Marathwada Mitra Mandal's

COLLEGE OF ENGINEERING, Karvenagar, Pune - 52

An Autonomous Institute affiliated to SPPU

Website: www.mmcoe.edu.in email id: mmcoe@mmcoe.edu.in

Honours Courses

The list of Honours courses to be offered from the academic year 2026-27.

Undergraduate Programme	
Offered by BoS	**Honours Courses**
Electrical Engineering	Next Generation Electric Vehicle
Mechanical Engineering	Automotive Technology
Electronics and Telecommunication Engineering	Automotive Embedded Systems
	VLSI Design and Technology
Information Technology	Block Chain Technology
	Internet of Things
	Game Design & Development
	Enterprise Linux Administration
Computer Engineering	Green Computing Engineering
	Quantum Computing
Artificial Intelligence and Data Science	Cloud Professionals

Dr. H. M. Jadhav
Associate Dean MDM, OE & Honours Degree

Dr. B. P. Vasgi
Dean Academics

Dr. S. R. Sakhare
Principal

Marathwada Mitra Mandal's College of Engineering

Karvenagar, Pune- 52

An Autonomous Institute Affiliated to SPPU

"येथे बहुतांचे हित"

Curriculum Structure and Syllabus

Honours

A.Y. 2026-27

Accredited with A++ grade by NAAC
Recipient of Best College award by SPPU | Accredited by NBA
Recognized under 2(f) and 12(B) of UGC Act 1956
www.mmcoe.edu.in

About Marathwada Mitra Mandal's College of Engineering

MMCOE has been recognised as the BEST COLLEGE by SPPU. MMCOE is one feather in the cap of Marathwada Mitra Mandal's Pune, from the academic year 2006 – 2007.

The Institute offers six undergraduate engineering programmes in Electrical Engineering, Mechanical Engineering, Electronics & Telecommunication Engineering, Information Technology, Computer Engineering, and Artificial Intelligence & Data Science. In addition, it offers six postgraduate programmes M.Tech (Electrical – Power Electronics & Drives), M.Tech (Mechanical – Design Engineering), M.Tech (Electronics & Telecommunication – VLSI & Embedded Systems), M.Tech (Information Technology – Data Science), M.Tech (Computer Engineering), along with a Ph.D. programme in Computer Engineering. It houses 2000+ students and 200 staff members. Along with the focus on core academics, college also takes care of overall personality development of the students.

The college has an active robotics cell, NCC, Training and Placement cell etc. The students are motivated to actively participate in co-curricular and extra-curricular activities. They have brought many laurels to the Institute in the events such as Vinodottam Karandak, Firodiya Karandak, Zest, Dexterity etc.

Institute Vision

To be a globally renowned institution through excellence in engineering education for sustainable and holistic development

Institute Mission

M1: Empower students with cutting-edge technologies and global competencies

M2: Foster culture of research and entrepreneurial mindset

M3: Imbibe social and professional values

M4: Provide an inclusive environment for lifelong learning

Department of Electrical Engineering

Vision and Mission of the Department

- **Vision**

To be center of excellence in electrical engineering by developing globally skilled professionals through innovation, sustainability and technological growth.

- **Mission**

M1: To provide quality education in electrical engineering through multidisciplinary, practical and industry oriented learning.

M2: To develop ethical and skilled engineers with expertise in emerging electrical technologies.

M3: To nurture lifelong learning and adaptability for addressing global, societal and future challenges.

- **Program Specific Outcomes (PSOs)**

After successfully completing the degree program. Electrical engineering graduates will be able to:

PSO1: Design and validate efficient electrical system through core and professional skills.

PSO2: Exhibit multidisciplinary skills in the area of Robotics, AI and Machine Learning for Electrical engineering applications.

PSO3: Formulate sustainable solutions for research in societal and industrial needs pertaining to green technology.

- **Program Outcomes (POs)**

PO1: Engineering Knowledge: Apply knowledge of mathematics, natural science, computing, engineering fundamentals and an engineering specialization as specified in WK1 to WK4 respectively to develop to the solution of complex engineering problems.

PO2: Problem Analysis: Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development. (WK1 to WK4)

PO3: Design/Development of Solutions: Design creative solutions for complex engineering problems and design/develop systems/components/processes to meet identified needs with consideration for the public health and safety, whole-life cost, net zero carbon, culture, society and environment as required. (WK5)

PO4: Conduct Investigations of Complex Problems: Conduct investigations of complex engineering problems using research-based knowledge including design of experiments, modelling, analysis & interpretation of data to provide valid conclusions. (WK8).

PO5: Engineering Tool Usage: Create, select and apply appropriate techniques, resources and modern engineering & IT tools, including prediction and modelling recognizing their limitations to solve complex engineering problems. (WK2 and WK6)

PO6: The Engineer and The World: Analyze and evaluate societal and environmental aspects while solving complex engineering problems for its impact on sustainability with reference to economy, health, safety, legal framework, culture and environment. (WK1, WK5, and WK7).

PO7: Ethics: Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national & international laws. (WK9)

PO8: Individual and Collaborative Team work: Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams.

PO9: Communication: Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations considering cultural, language, and learning differences

PO10: Project Management and Finance: Apply knowledge and understanding of engineering management principles and economic decision-making and apply these to one's own work, as a member and leader in a team, and to manage projects and in multidisciplinary environments.

PO11: Life-Long Learning: Recognize the need for, and have the preparation and ability for i) independent and life-long learning ii) adaptability to new and emerging technologies and iii) critical thinking in the broadest context of technological change. (WK8)

Curriculum for B. Tech. in Electrical Engineering with Honours in Next Generation Electric Vehicle

Department of Electrical Engineering

General Objectives and Scope:

The Honours programme in Electrical Engineering is designed to provide students with advanced and in-depth knowledge in specialized domains of electrical engineering beyond the regular undergraduate curriculum. The programme aims to strengthen conceptual understanding and analytical abilities, enabling students to explore emerging technologies and develop innovative solutions to real-world engineering problems.

Pursuing an honours degree in Electrical Engineering allows students to gain deeper expertise in areas such as power systems, renewable energy systems, smart grids, power electronics, electrical machines, control systems, and advanced electrical technologies. This extended academic engagement enhances the student's technical competence and research aptitude, preparing them to address complex engineering challenges in modern power and energy sectors.

Completing an honours degree in Electrical Engineering not only prepares students for higher education and research opportunities in fields such as power engineering, renewable energy systems, smart grid technologies, and automation, but also demonstrates to prospective employers a higher level of technical expertise and dedication

Why Next Generation Electric Vehicle?

Next Generation Electric Vehicles (EVs) are becoming essential due to the increasing need for sustainable, efficient, and environmentally friendly transportation systems. Conventional vehicles powered by internal combustion engines rely heavily on fossil fuels, which contribute to air pollution, greenhouse gas emissions, and depletion of natural resources. Next generation EVs provide an advanced alternative by utilizing electric power, improved battery technologies, and smart energy management systems.

Another important factor is energy efficiency and reduced dependence on fossil fuels. Electric vehicles convert electrical energy into mechanical energy more efficiently compared to conventional engines. With the integration of renewable energy sources such as solar and wind power, EVs can further reduce reliance on non-renewable energy resources.

Prerequisites:

- Basic Electrical Engineering
- Electrical Machines
- Power Electronics
- Energy Storage Systems
- Basic Electronics and Sensors
- Fundamentals of Renewable Energy

B. TECH honours IN Next Generation Electric Vehicle

(Semester IV, V, VI, VII, VIII)

Curriculum Structure

Course Code	Course Name	Teaching Scheme (Hrs/week)				Examination Scheme						Credits				
		L	P	T	ODL	CIE	ETE	TW	PR	OR	Total	L	P	T	ODL	Total
Semester IV																
EE24HON291R	Power Electronics Systems and Grid Integration for Electric Vehicles	3	-	-	-	40	60	-	-	-	100	3	-	-	-	3
EE24HON292R	Power Electronics Systems and Grid Integration for Electric Vehicles TUT	-	-	1	-	-	-	25	-	-	25	-	-	1	-	1
Semester V																
EE24HON341R	Vehicle Dynamics and control	3	-	-	-	40	60	-	-	-	100	3	-	-	-	3
EE24HON342R	Vehicle Dynamics and control TUT	-	-	1	-	-	-	25	-	-	25	-	-	1	-	1
Semester VI																
EE24HON391R	Sustainable Energy and EVs	3	-	-	-	40	60	-	-	-	100	3	-	-	-	3
EE24HON392R	Sustainable Energy and EVs TUT	-	-	1	-	-	-	25	-	-	25	-	-	1	-	1
Semester VII																
EE24HON441R	Smart Grid: Basics to Advanced Technologies -	3	-	-	-	40	60	-	-	-	100	3	-	-	-	3
EE24HON442R	NPTEL Course	-	-	-	1	-	-	25	-	-	25	-	-	-	1	1
Semester VIII																
EE24HON491R	Capstone Project / Case on Innovation and research in EV	-	8	-	-	-	-	100	-	-	100	-	4	-	-	4
Total		12	8	3	1	160	240	200	-	-	600	12	4	3	1	20

L- Lecture   P- Practical   T- Tutorial   CIE-Continuous Internal Evaluation
ETE- End Term Examination   TW- Term work   PR- Practical   OR- Oral
L : 1 Hr.= 1 credit   P: 2 Hr. = 1 Credit   T: 1 Hr. = 1 Credit   ODL-Open Distance Learning

Note: Mode of conduction of ETE will be decided by course coordinator in the supervision of BoS chairperson.

Second Year B. Tech. Electrical Engineering														
Semester: IV														
Course Code: EE24HON291R Course Name: Power Electronics Systems and Grid Integration for Electric Vehicles														
Teaching Scheme (Hours/Week)				**Examination Scheme**						**Credits**				
L	P	T	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	ODL	TOTAL
3	-		-	40	60	-	-	-	100	3	-	-	-	3

Prerequisite: Basic Electrical Engineering / Circuits, Power Electronics fundamentals, Power Systems fundamentals

Course Objectives:

- To understand the role of power semiconductor devices, power electronic components, and converter topologies used in electric vehicle (EV) powertrains.
- To analyse DC–DC converters, inverters, and advanced power converter topologies employed for EV propulsion and battery charging systems.
- To study EV charging architectures, including on-board and off-board chargers, power factor correction techniques, and fast-charging methods.
- To explore grid integration techniques for electric vehicles, including smart charging strategies, load management, and Vehicle-to-Grid (V2G) systems.
- To apply power electronic system concepts for effective interaction between electric vehicles and the electrical grid, considering grid stability, power quality, and ancillary services.

Course Outcomes:

After learning the course, the students will be able to:

CO1: Explain the operation and selection of power semiconductor devices and converter components used in EV applications.

CO2: Analyse the performance of DC–DC converters and inverter topologies used in EV propulsion systems.

CO3: Design and evaluate EV charging systems with appropriate power factor correction and fast-charging techniques.

CO4: Assess smart charging and V2G strategies for effective EV–grid interaction.

CO5: Apply power electronics principles to address grid stability, power quality, and ancillary service requirements in EV integration.

Unit	Contents	Duration (Hrs.)
1	**Introduction to Electric Vehicles and Power Electronics** Overview of Electric Vehicles: HEV, PHEV, BEV, FCEV, EV powertrain architecture and energy flow, Role of power electronics in EV systems, Power semiconductor devices: MOSFET, IGBT, SiC, GaN devices, Selection of devices for EV applications, Thermal management and protection in EV power electronics. Protection and Reliability Issues in EV Power Electronics Emerging Trends and Future Challenges.	8
2	**DC–DC Converters for EV Applications** Requirements of DC–DC converters in EVs, Non-isolated converters: Buck, Boost, Buck–Boost, Bidirectional DC–DC converters for battery interfacing, Isolated DC–DC converters: Flyback, Forward, Full-Bridge, Soft-switching techniques, Efficiency, control, and design considerations, Efficiency Analysis and Loss Modelling, Control Strategies for DC–DC Converters, Modelling, Simulation, and Practical Implementation	9
3	**Inverters and Motor Drive Systems for EVs** Role of inverters in EV propulsion, Voltage Source Inverter (VSI) for EV drives, PWM techniques for EV inverters, Multilevel inverters for high-power EV applications, Control of traction motors (BLDC, PMSM, Induction Motor), Regenerative braking	8

	and energy recovery, Efficiency and Loss Analysis of EV Inverters, EMI/EMC and Power Quality Issues, Advanced and Emerging Trends.	
4	**EV Charging Systems and Infrastructure** EV charging standards and levels (AC Level 1, 2, DC Fast Charging), On-board and off-board charger architectures, Power factor correction (PFC) converters, Fast-charging technologies and challenges, Wireless charging for EVs, Safety, isolation, and communication requirements.	7
5	**Grid Integration of Electric Vehicles** Impact of EV charging on power grid, Smart charging strategies and load management, Vehicle-to-Grid (V2G), Vehicle-to-Home (V2H), Vehicle-to-Building (V2B), Power quality issues: harmonics, voltage sag/swell, Grid stability and ancillary services using EVs, Standards and regulatory aspects of EV–grid integration	8
	Total Hours	40

Text Books

1. L. Ashok Kumar & S. Albert Alexander, “Power Converters for Electric Vehicles”,CRC Press,2020.
2. Kishan, Kannan, Reddy, Prabhakaran,“ Power Electronics for Electric Vehicles and Energy Storage: Emerging Technologies and Developments,” CRC Press,2023
3. Bimal K. Bose, “Power Electronics in Renewable Energy Systems and Smart Grid: Technology and Applications”,Wiley,2019

Reference Books

1. R. Krishnan, “Electric Drives: Modelling, Analysis and Control”,PHI,2007
2. Iqbal Hussain, “Electric and Hybrid Electric Vehicles Design Fundamentals”,CRC Press,2010.
3. Joseph Vithayathil,,“ Power Electronics, Principles and Applications”, McGraw Hi Series,2013
4. Vedam Subramaniam,“ Electric Drives: Concepts and Applications”, Tata McGraw Hill,2011.

Online References

1. NPTEL Course: Prof. G. Bhuvaneshwari,IIT,Delhi,“ Power Electronics”(https://nptel.ac.in/courses/13b5bfe96f3e).
2. NPTEL Course: Prof. Bhim Singh,IIT Delhi,“ Advanced Power Electronics”(https://nptel.ac.in/courses/ noc25_ee02)
3. NPTEL Course: Prof. Amit Jain, Prof. Avanish Tripathi ,IIT Delhi,“ EV - Vehicle Dynamics and Electric Motor Drives”(https://nptel.ac.in/courses/noc26_ee27)

Second Year B. Tech. Electrical Engineering

Semester-IV

Course Code: EE24HON292R **Course Name: Power Electronics Systems and Grid Integration for Electric Vehicles**

Teaching Scheme (Hours/Week)				Examination Scheme						Credits				
L	**P**	**T**	**ODL**	**CIE**	**ETE**	**TW**	**OR**	**PR**	**TOTAL**	**L**	**P**	**T**	**ODL**	**TOTAL**
-	-	1	-	-	-	25	-	-	25	-	-	1	-	1

Prerequisite: Basic Electrical Engineering / Circuits, Power Electronics fundamentals, Power Systems fundamentals

Course Objectives:
- Reinforce theoretical concepts through numerical problem solving
- Develop analytical skills in converter, inverter, and EV system design
- Introduce modelling, loss calculations, and real-world EV case studies
- Prepare students for examinations, industry applications, and research

Course Outcomes:
After learning the course, the students will be able to:
CO1: Explain the fundamentals of electric vehicle technologies including HEV, PHEV, BEV, and FCEV, and describe EV powertrain architecture, energy flow, and the role of power electronics in EV systems.
CO2: Analyze and select appropriate power semiconductor devices for EV applications
CO3: Design and evaluate DC–DC converter topologies (isolated and non-isolated) used in EVs
CO4: Analyze and implement inverter and motor drive systems for EV propulsion, including PWM techniques
CO5: Explain and evaluate EV charging systems and infrastructure, including on-board/off-board chargers

Tutorial	Contents	Duration (Hrs.)
1	**EV Classification and Powertrain Analysis.** • Compare HEV, PHEV, BEV, and FCEV architectures • Draw and explain EV powertrain block diagrams • Energy flow analysis during acceleration and braking • Short conceptual questions on role of power electronics	1
2	**Power Semiconductor Devices for EV Applications** • Comparative study of MOSFET, IGBT, SiC, and GaN devices • Device selection criteria for traction inverter and DC–DC converters • Switching and conduction loss calculations • Thermal and reliability considerations	1
3	**Thermal Management, Protection, and Reliability** • Thermal modelling and junction temperature calculation • Cooling techniques used in EV power electronics • Protection schemes: overcurrent, overvoltage, short-circuit • Reliability and failure analysis case study	1
4	**DC–DC Converter Requirements in EVs** • Functional requirements of DC–DC converters in EVs • Battery interfacing using bidirectional converters • Duty cycle and voltage conversion ratio problems • Efficiency evaluation of non-isolated converters	1
5	**Non-Isolated DC–DC Converters** • Buck, Boost, Buck–Boost converter analysis • Design-oriented numerical problems	1

	• Performance comparison for EV applications • Ripple and loss calculations	
6	**Isolated DC–DC Converters and Soft Switching** • **Flyback, Forward, and Full-Bridge converter operation** • **Transformer design basics and turns ratio calculations** • **Soft-switching techniques (ZVS, ZCS)** • **Efficiency and loss modelling problems**	2
7	**Modelling, Control, and Simulation of DC–DC Converters** • **Converter modelling concepts** • **Voltage-mode and current-mode control strategies** • **Controller parameter calculations** • **Simulation-based problem solving (MATLAB/Simulink)**	2
8	**Inverters for EV Propulsion** • **Voltage Source Inverter (VSI) operation** • **PWM techniques: SPWM and SVPWM** • **Modulation index and harmonic analysis** • **Switching loss and efficiency calculations**	2
9	**Motor Drive Systems and Regenerative Braking** • **BLDC, PMSM, and Induction motor control in EVs** • **Torque–speed characteristics analysis** • **Regenerative braking energy recovery calculations** • **Efficiency comparison of motor drive systems**	2
10	**EV Charging Systems and Grid Integration** • **EV charging standards and levels (AC, DC fast charging)** • **On-board vs off-board charger architectures** • **Power factor correction (PFC) converters** • **Smart charging, V2G/V2H/V2B, power quality issues**	2
	Total Hours	**15**

Sr.No.	**List of Tutorials**
1	Compare HEV, PHEV, BEV, and FCEV with respect to energy source, power electronics used, and typical applications.
2	An EV inverter operates at 20 kHz and 400 V. Justify the selection of a suitable power semiconductor device considering efficiency and thermal constraints.
3	Explain thermal management techniques used in EV power electronics. Calculate the junction temperature of a device given thermal resistance and power loss data.
4	Design a Buck converter for an EV auxiliary system with given specifications and calculate inductor and capacitor values.
5	Explain Zero Voltage Switching (ZVS) and Zero Current Switching (ZCS). How do soft-switching techniques improve efficiency in EV converters?
6	Develop a basic small-signal model of a DC–DC converter and explain its use in controller design.(Simulation in any kind of software)
7	Explain the operation of a three-phase Voltage Source Inverter (VSI) used in EVs. Derive the relationship between modulation index and output voltage.
8	Compare SPWM and SVPWM techniques for EV inverters with respect to DC bus utilization and harmonic content.
9	Explain the principle of regenerative braking in EVs and calculate the recoverable energy for a given vehicle speed and mass.

10	Explain EV charging levels and standards. Compare on-board and off-board chargers with suitable block diagrams.
Text Books	
	1. L. Ashok Kumar & S. Albert Alexander, "Power Converters for Electric Vehicles",CRC Press,2020. 2. Kishan, Kannan, Reddy, Prabhakaran," Power Electronics for Electric Vehicles and Energy Storage: Emerging Technologies and Developments," CRC Press,2023 3. Bimal K. Bose, "Power Electronics in Renewable Energy Systems and Smart Grid: Technology and Applications",Wiley,2019
Reference Books	
	1. R. Krishnan, "Electric Drives: Modelling, Analysis and Control",PHI,2007 2. Iqbal Hussain, "Electric and Hybrid Electric Vehicles Design Fundamentals",CRC Press,2010. 3. Joseph Vithayathil,," Power Electronics, Principles and Applications", McGraw Hi Series,2013 4. Vedam Subramaniam," Electric Drives: Concepts and Applications", Tata McGraw Hill,2011.
Online References	
	1. NPTEL Course: Prof. G. Bhuvaneshwari,IIT,Delhi," Power Electronics"(https://nptel.ac.in/courses/ 13b5bfe96f3e). 2. NPTEL Course: Prof. Bhim Singh,IIT Delhi," Advanced Power Electronics"(https://nptel.ac.in/courses/ noc25_ee02) 3. NPTEL Course: Prof. Amit Jain, Prof. Avanish Tripathi ,IIT Delhi," EV - Vehicle Dynamics and Electric Motor Drives"(https://nptel.ac.in/courses/noc26_ee27)

Third Year B. Tech. Electrical Engineering

Semester-V

Course Code: EE24HON341R **Course Name: Vehicle Dynamics and Control**

Teaching Scheme (Hours/Week)				Examination Scheme						Credits				
L	P	T	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	ODL	TOTAL
3	-	-	-	40	60	-	-	-	100	3	-	-	-	3

Prerequisite: Engineering Mechanics, Basic Dynamics,Fundamentals of Mechatronics,Matlab simulation

Course Objectives:
- Provide fundamental understanding of vehicle dynamics related to longitudinal, lateral, and vertical motion.
- Enable analysis of forces and moments acting on vehicles during different driving conditions.
- Introduce modelling and control of vehicle subsystems such as steering, braking, and suspension.
- Develop knowledge of modern vehicle control systems for safety and stability.
- Apply control strategies to improve vehicle performance, comfort, and safety.

Course Outcomes:
After learning the course, the students will be able to:
CO1: Explain vehicle coordinate systems, forces, and basic vehicle dynamic models.
CO2: Analyze longitudinal vehicle dynamics including acceleration and braking.
CO3: Model and analyze suspension and ride dynamics of vehicles
CO4: Apply lateral and yaw dynamics related to steering and handling.
CO5: Evaluate modern vehicle dynamics control systems using case studies.

Unit	Contents	Duration (Hrs.)
1	**Fundamentals of Vehicle Dynamics** Vehicle coordinate systems and reference frames, Degrees of freedom of a road vehicle, Forces acting on a vehicle,Tire–road interaction fundamentals,Vehicle mass distribution and center of gravity,Load transfer during acceleration and braking, Aerodynamic forces and drag, Rolling resistance, Road grade effects on vehicle motion.	8
2	**Longitudinal Vehicle Dynamics** Longitudinal motion equations of vehicles,Traction and braking forces, Acceleration performance analysisBraking dynamics and stopping distance,Weight transfer during braking,Tire slip ratio and longitudinal slip Anti-lock Braking System (ABS) fundamentals, Traction control system (TCS) basics,Regenerative braking concept (EV focus),	9
3	**Lateral and Yaw Dynamics** Steering system fundamentals,Vehicle handling characteristics,Tire cornering force and slip angle, bicycle (single-track) vehicle model, Understeer, oversteer, and neutral steer, Yaw rate dynamics,Lateral acceleration and stability,Lane change and cornering analysis, Vehicle stability limits,	8
4	**Vertical Dynamics and Suspension Systems** Ride comfort and road holding, Suspension system components, Quarter-car model, Sprung and unsprung mass dynamics, Passive suspension systems, Active and semi-active suspension, Damping characteristics, Vehicle vibration analysis, Suspension control objectives	7
5	**Vehicle Dynamics Control Systems** Introduction to vehicle control systems, Control objectives for vehicle dynamics, Electronic Stability Control (ESC),Yaw stability control, Active steering systems, Integrated chassis control, Control algorithms for vehicle dynamics, Sensor requirements for vehicle control, Case studies in vehicle dynamics control	8
	Total Hours	**40**

Text Books	
1. Thomas D. Gillespie, "Fundamentals of Vehicle Dynamics", SAE International,1999 2. Rajesh Rajamani," Vehicle Dynamics and Control,"Springer,2012. 3. Reza N. Jazar, "Vehicle Dynamics: Theory and Application", Springer,2008.	
Reference Books	
1. Wong, J. Y.," Theory of Ground Vehicles",Wiley,2008. 2. Milliken & Milliken, "Race Car Vehicle Dynamics",SAE,1999. 3. Genta & Morello," The Automotive Chassis", Springer,2008. 4. Bosh,"Automotive Handbook" Friedrichshafen and Karlsruhe,2022	
Online References	
1. NPTEL Course: Prof.R.Krishnakumar, IIT Madras," Vehicle Dynamics"(https://nptel.ac.in/courses/107106080). 2. NPTEL Course: Prof. Amit Jain ,IIT Delhi,"Electric Vehicles Part-I"(https://nptel.ac.in/courses/noc22_ee53) 3. NPTEL Course: Prof. Amit Jain, Prof. Avanish Tripathi,IIT Delhi," EV - Vehicle Dynamics and Electric Motor Drives"(https://nptel.ac.in/courses/noc26_ee27)	

Third Year B. Tech. Electrical Engineering

Semester- V

Course Code: EE24HON342R **Course Name: Vehicle Dynamics and Control**

Teaching Scheme (Hours/Week)				Examination Scheme						Credits				
L	**P**	**T**	**ODL**	**CIE**	**ETE**	**TW**	**OR**	**PR**	**TOTAL**	**L**	**P**	**T**	**ODL**	**TOTAL**
-	-	1	-	-	-	25	-	-	25	-	-	1	-	1

Prerequisite: Basic Electrical and Electronics Engineering Circuits, Power Electronics fundamentals, Power Systems fundamentals.

Course Objectives:

- Develop fundamental understanding of vehicle dynamics related to longitudinal, lateral, and vertical motions
- Enable students to analyze forces, moments, and tire–road interactions acting on a vehicle.
- Introduce mathematical modeling of vehicle subsystems such as steering, braking, and suspension.
- Provide knowledge of modern vehicle dynamics control systems for safety and stability.
- Apply control system concepts to improve vehicle performance, comfort, and handling.

Course Outcomes:

After learning the course, the students will be able to:

CO1: Explain vehicle coordinate systems, forces, and basic dynamic models of vehicles.

CO2: longitudinal vehicle dynamics including acceleration, braking, and load transfer.

CO3: Analyze lateral and yaw dynamics related to steering and handling behavior.

CO4: Model and analyze suspension systems and vertical vehicle dynamics.

CO5 Evaluate vehicle dynamics control strategies using real-world case studies.

Tutorial	**Contents**	**Duration (Hrs.)**
1	**Fundamentals of Vehicle Dynamics** ● Define vehicle coordinate systems and reference frames. ● Identify degrees of freedom in vehicle motion. ● List forces acting on a moving vehicle. ● Explain the concept of center of gravity. ● Numerical problems on tractive force calculation	2
2	**Longitudinal Vehicle Dynamics** ● Longitudinal equations of motion ● Traction and braking force modelling ● Acceleration performance analysis ● Braking dynamics and stopping distance ● Weight transfer during braking ● Tire slip ratio and longitudinal slip ● ABS fundamentals ● Traction Control System (TCS) ● Regenerative braking (EV perspective)	1
3	**Lateral and Yaw Dynamics** ● Steering system fundamentals ● Tire cornering force and slip angle ● Bicycle (single-track) vehicle model ● Understeer, oversteer, and neutral steer ● Yaw rate dynamics ● Lateral acceleration and vehicle stability ● Lane change and cornering analysis	1

	• Vehicle stability limits	
4	**Vertical Dynamics and Suspension Systems** • Ride comfort and road holding • Suspension system components • Quarter-car model • Sprung and unsprung mass dynamics • Passive suspension systems • Active and semi-active suspension • Damping characteristics • Vehicle vibration analysis • Suspension control objectives	1
5	**Vehicle Dynamics Control Systems** • Vehicle dynamics control objectives • Electronic Stability Control (ESC) • Yaw stability control • Active steering systems • Integrated chassis control • Control algorithms for vehicle dynamics • Sensor requirements	1
6	**Tire Modeling and Advanced Tire Dynamics** • Role of tires in vehicle dynamics • Tire force and moment generation • Longitudinal, lateral, and combined slip • Linear and nonlinear tire models • Magic Formula (Pacejka) tire model • Tire relaxation length and transient behavior • Road friction estimation techniques • Tire wear and temperature effects on dynamics	1
7	**Multibody Vehicle Dynamics and Modeling** • Vehicle as a multibody system • Coordinate systems and constraints • Kinematic and dynamic modeling approaches • Full-car models vs reduced-order models • Suspension kinematics and compliance • Modeling of steering and drivetrain subsystems • Introduction to vehicle dynamics simulation tools (MATLAB/Simulink, ADAMS/Car, CarSi	2
8	**Advanced Braking and Steering Systems** • Advanced braking system architectures • Electronic Brakeforce Distribution (EBD) • Brake-by-wire systems • Steer-by-wire systems • Four-wheel steering systems • Integrated braking and steering control • Redundancy and fault tolerance in chassis systems • Safety and regulatory aspects	2
9	**Vehicle Dynamics for Electric and Hybrid Vehicles** • Impact of electrification on vehicle dynamics • In-wheel motors and torque vectoring • Regenerative braking coordination	2

	• EV-specific traction and stability challenges • Torque distribution strategies for EVs • Chassis control with electric drivetrains • Thermal effects on dynamics components • Case studies of EV platform	
10	**Vehicle Dynamics for ADAS and Autonomous Vehicles** • Vehicle dynamics requirements for ADAS • Motion planning and trajectory tracking • Vehicle models for autonomous control • Lateral and longitudinal control strategies • Path tracking controllers (PID, LQR, MPC basics) • Interaction between perception, planning, and control • Stability and safety constraints • Real-world case studies in autonomous driving	2
	Total Hours	**15**

Sr.No.	**List of Tutorials**
1	Numerical problems on vehicle coordinate systems, DOF classification, and force decomposition.
2	Calculation of CG location, axle loads, and load transfer during acceleration and braking.
3	Estimation of aerodynamic drag, rolling resistance, and grade resistance for given driving conditions.
4	Longitudinal vehicle motion: acceleration performance and tractive effort calculations.
5	Braking dynamics: stopping distance, braking force distribution, and weight transfer analysis.
6	Regenerative braking in EVs: braking force sharing and energy recovery estimation.
7	Tire cornering force, slip angle calculation, and steering response analysis.
8	Bicycle model analysis: understeer, oversteer, yaw rate, and lateral acceleration computation.
9	Lane change and steady-state cornering analysis; vehicle stability limits.
10	Case study–based tutorial on ESC / yaw stability control / integrated chassis control using block diagrams and simulations.

Text Books

1. Thomas D. Gillespie, "Fundamentals of Vehicle Dynamics", SAE International,1999
2. Rajesh Rajamani," Vehicle Dynamics and Control,"Springer,2012.
3. Reza N. Jazar, "Vehicle Dynamics: Theory and Application", Springer,2008.
4. Pacejka, H., "Tire and Vehicle Dynamics", Elsvier,2014.

Reference Books

1. Wong, J. Y.," Theory of Ground Vehicles",Wiley,2008.
2. Milliken & Milliken, "Race Car Vehicle Dynamics",SAE,1999.
3. Genta & Morello," The Automotive Chassis", Springer,2008.
4. Bosh,"Automotive Handbook" Friedrichshafen and Karlsruhe,2022
5. Heißing, M. & Ersoy, M.," Chassis Handbook", Springer,2010.

Online References

1. NPTEL Course: Prof.R.Krishnakumar, IIT Madras," Vehicle Dynamics"(https://nptel.ac.in/courses/107106080).
2. NPTEL Course: Prof. Amit Jain ,IIT Delhi,"Electric Vehicles Part-I"(https://nptel.ac.in/courses/ noc22_ee53)
3. NPTEL Course: Prof. Amit Jain, Prof. Avanish Tripathi,IIT Delhi," EV - Vehicle Dynamics and Electric Motor Drives"(https://nptel.ac.in/courses/noc26_ee27)

Third Year B. Tech. Electrical Engineering

Semester-VI

Course Code: EE24HON391R **Course Name: SUSTAINABLE ENERGY AND ELECTRIC VEHICLES**

Teaching Scheme (Hours/Week)				Examination Scheme						Credits				
L	P	T	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	ODL	TOTAL
3	-	-	-	40	60	-	-	-	100	3	-	-	-	3

Prerequisite: Electrical circuits and machines, Energy conversion principles, Basic power electronics, Fundamentals of automobiles (desirable),Environmental studies / energy systems (basic)

Course Objectives:

- Understand the concepts of sustainability, energy transition, and climate impact.
- Analyze renewable energy sources and their integration with electric vehicles.
- Understand EV architecture, components, and energy storage systems.
- Evaluate charging infrastructure and grid interaction for EVs.
- Apply sustainable energy concepts to real-world EV applications and policies.

Course Outcomes:

After learning the course, the students will be able to:

CO1: Explain sustainability concepts and energy transition
CO2: Analyze renewable energy sources for EV integration
CO3: Describe EV architecture and energy storage systems
CO4: Evaluate EV charging methods and grid interaction
CO5: Assess policies and sustainability impact of EV adoption

Unit	Contents	Duration (Hrs.)
1	**Sustainable Energy and Energy Transition** Concept of sustainability,Global energy scenario and climate change, Carbon footprint and life-cycle assessment (LCA),Fossil fuels vs renewable energy sources,Energy transition pathways, Sustainable transportation systems, Role of EVs in decarbonisation.	8
2	**Renewable Energy Sources** Solar energy systems (PV basics, grid & off-grid), Wind energy systems, Biomass and biofuels,Small hydro and emerging renewables, Renewable energy storage systems, Intermittency and energy management, Integration of renewables with EV charging	9
3	**Electric Vehicle Fundamentals** Evolution of electric vehicles,Types of EVs: BEV, HEV, PHEV, FCEV,EV powertrain architecture,Electric motors used in EV, Power electronic converters in EV, Energy flow in EV systems, Comparison of EVs with ICE vehicles	8
4	**Energy Storage and EV Charging** Battery technologies (Li-ion, LFP, Solid-state),Battery characteristics and performance parameters, Battery management systems (BMS) Super capacitors and hybrid storage, EV charging levels and standards, Wireless charging basics, Battery swapping concepts.	7
5	**Grid Integration, Policy, and Sustainability** Impact of EVs on the power grid, Smart charging and demand response,Vehicle-to-Grid (V2G) and V2H, Renewable-powered EV charging stations, EV policies and incentives (India & global), Sustainability assessment of EVs, Future trends in sustainable mobility	8
	Total Hours	40

Text Books

1. Guzzella, L., Sciarretta, A., "Vehicle Propulsion Systems", Springer,2005.
2. Larminie, J., Lowry, J.," Vehicle Dynamics and Control,"Wiley,2012.
3. Ehsani, M., Gao, Y., Emadi, A.," Modern Electric, Hybrid Electric and Fuel Cell Vehicles," CRC Press,2018.

Reference Books

1. Dincer, I., Rosen, M.," Sustainable Energy Systems and Applications", Springer,2012
2. Hughes, A., Drury, B., "Electric Motors and Drives", Elsevier,2019.
3. Chan, C. C., Chau, K. T.," Modern Electric Vehicle Technology", Oxford University Press,2011.
4. Bosch Handbook," Automotive & EV systems",Bosch,2022.

Online References

1. NPTEL Course: Prof. R. Anandalakshmi, Prof. Vaibhav Vasant Goud , IIT Guwahati," Renewable Energy Engineering"(https://nptel.ac.in/courses/ noc26_ch26).
2. NPTEL Course: Prof. Amit Jain ,IIT Delhi,"Electric Vehicles Part-I"(https://nptel.ac.in/courses/ noc22_ee53)
3. NPTEL Course: Prof.Kaushal Kumar Jha ,IIT Madras," Battery Management Systems"(https://nptel.ac.in/courses/ 108106182)

Third Year B. Tech. Electrical Engineering

Semester-VI

Course Code: EE24HON392R **Course Name: SUSTAINABLE ENERGY AND ELECTRIC VEHICLES**

Teaching Scheme (Hours/Week)				Examination Scheme						Credits				
L	**P**	**T**	**ODL**	**CIE**	**ETE**	**TW**	**OR**	**PR**	**TOTAL**	**L**	**P**	**T**	**ODL**	**TOTAL**
-	-	1	-	-	-	25	-	-	25	-	-	1	-	1

Prerequisite: Electrical circuits and machines, Energy conversion principles, Basic power electronics, Fundamentals of automobiles (desirable),Environmental studies / energy systems (basic)

Course Objectives:

- Introduce students to sustainability concepts and carbon footprint analysis in transportation.
- Enable analysis of renewable energy sources for EV charging applications.
- Develop understanding of energy flow, efficiency, and performance of EV powertrains.
- Provide hands-on exposure to battery performance evaluation and charging calculations.
- Analyze policy,economic, and environmental impacts of EV adoption.

Course Outcomes:
After learning the course, the students will be able to:
CO1: Estimate renewable energy yield for EV charging applications
CO2: Analyze and compare energy flow in EV and ICE powertrains
CO3: Evaluate battery performance parameters and charging requirements
CO4: Design conceptual EV charging stations with renewable integration
CO5 Assess EV policies and sustainability impact using case studies.

Tutorial	Contents	Duration (Hrs.)
1	**Carbon Footprint Calculation and Life-Cycle Analysis of Vehicles.** ● Definition of carbon footprint ● Direct and indirect emissions ● Well-to-Wheel (WTW) vs Tank-to-Wheel (TTW) analysis ● Life-cycle stages: ● Manufacturing ● Operation ● End-of-life ● Carbon footprint comparison: ● ICE vs EV ● EV charged from grid vs renewable energy	2
2	**Energy Yield Estimation from Solar/Wind Sources for EV Charging** ● Solar PV energy yield calculation ● Wind energy power estimation ● Capacity factor and availability ● Seasonal variation of renewable sources ● 🎬 Energy demand of EV charging	1
3	**EV Powertrain Energy Flow Analysis and Comparison with ICE Vehicles** ● EV powertrain components and energy flow ● ICE powertrain energy losses ● Efficiency comparison ● Regenerative braking energy recovery	1

	● Energy consumption per km	
4	**Battery Performance Calculation – SOC, SOH, Charging TimeRide comfort and road holding.** ● Battery capacity and C-rate ● State of Charge (SOC) calculation ● State of Health (SOH) estimation ● Charging time estimation (AC/DC) ● Depth of Discharge (DoD)	2
5	**EV Charging Station Design with Renewable Integration** ● Types of EV charging stations ● Load estimation for charging station ● Solar-grid hybrid charging architecture ● Energy storage requirement ● 🎬 Basic layout and block diagram	1
6	**Smart Charging and V2G Numerical Problems** ● Smart charging principles ● Time-of-use pricing ● Load shifting and peak shaving ● Vehicle-to-Grid (V2G) concept ● Energy exchange calculations	2
7	**Case Study on EV Policies and Sustainability Impact** ● Overview of EV policies (India & global) ● FAME-II scheme ● State EV policies ● Incentives and subsidies ● 🎬 Sustainability and economic impact	1
8	**Mini Project on Sustainable Energy and EV Solutions** ● Problem identification ● Renewable-EV system design ● Energy and emission calculations ● Sustainability assessment ● 🎬 Report and presentation	1
9	**Energy Economics and Cost–Benefit Analysis of EVs and Renewable Integration** ● Capital cost vs operating cost of EVs ● Total Cost of Ownership (TCO) comparison: ● ICE vehicle vs EV ● Cost components: ● Vehicle cost ● Fuel/electricity cost ● Maintenance cost ● Battery replacement cost ● Economics of renewable-powered EV charging stations ● Payback period and break-even analysis ● Impact of subsidies and incentives on EV economics	2

10	**Energy Storage Systems and Second-Life EV Batteries** ● Overview of energy storage systems: ● Batteries ● Supercapacitors ● Hybrid storage ● Degradation mechanisms of EV batteries ● Concept of second-life batteries ● Challenges and safety considerations	2
	Total Hours	**15**

Sr.No.	**List of Tutorials**
1	Carbon footprint calculation and life-cycle analysis of vehicles..
2	Energy yield estimation from solar/wind sources for EV charging.
3	EV powertrain energy flow analysis and comparison with ICE vehicles.
4	Battery performance calculation: SOC, SOH, charging time estimation.
5	EV charging station design with renewable integration.
6	Smart charging and V2G numerical problems.
7	Case study on EV policies and sustainability impact.
8	Charging time calculation under different chargers
9	Impact analysis of government subsidy on EV cost
10	Mini project discussion on sustainable EV solutions.

Text Books

1. Guzzella, L., Sciarretta, A., "Vehicle Propulsion Systems", Springer,2005.
2. Larminie, J., Lowry, J.," Vehicle Dynamics and Control,"Wiley,2012.
3. Ehsani, M., Gao, Y., Emadi, A.," Modern Electric, Hybrid Electric and Fuel Cell Vehicles," CRC Press,2018.

Reference Books

1. Dincer, I., Rosen, M.," Sustainable Energy Systems and Applications", Springer,2012
2. Hughes, A., Drury, B., "Electric Motors and Drives", Elsevier,2019.
3. Chan, C. C., Chau, K. T.," Modern Electric Vehicle Technology", Oxford University Press,2011.
4. Bosch Handbook," Automotive & EV systems",Bosch,2022.

Online References

1. NPTEL Course: Prof. R. Anandalakshmi, Prof. Vaibhav Vasant Goud , IIT Guwahati," Renewable Energy Engineering"(https://nptel.ac.in/courses/ noc26_ch26).
2. NPTEL Course: Prof. Amit Jain ,IIT Delhi,"Electric Vehicles Part-I"(https://nptel.ac.in/courses/ noc22_ee53)
3. NPTEL Course: Prof.Kaushal Kumar Jha ,IIT Madras," Battery Management Systems"(https://nptel.ac.in/courses/ 108106182)

Semester-VII														
Course Code: EE24HON441R				Course Name: Smart Grid: Basics to Advanced Technologies										
Teaching Scheme (Hours/Week)				Examination Scheme						Credits				
L	P	T	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	ODL	TOTAL
3	-	-	-	40	60	-	-	-	100	3	-	-	-	3

Prerequisite: A basic understanding of Power Systems,Fundamentals of Power Electronics

Course Objectives:
- Introduce key concepts of smart grid architectures and components.
- Explore renewable energy integration and energy storage systems in smart grids.
- Understand monitoring systems such as PMUs and wide-area monitoring.
- Analyze microgrid operation and control strategies (AC, DC, and hybrid).
- Cover grid protection, islanding detection , demand response, and EV interface.

Course Outcomes:
After learning the course, the students will be able to:
CO1: Explain the architecture and technologies underlying modern smart grids.
CO2: Describe energy storage, phasor measurement, and monitoring tools used in grid operation.
CO3: Analyze microgrid operation and control modes, including AC and DC microgrids.
CO4: Evaluate protection mechanisms, cybersecurity issues, and demand response strategies.
CO5: of Understand integration renewables, EVs, and ancillary support mechanisms in smart grid.

Unit	Contents	Duration (Hrs.)
1	**Concept and Definition of Smart Grid** A Smart Grid is an advanced electricity network that uses digital communication, automation, sensing, and control technologies to efficiently generate, transmit, distribute, and consume electricity. Conventional grids, smart grids enable two-way power flow and two-way information flow, improving reliability, efficiency, and sustainability, Generation (centralized & distributed), Transmission network,Distribution system ,Smart meters, Sensors (PMUs, RTUs, Communication networks, Control centers, Energy management systems (EMS/DMS)	8
2	**Energy Storage Systems in Smart Grid** Battery Energy Storage Systems (BESS),Supercapacitors, Flywheels, Pumped hydro storage, Frequency regulation, Renewable smoothing, Backup power, Smart Grid Protection and Monitoring, Wide Area Monitoring Systems (WAMS),Phasor Measurement Units (PMUs), Adaptive and digital relays, Faster fault detection, Improved system stability.	9
3	**Microgrids and Distributed Generation** In grid-connected mode, In islanded mode, Microgrids, Reliability, Resilience, Demand response programs, Dynamic pricing, Smart appliances, Price signals, Grid conditions, Electric Vehicle Integration with Smart Grid, Distributed energy storage units, Smart charging,	8
4	**Advanced Smart Grid Topics** Placement of phasor measurement units Cyber security and resilience,Virtual inertia and ancillary support, Demand side management, Practical demonstrations (e.g., solar generation) and real case analyses	7
5	**Cyber security, Standards, and Future Trends** Data privacy, Secure communication, Protection against cyber-attacks, Standards, AI-based grid management, Blockchain for energy trading, Digital twins and autonomous grids.	8
	Total Hours	**40**

Text Books

1. A. Keyhani & M. Marwali , "Smart Power Grids", Springer,2011
2. Arun Phadke.," Computer Relaying for Power Systems," Second Edition,Wiley,2012.
3. Nikos Hatziargyriou.," Microgrids – Architecture and Control," Wiley,2013

Reference Books

1. Dincer, I., Rosen, M.," Sustainable Energy Systems and Applications", Springer,2012
2. Hughes, A., Drury, B., "Electric Motors and Drives", Elsevier,2019.
3. Yazdani & Iravani.," Voltage Sourced Converters in Power Systems", Wiley,2010
4. Fang Lin Luo & Hong Ye," Renewable Energy Systems",CRC Press,2012

Online References

1. NPTEL Course: Prof. N. P. Padhy, Prof. Premalata Jena, IIT Roorkee," Smart Grid:Basics to Advanced Technoligies"(https://nptel.ac.in/courses/ noc25_ee79).
2. NPTEL Course: Prof. Amit Jain ,IIT Delhi,"Electric Vehicles Part-I"(https://nptel.ac.in/courses/ noc22_ee53)
3. NPTEL Course: Prof.Kaushal Kumar Jha ,IIT Madras," Battery Management Systems"(https://nptel.ac.in/courses/ 108106182)

Final Year B. Tech. Electrical Engineering														
Semester-VII														
Course Code: EE24HON442R						**Course Name: Smart Grid: Basics to Advanced Technologies**								
Teaching Scheme (Hours/Week)				**Examination Scheme**						**Credits**				
L	**P**	**T**	**ODL**	**CIE**	**ETE**	**TW**	**OR**	**PR**	**TOTAL**	**L**	**P**	**T**	**ODL**	**TOTAL**
-	-	-	1	-	-	25	-	-	25	-	-	-	1	1

Prerequisite: A basic understanding of Power Systems**,** Fundamentals of Power Electronics**.**

Course Objectives:

- Introduce the concept, architecture, and need of smart grids.
- Explain advanced technologies enabling smart grid operation.
- Understand micro grid operation, control, and protection.
- Analyze renewable energy and energy storage integration.
- Study EV integration, demand response, cyber security, and future trends

Course Outcomes:
After learning the course, the students will be able to:
CO1: Explain smart grid concepts, components, and architecture.
CO2: Analyze renewable energy and energy storage integration in smart grids.
CO3: Apply microgrid and EV integration concepts to real-world systems..
CO4: Describe monitoring, protection, and communication technologies.
CO5 : Evaluate smart grid challenges, cybersecurity issues, and future technologies.

Tutorial	Contents	Duration (Hrs.)
1	**Introduction to Smart Grid (Basics)** *(Self-Learning Module + Recorded Lectures)* ● Evolution of conventional grid to smart grid ● Definition, features, and benefits of smart grid ● Smart grid architecture and functional layers ● Key components: smart meters, sensors, control centers ● Smart grid standards and policies (IEEE, IEC overview)	4
2	**Renewable Energy & Energy Storage Integration** *(Simulation-based learning / case study)* ● Distributed Energy Resources (DERs) ● Solar and wind integration challenges ● Energy storage technologies (BESS, flywheel, supercapacitor) ● Grid support functions of storage ● Forecasting and variability management	4
3	**Electric Vehicle Integration with Smart Grid** *(EV-focused module)* ● EV charging infrastructure ● Smart charging strategies ● Vehicle-to-Grid (V2G) concept ● Impact of EVs on grid stability ● EVs as distributed energy storage	2
4	**Monitoring, Protection, and Control** *(Virtual lab demonstrations)* ● Wide Area Monitoring Systems (WAMS) ● Phasor Measurement Units (PMUs) ● Smart grid protection challenges ● Digital relays and adaptive protection ● Self-healing and fault management	2

5	**Cybersecurity, Standards & Future Trends (Advanced)** *(Research-oriented learning)* ● Cyber threats in smart grids ● Data privacy and secure communication ● Smart grid standards (IEEE 1547, IEC 61850) ● AI, blockchain, and digital twins ● Future autonomous and resilient grids	3
	Total Hours	**15**

Text Books

1. A. Keyhani & M. Marwali , “Smart Power Grids”, Springer,2011
2. Arun Phadke.,“ Computer Relaying for Power Systems,” Second Edition,Wiley,2012.
3. Nikos Hatziargyriou.,“ Microgrids – Architecture and Control,” Wiley,2013

Reference Books

1. Dincer, I., Rosen, M.,“ Sustainable Energy Systems and Applications”, Springer,2012
2. Hughes, A., Drury, B., “Electric Motors and Drives”, Elsevier,2019.
3. Yazdani & Iravani.,“ Voltage Sourced Converters in Power Systems”, Wiley,2010
4. Fang Lin Luo & Hong Ye,“ Renewable Energy Systems”,CRC Press,2012

Online References

1. NPTEL Course: Prof. N. P. Padhy, Prof. Premalata Jena, IIT Roorkee,“ Smart Grid:Basics to Advanced Technoligies”(https://nptel.ac.in/courses/ noc25_ee79).
2. NPTEL Course: Prof. Amit Jain ,IIT Delhi,“Electric Vehicles Part-I”(https://nptel.ac.in/courses/ noc22_ee53)
3. NPTEL Course: Prof.Kaushal Kumar Jha ,IIT Madras,“ Battery Management Systems”(https://nptel.ac.in/courses/ 108106182)

Final Year B. Tech. Electrical Engineering														
Semester-VIII														
Course Code: EE24HON491R				Course Name: Capstone Project(Innovation and research in EV)										
Teaching Scheme (Hours/Week)				Examination Scheme						Credits				
L	P	T	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	ODL	TOTAL
-	8	-	-	-	-	100	-	-	100	-	4	-	-	4

Prerequisite: Electric Vehicles / Hybrid Vehicles, Power Electronics, Electrical Machines / Drives. Battery Energy Storage Systems. Dynamics, Fundamentals of Mechatronics, MATLAB simulation

Course Objectives:
- Expose students to next-generation EV technologies and architectures
- Enable analysis of forces and moments acting on vehicles during different driving conditions.
- Apply Prepare students for industry, startups, and higher studies such as AI, smart charging, V2G, and autonomy
- Develop research, innovation, and sustainability orientation.
- Prepare students for industry, startups, and higher studies.

Course Outcomes:
After learning the course, the students will be able to:
CO1: Identify and define problems related to next-generation EV systems
CO2: Design conceptual models of advanced EV architectures
CO3: Analyze energy, performance, and sustainability aspects
CO4: Apply modern tools and simulation for EV systems
CO5: Propose innovative solutions aligned with future mobility

Unit	Contents	Duration (Hrs.)
1	**Overview of Next Generation Electric Vehicles** ● Evolution of EV technologies ● Limitations of present-day EVs ● Characteristics of next-generation EVs ● Trends in global EV industry ● Role of sustainability and circular economy	20
2	**Advanced EV Powertrain Technologies** ● High-efficiency electric motors ● Multi-motor and in-wheel motor systems ● Advanced power electronics (SiC, GaN) ● High-voltage EV architectures (800V systems) ● Torque vectoring and advanced control	20
3	**Advanced Energy Storage and Charging** ● Solid-state batteries ● Ultra-fast charging technologies ● Wireless and dynamic charging ● Battery swapping ● Second-life and recycling of EV batteries	20
4	**Intelligent EV Systems** ● AI and ML applications in EVs ● Predictive energy management ● Smart BMS and thermal management ● Vehicle connectivity (V2X) ● Cybersecurity in EV systems	20
5	**EV Integration with Smart Grid and Future Mobility** ● Vehicle-to-Grid (V2G), V2H	25

	● Renewable energy-based EV charging ● Autonomous and connected EVs ● EVs in smart cities ● Policy, safety, and regulatory aspects	
	Total Hours	**105**
Sr.No.	**Capstone Project Topics**	
1	**Powertrain & Energy** ● Design of an 800V EV powertrainqq ● Comparative study of solid-state vs Li-ion batteries ● In-wheel motor-based EV architecture	
2	**Charging & Grid** ● Wireless EV charging system design ● V2G-enabled residential EV system ● Renewable-powered ultra-fast EV charging station	
3	**Intelligence & Control** ● AI-based battery SOC/SOH estimation ● Smart thermal management system for EV batteries ● Predictive energy management for EVs	
4	**Sustainability & Future Mobility** ● Life-cycle analysis of next-generation EVs ● Circular economy model for EV batteries ● EVs in autonomous and smart city frameworks	
Sr.NO.	**PROJECT POLICIES, RULES & REGULATIONS**	
1	**Purpose of the Project** The Honours Capstone Project is introduced to: ● Ensure **experiential and project-based learning** ● Promote **student involvement beyond classroom learning** ● Strengthen **industry relevance and research orientation** ● Achieve **full coverage of syllabus outcomes** through applied work	
2	**Eligibility Criteria** ● Only Honours **Program students** are eligible ● Students must have **passed all prerequisite courses** ● Enrollment permitted at the **beginning of the semester**	

3	**Group Formation Policy**
	● Capstone project may be:
	    ● **Individual** OR
	    ● **Group of maximum 2 students**
	● Interdisciplinary groups are **permitted and encouraged**
	● Each group shall be assigned **one faculty mentor**	

4	**Project Topic Selection Rules**
	● Topics must be **mapped to syllabus units**
	● Project shall cover **minimum two major units** from the syllabus
	● Topics may be:
	    ● Faculty proposed
	    ● Student proposed (with justification)
	● Innovation, sustainability, and next-generation relevance are mandatory	

<table>
<tr><td>5</td><td>Student Involvement & Engagement Policy

Students must demonstrate active involvement through:
Literature survey and problem identification
System design / modeling
Analytical or simulation-based study
Sustainability, energy, or performance evaluation
Report writing and presentation</td></tr>
<tr><td>6</td><td>Mentoring & Review Mechanism

Faculty Mentorship
Weekly mentoring sessions (minimum 4 hrs/week)
Progress tracked through:
Logbook
Review records
Interim submissions</td></tr>
<tr><td>7</td><td>Submission & Documentation Rules

Mandatory submissions include:
Approved project proposal
Literature review summary
Interim review reports
Final project report
Presentation slides</td></tr>
<tr><td>8</td><td>Industry / Research / Startup Linkage (Optional)

Students are encouraged to:
Undertake industry-defined problems
Use real-world datasets
Align projects with:
Startups
Research labs
Government EV initiatives</td></tr>
</table>

Text Books

1. Thomas D. Gillespie, “Fundamentals of Vehicle Dynamics”, SAE International,1999
2. Rajesh Rajamani,“ Vehicle Dynamics and Control,”Springer,2012.
3. Reza N. Jazar, “Vehicle Dynamics: Theory and Application”, Springer,2008.

Reference Books

1. Wong, J. Y.,“ Theory of Ground Vehicles”,Wiley,2008.
2. Milliken & Milliken, “Race Car Vehicle Dynamics”,SAE,1999.
3. Genta & Morello,“ The Automotive Chassis”, Springer,2008.
4. Bosh,“Automotive Handbook” Friedrichshafen and Karlsruhe,2022

Online References

1. NPTEL Course: Prof.R.Krishnakumar, IIT Madras,“ Vehicle Dynamics”(https://nptel.ac.in/courses/107106080).
2. NPTEL Course: Prof. Amit Jain ,IIT Delhi,“Electric Vehicles Part-I”(https://nptel.ac.in/courses/noc22_ee53)
3. NPTEL Course: Prof. Amit Jain, Prof. Avanish Tripathi,IIT Delhi,“ EV - Vehicle Dynamics and Electric Motor Drives”(https://nptel.ac.in/courses/noc26_ee27)

Marathwada Mitra Mandal's College of Engineering

Karvenagar, Pune- 52

An Autonomous Institute Affiliated to SPPU

"येथे बहुतांचे हित"

Curriculum Structure and Syllabus

Honours

A.Y. 2026-27

Accredited with A++ grade by NAAC
Recipient of Best College award by SPPU | Accredited by NBA
Recognized under 2(f) and 12(B) of UGC Act 1956
www.mmcoe.edu.in

About Marathwada Mitra Mandal’s College of Engineering

MMCOE has been recognised as the BEST COLLEGE by SPPU. MMCOE is one feather in the cap of Marathwada Mitra Mandal’s Pune, from the academic year 2006 – 2007.

The Institute offers six undergraduate engineering programmes in Electrical Engineering, Mechanical Engineering, Electronics & Telecommunication Engineering, Information Technology, Computer Engineering, and Artificial Intelligence & Data Science. In addition, it offers six postgraduate programmes M.Tech (Electrical – Power Electronics & Drives), M.Tech (Mechanical – Design Engineering), M.Tech (Electronics & Telecommunication – VLSI & Embedded Systems), M.Tech (Information Technology – Data Science), M.Tech (Computer Engineering), along with a Ph.D. programme in Computer Engineering. It houses 2000+ students and 200 staff members. Along with the focus on core academics, college also takes care of overall personality development of the students.

The college has an active robotics cell, NCC, Training and Placement cell etc. The students are motivated to actively participate in co-curricular and extra-curricular activities. They have brought many laurels to the Institute in the events such as Vinodottam Karandak, Firodiya Karandak, Zest, Dexterity etc.

Institute Vision

To be a globally renowned institution through excellence in engineering education for sustainable and holistic development

Institute Mission

M1: Empower students with cutting-edge technologies and global competencies

M2: Foster culture of research and entrepreneurial mindset

M3: Imbibe social and professional values

M4: Provide an inclusive environment for lifelong learning

Department of Mechanical Engineering

Vision and Mission of the Department

• **Vision**

To develop globally competent and socially committed mechanical engineers

• **Mission**

- ❖ Develop competent professionals through center of excellence
- ❖ Strengthen industry-academia collaboration for ethically responding real-world problems
- ❖ Provide an inclusive environment for entrepreneurship, research and innovation.
- ❖ Empower students for higher education and lifelong learning.

- **Program Specific Outcomes (PSOs)**

Graduates from our program will:

- ❖ PS0-I: Apply advanced technologies of mechanical and allied engineering to address the challenges faced in Industry 4.0.
- ❖ PS0-II: Contribute to the project in the core and associated domain by using add-on skills like CAD, CAM and CAE.
- ❖ PS0-III: Develop the ability to innovate socially relevant products and systems by effectively utilizing resources.

- **Program Outcomes (POs)**

PO1: Engineering Knowledge: Apply knowledge of mathematics, natural science, computing, engineering fundamentals and an engineering specialization as specified in WK1 to WK4 respectively to develop to the solution of complex engineering problems.

PO2: Problem Analysis: Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development. (WK1 to WK4)

PO3: Design/Development of Solutions: Design creative solutions for complex engineering problems and design/develop systems/components/processes to meet identified needs with consideration for the public health and safety, whole-life cost, net zero carbon, culture, society and environment as required. (WK5)

PO4: Conduct Investigations of Complex Problems: Conduct investigations of complex engineering problems using research-based knowledge including design of experiments, modelling, analysis & interpretation of data to provide valid conclusions. (WK8).

PO5: Engineering Tool Usage: Create, select and apply appropriate techniques, resources and modern engineering & IT tools, including prediction and modelling recognizing their limitations to solve complex engineering problems. (WK2 and WK6)

PO6: The Engineer and The World: Analyze and evaluate societal and environmental aspects while solving complex engineering problems for its impact on sustainability with reference to economy, health, safety, legal framework, culture and environment. (WK1, WK5, and WK7).

PO7: Ethics: Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national & international laws. (WK9)

PO8: Individual and Collaborative Team work: Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams.

PO9: Communication: Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations considering cultural, language, and learning differences

PO10: Project Management and Finance: Apply knowledge and understanding of engineering management principles and economic decision-making and apply these to one's own work, as a member and leader in a team, and to manage projects and in multidisciplinary environments.

PO11: Life-Long Learning: Recognize the need for, and have the preparation and ability for i) independent and life-long learning ii) adaptability to new and emerging technologies and iii) critical thinking in the broadest context of technological change. (WK8)

Curriculum for B. Tech. in Mechanical Engineering with Honours in Automotive Technology

Department of Mechanical Engineering

General Objectives and Scope:

The Honour Degree: Automotive Technology is designed to provide students with comprehensive knowledge of modern automotive systems and technologies.

Objectives

- To develop fundamental understanding of automotive systems such as engine, transmission, chassis, suspension, braking and steering systems.
- To provide knowledge of vehicle dynamics and performance parameters influencing stability, handling and ride comfort.
- To familiarize students with automotive electronics, sensors, actuators and electronic control systems used in modern vehicles.
- To provide hands-on practical experience through laboratory experiments related to automotive systems and diagnostics.
- To introduce students to advanced automotive technologies such as electric vehicles, hybrid vehicles, autonomous systems and intelligent vehicle technologies.
- To develop analytical and problem-solving skills through capstone projects involving data analysis, predictive analytics and real-world automotive applications.
- To encourage students to apply interdisciplinary knowledge of mechanical engineering, electronics and data analytics for solving automotive engineering problems.

Scope

- The scope of these subjects includes:
- Study of automotive components and subsystems
- Study of vehicle dynamics, stability, handling and braking performance
- Understanding of automotive electronic systems and control units (ECUs)
- Testing, diagnosis and performance evaluation of vehicles through laboratory experiments
- Exploration of emerging technologies such as electric mobility, hybrid powertrains, autonomous vehicles and connected vehicles
- Application of data analytics and predictive techniques in automotive engineering problems such as maintenance prediction, fault diagnosis and performance optimization.

Why Automotive Technology?

The automotive sector is one of the most significant industries worldwide and is rapidly evolving due to technological advancements and sustainability requirements. The study of automotive technology is important for the following reasons:

1. Rapid technological advancements: Modern vehicles incorporate advanced systems such as electronic control units, sensors, electric powertrains, and autonomous driving technologies.
2. Growing demand for electric and hybrid vehicles: The global shift toward electric mobility and sustainable transportation requires engineers with knowledge of advanced automotive technologies.
3. Integration of electronics and software in vehicles: Modern automobiles rely heavily on automotive electronics, embedded systems, and data communication networks.
4. Industry demand for skilled engineers: Automotive companies require engineers who can design, analyse, test and maintain complex automotive systems.
5. Improvement in vehicle safety and efficiency: Knowledge of automotive technology helps in developing safer, fuel-efficient and environmentally friendly vehicles.
6. Application of data analytics and intelligent systems: Advanced vehicles generate large amounts of data, and predictive analytics helps improve vehicle performance, maintenance planning, and reliability.

Thus, this Honour Degree collectively prepare students to meet the technical and industry requirements of modern automotive engineering and emerging mobility technologies.

Prerequisites :

- Engineering Mathematics -I & II, Engineering Graphics, Thermofluidic, Applied Thermofluidic, Kinematics of Machines
- Basics of Electrical Engineering, Basics of Electronics Technology, Automotive Engineering
- Engineering Mathematics -I & II, Engineering Graphics, Kinematics of Machines, Design of Machine Elements
- Engineering Mathematics -I & II, Engineering Graphics, Kinematics of Machines, Automotive Technology, Design of Machine Elements, Vehicle Dynamics, Automotive Electronics
- Engineering Mathematics -I & II, Engineering Graphics, Kinematics of Machines, Automotive Technology, Design of Machine Elements, Vehicle Dynamics, Automotive Electronics

B. TECH honors in AUTOMOTIVE TECHNOLOGY (Semester IV, V, VI, VII, VIII)

Curriculum Structure

Course Code	Course Name	Teaching Scheme (Hrs/week)				Examination Scheme						Credits				
		L	P	T	ODL	CIE	ETE	TW	PR	OR	Total	L	P	T	ODL	Total
Semester IV																
ME24HON291R	Automotive Engineering	3	-	-	-	40	60	-	-	-	100	3	-	-	-	3
ME24HON292R	Automotive Engineering Lab	-	-	2	-	-	-	50	-	-	50	-	-	2	-	2
Semester V																
ME24HON341R	Automotive Electronics	3	-	-	-	40	60	-	-	-	100	3	-	-	-	3
ME24HON342R	Automotive Electronics Lab	-	-	2	-	-	-	50	-	-	50	-	-	2	-	2
Semester VI																
ME24HON391R	Vehicle Dynamics	3	-	-	-	40	60	-	-	-	100	3	-	-	-	3
Semester VII																
ME24HON441R	Advanced Automotive Technologies	3	-	-	-	40	60	-	-	-	100	3	-	-	-	3
Semester VIII																
ME24HON491R	Capstone predictive analytics project	-	8	-	-	-	-	100	-	-	100	-	4	-	-	4
Total		12	8	4	-	160	240	200	-	-	600	12	4	4	-	20

L- Lecture
P- Practical
T- Tutorial
CIE-Continuous Internal Evaluation
ETE- End Term Examination
TW- Term work
PR- Practical
OR- Oral
L : 1 Hr.= 1 credit
P: 2 Hr. = 1 Credit
T: 1 Hr. = 1 Credit
ODL-Open Distance Learning

Note: For the semester VI, VII and VIII the ETE will be conducted by course coordinator in concern with BoS chairman.

Second Year B. Tech. Mechanical Engineering																
Semester- IV																
Course Code: ME24HON291R										Course Name: Automotive Engineering						
Teaching Scheme (Hours/Week)					Examination Scheme						Credits					
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	OL	ODL	TOTAL
3	-		-	-	40	60	-	-	-	100	3	-	-	-	-	3

Course Objectives:

- Identify vehicle structures, automotive subsystems, and their layout in automobiles.
- Explain the construction and working of automotive transmission systems including clutch, gearbox, and differential.
- Describe the operating principles of steering and suspension systems with respect to vehicle control and ride comfort.
- Explain basic braking principles and load transfer concepts to understand braking performance.

Course Outcomes:

After learning the course, the students will be able to:

CO1: Explain vehicle structures, chassis and body construction, and forces acting on automobiles.
CO2: Explain the working and types of automotive transmission systems.
CO3: Use steering and suspension principles to explain basic vehicle handling and ride comfort behavior.
CO4: Explain braking performance and fundamental braking principles.

Unit	Contents	Duration (Hrs.)
1	**Vehicle Structure** Automotive components, subsystems and their positions - chassis, frame and body - front, rear and four-wheel drives - operation and performance- forces on vehicles, traction force and tractive resistance-power required for automobiles - rolling, air and gradient resistance.	8
2	**Transmission System** Clutch: Types- diaphragm type clutch, single and multi-plate clutches — Gearbox: Types-constant mesh, sliding mesh and synchromesh gearbox, layout of gearbox, gear selector and shifting mechanism, overdrive, hydraulic coupling, automatic transmission, propeller shaft, universal joint. slip joint differential and rear axle arrangement.	8
3	**Steering System** Front axle — types and construction, steering system types, Ackermann principle, Davis, steering, wheel geometry, steering gear, steering gearboxes, steering linkages, power, caster, camber, toe-in, toe-out, wheel alignment and balancing.	8
4	**Suspension** Types - front and rear suspension, conventional and independent type suspension, leaf springs, coil springs, dampers, torsion bars, stabilizer bars, arms, air suspension systems, active suspension systems, wheels and tyres. Braking System, Load transfer, brake force distribution, stopping distance, types of brakes - disc & drum brakes, actuation - mechanical, hydraulic, air, engine brakes, anti-lock braking system (ABS), electronic brake force distribution (EBD), traction control system (TCS), electronic	7
5	**Ergonomics, Comfort and Safety** Ergonomics: Regulations and requirements, passenger and driver's cabin, dashboard equipment arrangement, positioning of operational controls, human factors, pedal positioning. Comfort: Regulations and requirements – ride and vehicle handling, HVAC, seating and upholstery. Safety: Active and passive safety, concept of crumple zone, safety sandwich construction, passenger and occupant safety – testing.	8

Total Hours	39

Text Books

1. Dr. Kirpal Singh, *Automobile Engineering*, Vol. I & II, Standard Publishers Distributors, New Delhi, 2021.
2. V. A. W. Hillier and David R. Rogers, *Hillier's Fundamentals of Motor Vehicle Technology*, Book 1, 6th Edition, Nelson Thornes Ltd., UK, 2011.
3. William H. Crouse and Donald L. Anglin, *Automotive Mechanics*, 10th Edition, McGraw Hill Education, 2017.

Reference Books

1. Robert Bosch GmbH, *Automotive Handbook*, 11th Edition, John Wiley and Sons, England, 2022.
2. G. B. S. Narang, *Automobile Engineering*, Khanna Publishers, New Delhi, 2012.
3. Heinz Heisler, *Advanced Vehicle Technology*, 2nd Edition, Butterworth-Heinemann, Elsevier, 2002.

Online References

1. NPTEL Course: Prof. C. S. Shankar Ram, IIT Madras, "Fundamentals of Automotive Systems" (https://nptel.ac.in/courses/107106088).
2. NPTEL Course: Prof. S. S. Soman, IIT Bombay, "Vehicle Dynamics" (https://nptel.ac.in/courses/112101096).
3. NPTEL Course: Prof. S. K. Saha, IIT Delhi, "Introduction to Robotics and Mechanical Systems" (https://nptel.ac.in/courses/112104298).

Second Year B. Tech. Mechanical Engineering

Semester- IV

Course Code: ME24HON292R | **Course Name: Automotive Engineering Lab**

Teaching Scheme (Hours/Week)					Examination Scheme						Credits					
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	OL	ODL	TOTAL
-	-	2	-	-	-	-	50	-	-	50	-	-	2	-	-	2

Course Objectives:

- Demonstrate the physical assembly, disassembly, and preventive maintenance steps for key automotive mechanisms.
- Explain the fundamental principles behind hydraulic/pneumatic layouts, and working mechanisms of vehicles.
- Illustrate the integration of ergonomic concepts and alternative materials within automotive component design.
- Apply standard methods for the interpretation and preparation of automobile production drawings.

Course Outcomes:

After learning the course, the students will be able to:

CO1: Apply physical assembly, disassembly, and preventive maintenance procedures to IC engines and gearboxes.

CO2: Design hydraulic/pneumatic systems, and working models for automotive applications.

CO3: Select appropriate alternative materials and ergonomic parameters for automotive components.

CO4: Draft detailed production drawings of automobile parts per BIS standards.

Expt No.	Contents	Duration (Hrs.)
1	Assembly and disassembly of gear box and IC engine.	4
2	Development and demonstration of working/animation models of any automotive mechanism.	2
3	Design a circuit of hydraulic/pneumatic system of 4 wheelers and its verification.	4
4	Preventive maintenance of automotives.	4
5	Use of ergonomic principles for the design of hand tools, control in automobile dashboards, etc.	2
6	Use of alternative materials in the construction of automobile components.	2
7	Interpretation of Drawings; Exercises in identifying the type of production, extracting important functional dimensions, checking the number of parts in an assembly of various automobile components.	4
8	Exercises in preparation of detailed production drawings as per BIS standard of simple automobile parts.	4
	Total Hours	**26**

Text Books

1. Dr. Kirpal Singh, Automobile Engineering, Vol. I- II, Standard Publishers, 2021
2. N. D. Bhatt and V. M. Panchal, Machine Drawing, 51st Edition, Charator Publishers, 2022
3. V. D. Bhise, Ergonomics in the Automotive Design Process, 2nd Edition, CRC Press, UK, 2024.

Reference Books

1. R. B. Gupta, Automobile Engineering, Satya Prakashan, New Delhi, 2016.

2. K. L. Narayana, P. Kannaiah, K. Venkata Reddy, Machine Drawing, 15th Edition, New Age International Publication, 2009.
3. Robert Bosch Gmbh, Automotive Handbook, 11th Edition, John Wiley and Sons, England, 2022.

Online References

1. NPTEL Course: Prof. C. S. Shankar Ram, IIT Madras, "Fundamentals of Automotive Systems" (https://nptel.ac.in/courses/107106088).
2. NPTEL Course: Prof. Somashekhar S, IIT Madras, "Oil Hydraulics and Pneumatics" (https://nptel.ac.in/courses/112106300)
3. NPTEL Course: Prof. P.S. Robi, IIT Guwahati, "Introduction to Engineering Drawing" (https://nptel.ac.in/courses/112103019)

Third Year B. Tech. Mechanical Engineering																
Semester-V																
Course Code: ME24HON341R											**Course Name: Automotive Electronics**					
Teaching Scheme (Hours/Week)					**Examination Scheme**						**Credits**					
L	**P**	**T**	**OL**	**ODL**	**CIE**	**ETE**	**TW**	**OR**	**PR**	**TOTAL**	**L**	**P**	**T**	**OL**	**ODL**	**TOTAL**
3	-		-	-	40	60	-	-	-	100	3	-	-	-	-	3

Course Objectives:

- Introduce the fundamentals, evolution, classification, and environmental significance of electric and hybrid vehicles.
- Explain automotive electrical systems including power generation, storage, distribution, communication networks, and electromagnetic compatibility.
- Describe energy storage systems used in electric vehicles with respect to battery types, characteristics, performance, and safety.
- Explain the working principles of starting, charging, and ignition systems used in modern automobiles.
- Explain the basic concepts of automotive sensors and actuators used for engine management and chassis control systems.

Course Outcomes:

After learning the course, the students will be able to:

CO1: Explain the concepts, classification, advantages, challenges, and environmental impact of electric and hybrid vehicles.

CO2: Explain automotive electrical and electronic systems, vehicle communication networks, and EMC requirements.

CO3: Describe various energy storage systems used in electric vehicles and their performance, testing, and recycling aspects.

CO4: Explain the working of starting, charging, and ignition systems used in automobiles.

CO5: Explain the working principles of automotive sensors and actuators for engine and chassis control applications.

Unit	Contents	Duration (Hrs.)
1	**Introduction to Electric and Hybrid Vehicle** History and evolution of Electric Vehicles, Comparison of Electric with Internal Combustion, Engine Vehicles, Limitations of IC Engine Vehicles (ICEV), Exhaust Emission and Global warming, Environmental importance of Hybrid and Electric Vehicles, Overview of EV Challenges, Classification, Overview of EV Technologies, Advantages and Disadvantages, Economic and Environmental impacts of using Electrical Vehicles, Emerging Technologies for Electric Vehicle Drives, Case Studies of Two-Wheeler, Three-Wheeler, and Four-Wheeler Electric Vehicles, Brief introduction to Autonomous and self-driving Vehicles.	8
2	**Automotive Electrical Systems** Introduction to Automotive electricity generation, storage & distribution systems, wiring harness, 12/24/42 volt system, Connectors and its types, positive earth and negative earth, earth return and insulated return systems, Introduction of Controlled Area Networks (CAN) and LIN Communication. Instrument Cluster, types of indication in the cluster, Driver information systems. Electromagnetic Interference (EMI), Electromagnetic Susceptibility (EMS), Electromagnetic Compatibility (EMC)- Need, Types, Methods of Coupling, Sources of EMI, Testing Methods, Related ISO / CISPR / SAE / AIS Standards and Few Case studies.	8
3	**Energy Storage Systems** Types of batteries- Lead acid battery, Nickel-based batteries -Nickel Manganese Cobalt (NMC), Nickel Cobalt Aluminum (NCA), sodium-based batteries, and Lithium-based batteries – Li-ion & Li-poly, Lithium-Cobalt oxide (LCO), Li-ion Manganese Oxide	8

	(LMO), Lithium iron Phosphate (LFP), metal-air battery, zinc chloride battery, Ultracapacitors, Battery-characteristics & parameters, Battery ratings, Battery Performance, Battery capacities, Battery efficiency, Battery tests, Battery failures, Recycling of batteries.	
4	**Starting, Charging and Ignition System** Starting system - requirements, principle and construction of starter motor, types of starters, starter motor drives, switches, starter motor characteristics, and design considerations. Charging system - construction and working of alternator, rectification, types of voltage regulators,Cutout relay, alternator characteristics, and design considerations. Ignition System- Battery ignition system, components details and working, Electronic and distributorless ignition systems, coil-on-plug ignition systems, Spark plugs, types, construction & characteristics.	7
5	**Automotive Sensors and Actuators** Engine sensors: Manifold Absolute Pressure sensor, Knock sensor, Coolant and Intake Air temperature Sensor, Exhaust Oxygen level sensor, Throttle position sensor, Accelerator pedal position sensor and Crankshaft position sensor, Mass Air flow sensor. Chassis Control :- Steering wheel angle sensor, Vibration and acceleration sensors, Speed and RPM sensors, torque sensors, Load Cell, Proximity Sensor, Rain Sensor, Crash Sensor for actuation of SRS Airbag. Actuators- Solenoids, stepper motors and relays, piezo actuators.	8
	Total Hours	39

Text Books

1. Iqbal Husain, *Electric and Hybrid Vehicles: Design Fundamentals*, 2nd Edition, CRC Press, Taylor & Francis Group, USA, 2011.
2. James Larminie and John Lowry, *Electric Vehicle Technology Explained*, 2nd Edition, John Wiley and Sons Ltd., UK, 2012.
3. Mehrdad Ehsani, Yimin Gao and Ali Emadi, *Modern Electric, Hybrid Electric, and Fuel Cell Vehicles: Fundamentals, Theory and Design*, 3rd Edition, CRC Press, 2018.

Reference Books

1. Robert Bosch GmbH, *Automotive Handbook*, 11th Edition, John Wiley and Sons, England, 2022.
2. Ron Hodkinson and John Fenton, *Lightweight Electric/Hybrid Vehicle Design*, Butterworth-Heinemann, Elsevier, 2001.
3. Sandeep Dhameja, *Electric Vehicle Battery Systems*, 2nd Edition, Newnes, Elsevier, 2019.

Online References

1. NPTEL Course: Prof. S. Williamson, IIT Madras, "Electric Vehicles – Part 1" (https://nptel.ac.in/courses/108106170).
2. NPTEL Course: Prof. S. Williamson, IIT Madras, "Electric Vehicles – Part 2" (https://nptel.ac.in/courses/108106171).
3. NPTEL Course: Prof. Arindam Basu, IIT Kharagpur, "Hybrid Electric Vehicles" (https://nptel.ac.in/courses/108105154).

Third Year B. Tech. Mechanical Engineering

Semester-V

Course Code: ME24HON342R **Course Name: Automotive Electronics Lab**

Teaching Scheme (Hours/Week)					Examination Scheme						Credits					
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	OL	ODL	TOTAL
-	-	2	-	-	-	-	50	-	-	50	-	-	2	-	-	2

Course Objectives:

- To identify and troubleshoot traditional and high-voltage automotive wiring architectures.
- To understand the logic behind electronic control systems like BMS and Regenerative Braking.
- To design robust wiring harnesses for specialized vehicles (ATV/Formula).

Course Outcomes:

After learning the course, the students will be able to:

CO-1: Explain the architectural differences and safety protocols between standard 12V car wiring, high-voltage EV systems, and rugged racing ATV harnesses.

CO-2: Describe the functional logic of power management systems, specifically how the BMS protects battery cells and how regenerative braking recovers energy.

CO-3: Discuss the role of the Vehicle Control Unit (VCU) in managing electronic signals and the impact of emerging trends in smart vehicle electronics.

Expt No.	Contents	Duration (Hrs.)
1	Study of Car electrical wiring system	4
2	Study Electric vehicle wiring system	4
3	Study of complete working of vehicle	2
4	Study EV Regenerative Braking Control	2
5	Study of Battery Management System	2
6	Study of wiring harness for ATV/Formula vehicle	4
7	Seminar Presentation on any topic related to "Advanced Automobile Electronics"	4
8	Industrial Visit	4
	Total Hours	**26**

Text Books

1. Robert Bosch GmbH, *Automotive Handbook*, 11th Edition, John Wiley and Sons, England, 2022.
2. Tom Denton, *Automobile Electrical and Electronic Systems*, 5th Edition, Routledge, Taylor & Francis Group, UK, 2017.
3. Iqbal Husain, *Electric and Hybrid Vehicles: Design Fundamentals*, 2nd Edition, CRC Press, Taylor & Francis Group, USA, 2011.

Reference Books

1. James Larminie and John Lowry, *Electric Vehicle Technology Explained*, 2nd Edition, John Wiley and Sons Ltd., UK, 2012.
2. Mehrdad Ehsani, Yimin Gao and Ali Emadi, *Modern Electric, Hybrid Electric, and Fuel Cell Vehicles: Fundamentals, Theory and Design*, 3rd Edition, CRC Press, 2018.
3. Sandeep Dhameja, *Electric Vehicle Battery Systems*, 2nd Edition, Newnes, Elsevier, 2019.

Online References

1. NPTEL Course: Prof. S. Williamson, IIT Madras, "Electric Vehicles – Part 1" (https://nptel.ac.in/courses/108106170).
2. NPTEL Course: Prof. S. Williamson, IIT Madras, "Electric Vehicles – Part 2" (https://nptel.ac.in/courses/108106171).
3. NPTEL Course: Prof. S. K. Saha, IIT Delhi, "Introduction to Mechatronics" (https://nptel.ac.in/courses/112104298).

Third Year B. Tech. Mechanical Engineering																
Semester- VI																
Course Code: ME24HON391R								Course Name: Vehicle Dynamics								
Teaching Scheme (Hours/Week)					Examination Scheme						Credits					
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	OL	ODL	TOTAL
3	-		-	-	40	60	-	-	-	100	3	-	-	-	-	3

Course Objectives:

- To describe fundamental concepts, coordinate systems, and motion variables used in vehicle dynamics.
- To explain tyre construction, tyre mechanics, and tyre–road interaction affecting vehicle performance.
- To interpret suspension system geometry and its influence on ride and handling characteristics.
- To explain braking dynamics and brake force distribution under different driving conditions.
- To summarize steering systems and vehicle handling characteristics under steady-state and transient conditions.

Course Outcomes:

After learning the course, the students will be able to:

CO1: Describe vehicle motion, coordinate systems, force systems, and dynamic axle load transfer.
CO2: Explain tyre mechanics and slip characteristics affecting traction and cornering.
CO3: Interpret suspension system dynamics and geometry influencing ride and handling.
CO4: Explain braking dynamics, stopping distance, and brake force distribution.
CO5: Summarize steering system models and vehicle handling behaviour under steady-state and transient conditions.

Unit	Contents	Duration (Hrs.)
1	**Introduction to vehicle dynamics** lumped mass, vehicle fixed coordinate system, motion variables, earth fixed coordinate system, Euler angles, force system acting on a rigid vehicle, Dynamics axle loads: static loads on level ground, low speed acceleration, loads on grades, rigid body translation and rotational dynamics.	8
2	**Tyre Mechanics** Tire Construction, Size and Load Rating, Terminology and Axis System, Tractive and Cornering Properties, Camber Thrust, Aligning Moment, Combined Braking and Cornering, Conicity and Ply Steer, Slip, Skid, Rolling Resistance, Elastic Band Model for longitudinal slip, Simple model for lateral slip, Combined longitudinal/lateral slip (friction ellipse), Taut string model for lateral slip, Magic Tire Formula	9
3	**Suspension** Degree of freedom and motion path, instant centre, solid axles, anti squat and anti pitch geometry, anti dive suspension geometry, roll centre geometry, active suspension, castor theory.	8
4	**Braking** Constant Deceleration, Stopping distance and time, Braking Sources, Brake Factor, Braking Efficiency, Ideal brake force distribution: Braking Applied To Rear Wheels, Front Wheels And All Four Wheels, On Straight And Curved Path.	7
5	**The Steering System:** The Steering Linkages, Steering System Forces and Moments, Steering System Models, Steering Geometry, Steady Handling (2 DOF steadystate model), Understeer and Oversteer, Effect of Tire Camber and Vehicle Roll (3 DOF steady-state model),	8

	Transient Handling and Directional Stability (2 DOF unsteady model), Effect of Vehicle Roll on Transient Handling (3 DOF unsteady model), Steady-State and Transient Handling of Articulated Vehicles.	
	Total Hours	40

Text Books

1. Rajesh Rajamani, *Vehicle Dynamics and Control*, 2nd Edition, Springer, USA, 2012.
2. Thomas D. Gillespie, *Fundamentals of Vehicle Dynamics*, SAE International, Warrendale, USA, 1992.
3. Hans B. Pacejka, *Tire and Vehicle Dynamics*, 3rd Edition, Butterworth-Heinemann, Elsevier, 2012

Reference Books

1. Reza N. Jazar, *Vehicle Dynamics: Theory and Application*, 2nd Edition, Springer, 2017.
2. J. Y. Wong, *Theory of Ground Vehicles*, 4th Edition, John Wiley and Sons, USA, 2008.
3. Giancarlo Genta and Lorenzo Morello, *The Automotive Chassis: Engineering Principles*, Springer, 2009.

Online References

1. NPTEL Course: Prof. S. S. Soman, IIT Bombay, "Vehicle Dynamics" (https://nptel.ac.in/courses/112101096).
2. NPTEL Course: Dr. R. Krishnakumar, IIT Madras, "Introduction to Vehicle Dynamics (https://nptel.ac.in/courses/107106080).
3. NPTEL Course: Prof. S. K. Saha, IIT Delhi, "Mechanics of Machines" (https://nptel.ac.in/courses/112104121).

Final Year B. Tech. Mechanical Engineering																
Semester-VII																
Course Code: ME24HON441R								Course Name: Advanced Automotive Technologies								
Teaching Scheme (Hours/Week)					Examination Scheme						Credits					
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	OL	ODL	TOTAL
3	-		-	-	40	60	-	-	-	100	3	-	-	-	-	3

Course Objectives:

- To explain modern trends and technological advancements in internal combustion engines with reference to efficiency and emission norms.
- To describe the need for alternative fuels and their role in sustainable automotive applications.
- To explain modern automotive HVAC systems, components, refrigerants, and control strategies used in conventional and electric vehicles.
- To describe advanced driver assistance systems and automotive safety and security technologies.
- To explain the fundamental concepts of autonomous vehicle technologies and their operational challenges.

Course Outcomes:

After learning the course, the students will be able to:

CO1: Explain modern IC engine technologies including downsizing, advanced combustion concepts, emission regulations, and noise norms.

CO2: Describe advancements in alternative fuels such as biofuels, hydrogen, synthetic fuels, and gaseous fuels and their environmental impact.

CO3: Explain modern automotive HVAC systems, low-GWP refrigerants, electronic controls, and energy-efficient HVAC strategies for EVs and HEVs.

CO4: Describe advanced driver assistance systems, vehicle safety features, and automotive security systems used in modern vehicles.

CO5: Explain autonomous vehicle technologies including levels of automation, sensors, perception concepts, and regulatory considerations.

Unit	Contents	Duration (Hrs.)
1	**Modern Trends in IC Engines** Engine downsizing and application of turbocharging and supercharging, Lean Burning and Adiabatic concepts, Rotary Engines, Modification in I.C engines to suit Bio – fuels, HCCI and GDI concepts, BS-VI Emission Norms, Noise Pollution Norms. Hydrogen powered fuel cell Vehicles.	8
2	**Advancements in Alternative Fuels** Need for alternative fuels and limitations of conventional fuels Advanced biofuels: second- and third-generation biofuels, biodiesel, bioethanol blends Hydrogen as automotive fuel: H_2-ICE, fuel cell vehicles, green hydrogen and storage methods Synthetic fuels and e-fuels: carbon-neutral concepts and engine compatibility Gaseous fuels: CNG, LNG, biogas, biomethane, HCNG Alcohol and emerging fuels: methanol, ethanol, ammonia Environmental impact and future role of alternative fuels in sustainable mobility	9
3	**Modern Automotive HVAC** Evolution of automotive HVAC systems, Performance requirements in modern vehicles, HVAC role in comfort, safety, and energy efficiency,	8

	Variable displacement compressors, Electric compressors for EVs, Micro-channel condensers and high-efficiency evaporators, Electronic expansion valves (EEV), Advanced cabin air filtration systems. Low-GWP refrigerants: R1234yf, CO_2 (R744) HVAC systems, Environmental impact and safety aspects, Automotive HVAC regulations and standards Automatic climate control (ACC), Dual-zone and multi-zone HVAC systems, Sensor-based HVAC control, AI-assisted and predictive climate control, Integration with vehicle electronics (CAN, LIN) HVAC challenges in EVs and HEVs, Heat pump-based HVAC systems, Battery thermal management integration, Energy-efficient HVAC strategies	
4	**Advanced Driver Assist Systems** ABS system with layout and working, Driver state monitoring (DSM), Supplementary Restraint System of air bag system, Seat belts- Retractor, Pretensioner and Load Limiter, Adaptive Cruise control, Automatic Emergency Braking, Collision avoidance system. Vehicle security systems alarms, Keyless Entry system, Introduction to Global Positioning Systems, Lane Departure Warning System, Tire Pressure Monitoring System, Smart parking assist system (SPAS), Advance Front Lighting Systems (AFS), Head-up Display (HUD).	7
5	**Autonomous Vehicle Technologies** Evolution and need for autonomous vehicles, Levels of vehicle automation and classification, Sensors used in autonomous vehicles: camera, radar, LiDAR, ultrasonic sensors, Perception systems and sensor fusion fundamentals, Localization, mapping, and path-planning concepts (overview), Vehicle control strategies for autonomous operation, Safety, ethical, and regulatory challenges in autonomous driving	8
	Total Hours	40

Text Books

1. V. Ganesan, *Internal Combustion Engines*, 4th Edition, McGraw Hill Education, New Delhi, 2012.
2. James Larminie and John Lowry, *Electric Vehicle Technology Explained*, 2nd Edition, John Wiley and Sons Ltd., UK, 2012.
3. Tom Denton, *Automobile Electrical and Electronic Systems*, 5th Edition, Routledge, Taylor & Francis Group, UK, 2017.

Reference Books

1. Richard Stone, *Introduction to Internal Combustion Engines*, 4th Edition, Palgrave Macmillan, UK, 2012.
2. Mehrdad Ehsani, Yimin Gao and Ali Emadi, *Modern Electric, Hybrid Electric, and Fuel Cell Vehicles: Fundamentals, Theory and Design*, 3rd Edition, CRC Press, 2018.
3. Rajesh Rajamani, *Vehicle Dynamics and Control*, 2nd Edition, Springer, USA, 2012.
4. Robert Bosch GmbH, *Automotive Handbook*, 11th Edition, John Wiley and Sons, England, 2022.

Online References

1. NPTEL Course: Prof. V. Ganesan, IIT Madras, "Internal Combustion Engines" (https://nptel.ac.in/courses/112106133).
2. NPTEL Course: Prof. S. Williamson, IIT Madras, "Electric Vehicles" (https://nptel.ac.in/courses/108106170).
3. NPTEL Course: Prof. S. S. Soman, IIT Bombay, "Vehicle Dynamics" (https://nptel.ac.in/courses/112101096).

Final Year B. Tech. Mechanical Engineering																
Semester-VIII																
Course Code: ME24HON491R								**Course Name: Capstone predictive analytics project**								
Teaching Scheme (Hours/Week)					**Examination Scheme**						**Credits**					
L	**P**	**T**	**OL**	**ODL**	**CIE**	**ETE**	**TW**	**OR**	**PR**	**TOTAL**	**L**	**P**	**T**	**OL**	**ODL**	**TOTAL**
-	8	-	-	-	-	-	100	-	-	100	-	4	-	-	-	4

Course Objectives:

- Expose students to predictive analytics and data-driven decision making in automotive systems.
- Enable students to analyze vehicle performance, reliability, and operational data using statistical and machine learning techniques.
- Prepare students to develop predictive models for automotive applications such as maintenance, energy management, and safety.
- Encourage innovation, research, and interdisciplinary integration using AI, IoT, and connected vehicle technologies.
- Prepare students for industry, research, startups, and higher studies in automotive data analytics and intelligent mobility.

Course Outcomes:

After learning the course, the students will be able to:

CO1: Identify and formulate automotive problems suitable for predictive analytics.
CO2: Collect, preprocess, and analyze automotive datasets.
CO3: Develop predictive models using statistical and machine learning techniques.
CO4: Apply modern data analytics tools for automotive performance and reliability prediction.
CO5: Propose data-driven solutions for intelligent and sustainable mobility.

Unit	Contents	Duration (Hrs.)
1	**Introduction to Predictive Analytics in Automotive Systems** Overview of data analytics in automotive industry Types of analytics: descriptive, diagnostic, predictive, prescriptive Automotive data sources: sensors, ECUs, telematics, IoT Importance of predictive analytics in mobility and transportation Data-driven decision making in automotive engineering	10
2	**Data Collection and Preprocessing** Data acquisition from vehicle sensors and telematics systems Data cleaning and preprocessing techniques Handling missing data and noise filtering Feature extraction and selection Visualization and exploratory data analysis	15
3	**Machine Learning Techniques for Automotive Prediction** Regression models for vehicle performance prediction Classification techniques for fault detection Time-series analysis for predictive maintenance Introduction to deep learning concepts Model training, validation, and evaluation	25

<table>
<tr><td>4</td><td>Applications of Predictive Analytics in Automotive Systems
Predictive maintenance of vehicle components
Battery health prediction in EVs
Energy consumption and range prediction
Driver behavior analysis and safety prediction
Traffic and mobility forecasting</td><td>25</td></tr>
<tr><td>5</td><td>Tools, Platforms, and Case Studies
Data analytics tools (Python, MATLAB, R)
Automotive datasets and simulation environments
Case studies from automotive industry
Integration with IoT, connected vehicles, and smart mobility
Ethical, privacy, and regulatory considerations</td><td>25</td></tr>
<tr><td colspan="2" align="right">Total Hours</td><td>100</td></tr>
<tr><td>Sr. No.</td><td>Capstone Project Topics</td><td></td></tr>
<tr><td>1</td><td colspan="2">Data Analytics for Vehicle Performance
Vehicle fuel efficiency prediction using machine learning
Predictive analysis of vehicle acceleration and performance
Data-driven vehicle dynamics modeling</td></tr>
<tr><td>2</td><td colspan="2">Predictive Maintenance
Engine fault prediction using sensor data
Predictive maintenance for braking systems
Failure prediction of automotive components</td></tr>
<tr><td>3</td><td colspan="2">Electric Vehicle Analytics
EV battery State of Health (SOH) prediction
Electric vehicle range prediction model
Charging demand forecasting for EV fleets</td></tr>
<tr><td>4</td><td colspan="2">Smart Mobility and Connected Vehicles
Driver behavior prediction using telematics data
Traffic flow prediction using AI models
Vehicle usage analytics for fleet management</td></tr>
<tr><td>Sr. No.</td><td colspan="2">CAPSTON PROJECT POLICIES, RULES & REGULATIONS</td></tr>
<tr><td>1</td><td colspan="2">The Capstone Predictive Analytics Project is introduced to provide students with an opportunity to apply theoretical knowledge to real-world automotive problems using data analytics techniques. The project aims to:
Promote experiential, project-based, and research-oriented learning.
Encourage integration of automotive engineering with predictive analytics, AI, and data science.
Develop students' ability to analyze complex engineering data and derive meaningful insights.
Strengthen innovation, sustainability, and industry relevance in automotive technology.
Facilitate development of problem-solving, teamwork, and technical communication skills.</td></tr>
<tr><td>2</td><td colspan="2">Eligibility Criteria
The Capstone Project is mandatory for students enrolled in the Honours Degree in Automotive Technology.</td></tr>
</table>

	• Students must have successfully completed the prerequisite courses related to automotive systems and data analytics. • Registration for the project shall be completed at the beginning of the semester as per institutional guidelines.
3	• **Group Formation Policy** • The Capstone Project may be undertaken individually or in groups of a maximum of two students. • Interdisciplinary collaboration with students from related domains such as Artificial Intelligence, Data Science, Electronics, or Computer Engineering is encouraged. • Each project group will be assigned a faculty mentor responsible for guiding and monitoring project

4	**Project Topic Selection**
	• Project topics must be relevant to predictive analytics applications in automotive systems.
	• The project should ideally integrate automotive engineering concepts with data analytics or machine learning techniques.
	• Topics may be:
	    ○ Faculty-proposed, or
	    ○ Student-proposed, subject to approval by the department.
	• The project should demonstrate innovation, analytical depth, and practical relevance to modern automotive technologies.

5	**Student Engagement and Work Requirements** Students are expected to actively participate in all stages of the project, including: • Problem identification and literature review • Data collection, preprocessing, and exploratory analysis • Model development using predictive analytics techniques • Implementation using suitable tools and platforms • Performance evaluation and interpretation of results • Documentation, technical reporting, and presentation
6	**Mentoring and Review Mechanism** **Faculty Mentorship** Each project group shall work under the supervision of a faculty mentor, who will provide guidance on: • Project planning and methodology • Technical implementation • Data analysis and interpretation • Documentation and presentation **Progress Monitoring** Project progress will be evaluated through periodic reviews, including: • Project proposal approval • Literature review presentation • Mid-term progress review • Final project evaluation
7	**Submission and Documentation Requirements** Students are required to submit the following documents during the course of the project: • Approved project proposal • Literature review report • Dataset description and methodology

	• Interim progress reports • Final project report • Presentation slides and demonstration materials All documentation should follow institutional formatting and plagiarism guidelines.
8	**Industry / Research / Startup Linkages (Optional)** Students are encouraged to undertake projects in collaboration with: • Automotive industries and OEMs • Mobility and EV startups • Research laboratories and innovation centers • Government or smart mobility initiatives Such collaborations can provide exposure to real-world datasets, practical challenges, and emerging automotive technologies

Text Books

1. Trevor Hastie, Robert Tibshirani, Jerome Friedman, *The Elements of Statistical Learning*, Springer, 2009.
2. Aurélien Géron, *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*, O'Reilly Media, 2019.
3. Rajesh Rajamani, *Vehicle Dynamics and Control*, Springer, 2012.

Reference Books

1. Ian Goodfellow, Yoshua Bengio, Aaron Courville, *Deep Learning*, MIT Press, 2016.
2. Kevin Murphy, *Machine Learning: A Probabilistic Perspective*, MIT Press, 2012.
3. Bosch, *Automotive Handbook*, Robert Bosch GmbH, 2022.

Online References

1. NPTEL Course: *Machine Learning* – IIT Madras / IIT Kharagpur
   https://nptel.ac.in/courses/106106139
2. NPTEL Course: Prof. S. Williamson, IIT Madras, "Electric Vehicles"
   (https://nptel.ac.in/courses/108106170).
3. NPTEL Course: Prof. S. S. Soman, IIT Bombay, "Vehicle Dynamics"
   (https://nptel.ac.in/courses/112101096).
4. Kaggle Automotive and Mobility Datasets
   https://www.kaggle.com/datasets
5. SAE International Technical Papers and Automotive Research Resources
   https://www.sae.org/

Marathwada Mitra Mandal's College of Engineering

Karvenagar, Pune- 52

An Autonomous Institute Affiliated to SPPU

"येथे बहुतांचे हित"

Curriculum Structure and Syllabus

Honours

A.Y. 2026-27

Accredited with A++ grade by NAAC
Recipient of Best College award by SPPU | Accredited by NBA
Recognized under 2(f) and 12(B) of UGC Act 1956
www.mmcoe.edu.in

About Marathwada Mitra Mandal's College of Engineering

MMCOE has been recognised as the BEST COLLEGE by SPPU. MMCOE is one feather in the cap of Marathwada Mitra Mandal's Pune, from the academic year 2006 – 2007.

The Institute offers six undergraduate engineering programmes in Electrical Engineering, Mechanical Engineering, Electronics & Telecommunication Engineering, Information Technology, Computer Engineering, and Artificial Intelligence & Data Science. In addition, it offers six postgraduate programmes M.Tech (Electrical – Power Electronics & Drives), M.Tech (Mechanical – Design Engineering), M.Tech (Electronics & Telecommunication – VLSI & Embedded Systems), M.Tech (Information Technology – Data Science), M.Tech (Computer Engineering), along with a Ph.D. programme in Computer Engineering. It houses 2000+ students and 200 staff members. Along with the focus on core academics, college also takes care of overall personality development of the students.

The college has an active robotics cell, NCC, Training and Placement cell etc. The students are motivated to actively participate in co-curricular and extra-curricular activities. They have brought many laurels to the Institute in the events such as Vinodottam Karandak, Firodiya Karandak, Zest, Dexterity etc.

Institute Vision

To be a globally renowned institution through excellence in engineering education for sustainable and holistic development

Institute Mission

M1: Empower students with cutting-edge technologies and global competencies

M2: Foster culture of research and entrepreneurial mindset

M3: Imbibe social and professional values

M4: Provide an inclusive environment for lifelong learning

Department of
Electronics and Telecommunication Engineering

Vision and Mission of the Department

- **Vision**

To cultivate globally competent Electronics and Telecommunication Engineers for developing sustainable solutions

- **Mission**

M1: To empower students with foundational and emerging technologies in Electronics and Telecommunication Engineering, fostering global competencies
M2: To foster a research and collaborative learning ecosystem that encourages entrepreneurship, holistic and sustainable development
M3: To inculcate ethical values, leadership, adaptability, and a lifelong learnability in a diverse, global engineering.

- **Program Specific Outcomes (PSOs)**

PSO1: To apply principles of Embedded Systems, VLSI, Communication, and Automation using modern hardware and software tools to develop effective E&TC solutions.
PSO2: To design and develop multidisciplinary, innovative, sustainable solutions through project-based learning to address industrial, societal, and environmental needs.

- **Program Outcomes (POs)**

PO1: Engineering Knowledge: Apply knowledge of mathematics, natural science, computing, engineering fundamentals and an engineering specialization as specified in WK1 to WK4 respectively to develop to the solution of complex engineering problems.

PO2: Problem Analysis: Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development. (WK1 to WK4)

PO3: Design/Development of Solutions: Design creative solutions for complex engineering problems and design/develop systems/components/processes to meet identified needs with consideration for the public health and safety, whole-life cost, net zero carbon, culture, society and environment as required. (WK5)

PO4: Conduct Investigations of Complex Problems: Conduct investigations of complex engineering problems using research-based knowledge including design of experiments, modelling, analysis & interpretation of data to provide valid conclusions. (WK8).

PO5: Engineering Tool Usage: Create, select and apply appropriate techniques, resources and modern engineering & IT tools, including prediction and modelling recognizing their limitations to solve complex engineering problems. (WK2 and WK6)

PO6: The Engineer and The World: Analyze and evaluate societal and environmental aspects while solving complex engineering problems for its impact on sustainability with reference to economy, health, safety, legal framework, culture and environment. (WK1, WK5, and WK7).

PO7: Ethics: Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national & international laws. (WK9)

PO8: Individual and Collaborative Team work: Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams.

PO9: Communication: Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations considering cultural, language, and learning differences

PO10: Project Management and Finance: Apply knowledge and understanding of engineering management principles and economic decision-making and apply these to one’s own work, as a member and leader in a team, and to manage projects and in multidisciplinary environments.

PO11: Life-Long Learning: Recognize the need for, and have the preparation and ability for i) independent and life-long learning ii) adaptability to new and emerging technologies and iii) critical thinking in the broadest context of technological change. (WK8)

Curriculum for B. Tech. in Electronics and Telecommunication Engineering with Honours in Automotive Embedded Systems

Department of Electronics and Telecommunication Engineering

General Objectives and Scope:

The Automotive Embedded Systems honours program is designed to provide students with an in-depth understanding of modern computing systems with a particular emphasis on embedded and real-time applications. The program aims to equip students with advanced knowledge of computer organization, system software, embedded architectures and communication protocols used in modern intelligent systems. Through this specialized track, students are exposed to effective coding practices, system-level design, and performance optimization techniques that are essential to developing reliable, high-performance embedded solutions. The course emphasizes developing strong programming skills using C/C++, adopting coding standards, and using modern development tools such as debuggers, build systems, testing frameworks, and simulation environments. Students will also explore techniques for writing optimized code, debugging programs, and ensuring software reliability through unit testing and systematic development practices. In addition, the program provides hands-on exposure to the design of microcontroller-based systems using industry-relevant platforms such as the STM32 microcontroller, allowing students to program peripherals, develop interrupt service routines, and implement multitasking applications using real-time operating systems such as FreeRTOS. The specialized program also introduces students to integrated networking concepts and automotive communications technologies, including CAN, LIN, Automotive Ethernet, and service-oriented communications protocols used in modern vehicles. By completing this specialized program, students develop strong analysis, programming, and systems design skills required for careers in embedded systems, IoT, automotive electronics, and industrial automation. The program also prepares students for advanced research, graduate study, and innovation in the emerging areas of embedded and cyber-physical systems.

Why Automotive Embedded Systems ?

Automotive embedded systems have become a crucial component of modern vehicles because everything is moving toward smarter and more connected vehicles. Vehicles today depend on all this electronics and software to handle stuff like engine control, brakes, safety things, entertainment inside the car, and those driver assistance features. It improves how modern vehicles run, keep people safer, make it more efficient, and just better for the driver overall.

With sensors popping up everywhere and networks connecting parts, these systems have to deal with a lot of complicated tasks quickly and without messing up. Reliability and low delay are super important here. Things like CAN and LIN networks, plus Automotive Ethernet, they all help different control units in the modern vehicles talk to each other smoothly. Service protocols make the data sharing even better. Designing systems that work with these standards seems essential, but it gets tricky figuring out the best architecture for everything.

Furthermore, the rapid advancement of electric vehicles, connected vehicles, and autonomous driving technologies has significantly increased the demand for engineers skilled in automotive embedded systems. These technologies require expertise in microcontroller programming, real-time operating systems, embedded networking, and system-level optimization to ensure safe and efficient operation. As a result, the study of automotive embedded systems provides students with valuable industry-relevant skills and prepares them for careers in automotive electronics, intelligent transportation systems, and next-generation mobility technologies.

Prerequisites :

1. Basic knowledge of C/C++ programming
2. Fundamentals of digital electronics and microcontrollers
3. Understanding of computer organization and architecture
4. Basic concepts of operating systems
5. Fundamentals of data communication and embedded systems

B. TECH honours IN AUTOMOTIVE EMBEDDED SYSTEMS (Semester IV, V, VI, VII, VIII)

Curriculum Structure

Course Code	Course Name	Teaching Scheme (Hrs/week)				Examination Scheme						Credits				
		L	P	T	ODL	CIE	ETE	TW	PR	OR	Total	L	P	T	ODL	Total
Semester IV																
ET24HON291R	Foundations of Computer Systems	3	-	-	-	40	60	-	-	-	100	3	-	-	-	3
ET24HON292R	Foundations of Computer Systems Lab	-	2	-	-	-	-	25	-	-	25	-	1	-	-	1
Semester V																
ET24HON341R	Embedded Programming	3	-	-	-	40	60	-	-	-	100	3	-	-	-	3
ET24HON342R	Embedded Programming Lab	-	2	-	-	-	-	25	-	-	25	-	1	-	-	1
Semester VI																
ET24HON391R	Embedded Linux	3	-	-	-	40	60	-	-	-	100	3	-	-	-	3
ET24HON392R	Embedded Linux Lab	-	2	-	-	-	-	25	-	-	25	-	1	-	-	1
Semester VII																
ET24HON441R	Automotive Communication Networks	3	-	-	-	40	60	-	-	-	100	3	-	-	-	3
ET24HON442R	Automotive Communication Networks Lab	-	2	-	-	-	-	25	-	-	25	-	1	-	-	1
Semester VIII																
ET24HON491R	**Capstone** Project	-	8	-	-	-	-	100	-	-	100	-	4	-	-	4
Total		**12**	**16**	-	-	**160**	**240**	**200**	-	-	**600**	**12**	**8**	-	-	**20**

L- Lecture	P- Practical	T- Tutorial	CIE-Continuous Internal Evaluation
ETE- End Term Examination	TW- Term work	PR- Practical	OR- Oral
L : 1 Hr.= 1 credit	P: 2 Hr. = 1 Credit	T: 1 Hr. = 1 Credit	ODL-Open Distance Learning

Second Year B. Tech Electronics and Telecommunication Engineering															
Semester IV															
Course Code: ET24HON291R						Course Name: Foundations of Computer Systems									
Teaching Scheme (Hours/Week)					Examination Scheme						Credits				
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	ODL	TOTAL
3	-	-	-	-	40	60	-	-	-	100	3	-	-	-	3

Course Objectives:
- To explore tools and techniques of writing efficient and optimized codes.
- To understand computer organization and architecture.
- To describe the principles of computer architecture.
- To understand the fundamentals of system software and the role of the operating system in achieving parallel processing
- To understand the ways of Power and Performance Optimization in a computer system.

Course Outcomes:

After completing the course, the students will be able to:

CO1: Apply programming tools and optimization techniques to develop efficient C/C++ programs.

CO2: Analyze software requirements and apply SDLC and testing principles for quality software development.

CO3: Explain the organization and functioning of CPU, memory, and I/O subsystems in a computer system.

CO4: Compare computer architectures and analyze parallel processing in multi-core systems.

CO5: Analyze operating system functions related to process, memory, and I/O management with performance considerations.

Unit	Contents	Duration (Hrs.)
1	**Programming Fundamentals and Tools** Introduction to GNU Toolchain (GCC, GDB, Makefiles), Concepts of C and C++ programming, coding standards and code analysis, writing efficient and optimized codes, use of GenAI tools for writing, debugging and testing codes.	8
2	**Software Engineering** Introduction, Software Development Life Cycle, Requirement gathering and analysis, Use Case Modeling. Software Testing and Quality Assurance – levels of testing, testing techniques.	8
3	**Introduction to Computer Organization** Overview of computer organization, Brief history of computer development; Hardware Components - CPU: structure, function, and types, Memory - main memory, virtual memory, caching, I/O devices -peripherals, interfaces; Memory Management – memory organization, Virtual memory - concept, advantages, and disadvantages, Caching - concept, types, and cache hierarchies; Input/Output Systems – peripherals and Interfaces.	8
4	**Introduction to Computer Architecture** Overview and history of computer architecture, CISC, RISC, and ARM Architectures, Bus Architecture - Bus types: system bus, I/O bus, Bus protocols - synchronous and asynchronous buses; Parallel Processing and Multi-Core Architecture.	8
5	**System Software** Introduction to Operating System - Operating System Concept, Roles, and Operations, process, thread, and memory management, process scheduling,	8

	memory allocation, and I/O management, booting, login, and system calls, types of operating system.	
	Total Hours	40

Text Books

1. B. W. Kernighan and D. M. Ritchie, "The C Programming Language", Prentice Hall, 2nd Edition.
2. P. Deitel and H. Deitel, "C++ How to Program", Pearson Education, 10th Edition, 2016.
3. R. Stallman, R. Pesch, and S. Shebs, "Debugging with GDB:The GNU Source-Level Debugger", Free Software Foundation, 9th Edition, 2002.
4. R. Bast and R. Di Remigio, "CMake Cookbook: Building, Testing, and Packaging Modular Software with Modern CMake", Packt Publishing Ltd., 1st Edition, 2018.

Reference Books

1. J. P. Hayes, "Computer Architecture and Organisation", Tata McGraw Hill, 2022.
2. S. Harris and D. Harris, "Digital Design and Computer Architecture: RISC-V Edition", Morgan Kaufmann, 2021.
3. H. M. Deitel, P. Deitel, and D. R. Choffnes, "Operating Systems", Prentice Hall, 3rd Edition, 2003.

Online References

1. NPTEL Course: Prof. Indranil Sengupta and Prof. Kamalika Datta, IIT Kharagpur, "Computer Architecture and Organization" [https://nptel.ac.in/courses/106105163/]
2. NPTEL Course: Prof. Anupam Basu, IIT Kharagpur, "Problem Solving Through Programming in C" [https://nptel.ac.in/courses/106105171/]

Second Year B. Tech Electronics and Telecommunication Engineering

Semester IV

Course Code: ET24HON292R **Course Name: Foundations of Computer Systems Lab**

Teaching Scheme (Hours/Week)					Examination Scheme						Credits				
L	P	T	OL	ODL	CIE	ETE	TW	PR	OR	TOTAL	L	P	T	ODL	TOTAL
-	2	-	-	-	-	-	25	-	-	25	-	1	-	-	1

Course Objectives:

- To apply coding guidelines and best practices to write high quality code
- To use GDB to identify and fix errors in C/C++ programs
- To understand the basics of CMake and its applications in building C/C++ projects
- To use the Google Test framework to write and execute unit tests for C/C++ code.
- To utilize Generative AI tools to improve code writing, debugging and testing efficiency.
- To explore simulation tools to model and analyze computer systems and architectures

Course Outcomes:

After completing the course, the students will be able to:

CO1: Apply coding guidelines and Generative AI tools to develop high-quality C/C++ programs.
CO2: Debug and test C/C++ programs using GDB and Google Test frameworks.
CO3: Configure and build cross-platform C/C++ projects using CMake.
CO4: Simulate and analyze CPU execution, memory management, and multi-core processing using system simulation tools and develop a minor project.

Week	Contents	Hrs
1	Writing C/C++ programs considering coding guidelines and code analysis. Use of Generative AI tools for efficient code writing.	4
2	Debugging code with GDB. Use of GenAI tools for efficient code debugging.	4
3	Understanding CMake for cross-platform project configuration and builds.	4
4	Performing unit testing with Google Test framework. Use of GenAI tools for testing.	4
5	Simulate CPU execution, memory management, multi core task execution and bus communication using tools like Multi2Sim etc.	4
6	Development of Minor project in group adhering to the learned	6
	Total Hours	**26**

Text Books

1. B. W. Kernighan and D. M. Ritchie, **"The C Programming Language"**, Prentice Hall, 2nd Edition, 1988.
2. P. Deitel and H. Deitel, **"C++ How to Program"**, Pearson Education, 10th Edition, 2016.

Reference Book

1. R. Stallman, R. Pesch, and S. Shebs, "Debugging with GDB", Free Software Foundation, 1988.
2. R. Bast and R. Di Remigio, "CMake Cookbook: Building, Testing, and Packaging Modular Software with Modern CMake", Packt Publishing Ltd., 1st Edition, 2018.
3. GoogleTest User's Guide, Online Documentation.[https://google.github.io/googletest/]
4. CMake: A Powerful Software Build System, Official Documentation.[https://cmake.org/]
5. Multi2Sim: A Heterogeneous System Simulator, Official Website. [http://www.multi2sim.org/]

Online References

1. NPTEL Course: Prof. Anupam Basu, IIT Kharagpur, "Problem Solving Through Programming in C" [https://nptel.ac.in/courses/106105171/]
2. NPTEL Course: Prof. P. P. Chakrabarti, IIT Kharagpur, "Programming and Data Structures" [https://nptel.ac.in/courses/106105085/]

Third Year B. Tech Electronics and Telecommunication Engineering															
Semester V															
Course Code: ET24HON341R							Course Name: Embedded Programming								
Teaching Scheme (Hours/Week)					Examination Scheme						Credits				
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	ODL	TOTAL
3	-	-	-	-	40	60	-	-	-	100	3	-	-	-	3

Prerequisite:

Course Objectives:

- To understand the fundamentals of embedded systems and interpret technical documents.
- To understand and explain the build and development tools used in embedded systems design.
- To understand the function and program the microcontroller peripherals.
- To write interrupt service routines while using microcontroller peripherals.
- To understand RTOS internals and performance metrics.

Course Outcomes:

After completing the course, the students will be able to:

CO1: Explain embedded system fundamentals using technical documentation.
CO2: Use embedded development toolchains for building and linking applications.
CO3: Program microcontroller peripherals for embedded applications.
CO4: Implement interrupt service routines for microcontroller peripherals.
CO5: Apply RTOS concepts for real-time embedded applications.

Unit	Contents	Duration (Hrs.)
1	**Introduction to Embedded Systems** What are Embedded Systems and characteristics, building blocks of Embedded Systems, various applications of Embedded Systems, Difference between Desktop Vs Embedded application development process, role of microprocessor and microcontroller in Embedded System, Understanding Block Diagram, Schematic Diagram, Datasheet, User manual, Application Notes	8
2	**Embedded Systems Development Tool Chain** Build and development tools, Build Process, role of Linker, understanding object files, Elf Generation / Compilation/ Linker Error Resolution, tools used to extract binary code from elf, converting to different formats like plain binary, S-record, Concept of Make file and Complex Make file, Big Endian vs Little Endian, Memory mapping, understanding Startup code, Reset Code, Bootloader.	8
3	**Microcontroller Peripherals and Programming** Understanding and programming STM32 microcontroller for GPIO, UART, Timer, PWM, ADC, UART, I2C and SPI Protocol.	8
4	**Microcontroller Peripheral Interrupts and Programming** Interrupt Vector Table, Interrupt Priorities, Nested Interrupts, Writing ISR in assembly, and Embedded C, interrupt latency, Write interrupt service routines for various peripherals and operations.	8
5	**Real Time Operating System** Introduction to Real-Time Concepts, RTOS Internals & Real Time Scheduling, Performance Metrics of RTOS, Task Specifications, Schedulability Analysis, Application Programming on RTOS, Configuring and Porting of RTOS.	8

Total Hours	40

Text Books

1. K. V. K. Prasad, "Embedded / Real-Time Systems: Concepts, Design and Programming Black Book", Dreamtech Press, 2003.
2. Jonathan W. Valvano, "Embedded Systems: Introduction to ARM Cortex-M Microcontrollers", Pearson Education, 5th Edition, 2018.
3. Wayne Wolf, "Computers as Components: Principles of Embedded Computing System Design", Morgan Kaufmann, 4th Edition, 2016.

Reference Books

1. STM32F401 Microcontroller Documentation, STMicroelectronics, PDF Documentation. [https://www.st.com/en/microcontrollers-microprocessors/stm32f401/documentation.html]
2. FreeRTOS Documentation, Official Online Documentation.

Online References

1. NPTEL Course: Prof. P. R. Sinha, IIT Delhi, "Embedded Systems" [https://nptel.ac.in/courses/108102045/]
2. NPTEL Course: Prof. Santanu Chattopadhyay, IIT Kharagpur, "Embedded Systems – Design Verification and Test" [https://nptel.ac.in/courses/106105171/]
3. NPTEL Course: Prof. Indranil Banerjee, IIT Kharagpur, "Real-Time Systems" [https://nptel.ac.in/courses/106105215/]

Third Year B. Tech Electronics and Telecommunication Engineering															
Semester V															
Course Code: ET24HON342R						Course Name: Embedded Programming Lab									
Teaching Scheme (Hours/Week)					Examination Scheme						Credits				
L	P	T	OL	ODL	CIE	ETE	TW	PR	OR	TOTAL	L	P	T	ODL	TOTAL
-	2	-	-	-	-	-	25	-	-	25	-	1	-	-	1

Course Objectives:

- To interpret and Analyze Technical Documents.
- To utilize build and development tools and processes to design, develop, and implement projects using the STM32 microcontroller.
- To write efficient C code to program and interface with various STM32 peripherals, including GPIO, UART, Timer, PWM, ADC, I2C, and SPI protocols.
- To develop interrupt service routines to handle peripheral operations and events, ensuring efficient and reliable system performance.
- To design and implement multitasking operations using the FreeRTOS.

Course Outcomes:

After completing the course, the students will be able to:
CO1: Interpret and analyze STM32 technical documents for embedded system design.
CO2: Use embedded build and development tools to develop STM32-based applications.
CO3: Program and interface STM32 peripherals using embedded C.
CO4: Implement interrupt service routines for reliable peripheral operation.
CO5: Design multitasking embedded applications using FreeRTOS.

Week	Contents	Hrs
1	Interpret and understand various technical documents, including Block Diagrams, Schematic Diagrams, Datasheets, User Manuals, and Application Notes related to the STM32. 2	4
2	Understanding build and development tools and processes involved in working with the STM32.	4
3	Program the STM32 microcontroller to utilize its peripherals, such as GPIO, UART, Timer, PWM, ADC, I2C, and SPI protocols.	4
4	Write interrupt service routines to handle peripheral operations and events.	4
5	Perform multitasking operations using FreeRTOS, by creating tasks and using inter process communication and synchronization mechanisms.	4
6	Development of Minor project in group adhering to the learned concepts.	6
	Total Hours	**26**

Text Books

1. K. V. K. Prasad, "Embedded / Real-Time Systems: Concepts, Design and Programming Black Book", Dreamtech Press, 2003.
2. Jonathan W. Valvano, "Embedded Systems: Introduction to ARM Cortex-M Microcontrollers", Pearson Education, 5th Edition, 2018.
3. Wayne Wolf, "Computers as Components: Principles of Embedded Computing System Design", Morgan Kaufmann, 4th Edition, 2016.

Reference Book

1. STM32F401 Microcontroller Documentation, STMicroelectronics, PDF Documentation. [https://www.st.com/en/microcontrollers-microprocessors/stm32f401/documentation.html]
2. FreeRTOS Documentation, Official Online Documentation. [https://www.freertos.org/Documentation/00-Overview]
3. Muhammad Ali Mazidi, Sarmad Naimi, and Sepehr Naimi, "ARM Assembly Language Programming", MicroDigitalEd, 2016.
4. Joseph Yiu, "The Definitive Guide to ARM® Cortex®-M3 and Cortex®-M4 Processors", Newnes, 3rd Edition, 2014.

Online References

1. NPTEL Course: Prof. Indranil Sengupta and Prof. Kamalika Datta, IIT Kharagpur, "Embedded System Design with ARM" [https://nptel.ac.in/courses/106105193/]
2. NPTEL Course: Prof. P. R. Sinha, IIT Delhi, "Embedded Systems" [https://nptel.ac.in/courses/108102045/]
3. NPTEL Course: Prof. V. Kamakoti, IIT Madras, "Microprocessors and Microcontrollers" [https://nptel.ac.in/courses/108106084/]

Third Year B. Tech Electronics and Telecommunication Engineering															
Semester VI															
Course Code: ET24HON391R						Course Name: Embedded Linux									
Teaching Scheme (Hours/Week)					Examination Scheme						Credits				
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	ODL	TOTAL
3	-	-	-	-	40	60	-	-	-	100	3	-	-	-	3

Course Objectives:

- Understand Linux OS Fundamental.
- Design and Develop POSIX Thread-Based Applications.
- Understand the principles and terminologies of Embedded Linux based on Yocto.
- Apply the BSP philosophy to configure and develop Yocto Project for application development.
- Develop and Debug Embedded Linux Device Drivers.

Course Outcomes:

After completing the course, the students will be able to:

CO1: Explain Linux OS fundamentals and internal subsystems.
CO2: Design and develop multi-threaded applications using POSIX APIs.
CO3: Explain Embedded Linux concepts and build systems based on the Yocto Project.
CO4: Apply BSP concepts to configure and develop Embedded Linux applications using Yocto.
CO5: Develop and debug Embedded Linux device drivers for peripheral interfaces.

Unit	Contents	Hrs
1	**Introduction to Linux** Introduction to Embedded Operating Systems and Linux, Bootloader, Kernel, Root File System, Process Management, Inter-process Communication & Synchronization, Memory Management, I/O subsystem.	8
2	**POSIX Thread Programming** POSIX Thread Programming, POSIX Semaphores, Mutexes, Conditional Variables, Barriers, Message Queues, Shared Memory, Debugging and Testing of Multi-threaded Applications, Socket Programming.	8
3	**Embedded Linux based on Yocto** Introduction to the Yocto Project, setting and configuration of build environment, Yocto Project architecture and components, OpenEmbedded Build System, BitBake Build Engine, Poky.	8
4	**Board Support Packages and Application Development** BSP Philosophy, building with a BSP, Prebuilt Binaries and configuration files, Recipe Files, creating a Yocto Project BSP, Creating Bootable Media Images. Setting Up a Yocto Project ADT, Preparing for Application Development and building an application and launching with QEMU.	8
5	**Embedded Device Drivers** The Embedded Linux Software Eco-System, Linux Kernel Modules and Module Programming, Char Device Drivers, Kernel Internals: Dynamic memory allocations, Handling Delays, Timers, Synchronization, Locking, I/O Memory and Ports, Interrupts, Deferred Executions, Driver De- bugging Techniques, Drivers for GPIO, I2C, and SPI, Pseudo Filesystems (procfs, sysfs).	8
	Total Hours	40

Text Books

3. C. Hallinan, "Embedded Linux Primer: A Practical Real-World Approach", Pearson Education India, 2nd Edition, 2011.
4. R. J. Streif, "Embedded Linux Systems with the Yocto Project", Prentice Hall Press, 2016.

Reference Book

1. J. Corbet, A. Rubini, and G. Kroah-Hartman, **"Linux Device Drivers"**, 3rd Edition, O'Reilly Media.
2. Karim Yaghmour, **"Building Embedded Linux Systems"**, O'Reilly Media, 2nd Edition.
3. Jonathan Corbet, **"Linux Kernel Development"**, Addison-Wesley Professional, 3rd Edition.

Online References

1. **NPTEL Course:** *Linux Basics-I* — fundamentals of Linux OS, directory structure, commands, shell basics, and scripting (useful foundation for POSIX/Embedded Linux) [https://nptel.ac.in/courses/117106113/]
2. **NPTEL Course:** *Embedded System Design with ARM*, IIT Kharagpur — covers embedded system design concepts important for system understanding in Embedded Linux context [https://nptel.ac.in/courses/106105193/

Third Year B. Tech Electronics and Telecommunication Engineering															
Semester VI															
Course Code: ET24HON392R						Course Name: Embedded Linux Lab									
Teaching Scheme (Hours/Week)					Examination Scheme						Credits				
L	P	T	OL	ODL	CIE	ETE	TW	PR	OR	TOTAL	L	P	T	ODL	TOTAL
-	2	-	-	-	-	-	25	-	-	25	-	1	-	-	1

Course Objectives:

- To demonstrate Proficiency in understanding and applying Linux Commands.
- To design and Implement POSIX Thread-Based Applications.
- To configure and Customize Embedded Linux using Yocto.
- To develop, Build, and Deploy embedded Linux applications on Embedded Hardware.
- To understand the process of developing character and GPIO drivers.

Course Outcomes:

After completing the course, the students will be able to:

CO1: Apply Linux commands and utilities for effective system-level operations.
CO2: Design and implement multi-threaded applications using POSIX threads and IPC mechanisms.
CO3: Configure, build, and deploy Embedded Linux systems and applications using the Yocto Project
on QEMU and embedded hardware.
CO4: Develop and test character and GPIO device drivers using Linux kernel internals.

Unit	Contents	Hrs
1	Gain a comprehensive understanding of various Linux commands and their applications in different scenarios.	4
2	Learn to harness the power of POSIX threads and Inter-Process Communication (IPC) mechanisms to demonstrate multi threaded operations, ensuring efficient and synchronized execution of tasks.	4
3	Set up and configure Embedded Linux using the Yocto Project.	4
4	Develop, build, and launch Embedded Linux applications on QEMU and BeagleBone Hardware modules, showcasing the integration of software and hardware components in embedded systems.	4
5	Leveraging kernel internals create character device drivers and develop drivers for GPIO.	4
6	Development of Minor project in group adhering to the learned concepts.	6
	Total Hours	**26**

Text Books

1. C. Hallinan, "Embedded Linux Primer: A Practical Real-World Approach", Pearson Education India, 2nd Edition, 2011.
2. R. J. Streif, "Embedded Linux Systems with the Yocto Project", Prentice Hall Press, 2016.

Reference Book

1. J. Corbet, A. Rubini, and G. Kroah-Hartman, "Linux Device Drivers", 3rd Edition, O'Reilly Media.
2. Karim Yaghmour, "Building Embedded Linux Systems", O'Reilly Media, 2nd Edition.
3. Jonathan Corbet, "Linux Kernel Development", Addison-Wesley Professional, 3rd Edition.

Online References

1. NPTEL Course: Prof. P. P. Chakrabarti, IIT Kharagpur, "Linux Basics and System Programming" [https://nptel.ac.in/courses/106105214/]
2. NPTEL Course: Prof. Indranil Sengupta and Prof. Kamalika Datta, IIT Kharagpur, "Embedded System Design with ARM" [https://nptel.ac.in/courses/106105193/]

Third Year B. Tech Electronics and Telecommunication Engineering															
Semester VII															
Course Code: ET24HON441R								Course Name: Automotive Communication Networks							
Teaching Scheme (Hours/Week)					Examination Scheme						Credits				
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	ODL	TOTAL
3	-	-	-	-	40	60	-	-	-	100	3	-	-	-	3

Course Objectives:

- Explain the importance and evolution of automotive networks.
- Understand the Controller Area Network (CAN), Local Interconnect Network (LIN) protocols and their role in automotive networks.
- Understand the role of the internet protocol and ethernet in advanced automotive systems.
- Understand the role of Diagnostic over IP and SOME/IP in optimizing performance in automotive systems.

Course Outcomes:

After completing the course, the students will be able to:

CO1: Explain the evolution, types, and architectures of automotive communication networks.
CO2: Analyze CAN, CAN-FD, and CAN-XL protocols including frames, error handling, and DBC concepts.
CO3: Explain LIN protocol operation, frame structure, and its applications in automotive systems.
CO4: Explain Automotive Ethernet standards, topologies, and frame formats for in-vehicle networking.
CO5: Apply Internet protocols and service-oriented communication concepts such as DoIP and SOME/IP in automotive systems.

Unit	Contents	Hrs
1	**Introduction to Automotive Protocols** Overview of automotive networks and their importance, History and evolution of automotive networks, Types of automotive networks - CAN, LIN, FlexRay, MOST, Ethernet, Network topology and architecture, Introduction to OSI layers.	8
2	**Controller Area Network** Introduction to CAN protocol, CANFD, CAN in automotive systems, CAN frame structure and message formatting, CAN bus topology and wiring, CAN error handling and fault tolerance. Introduction to CANXL. Introduction to CAN Database with DBC file.	8
3	**Local Interconnect Network** Introduction to LIN protocol, LIN applications in automotive systems, LIN frame structure and message formatting, LIN bus topology and wiring, handling and fault tolerance.	8
4	**Ethernet in Automotive Systems** Introduction to Ethernet protocol, Ethernet applications in automotive systems, Ethernet network topology, IEEE 100 BASE T1, IEEE 1000 BASE T1, IEEE 10 BASE T1S, Ethernet Frame, VLAN.	8
5	**Internet Protocols and Ethernet Applications** Internet Protocol basics and properties, IPv4, IPv6, network and host address, TCP and UDP, Diagnostic over IP, SOME/IP.	8
	Total Hours	40

Text Books

1. K. Matheus and T. Königseder, "Automotive Ethernet", Cambridge University Press, 2021.

2. R. Bosch GmbH, "CAN Protocols", Bosch Semiconductors
Reference Book
1. CAN Bus Explained, CCS Electronics [https://www.csselectronics.com/pages/can-bus-simple-intro-tutorial] 2. LIN Standards and Specifications, LIN Consortium — Official LIN protocol standards. [https://www.lin-cia.org/standards/] 3. OSI Model and Ethernet Basics Documentation, IEEE (online resources for automotive Ethernet standards). 4. Internet Protocol (IPv4/IPv6) & SOME/IP Overview, online protocol guides/tutorials
Online References
1. NPTEL Course: Fundamentals of Automotive Systems, IIT Madras [https://nptel.ac.in/courses/107106088/] 2. NPTEL Course: Communication Networks, IIT Kharagpur [https://onlinecourses.nptel.ac.in/noc24_ee71/preview]

Semester VII															
Course Code: ET24HON442R					Course Name: Automotive Communication Networks Lab										
Teaching Scheme (Hours/Week)					Examination Scheme						Credits				
L	P	T	OL	ODL	CIE	ETE	TW	PR	OR	TOTAL	L	P	T	ODL	TOTAL
-	2	-	-	-	-	-	25	-	-	25	-	1	-	-	1

Course Objectives:

- To design and implement a CAN-based system.
- Implement a LIN-based system
- Integrate CAN communication with Enhanced Ethernet Switch.
- Use Python scripts for CAN communication in ANDi.
- Send and receive SOME/IP messages using Media converter.

Course Outcomes:

After completing the course, the students will be able to:

CO1: Design and implement CAN-based systems and analyze errors and fault-tolerance mechanisms.
CO2: Implement and validate LIN-based communication systems for automotive applications.
CO3: Integrate CAN communication with enhanced automotive Ethernet switches and develop Python-based CAN applications using ANDi.
CO4: Send and receive TCP/UDP and SOME/IP messages using automotive media converters and develop a minor integrated project.

Unit	Contents	Hrs
1	Implement a CAN-based system. Investigate the effects of errors and faults on the CAN network and implement fault-tolerance mechanisms.	4
2	Implement a LIN-based system.	4
3	CAN communication using Enhance Ethernet Switch.	4
4	CAN communication using Python Script in ANDi.	4
5	Sending and Receiving TCP/UDP/ SOME/IP messages using Media converter.	4
6	Development of Minor project in group adhering to the learned concepts.	6
	Total Hours	**26**

Text Books

1. K. Matheus and T. Königseder, "Automotive Ethernet", Cambridge University Press, 2021 — Provides context for automotive Ethernet and SOME/IP communications.
2. Dominique Paret, "Multiplexed Networks for Embedded Systems: CAN, LIN, FlexRay, Safe-by-Wire", Society of Automotive Engineers, Wiley, 2007
3. Colt Correa, Charles M. Kozierok, Robert B. Boatright, and Jeffrey Quesnelle, "Automotive Ethernet — The Definitive Guide", Intrepid Control Systems, 2014

Reference Book

1. CAN Protocols — Technical Specification, Bosch Semiconductors (official CAN protocol overview and guidelines). [https://www.bosch-semiconductors.com/products/ip-modules/can-protocols/]
2. LIN Standards and Specifications, LIN Consortium (official LIN protocol standards and documentation). [https://www.lin-cia.org/standards/]
3. ANDi Hardware and Software Reference Documentation — Technical reference material for ANDi automotive network interface
4. ANDi Reference Document — Official ANDi system reference for CAN communication scripting (ISO 11898 CAN Protocol Standard — Official CAN bus standard documentation (ISO/SAE standard reference).

Online References
1. NPTEL Course: Fundamentals of Automotive Systems, IIT Madras [https://nptel.ac.in/courses/107106088/]   2. NPTEL Course: Embedded Systems, IIT Delhi [https://nptel.ac.in/courses/108102045/]

Final Year B. Tech Electronics and Telecommunication Engineering															
Semester VIII															
Course Code: ET24HON491R								Course Name: Capstone Project							
Teaching Scheme (Hours/Week)					Examination Scheme						Credits				
L	P	T	OL	ODL	CIE	ETE	TW	PR	OR	TOTAL	L	P	T	ODL	TOTAL
-	8	-	-	-	-	-	100	-	-	100	-	4	-	-	4

Course Objectives:

- To test students' knowledge of course implementation.
- To make students ready for designing the basic automotive electronic subsystems for monitoring and control of vehicle parameters using embedded systems.

Course Outcomes:

After completing the course, the students will be able to:

CO1: Analyze and develop automotive electronic circuits and embedded control programs for Real-time applications.

CO2: Demonstrate the ability to integrate sensors, controllers, and communication interfaces in automotive electronic applications..

Detailed Guidelines:

- The students are encouraged to take projects for developing software solutions and hardware platforms using the concept of course taken under the certification.
- The project should be individual and preferably from Industry.
- The project shall be on state-of-the-art topics selected by students and approved by the authority. The student must submit the project report in the prescribed standard format, duly certified by the concerned Guide and the Head of the Department/Institute, confirming the satisfactory completion of the work.
- Authorities/ examiner (optional) along with a guide would be assessing the project work based on various parameters which may include- Topic selection, Contents and Presentation, regularity, Punctuality and Timely Completion, Question and Answers, and Report

Total Hours	**104**

Marathwada Mitra Mandal's College of Engineering

Karvenagar, Pune- 52

An Autonomous Institute Affiliated to SPPU

"येथे बहुतांचे हित"

Curriculum Structure and Syllabus

Honours

A.Y. 2026-27

Accredited with A++ grade by NAAC
Recipient of Best College award by SPPU | Accredited by NBA
Recognized under 2(f) and 12(B) of UGC Act 1956
www.mmcoe.edu.in

About Marathwada Mitra Mandal's College of Engineering

MMCOE has been recognised as the BEST COLLEGE by SPPU. MMCOE is one feather in the cap of Marathwada Mitra Mandal's Pune, from the academic year 2006 – 2007.

The Institute offers six undergraduate engineering programmes in Electrical Engineering, Mechanical Engineering, Electronics & Telecommunication Engineering, Information Technology, Computer Engineering, and Artificial Intelligence & Data Science. In addition, it offers six postgraduate programmes M.Tech (Electrical – Power Electronics & Drives), M.Tech (Mechanical – Design Engineering), M.Tech (Electronics & Telecommunication – VLSI & Embedded Systems), M.Tech (Information Technology – Data Science), M.Tech (Computer Engineering), along with a Ph.D. programme in Computer Engineering. It houses 2000+ students and 200 staff members. Along with the focus on core academics, college also takes care of overall personality development of the students.

The college has an active robotics cell, NCC, Training and Placement cell etc. The students are motivated to actively participate in co-curricular and extra-curricular activities. They have brought many laurels to the Institute in the events such as Vinodottam Karandak, Firodiya Karandak, Zest, Dexterity etc.

Institute Vision

To be a globally renowned institution through excellence in engineering education for sustainable and holistic development

Institute Mission

M1: Empower students with cutting-edge technologies and global competencies

M2: Foster culture of research and entrepreneurial mindset

M3: Imbibe social and professional values

M4: Provide an inclusive environment for lifelong learning

Department of
Electronics and Telecommunication
Engineering

Vision and Mission of the Department

- **Vision**

To cultivate globally competent Electronics and Telecommunication Engineers for developing sustainable solutions

- **Mission**

Our Mission is to:
M1: To empower students with foundational and emerging technologies in Electronics and Telecommunication Engineering, fostering global competencies
M2: To foster a research and collaborative learning ecosystem that encourages entrepreneurship, holistic and sustainable development
M3: To inculcate ethical values, leadership, adaptability, and a lifelong learnability in a diverse, global engineering

- **Program Specific Outcomes (PSOs)**

Students will be able,
PSO1: To apply principles of Embedded Systems, VLSI, Communication, and Automation using modern hardware and software tools to develop effective E&TC solutions.
PSO2: To design and develop multidisciplinary, innovative, sustainable solutions through project-based learning to address industrial, societal, and environmental needs.

- **Program Outcomes (POs)**

PO1: Engineering Knowledge: Apply knowledge of mathematics, natural science, computing, engineering fundamentals and an engineering specialization as specified in WK1 to WK4 respectively to develop the solution of complex engineering problems.

PO2: Problem Analysis: Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development. (WK1 to WK4)

PO3: Design/Development of Solutions: Design creative solutions for complex engineering problems and design/develop systems/components/processes to meet identified needs with consideration for the public health and safety, whole-life cost, net zero carbon, culture, society and environment as required. (WK5)

PO4: Conduct Investigations of Complex Problems: Conduct investigations of complex engineering problems using research-based knowledge including design of experiments, modelling, analysis & interpretation of data to provide valid conclusions. (WK8).

PO5: Engineering Tool Usage: Create, select and apply appropriate techniques, resources and modern engineering & IT tools, including prediction and modelling recognizing their limitations to solve complex engineering problems. (WK2 and WK6)

PO6: The Engineer and The World: Analyze and evaluate societal and environmental aspects while solving complex engineering problems for its impact on sustainability with reference to economy, health, safety, legal framework, culture and environment. (WK1, WK5, and WK7).

PO7: Ethics: Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national & international laws. (WK9)

PO8: Individual and Collaborative Team work: Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams.

PO9: Communication: Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations considering cultural, language, and learning differences

PO10: Project Management and Finance: Apply knowledge and understanding of engineering management principles and economic decision-making and apply these to one's own work, as a member and leader in a team, and to manage projects and in multidisciplinary environments.

PO11: Life-Long Learning: Recognize the need for, and have the preparation and ability for i) independent and life-long learning ii) adaptability to new and emerging technologies and iii) critical thinking in the broadest context of technological change. (WK8)

Curriculum for B. Tech. in Electronics and Telecommunication Engineering with Honours in VLSI Design and Technology

Department of Electronics and Telecommunication Engineering

General Objectives and Scope:

Honours degrees offer a more in-depth study of the field in which one is pursuing honours. It might help the student obtain a greater chance at a decent employment than someone with a generic degree. The honours degree entails extensive study that culminates in a research potential in a student.

Completing an honours degree not only prepares one for higher education in a specific field, but also demonstrates to prospective employers a greater capacity for independent learning and research, as well as better problem-solving, critical thinking, independent learning, and communication abilities.

A bachelor's degree with honours will increase employability. This demonstrates to a recruiter that you can work diligently to earn good grades. It demonstrates ambition, self-discipline, and organization, among other excellent characteristics.

Why VLSI Design and Technology?

The significance of offering an honours course in VLSI Design and Technology lies in enhancing students' technical depth, research orientation, and industry readiness in a rapidly evolving semiconductor domain. The key points are:

1. Advanced Knowledge and Specialization

An honours course provides students with deeper exposure to advanced topics in VLSI such as low-power design, system-on-chip (SoC), FPGA/ASIC implementation, and semiconductor fabrication technologies. This helps students develop specialized expertise beyond the regular curriculum.

2. Alignment with Industry Requirements

The semiconductor industry requires engineers with strong design and verification skills. An honours course equips students with practical exposure to tools such as Cadence, Vivado, or Synopsys, making them more employable in VLSI design companies.

3. Research Orientation

Honours programs encourage students to engage in mini-projects, research papers, and design challenges. This develops analytical thinking and motivates students to pursue higher studies or research in microelectronics and integrated circuit design.

4. Hands-on Skill Development

Students gain practical experience in digital design, CMOS circuits, hardware description languages (Verilog/VHDL), and FPGA prototyping, enabling them to bridge the gap between theory and real-world hardware implementation.

5. Innovation and Entrepreneurship

Exposure to advanced design concepts can help students develop innovative hardware solutions, start semiconductor-based startups, or contribute to indigenous chip design initiatives.

6. Contribution to National Semiconductor Mission

With initiatives such as India Semiconductor Mission and chip design programs, trained VLSI engineers are in high demand. Offering this honours course helps prepare skilled manpower for the growing semiconductor ecosystem.

7. Improved Academic Profile

Honours courses enhance the academic credentials of students by providing additional credits, advanced coursework, and specialization, which is beneficial for both higher education and career opportunities.

Prerequisites :

- Digital Electronics
- Semiconductor Physics

B. TECH HONOURS IN VLSI DESIGN AND TECHNOLOGY (Semester IV, V, VI, VII, VIII)

Curriculum Structure

Course Code	Course Name	Teaching Scheme (Hrs/week)				Examination Scheme						Credits				
		L	P	T	ODL	CIE	ETE	TW	PR	OR	Total	L	P	T	ODL	Total
Semester IV																
ET24HON293R	System Design using FPGA	3	-	-	-	40	60	-	-	-	100	3	-	-	-	3
ET24HON294R	System Design using FPGA Lab	-	2	-	-	-	-	25	-	-	25	-	1	-	-	1
Semester V																
ET24HON343R	VLSI Testing and Verification	3	-	-	-	40	60	-	-	-	100	3	-	-	-	3
ET24HON344R	VLSI Testing and Verification Lab	-	2	-	-	-	-	25	-	-	25	-	1	-	-	1
Semester VI																
ET24HON393R	VLSI Manufacturing Eco System	3	-	-	-	40	60	-	-	-	100	3	-	-	-	3
Semester VII																
ET24HON443R	Analog CMOS Design	3	-	-	-	40	60	-	-	-	100	3	-	-	-	3
ET24HON444R	Analog CMOS Design Lab	-	2	-	-	-	-	25	-	-	25	-	1	-	-	1
Semester VIII																
ET24HON493R	Capstone Project	-	10	-	-	-	-	75	-	50	125	-	-	5	-	5
Total		12	16	-	-	160	240	150	-	50	600	12	3	5	-	20

L- Lecture	P- Practical	T- Tutorial	CIE-Continuous Internal Evaluation
ETE- End Term Examination	TW- Term work	PR- Practical	OR- Oral
L : 1 Hr.= 1 credit	P: 2 Hr. = 1 Credit	T: 1 Hr. = 1 Credit	ODL-Open Distance Learning

Second Year B. Tech Electronics and Telecommunication Engineering

Semester- IV

Course Code: ET24HON293R **Course Name: System Design using FPGA**

Teaching Scheme (Hours/Week)					Examination Scheme						Credits				
L	P	T	OL	ODL	CIE	ETE	TW	PR	OR	TOTAL	L	P	T	ODL	TOTAL
3	-	-	-	-	40	60	-	-	-	100	3	-	-	-	3

Course Objectives:

- Introduce FPGA architectures and their role in VLSI system design.
- Develop skills to design digital circuits using HDL for FPGA implementation.
- Apply FPGA design methodology including synthesis, timing analysis, and implementation.
- Demonstrate verification and simulation techniques for FPGA-based designs.
- Implement a VLSI system

Course Outcomes:

After completing the course, the students will be able to:

CO1: Explain the architecture and operation of FPGA devices for VLSI applications.

CO2: Design combinational and sequential circuits using HDL for FPGA implementation.

CO3: Perform FPGA synthesis, timing analysis, and implementation for digital systems.

CO4: Verify and simulate FPGA designs using testbenches and functional verification techniques.

CO5: Implement a VLSI system using subsystems like counters, ALUs, memory, and FSMs on FPGA

Unit	Contents	Hrs.
1	**Introduction to FPGA and VLSI Design** Overview of VLSI design flow, FPGA architecture: CLBs, IOBs, routing resources, LUTs, FPGA vs ASIC comparison, Design entry methods: HDL-based and schematic-based	8
2	**Digital Design with HDL** Introduction to Verilog for FPGA design, Combinational and sequential circuits design ,Simulation and synthesis concepts, Behavioral, dataflow, and structural modeling	8
3	**FPGA Design Methodology** FPGA design flow: design entry , synthesis , implementation , bitstream generation, Timing analysis and constraints, Floorplanning and placement, Introduction to partial reconfiguration.	8
4	**FPGA Verification** Testbench creation for FPGA designs, Functional verification using simulation, Introduction to FPGA-based prototyping, Emerging trends: SoC integration, High-Level Synthesis (HLS), SystemVerilog for FPGA design	8
5	**VLSI Subsystems Implementation** Design of counters, registers, multiplexers, adders, and ALUs on FPGA, Memory elements: RAM, ROM, FIFO implementation, Design of FSMs (Mealy and Moore machines), Pipeline and parallelism in FPGA design	8
	Total Hours	**40**

Text Books

1. Zainalabedin Navabi, Verilog Digital System Design, McGraw-Hill, 2nd Edition, 2009.
2. Stephen Brown and Zvonko Vranesic, Fundamentals of Digital Logic with VHDL Design, McGraw-Hill, 3rd Edition, 2009.
3. John F. Wakerly, Digital Design: Principles and Practices, 5th Edition, Pearson, 2018.

Reference Books
1. Charles H. Roth, Digital System Design Using FPGA: Implementation of Arithmetic and Control Systems, Cengage, 2010. 2. Wayne Wolf, FPGA-Based System Design, Prentice Hall, 2004. 3. Michael D. Ciletti, Advanced Digital Design with the Verilog HDL, 2nd Edition, Prentice Hall, 2003. 4. Pong P. Chu, FPGA Prototyping By VHDL Examples, 2nd Edition, Wiley, 2008.
Online References
1. https://www.magmio.com/news/154-fpga-use-cases 2. NPTEL Course "Hardware modeling usingVerilog" https://onlinecourses.nptel.ac.in/noc23_cs76/preview

Second Year B. Tech Electronics and Telecommunication Engineering

Semester- IV

Course Code: ET24HON294R Course Name: System Design using FPGA Lab

Teaching Scheme (Hours/Week)					Examination Scheme						Credits				
L	P	T	OL	ODL	CIE	ETE	TW	PR	OR	TOTAL	L	P	T	ODL	TOTAL
-	2	-	-	-	-	-	25	-	-	25	-	1	-	-	1

Course Objectives:

- Introduce FPGA architectures and their role in VLSI system design.
- Develop skills to design digital circuits using HDL for FPGA implementation.
- Apply FPGA design methodology including synthesis, timing analysis, and implementation.
- Implement VLSI subsystems like counters, ALUs, memory, and FSMs on FPGA.
- Demonstrate verification and simulation techniques for FPGA-based designs.

Course Outcomes:

After completing the course, the students will be able to:

CO1: Explain the architecture and operation of FPGA devices for VLSI applications.

CO2: Design combinational and sequential circuits using HDL for FPGA implementation.

CO3: Perform FPGA synthesis, timing analysis, and implementation for digital systems.

CO4: Develop and implement VLSI subsystems and finite state machines on FPGA.

CO5: Verify and simulate FPGA designs using testbenches and functional verification techniques.

Unit	Contents	Hrs.
1	Finite state machine	4
2	Case study-1: Traffic Light controller	4
3	Case study -2 Vending machine	6
4	Case study- 3: Automotive Electronics applications	6
5	Case study- 4: Image Processing and Computer vision applications	6
	Total Hours	**26**

Text Books

1 .Zainalabedin Navabi, Verilog Digital System Design, McGraw-Hill, 2nd Edition, 2009.
2. Stephen Brown and Zvonko Vranesic, Fundamentals of Digital Logic with VHDL Design, McGraw-Hill, 3rd Edition, 2009.
3. John F. Wakerly, Digital Design: Principles and Practices, 5th Edition, Pearson, 2018.

Reference Books

1 .Charles H. Roth, Digital System Design Using FPGA: Implementation of Arithmetic and Control Systems, Cengage, 2010.
2. Wayne Wolf, FPGA-Based System Design, Prentice Hall, 2004.
3. Michael D. Ciletti, Advanced Digital Design with the Verilog HDL, 2nd Edition, Prentice Hall, 2003.
4. Pong P. Chu, FPGA Prototyping By VHDL Examples, 2nd Edition, Wiley, 2008.

Online References

1. https://www.magmio.com/news/154-fpga-use-cases
2.NPTEL Course "Hardware modeling usingVerilog"
https://onlinecourses.nptel.ac.in/noc23_cs76/preview

Third Year B. Tech Electronics and Telecommunication Engineering

Semester- V

Course Code: ET24HON343R Course Name: VLSI Testing and Verification

Teaching Scheme (Hours/Week)					Examination Scheme						Credits				
L	P	T	OL	ODL	CIE	ETE	TW	PR	OR	TOTAL	L	P	T	ODL	TOTAL
3	-	-	-	-	40	60	-	-	-	100	3	-	-	-	3

Course Objectives:

- Explain the fundamentals of digital verification and the role of SystemVerilog in hardware verification.
- Apply System Verilog language constructs for developing verification testbenches.
- Implement object-oriented programming concepts in SystemVerilog for advanced verification.
- Analyze assertions, constrained randomization, and functional coverage techniques.
- Develop verification environments using SystemVerilog for digital designs.

Course Outcomes:

After completing the course, the students will be able to:

CO1: Explain testing and verification concepts and SystemVerilog language features

CO2: Develop basic testbenches using SystemVerilog constructs such as interfaces, tasks, and functions.

CO3: Apply object-oriented programming concepts to create reusable verification components.

CO4: Design verification environments for testing and validating digital systems

CO5:Explain VLSI test structures and fault modeling fundamentals

Unit	Contents	Hrs.
1	**Introduction to VLSI testing and Verification** Introduction to VLSI testing and verification, need for testing and verification, verification methodologies, overview of Verilog vs SystemVerilog, SystemVerilog data types, variables, arrays, structures, enumerations, procedural blocks, tasks and functions, interfaces.	8
2	**SystemVerilog for Design and Testbench** Modules and interfaces in SystemVerilog, clocking blocks, program blocks, testbench architecture, stimulus generation, procedural statements, packages, randomization basics, constraints and distribution.	8
3	**Functional Verification Concepts** Assertions in SystemVerilog, immediate and concurrent assertions, writing properties and sequences, coverage concepts, functional coverage, covergroups, coverpoints, cross coverage, coverage-driven verification,Object-oriented programming in SystemVerilog, classes, objects, inheritance, polymorphism, virtual methods, mailboxes, semaphores, inter-process communication, transaction-based verification, testbench components	8
4	**Verification Methodology and Applications** Introduction to verification methodologies, layered testbench architecture, driver, monitor, scoreboard, stimulus generation techniques, debugging and verification strategies, introduction to UVM concepts, practical case studies of digital circuit verification.	8
5	**VLSI testing and analysis** Verification and testing process comparison, Need of testing.Functional modeling at the logic and the register level, Structural models, Level of modeling. Unknown logic values different delay models, Hazard Detection, Design for testability, Scan and Boundary architectures, Built-in Self-test (BIST), Test pattern generation for BIST, analog test bus standard. Testability measure (SCOAP) , trade- offs, Ad hoc	8

	Design for Testability techniques, Self-testing circuits for systems, memory & processor testing, automatic test pattern generation and TAP Controller, JTAG. Introduction to Fault modeling and simulation,Types of faults, Fault sampling and statistical fault analysis	
	Total Hours	**40**

Text Books

1. Abramovici M, Breuer M A and Friedman A D, "Digital systems and Testable Design", 1st Edition, Jaico Publications, 1994
2. Kropf T, "Introduction to Formal Hardware Verification", Springer Publications, 2008
3. Stuart Sutherland, Simon Davidman and Peter Flake, "SystemVerilog for Design: A Guide to Using System Verilog for Hardware Design and Modeling", 2nd Edition, Springer, 2010
4. Zainalabedin Navabi, "Verilog Digital System Design Register Transfer Level Synthesis, Testbench and Verification", 1st edition McGraw- Hill, 2005

Reference Books

1. Janick Bergeron, Kluwer, "Writing Test benches: Functional Verification of HDL Models", 2nd Edition, Academic Publishers, 2003
2. Andreas S. Meyer, "Principles of Functional Verification", Elsevier Science, 2004
3. Donald Thomas, "Logic Design and Verification Using System Verilog, CreateSpace", Independent Publishing Platform, 2014
4. Harry D. Foster, Adam C. Krolnik, David J. Lacey, Kluwer, "Assertion-Based Design", 2nd Edition, Academic Publishers, 2004

Online References

1. Digital VLSI Testing By Prof. Santanu Chattopadhya, IIT Kharagpur [https://archive.nptel.ac.in/noc/courses/noc17/SEM1/noc17-ec02/]

Third Year B. Tech Electronics and Telecommunication Engineering

Semester- V

Course Code: ET24HON344R Course Name: VLSI Testing and Verification Lab

Teaching Scheme (Hours/Week)					Examination Scheme						Credits				
L	P	T	OL	ODL	CIE	ETE	TW	PR	OR	TOTAL	L	P	T	ODL	TOTAL
-	2	-	-	-	-	-	25	-	-	25	-	1	-	-	1

Course Objectives:

- Introducing basic SystemVerilog language features.
- Learn to design a test plan for given digital systems and verify it using SystemVerilog testbenches.
- Apply object-oriented programming concepts in SystemVerilog for verification purposes.
- Implement basic assertion-based verification to ensure correct operation of digital circuits.
- Learn test structures and modeling concepts

Course Outcomes:

After completing the course, the students will be able to:

CO1: Demonstrate the use of SystemVerilog data types, operators, and expressions.

CO2: Write modular SystemVerilog programs using tasks and functions.

CO3: Develop testbenches to verify combinational circuits like Half Adder and Full Adder.

CO4: Implement classes and objects in SystemVerilog to represent verification components.

CO5: Apply assertion-based verification techniques to check the functionality of digital circuits such as multiplexers.

Unit	Contents	Hrs.
1	Hands on with System Verilog language and features	4
2	A)Create a testbench in SystemVerilog to verify a Half Adder or Full Adder circuit. B)Develop a SystemVerilog program using tasks and functions for adder based Arithmetic and Logic Unit(ALU).	6
3	Write a SystemVerilog program to demonstrate classes and objects (basic OOP concept).	4
4	Implement 2 B) above using UVM methodology	6
5	Mini Project: Simulation of JTAG module using system Verilog	6
	Total Hours	**26**

Text Books

1. Abramovici M, Breuer M A and Friedman A D, "Digital systems and Testable Design", 1st Edition, Jaico Publications, 1994
2. Kropf T, "Introduction to Formal Hardware Verification", Springer Publications, 2008
3. Stuart Sutherland, Simon Davidman and Peter Flake, "SystemVerilog for Design: A Guide to Using System Verilog for Hardware Design and Modeling", 2nd Edition, Springer, 2010
4. Zainalabedin Navabi, "Verilog Digital System Design Register Transfer Level Synthesis, Testbench and Verification", 1st edition McGraw- Hill, 2005

Reference Books

1. Janick Bergeron, Kluwer, "Writing Test benches: Functional Verification of HDL Models", 2nd Edition, Academic Publishers, 2003
2. Andreas S. Meyer, "Principles of Functional Verification", Elsevier Science, 2004
3. Donald Thomas, "Logic Design and Verification Using System Verilog, CreateSpace", Independent Publishing Platform, 2014

4. Harry D. Foster, Adam C. Krolnik, David J. Lacey, Kluwer, "Assertion-Based Design", 2nd Edition, Academic Publishers, 2004
Online References
1. NOC | VLSI Design Verification and test (nptel.ac.in) :http://nptel.ac.in 2. Digital VLSI Testing - Course (nptel.ac.in)

Third Year B. Tech Electronics and Telecommunication Engineering

Semester- VI

Course Code: ET24HON393R **Course Name: VLSI Manufacturing Eco System**

Teaching Scheme (Hours/Week)					Examination Scheme						Credits				
L	P	T	OL	ODL	CIE	ETE	TW	PR	OR	TOTAL	L	P	T	ODL	TOTAL
3	-	-	-	-	40	60	-	-	-	100	3	-	-	-	3

Course Objectives:

- Introduce evolution of Global Landscape and Semiconductor Economics
- Introduction to VLSI fabrication process and advanced technologies
- Overview of VLSI assembly,packaging and testing technology

Course Outcomes:

After completing the course, the students will be able to:

CO1: Describe semiconductor ecosystem

CO2: Explain VLSI fabrication steps and require equipments

CO3: Explain overview of advanced semiconductor technologies .

CO4: Explain VLSI assembly,packaging and testing technology

CO5: Grasp Indian Context of Semiconductor Industries and trends

Unit	Contents	Hrs.
1	**Global Landscape and Semiconductor Economics** Evolution of semiconductor industries and VLSI technology, evolution of IDMs (INTEL) to fabless -foundry model(NVDIA-TSMC),semiconductor supply chain and ecosystem- roles of fabless companies,EDA tool companies,VLSI IP companies, semiconductor foundries, overview of semiconductor raw materials,OSAT, semiconductor market trends and economics.	8
2	**VLSI Wafer Fabrication and Ecosystem** Review of CMOS fabrication process, Substrate manufacturing, Clean rooms and contamination control ,Lithography, fabrication steps: oxidation, photolithography, diffusion, ion implantation, deposition techniques (CVD, PVD), etching (wet and dry), planarization (CMP).	8
3	**Advanced Semiconductor technologies and fabrication** Semiconductor materials like GaN, SiC etc. and devices based on them ,specialized power devices, advanced technologies (FinFET, 3D ICs etc.), technology for memories,High Bandwidth Memory(HBM).	8
4	**Assembly, Packaging and Testing** Wafer sorting and testing,Semiconductor packaging concepts, types of packages (DIP, QFP, BGA, CSP), packaging technologies, reliability issues, chip testing, burn-in testing, yield and quality control.	9
5	**Semiconductor Industry and Emerging Trends** Semiconductor design ecosystem, EDA tools and IP providers, fabrication facilities, Semiconductor manufacturing in India, government initiatives, Future trends in VLSI manufacturing.	7
	Total Hours	**40**

Text Books

1. Peter Van Zant, Microchip Fabrication: A Practical Guide to Semiconductor Processing, 6th Edition, McGraw-Hill Education, 2014.

2. Michael Quirk and Julian Serda, Semiconductor Manufacturing Technology, 2nd Edition, Pearson Education, 2001.

Reference Books

1. S. Wolf and R. N. Tauber, Silicon Processing for the VLSI Era, Volume 1: Process Technology, Lattice Press.

2. Sorab K. Ghandhi, VLSI Fabrication Principles: Silicon and Gallium Arsenide, 2nd Edition, Wiley India. 3. S. M. Sze and Ming-Kwei Lee, Semiconductor Devices: Physics and Technology, 3rd Edition, Wiley India, 2012.

Online References

1. http://digimat.in/nptel/courses/video/108101089/L21.html
2. Online resources related to latest manufacturing technologies

Final Year B. Tech Electronics and Telecommunication Engineering

Semester- VII

Course Code: ET24HON443R **Course Name: Analog CMOS Design**

Teaching Scheme (Hours/Week)					Examination Scheme						Credits				
L	P	T	OL	ODL	CIE	ETE	TW	PR	OR	TOTAL	L	P	T	ODL	TOTAL
3	-	-	-	-	40	60	-	-	-	100	3	-	-	-	3

Course Objectives:

- Explain MOS transistor operation, modeling, and characteristics in analog ICs.
- Study CMOS analog sub-circuits such as MOS switches, current mirrors, and reference circuits.
- Analyze and design CMOS amplifier configurations.
- Learn the design and compensation techniques of CMOS operational amplifiers.
- Explain the operation and performance of commonly used analog systems.

Course Outcomes:

After completing the course, the students will be able to:
CO1: Explain MOS device operation, modeling, and passive components in CMOS technology.
CO2: Analyze and design CMOS sub-circuits like current mirrors, and sources.
CO3: Evaluate different CMOS amplifier architectures and their performance.
CO4: Design CMOS operational amplifiers and analyze stability and PSRR.
CO5: Design of OP Amp based analog systems

Unit	Contents	Hrs.
1	**Analog Design Foundation** Understanding of R,L,C, V and I based circuits,Overview of MOS Transistor, Passive Components- Capacitor & Resistor, Integrated circuit Layout, CMOS Device Modeling - Simple MOS Large-Signal Model, Other Model Parameters, Small-Signal Model for the MOS Transistor, Computer Simulation Models.	8
2	**Analog CMOS Sub-Circuits** MOS Diode, MOS Active Resistor, Current Sinks and Sources, Current Mirrors, Degeneration, Cascode current Mirrors ,Common Source, Common Gate , Source Follower, Cascode Amplifiers , Analysis of Input Impedance , output impedance and Gain	8
3	**Fundamentals of CMOS Amplifiers** Differential Amplifiers basics,Design of CMOS Op Amps, Design of Two-Stage Op Amps, input -output impedances, Gain,Frequency response analysis	8
4	**CMOS Operational Amplifiers** Two-Stage Op Amps, OP Amp in feedback, Compensation of Op Amps,Measurement techniques of OP Amp parameters	8
5	**Analog Systems** Current and Voltage References, Band gap Reference architectures, Voltage regulators,Comparators and discrete time comparators	8
	Total Hours	**40**

Text Books

1.Design of Analog CMOS Integrated Circuits- Second edition,Behzad Razavi, TMH Edition.
2.CMOS Analog Circuit Design - Philip E. Allen and Douglas R. Holberg, Oxford University Press, International Second Edition/Indian Edition, 2010.

Reference Books

1. Franco Maloberti Texas A & M University, U.S.A. and University of Pavia, Italy, "Analog Design for CMOS VLSI Systems" 2. Analog Integrated Circuit Design- David A. Johns, Ken Martin, Wiley Student Edn, 2013. 3. CMOS: Circuit Design, Layout and Simulation- Baker, Li and Boyce, PHI.
Online References
1. https://onlinecourses.nptel.ac.in/noc23_ee142/preview

Final Year B. Tech Electronics and Telecommunication Engineering

Semester- VII

Course Code: ET24HON444R **Course Name: Analog CMOS Design Lab**

Teaching Scheme (Hours/Week)					Examination Scheme						Credits				
L	P	T	OL	ODL	CIE	ETE	TW	PR	OR	TOTAL	L	P	T	ODL	TOTAL
-	2	-	-	-	-	-	25	-	-	25	-	1	-	-	1

Course Objectives:

- To introduce students to CMOS fundamentals
- To develop the ability to design analog CMOS circuits
- To develop fundamentals of OPAMP design
- To introduce students to fundamentals of analog systems

Course Outcomes:

After completing the course, the students will be able to:
CO1: Explain CMOS Fundamentals.
CO2: Explain analog CMOS components.
CO3: Design OPAMP Circuits
CO4: Design of analog systems

Unit	Contents	Hrs.
1	**Introduction to Cadence design tools for schematic and simulation** PART – A (Analog Experiments-using EDA tools like Cadence Virtuoso/Synopsys) Common source amplifier schematic for a given gain , Design ,schematic capture and simulate for the dc transfer function , ac analysis and comment on results.	4
2	More different types of analysis for above designs(Frequency response,transient,cascode etc.)	4
3	Part-1 Design of 7 T OPAMP - design schematic capture, functional simulation	4
4	Part-2 Design of 7 T OPAMP - simulation for closed loop and open loop performance	4
5	Part B - Mini project (Any one) Specifications,Design and verification of Bandgap reference/ Specifications,Design and verification of Voltage regulator(LDO)/ Specifications,Design and verification of Comparator	10
	Total Hours	**26**

Text Books

1. Design of Analog CMOS Integrated Circuits- Second edition,Behzad Razavi, TMH Edition.
2. CMOS Analog Circuit Design - Philip E. Allen and Douglas R. Holberg, Oxford University Press, International Second Edition/Indian Edition, 2010.

Reference Books

1. Franco Maloberti Texas A & M University, U.S.A. and University of Pavia, Italy, "Analog Design for CMOS VLSI Systems"
2. Analog Integrated Circuit Design- David A. Johns, Ken Martin, Wiley Student Edn, 2013.
3. CMOS: Circuit Design, Layout and Simulation- Baker, Li and Boyce, PHI.

Online References

1. https://onlinecourses.nptel.ac.in/noc18_ee04
2. https://nptel.ac.in/content/syllabus_pdf/117106030.pdf
3. http://www.infocobuild.com/education/audio-video
4. courses/electronics/AnalogICDesign-IIT-Madras/lecture-29.html

Final Year B. Tech Electronics and Telecommunication Engineering

Semester- VIII

Course Code: ET24HON493R **Course Name: Capstone Project**

Teaching Scheme (Hours/Week)					Examination Scheme						Credits				
L	P	T	OL	ODL	CIE	ETE	TW	PR	OR	TOTAL	L	P	T	ODL	TOTAL
-	10	-	-	-	-	-	75	-	50	125	-	-	5	-	5

Course Objectives:

- To develop a practical VLSI project for a specific application ,literature survey ,finalize title,scope and specifications
- To apply VLSI concepts to design, simulate, and verify VLSI circuits and subsystems for the specifications
- To demonstrate professionalism with ethics,present effective communication skills and relate VLSI engineering challenges to societal needs.

Course Outcomes:

After completing the course, the students will be able to:

CO1: Undertake real-world problem identification, formulation, and solution in VLSI system design

CO2: Develop detailed (system and block by block) Engineering specifications

CO3: Design efficient and optimized VLSI circuits and subsystems using a systematic and methodical approach

CO4: Demonstrate a strong technical foundation in VLSI design and verification techniques

CO5: Demonstrate effective communication skills, teamwork, and professional ethics in VLSI engineering projects.

Unit	Contents	Hrs.
-	**Project Overview** Introduction to project work Objectives and expected outcomes Importance of research, review, and demonstration Publication and documentation basics	10
	Stage 1: Project Identification & Literature Survey Problem identification Selection of project topic Literature review / survey of existing work Feasibility analysis **Deliverable:** Project proposal document (Synopsis – I)	20
	Stage 2: Design & Specifications System design and architecture Block diagram and workflow Tools, technologies, and specifications Methodology selection **Deliverable:** Design document (Synopsis – II)	20
	Stage 3: Implementation (Programming & Development) Coding and module development Integration of components Testing methods and debugging Iterative improvements	30

	Deliverable: Working model / prototype	
	Stage 4: Report Writing & Publication Final report preparation Documentation standards20 Presentation of results Paper writing / publication (if applicable) **Deliverable:** Final project report	20
	Total Hours	**100**

Marathwada Mitra Mandal's College of Engineering

Karvenagar, Pune- 52

An Autonomous Institute Affiliated to SPPU

"येथे बहुतांचे हित"

Curriculum Structure and Syllabus

Honours

A.Y. 2026-27

Accredited with A++ grade by NAAC
Recipient of Best College award by SPPU | Accredited by NBA
Recognized under 2(f) and 12(B) of UGC Act 1956
www.mmcoe.edu.in

About Marathwada Mitra Mandal's College of Engineering

MMCOE has been recognised as the BEST COLLEGE by SPPU. MMCOE is one feather in the cap of Marathwada Mitra Mandal's Pune, from the academic year 2006 – 2007.

The Institute offers six undergraduate engineering programmes in Electrical Engineering, Mechanical Engineering, Electronics & Telecommunication Engineering, Information Technology, Computer Engineering, and Artificial Intelligence & Data Science. In addition, it offers six postgraduate programmes M.Tech (Electrical – Power Electronics & Drives), M.Tech (Mechanical – Design Engineering), M.Tech (Electronics & Telecommunication – VLSI & Embedded Systems), M.Tech (Information Technology – Data Science), M.Tech (Computer Engineering), along with a Ph.D. programme in Computer Engineering. It houses 2000+ students and 200 staff members. Along with the focus on core academics, college also takes care of overall personality development of the students.

The college has an active robotics cell, NCC, Training and Placement cell etc. The students are motivated to actively participate in co-curricular and extra-curricular activities. They have brought many laurels to the Institute in the events such as Vinodottam Karandak, Firodiya Karandak, Zest, Dexterity etc.

Institute Vision

To be a globally renowned institution through excellence in engineering education for sustainable and holistic development

Institute Mission

M1: Empower students with cutting-edge technologies and global competencies

M2: Foster culture of research and entrepreneurial mindset

M3: Imbibe social and professional values

M4: Provide an inclusive environment for lifelong learning

Department of Information Technology

Vision and Mission of the Department

- **Vision**

  To emerge as a globally competent center in education, research and innovation in Information Technology, fostering sustainable development with a holistic vision.

- **Mission**

  M1 : To impart IT education that develops globally competent professionals

  M2 : To promote innovation through research and entrepreneurship.

  M3 : To encourage lifelong learning and a holistic commitment towards societal development.

- **Program Specific Outcomes (PSOs)**

  Information Technology graduates will be able to :

  - Design and develop information systems using principles of computing to solve real-world problems.
  - Integrate emerging IT technologies to innovate and improve technological ecosystems.

Program Outcomes (POs)

PO1: Engineering Knowledge: Apply knowledge of mathematics, natural science, computing, engineering fundamentals and an engineering specialization as specified in WK1 to WK4 respectively to develop to the solution of complex engineering problems.

PO2: Problem Analysis: Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development. (WK1 to WK4)

PO3: Design/Development of Solutions: Design creative solutions for complex engineering problems and design/develop systems/components/processes to meet identified needs with consideration for the public health and safety, whole-life cost, net zero carbon, culture, society and environment as required. (WK5)

PO4: Conduct Investigations of Complex Problems: Conduct investigations of complex engineering problems using research-based knowledge including design of experiments, modelling, analysis & interpretation of data to provide valid conclusions. (WK8).

PO5: Engineering Tool Usage: Create, select and apply appropriate techniques, resources and modern engineering & IT tools, including prediction and modelling recognizing their limitations to solve complex engineering problems. (WK2 and WK6)

PO6: The Engineer and The World: Analyze and evaluate societal and environmental aspects while solving complex engineering problems for its impact on sustainability with reference to economy, health, safety, legal framework, culture and environment. (WK1, WK5, and WK7).

PO7: Ethics: Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national & international laws. (WK9)

PO8: Individual and Collaborative Team work: Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams.

PO9: Communication: Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations considering cultural, language, and learning differences

PO10: Project Management and Finance: Apply knowledge and understanding of engineering management principles and economic decision-making and apply these to one's own work, as a member and leader in a team, and to manage projects and in multidisciplinary environments.

PO11: Life-Long Learning: Recognize the need for, and have the preparation and ability for i) independent and life-long learning ii) adaptability to new and emerging technologies and iii) critical thinking in the broadest context of technological change. (WK8)

Curriculum for B. Tech. in Information Technology with Honours in Blockchain Technology

Department of Information Technology

General Objectives and Scope:

The Honours Degree Program aims to provide students with advanced knowledge and deeper understanding in their chosen field of study. It focuses on developing critical thinking, analytical abilities, and research skills through specialized courses and practical learning. The program encourages independent learning and innovation, enabling students to gain expertise beyond the regular undergraduate curriculum.

The scope of the honours degree includes opportunities for research, specialization, and skill development. Students may undertake projects or dissertations that enhance their practical and analytical abilities. The program also prepares students for higher education, research careers, and better employment opportunities in industry and academia.

Why Blockchain Technology?

Blockchain Technology is offered as an honours course to provide students with advanced knowledge of decentralized systems and emerging digital technologies. The objective is to help students understand the concepts of distributed ledgers, cryptography, smart contracts, and secure data management. It also develops analytical and technical skills required to design transparent, secure, and efficient digital solutions for modern applications.

The scope of blockchain technology is rapidly expanding across various sectors such as finance, supply chain management, healthcare, and cybersecurity. Studying blockchain as an honours subject enables students to explore innovative applications, conduct research, and develop practical solutions using decentralized technologies, thereby preparing them for advanced studies and emerging career opportunities in the technology industry.

Prerequisites :

1. Data Structures
2. Computer Networks
3. Database Management Systems
4. Operating Systems
5. Cryptography and Network Security
6. Object Oriented Programming
7. Distributed Systems
8. Basics of Programming (Python / Java / C++)

B. TECH HONORS IN BLOCK CHAIN TECHNOLOGY (Semester IV, V, VI, VII, VIII)

Curriculum Structure

Course Code	Course Name	Teaching Scheme (Hrs/week)				Examination Scheme						Credits				
		L	P	T	ODL	CIE	ETE	TW	PR	OR	Total	L	P	T	ODL	Total
Semester IV																
IT24HON291R	Fundamentals of Block chain	3	-	-	-	40	60	-	-	-	100	3	-	-	-	**3**
IT24HON292R	Blockchain Technology Laboratory	-	4	-	-	-	-	25	-	25	50	-	2	-	-	**2**
Semester V																
IT24HON341R	Blockchain Architecture and Design	3	-	-	-	40	60	-	-	-	100	3	-	-	-	**3**
IT24HON342R	Blockchain Developme nt Laboratory	-	4	-	-	-	-	25	-	25	50	-	2	-	-	**2**
Semester VI																
IT24HON391R	Blockchain Application Development and Security Engineering	2	-	-	-	40	60	-	-	-	100	2	-	-	-	**2**
IT24HON392R	Blockchain Application Developme nt and Security Engineering Laboratory	-	2	-	-	-	-	25	-	-	25	-	1	-	-	**1**
Semester VII																
IT24HON441R	Blockchain Applications and Emerging Technologies	2	-	-	-	40	60	-	-	-	100	2	-	-	-	**2**
IT24HON442R	Seminar	-	2	-	-	-	-	25	-	-	25	-	1	-	-	**1**
Semester VIII																
IT24HON491R	Capstone Project	-	8	-	-	-	-	25	-	100	125	-	4	-	-	**4**
Total		**10**	**20**	**-**	**-**	**160**	**240**	**125**	**-**	**150**	**675**	**10**	**10**	**-**	**-**	**20**

L- Lecture P- Practical T- Tutorial CIE-Continuous Internal Evaluation
ETE- End Term Examination TW- Term work PR- Practical OR- Oral
L : 1 Hr.= 1 credit P: 2 Hr. = 1 Credit T: 1 Hr. = 1 Credit ODL-Open Distance Learning

Second Year B.Tech Information Technology

Semester- IV

Course Code: IT24HON291R **Course Name: Fundamentals of Blockchain**

Teaching Scheme (Hours/Week)					Examination Scheme						Credits				
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	ODL	TOTAL
3	-	-	-	-	40	60	-	-	-	100	3	-	-	-	3

Course Objectives:

- Introduce the fundamental concepts, architecture, and working principles of blockchain and distributed ledger technologies.
- Explain cryptographic foundations and consensus mechanisms used to ensure security, transparency, and immutability in blockchain systems.
- Provide hands-on exposure to blockchain platforms such as Ethereum and Hyperledger Fabric.
- Enable students to design, develop, and deploy smart contracts and decentralized applications (DApps) using Solidity.
- Analyze real-world blockchain applications, benefits, risks, and regulatory challenges across multiple domains.

Course Outcomes:

After learning the course, the students will be able to:

CO1: Explain the **fundamental concepts of blockchain technology**, including evolution, Distributed Ledger Technology (DLT), key characteristics, and types of blockchain.

CO2: Analyze and compare different blockchain platforms

CO3: Describe the **blockchain architecture**, including layered structure, core components, nodes, mining, forks, and the role of the Ethereum Virtual Machine (EVM).

CO4: Develop a basic understanding of smart contracts and decentralized applications (DApps) using Solidity, including functions, data types, gas mechanism, and wallet interaction.

CO5: Evaluate the **benefits, limitations, risks, and real-world applications of blockchain**, including enterprise use cases, security concerns, and legal/regulatory challenges.

Unit	Contents	Duration (Hrs.)
1	**Introduction to Blockchain** Evolution of Blockchain Technology, Basics of Distributed Ledger Technology (DLT), Centralized vs Distributed systems, Characteristics of DLT: decentralization, transparency, immutability,Types of Distributed Ledgers, Key Concepts and Characteristics of Blockchain, Types of Blockchain, Blockchain vs Traditional Databases	7
2	**Blockchain Platforms:** Introduction to Blockchain Platforms, Overview of Major Blockchain Platforms (Ethereum, Hyperledger Fabric,Binance Smart Chain,Corda (basic idea)), Ethereum Platform,Hyperledger Fabric,Comparison of Blockchain Platforms, Selection Criteria for Blockchain Platforms	8
3	**Blockchain Architecture :** Overview of how blockchain systems are structured, Layered Architecture of Blockchain,Core Components of Blockchain Architecture,Genesis Block and	7

	Chain Formation, Nodes and Mining, Forks in Blockchain, Ethereum Virtual Machine (EVM) (Overview)	
4	**Smart Contracts & DApps** Introduction to Smart Contracts,Advantages & Limitations of Smart Contracts,Ethereum & Solidity Programming, Ethereum Basics, Introduction to Solidity Programming, Smart Contract Functions & Data Types,Decentralized Applications (DApps),Popular DApps & Their Use Cases,Gas Fees and Transaction:Understanding Gas in Ethereum, Use of wallet.	9
5	**Merits and Demerits of Blockchain and Applications** Selection Criteria for Blockchain platform for Applications, Blockchain and Enterprise – A Technology of Coordination, Risks and Limitations of Blockchain: Privacy, Security Risks of Blockchain, The "Evil Sides" of Blockchain and Legal Regulations for Blockchain: Ransomware, Money Laundering. Benefits of Blockchain in various scenarios.	8
	Total Hours	**39**

Text Books

1. Mastering Blockchain - Third Edition - Imran Bashir (Author, Contributor)
2. Cryptography and Network Security - Principles and Practice | Seventh Edition | By Pearson- Stallings William (Author)
3. Blockchain Revolution: How the Technology Behind Bitcoin and Other Cryptocurrencies is Changing the World-by Don Tapscott

Reference Books

1. Blockchain Revolution: How the Technology Behind Bitcoin and Other Cryptocurrencies is Changing the World-by Don Tapscott
2. Mastering Ethereum: Building Smart Contracts and DApps-by Andreas Antonopoulos
3. Blockchain Applications: A Hands-On Approach-by Arshdeep Bagha

Online References

1. Ethereum Docs – For Solidity & Smart Contract Development
2. Bitcoin Whitepaper – Original research paper by Satoshi Nakamoto
3. Hyperledger Fabric Documentation – For Private Blockchain Networks

Second Year B.Tech Information Technology															
Semester- IV															
Course Code: IT24HON292R								**Course Name: Blockchain Laboratory**							
Teaching Scheme (Hours/Week)					**Examination Scheme**						**Credits**				
L	**P**	**T**	**OL**	**ODL**	**CIE**	**ETE**	**TW**	**OR**	**PR**	**TOTAL**	**L**	**P**	**T**	**ODL**	**TOTAL**
-	4	-	-	-	-	-	25	-	25	50	-	2	-	-	2

Course Objectives:

- To introduce the fundamental concepts of **blockchain technology** and **distributed ledger systems**.
- To understand the **architecture, components, and working principles** of blockchain networks.
- To explore major blockchain platforms such as Ethereum and Hyperledger Fabric.
- To develop and deploy **smart contracts using Solidity** and understand their role in decentralized applications.
- To analyze **real-world applications, benefits, limitations, and security aspects** of blockchain technology.

Course Outcomes:

After learning the course, the students will be able to:

CO1: Understand the basic concepts of blockchain, distributed ledger technology, and blockchain architecture.

CO2: Explain different blockchain types and block structure used in blockchain networks.

CO3: Analyze the working and features of blockchain platforms such as Ethereum and Hyperledger Fabric.

CO4: Design and implement smart contracts using Solidity and deploy them on blockchain networks.

CO5: Develop simple **decentralized applications (DApps)** and evaluate blockchain applications in real-world scenarios considering security and regulatory issues.

Sr. No	List of Experiments	Duration (Hrs.)
1	Installation and setup of Blockchain development environment using Node.js, Ganache and MetaMask wallet.	4
2	Study of blockchain structure and implementation of a simple blockchain with blocks, hashing and linking using programming language.	4
3	Study of layered architecture of blockchain and identification of different layers such as data, network, consensus and application layers.	4
4	Understanding nodes and mining by creating accounts and transactions using Ganache local blockchain.	4
5	Study and demonstration of smart contract execution using Remix IDE and observing gas usage and transaction logs.	4
6	Demonstration of asset transfer application using Hyperledger Fabric test network.	4

7	Writing and deploying a basic smart contract using Solidity in Remix IDE	4
8	Implementation of smart contracts with functions, data types and arrays using Solidity.	4
9	Development of a simple decentralized application (DApp) interacting with a smart contract using MetaMask wallet.	4
10	Mini Project on Block chain	12
	Total Hours	**52**

Text Books

1. Mastering Blockchain - Third Edition - Imran Bashir (Author, Contributor)
2. Cryptography and Network Security - Principles and Practice | Seventh Edition | By Pearson- Stallings William (Author)
3. Blockchain Revolution: How the Technology Behind Bitcoin and Other Cryptocurrencies is Changing the World-by Don Tapscott

Reference Books

1. Blockchain Revolution: How the Technology Behind Bitcoin and Other Cryptocurrencies is Changing the World-by Don Tapscott
2. Mastering Ethereum: Building Smart Contracts and DApps-by Andreas Antonopoulos
3. Blockchain Applications: A Hands-On Approach-by Arshdeep Bagha

Online References

1. Ethereum Docs – For Solidity & Smart Contract Development
2. Bitcoin Whitepaper – Original research paper by Satoshi Nakamoto
3. Hyperledger Fabric Documentation – For Private Blockchain Networks

Third Year B.Tech Information Technology														
Semester- V														
Course Code: IT24HON341R							**Course Name: Blockchain Architecture and Design**							
Teaching Scheme (Hours/Week)					**Examination Scheme**						**Credits**			
L	**P**	**T**	**OL**	**ODL**	**CIE**	**ETE**	**TW**	**OR**	**PR**	**TOTAL**	**L**	**P**	**T**	**ODL**	**TOTAL**
3	-	-	-	-	40	60	-	-	-	100	3	-	-	-	3

Course Objectives:

- To understand the architecture and core components of blockchain systems.
- To study the design principles of decentralized networks and distributed ledgers.
- To explore consensus mechanisms and security aspects in blockchain.
- To analyze the design of smart contracts and blockchain applications.

Course Outcomes:

After learning the course, the students will be able to:

CO1: Explain the fundamentals of blockchain networks, including their architecture, components, and working principles.

CO2: Analyze blockchain network operations such as **transaction lifecycle, mining process, consensus mechanisms (PoW, PoS, DPoS, BFT), and fork management**.

CO3: Analyze the role of miners and consensus mechanisms in maintaining trust and security in distributed networks.

CO4:Develop and evaluate basic blockchain systems using **Python and JavaScript**, including block creation, hashing, proof-of-work, and chain validation.

CO5: Assess security and scalability challenges in blockchain systems, including attacks (double spending, 51%), and explore solutions such as Layer 2 technologies.

Unit	Contents	Duration (Hrs.)
1	**Fundamentals of Blockchain Networks** Fundamentals of Blockchain Networks :Definition and working of blockchain networks,Peer-to-peer (P2P) network architecture, Nodes and Mining , Understanding Nodes in a Blockchain Network,Mining in Blockchain, Forks in Blockchain,Notable Forks in Blockchain History,Fork Governance & Impact on Blockchain Networks, Ethereum Virtual Machine.	7
2	**Blockchain Network Components** Nodes (Full node, Light node, Miner node),Transactions and transaction lifecycle,Blocks and chain formation,Cryptographic hashing (conceptual), Types of Blockchain Networks,Public blockchain,Private blockchain,Consortium blockchain,Hybrid blockchain,Working of Blockchain Network,Step-by-step transaction flow,Block validation and addition to chain,Role of nodes in maintaining the ledger	8
3	**Mining in Blockchain** **Purpose of Mining, Miners , working of mining ,consensus mechanism used in mining,** Need for Consensus in Distributed Systems, Proof of Work (PoW), Proof of Stake (PoS), Delegated Proof of Stake (DPoS), Byzantine Fault Tolerance (BFT).**Advantages and limitations of Mining**	8

4	**Blockchain Implementation and Language Comparison** Basic blockchain simulation in Python, Creating blocks and chain structure, Implementing hashing using libraries (hashlib) Simple proof-of-work implementation, Verifying chain integrity **Blockchain Implementation using JavaScript:**Blockchain development using JavaScript (Node.js), Creating block and blockchain classes, Implementing hashing (crypto module), Simulating transactions and mining, Basic API interaction, Comparison: Python vs JavaScript vs Solidity, Integration Perspective in real DApp development.	8
5	**Security and Scalability** Security Issues in Blockchain, Double Spending Problem, 51% Attack, Blockchain Scalability Challenges, Layer 2 Solutions and Future Trends.	8
	Total Hours	**39**

Text Books

1. Mastering Blockchain - Third Edition - Imran Bashir (Author, Contributor)
2. Cryptography and Network Security - Principles and Practice | Seventh Edition | By Pearson-Stallings William (Author)
3. Blockchain Revolution: How the Technology Behind Bitcoin and Other Cryptocurrencies is Changing the World-by Don Tapscott

Reference Books

1. Blockchain Revolution: How the Technology Behind Bitcoin and Other Cryptocurrencies is Changing the World-by Don Tapscott
2. Mastering Ethereum: Building Smart Contracts and DApps-by Andreas Antonopoulos
3. Blockchain Applications: A Hands-On Approach-by Arshdeep Bagha
4. *Blockchain Basics: A Non-Technical Introduction in 25 Steps* – **Daniel Drescher**.

Online References

1. Ethereum Docs – For Solidity & Smart Contract Development
2. Bitcoin Whitepaper – Original research paper by Satoshi Nakamoto
3. Hyperledger Fabric Documentation – For Private Blockchain Networks
4. Solidity Documentation – https://docs.soliditylang.org/
5. IBM Blockchain Resources – https://www.ibm.com/blockchain/resources

Third Year B.Tech Information Technology

Semester- V

Course Code: IT24HON342R **Course Name: Blockchain Development Laboratory**

Teaching Scheme (Hours/Week)					Examination Scheme						Credits				
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	ODL	TOTAL
-	4	-	-	-	-	-	25	-	25	50	-	2	-	-	2

Course Objectives:

- To provide hands-on experience in blockchain development.
- To implement blockchain concepts using development tools.
- To develop and deploy smart contracts.
- To build simple decentralized applications (DApps).

Course Outcomes:

After learning the course, the students will be able to:

CO1: Install and configure blockchain development environments.

CO2: Implement blockchain transactions and wallet operations.

CO3: Develop and deploy smart contracts.

CO4: Build simple blockchain-based applications.

Sr. No	List of Experiments	Duration (Hrs.)
1	Implementation of smart contract for voting system	6
2	Development of blockchain-based asset transfer application	6
3	Implementation of blockchain-based document verification system	6
4	Implementation of basic blockchain structure using Python	6
5	Blockchain Implementation using JavaScript (Node.js)	6
6	To develop a web interface that interacts with a deployed smart contract using Web3.js and a wallet.	6
7	To create a private blockchain network with your own cryptocurrency.	6
8	Mini Project on Blockchain-Based Asset Transfer Application	10
	Total Hours	**52**

Text Books

1. Mastering Blockchain - Third Edition - Imran Bashir (Author, Contributor)
2. Cryptography and Network Security - Principles and Practice | Seventh Edition | By Pearson-Stallings William (Author)
3. Blockchain Revolution: How the Technology Behind Bitcoin and Other Cryptocurrencies is Changing the World-by Don Tapscott

Reference Books

1. Blockchain Revolution: How the Technology Behind Bitcoin and Other Cryptocurrencies is Changing the World-by Don Tapscott
2. Mastering Ethereum: Building Smart Contracts and DApps-by Andreas Antonopoulos
3. Blockchain Applications: A Hands-On Approach-by Arshdeep Bagha
4. Blockchain Basics: A Non-Technical Introduction in 25 Steps – **Daniel Drescher**.

Online References

1. Ethereum Docs – For Solidity & Smart Contract Development
2. Bitcoin Whitepaper – Original research paper by Satoshi Nakamoto
3. Hyperledger Fabric Documentation – For Private Blockchain Networks
4. Solidity Documentation – https://docs.soliditylang.org/
5. IBM Blockchain Resources – https://www.ibm.com/blockchain/resources

Third Year B. Tech Information Technology

Semester VI

Course Code: IT24HON391R **Course Name: Blockchain Application Development and Security Engineering**

Teaching Scheme (Hours/Week)					Examination Scheme						Credits			
L	P	T	OL	ODL	CIE	ETE	TW	PR	OR	TOTAL	L	P	T	TOTAL
2	-	-	-	-	40	60	-	-	-	100	2	-	-	2

Course Objectives:

- To understand advanced Solidity concepts for smart contract development.
- To use tools like Hardhat, Truffle, and Ganache for development and testing.
- To design and develop DApps using Web3.js.
- To deploy and optimize smart contracts on Ethereum.
- To analyze security and apply auditing techniques in blockchain systems.

Course Outcomes:

After completing the course, the students will be able to:

CO1: Apply advanced Solidity programming concepts including data structures, inheritance, modifiers, and design patterns to develop efficient smart contracts.

CO2: Use blockchain development frameworks such as Hardhat, Truffle, and Ganache to develop, test, and deploy smart contracts in a local blockchain environment.

CO3: Design and implement decentralized applications (DApps) using Web3 technologies and integrate smart contracts with frontend frameworks and wallet services.

CO4: Deploy smart contracts on Ethereum test networks, interact with blockchain applications using APIs, and apply techniques for optimizing gas usage and upgrading contracts.

CO5: Analyze security mechanisms in blockchain applications, identify smart contract vulnerabilities, and apply auditing and optimization techniques to ensure secure blockchain systems.

Unit	Contents	Hrs.
1	**Advanced Solidity Programming** Advanced data structures in Solidity, Structs, mappings and modifiers, Events and error handling, Inheritance and libraries in Solidity, Smart contract design patterns.	6
2	**Blockchain Development Frameworks** Hardhat framework, Truffle framework, Ganache local blockchain,Study of Blockchain Development Tools, Testing smart contracts, Comparative.	6
3	**Decentralized Application (DApp) Architecture and Web 3** Architecture of decentralized applications, Components of DApps (frontend, smart contract, blockchain network), Interaction between frontend and smart contracts, Web3 Technology Stack, Integration with frontend frameworks (React / Node.js), Wallet Integration.	6
4	**Deployment and Optimization** Deploying contracts on Ethereum test networks, Gas optimization techniques, Upgradable smart contracts, Interacting with smart contracts using APIs.	6
5	**Security Mechanisms in Blockchain Applications and Optimization** Introduction to Smart Contract Security, Common Smart Contract Vulnerabilities, Reentrancy Attacks and Prevention Techniques, Integer Overflow and Security Risks, Smart Contract Auditing Basics.	6
	Total Hours	**30**

Note for Course Coordinator

The End Term Examination (ETE) of 60 marks will be conducted at the department level. The mode, tools, and methodology for conducting the examination will be decided by the Course Coordinator, with due approval from the Chairperson of the Board of Studies (BoS).

Text Books

1. Mastering Blockchain - Third Edition - Imran Bashir (Author, Contributor)
2. Blockchain Basics: A Non-Technical Introduction in 25 Steps – Daniel Drescher
3. Blockchain Revolution: How the Technology Behind Bitcoin and Other Cryptocurrencies is Changing the World – Don Tapscott & Alex Tapscott

Reference Book

1. Bitcoin and Cryptocurrency Technologies – Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller, Steven Goldfeder
2. Blockchain and Distributed Ledger Technology Use Cases: Applications and Industry Adoption – Horst Treiblmaier & Roman Beck
3. Building Blockchain Projects – Narayan Prusty
4. Mastering Ethereum – Andreas M. Antonopoulos & Gavin Wood
5. Handbook of Blockchain, Digital Finance, and Inclusion – David Lee Kuo Chuen & Robert Deng
6. The Basics of Bitcoins and Blockchains – Antony Lewis
7. Hyperledger Fabric for Blockchain Applications – Huaqiang Yuan

Online References

1. Blockchain 101 (IBM Developer Guide)
2. Ethereum.org – Introduction to Blockchain
3. **Cryptographic Hash Functions (SHA-256, Keccak-256) – University of Waterloo**
4. Blockchain Security (OWASP Guide)
5. Hyperledger Fabric Documentation
6. Quorum Blockchain (Enterprise Ethereum)

Third Year B. Tech Information Technology															
Semester- VI															
Course Code: IT24HON392R							**Course Name: Blockchain Application Development and Security Engineering Laboratory**								
Teaching Scheme (Hours/Week)					**Examination Scheme**						**Credits**				
L	**P**	**T**	**OL**	**ODL**	**CIE**	**ETE**	**TW**	**OR**	**PR**	**TOTAL**	**L**	**P**	**T**	**ODL**	**TOTAL**
-	2	-	-	-	-	-	25	-	-	25	-	1	-	-	1

Course Objectives:

- To provide hands-on experience in developing **smart contracts using Solidity**.
- To familiarize students with **blockchain development tools and frameworks** such as Hardhat, Truffle, and Ganache.
- To enable students to **integrate smart contracts with decentralized applications (DApps)** using Web3.js or Ethers.js.
- To develop skills in **deploying and interacting with smart contracts on Ethereum test networks** using MetaMask.

Course Outcomes:

After completing the course, the students will be able to:

CO1: Apply advanced Solidity programming concepts including data structures, inheritance, modifiers, and design patterns to develop efficient smart contracts.

CO2: Use blockchain development frameworks such as Hardhat, Truffle, and Ganache to develop, test, and deploy smart contracts in a local blockchain environment.

CO3: Design and implement decentralized applications (DApps) using Web3 technologies and integrate smart contracts with frontend frameworks and wallet services.

CO4: Deploy smart contracts on Ethereum test networks, interact with blockchain applications using APIs, and apply techniques for optimizing gas usage and upgrading contracts.

CO5: Analyze security mechanisms in blockchain applications, identify smart contract vulnerabilities, and apply auditing and optimization techniques to ensure secure blockchain systems.

Sr. No	List of Experiments	Duration (Hrs.)
1	Smart Contract Development using Solidity Write and deploy a basic smart contract using Solidity that demonstrates the use of structs, mappings, events, and modifiers using Remix IDE.	6
2	Blockchain Development Environment Setup Install and configure Hardhat / Truffle development framework and create a local blockchain network using Ganache to compile and test a smart contract.	6
3	Decentralized Application (DApp) Integration Develop a simple DApp interface that connects a smart contract with a frontend application and interacts with it using Web3.js or Ethers.js.	6
4	Smart Contract Deployment Deploy a smart contract on an Ethereum test network (e.g., Sepolia / Goerli) and interact with it through MetaMask wallet.	8
	Total Hours	**26**

Text Books

1. Mastering Blockchain - Third Edition - Imran Bashir (Author, Contributor)

2. Blockchain Basics: A Non-Technical Introduction in 25 Steps – Daniel Drescher

3. Blockchain Revolution: How the Technology Behind Bitcoin and Other Cryptocurrencies is Changing the World – Don Tapscott & Alex Tapscott

Reference Books

1. Bitcoin and Cryptocurrency Technologies – Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller, Steven Goldfeder
2. Blockchain and Distributed Ledger Technology Use Cases: Applications and Industry Adoption – Horst Treiblmaier & Roman Beck
3. Building Blockchain Projects – Narayan Prusty
4. Mastering Ethereum – Andreas M. Antonopoulos & Gavin Wood
5. Handbook of Blockchain, Digital Finance, and Inclusion – David Lee Kuo Chuen & Robert Deng
6. The Basics of Bitcoins and Blockchains – Antony Lewis
7. Hyperledger Fabric for Blockchain Applications – Huaqiang Yuan

Online References

1. Blockchain 101 (IBM Developer Guide)
2. Ethereum.org – Introduction to Blockchain
3. Cryptographic Hash Functions (SHA-256, Keccak-256) – University of Waterloo
4. Blockchain Security (OWASP Guide)
5. Hyperledger Fabric Documentation
6. Quorum Blockchain (Enterprise Ethereum)

Final Year B.Tech Computer Engineering															
Semester- VII															
Course Code: IT24HON441R							**Course Name: Blockchain Applications and Emerging Technologies**								
Teaching Scheme (Hours/Week)					**Examination Scheme**						**Credits**				
L	**P**	**T**	**OL**	**ODL**	**CIE**	**ETE**	**TW**	**OR**	**PR**	**TOTAL**	**L**	**P**	**T**	**ODL**	**TOTAL**
2	-	-	-	-	40	60	-	-	-	100	2	-	-	-	**2**

Course Objectives:

- Understand practical applications of blockchain across industries.
- Analyze real-world blockchain implementation case studies.
- Evaluate blockchain platforms and enterprise solutions.
- Identify challenges, scalability issues, and regulatory aspects.
- Develop ability to design blockchain-based solutions for industry problems.

Course Outcomes:

After learning the course, the students will be able to:

CO1: Explain the integration of blockchain technology with emerging technologies such as Internet of Things (IoT), Artificial Intelligence, cloud computing, and decentralized storage systems.

CO2: Analyze the role of blockchain in financial services (FinTech) including cryptocurrency, decentralized finance (DeFi), digital payments, asset tokenization, and central bank digital currencies.

CO3: Evaluate blockchain applications in supply chain and logistics for improving transparency, traceability, and automation using smart contracts.

CO4: Assess the use of blockchain in government services, healthcare systems, and digital identity management with emphasis on security, privacy, and trust.

CO5: Examine emerging blockchain applications and future trends such as NFTs, Web3 ecosystem, metaverse, decentralized organizations (DAO), scalability solutions, and regulatory challenges.

Unit	Contents	Duration (Hrs.)
1	**Blockchain Integration with Emerging Technologies** Blockchain with Internet of Things (IoT), Blockchain with Artificial Intelligence, Blockchain and Cloud Computing, Decentralized storage systems (IPFS)	6
2	**Blockchain in Financial Services (FinTech)** Cryptocurrency and digital payments, Decentralized Finance (DeFi), Cross-border payments using blockchain, Tokenization of assets, Central Bank Digital Currency (CBDC) **Industry Case Studies:** Ripple for international payments, JPMorgan Onyx blockchain platform, India's digital rupee initiative, DeFi platforms (Uniswap, Aave)	6
3	**Blockchain in Supply Chain and Logistics** Supply chain transparency and traceability, Provenance tracking using blockchain, Smart contracts for logistics automation, Integration with IoT in supply chain **Industry Case Studies:** Walmart food traceability using blockchain, IBM Food Trust, Maersk TradeLens platform, Pharmaceutical drug tracking	6
4	**Blockchain in Government, Healthcare and Digital Identity**	6

	Digital identity management, Blockchain in e-governance, Healthcare record management, Blockchain for voting systems, Data security and privacy **Industry Case Studies:** Estonia e-Governance blockchain system, Aadhaar + blockchain possibilities in India, Medicalchain healthcare platform, Blockchain voting pilots	
5	**Emerging Blockchain Applications and Future Trends** Blockchain in NFTs and digital assets, Blockchain in gaming and metaverse, Blockchain in education and certificates, Web3 ecosystem and decentralized internet, Scalability solutions (Layer 2, Polygon, rollups), Regulatory and legal challenges **Industry Case Studies:** OpenSea NFT marketplace, Polygon network ecosystem, Blockchain based academic certificates, Decentralized Autonomous Organizations (DAO)	6
	Total Hours	**30**

Note for Course Coordinator

The End Term Examination (ETE) of 60 marks will be conducted at the department level. The mode, tools, and methodology for conducting the examination will be decided by the Course Coordinator, with due approval from the Chairperson of the Board of Studies (BoS).

Text Books

1. Mastering Blockchain - Third Edition - Imran Bashir (Author, Contributor)
2. Blockchain Basics: A Non-Technical Introduction in 25 Steps – Daniel Drescher
3. Blockchain Revolution: How the Technology Behind Bitcoin and Other Cryptocurrencies is Changing the World – Don Tapscott & Alex Tapscott

Reference Books

1. Bitcoin and Cryptocurrency Technologies – Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller, Steven Goldfeder
2. Blockchain and Distributed Ledger Technology Use Cases: Applications and Industry Adoption – Horst Treiblmaier & Roman Beck
3. Building Blockchain Projects – Narayan Prusty
4. Mastering Ethereum – Andreas M. Antonopoulos & Gavin Wood
5. Handbook of Blockchain, Digital Finance, and Inclusion – David Lee Kuo Chuen & Robert Deng
6. The Basics of Bitcoins and Blockchains – Antony Lewis
7. Hyperledger Fabric for Blockchain Applications – Huaqiang Yuan

Online References

1. Blockchain 101 (IBM Developer Guide)
2. Ethereum.org – Introduction to Blockchain
3. Cryptographic Hash Functions (SHA-256, Keccak-256) – University of Waterloo
4. Blockchain Security (OWASP Guide)
5. Hyperledger Fabric Documentation
6. Quorum Blockchain (Enterprise Ethereum)

<table>
<tr><th colspan="16">Final Year B.Tech Information Technology</th></tr>
<tr><td colspan="16">Semester- VII</td></tr>
<tr><td colspan="11">Course Code: IT24HON442R</td><td colspan="5">Course Name: Seminar</td></tr>
<tr><th colspan="5">Teaching Scheme (Hours/Week)</th><th colspan="6">Examination Scheme</th><th colspan="5">Credits</th></tr>
<tr><th>L</th><th>P</th><th>T</th><th>OL</th><th>ODL</th><th>CIE</th><th>ETE</th><th>TW</th><th>OR</th><th>PR</th><th>TOTAL</th><th>L</th><th>P</th><th>T</th><th>ODL</th><th>TOTAL</th></tr>
<tr><td>-</td><td>2</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>25</td><td>-</td><td>-</td><td>25</td><td>-</td><td>1</td><td>-</td><td>-</td><td>1</td></tr>
</table>

Course Objectives:

- To collect and study literature related to Blockchain Technology.
- To summarize the literature to understand current developments in blockchain.
- To identify research gaps and scope for future work in blockchain.
- To present ideas for a proposed blockchain-based project.
- To report the literature review and proposed work in a scientific manner.

Course Outcomes:

After learning the course, the students will be able to:

CO1: Understand the fundamental concepts and architecture of **Blockchain Technology**.
CO2: Explain the working principles of distributed ledger, cryptography, and consensus mechanisms.
CO3: Analyze various blockchain platforms and their applications.
CO4: Present and document blockchain-based ideas or projects effectively.

Guidelines for Blockchain Seminar Selection and Presentation

1. The student shall identify seminar topics related to Blockchain Technology and its applications by referring to recent technological trends and developments, in consultation with the industry (if applicable) and the institute guide.

2. The student must review sufficient literature such as reference books, journal articles, conference papers, white papers, technical magazines, and web resources related to blockchain technologies.

3. Seminar topics should focus on recent developments in blockchain, such as smart contracts, decentralized finance (DeFi), blockchain security, distributed ledger technology, NFTs, and blockchain-based applications. The guide should approve the topic after evaluating the relevance, research techniques used, comparative analysis of existing solutions, and tools used by researchers.

4. Research articles may be referred from digital libraries such as IEEE, ACM, ScienceDirect, Springer, Elsevier, IETE, CSI, or freely available digital libraries such as Digital Library of India, National Science Digital Library, CiteSeerX, ResearchGate, Open J-Gate, and WorldwideScience.

5. Each student shall present the seminar individually for 20–25 minutes in English, followed by a Question and Answer session.

6. The guide should ensure that the student performs proper literature survey and critical review related to blockchain technologies.

7. The guide should provide appropriate instructions for effective presentation and technical explanation of blockchain concepts.

8. Attendance of all other students in the class during seminar presentations is mandatory.

Suggested Timeline for Blockchain Seminar

1. Week 01: Introduction to technical papers related to blockchain and guidance on how and where to search research papers.
2. Week 02: Download at least four technical research papers on blockchain technology, get them approved by the guide, and prepare an abstract summary.
3. Week 03–04: Read and analyze the research papers in detail focusing on problem statement, blockchain techniques used, experimental results, and conclusions.
4. Week 05: Review of the studied papers by the Guide / Panel.
5. Week 06–07: Explore alternative blockchain techniques or improvements that can enhance performance, scalability, or security.
6. Week 08–09: Prepare seminar presentation including:
    - Topic and its significance
    - Research problem
    - Studied blockchain solutions
    - Strengths and weaknesses of each solution
    - Comparison of approaches
    - Future research directions
    - Probable project idea based on blockchain
    - Tentative project plan
7. Week 10: Prepare and write the Blockchain Seminar Report.
8. Week 11: Deliver the seminar presentation before the Guide / Panel.
9. Week 12: Verification and final submission of seminar report.

Guidelines for Blockchain Seminar Report

1. Each student shall submit two copies of the seminar report prepared using appropriate software tools as per the prescribed format, duly signed by the Guide and Head of the Department / Principal.

2. The seminar report (20–25 pages) should include:
    a. Title Page with topic title, student name, exam seat number / roll number, guide name, department name, institution, year, and university.
    b. Seminar Approval Sheet / Certificate
    c. Abstract and Keywords
    d. Acknowledgements
    e. Table of Contents, List of Figures, List of Tables, and Nomenclature
    f. Chapters covering:
        i. Introduction and organization of the report
        ii. Literature Survey related to blockchain technology
        iii. Motivation, purpose, scope, and objectives of the seminar
        iv. Details of blockchain architecture / technology / analysis (if applicable) Discussion and conclusions
        v. Bibliography / References (IEEE format)
        vi. Plagiarism Check Report
3. Students are encouraged to use open-source tools for report writing, reference citation, and plagiarism detection.

Final Year B.Tech Information Technology

Semester- VIII

Course Code: IT24HON491R **Course Name: Capstone Project**

Teaching Scheme (Hours/Week)					Examination Scheme						Credits				
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	ODL	TOTAL
-	8	-	-	-	-	-	25	100	-	125	-	4	-	-	4

Course Objectives:

- To apply blockchain knowledge to solve real-world problems.
- To develop skills in system design and implementation.
- To encourage research and innovation in blockchain applications.
- To improve project documentation and presentation skills.

Course Outcomes:

After learning the course, the students will be able to:

CO1: Identify and analyze real-world problems suitable for blockchain solutions.

CO2: Design and implement a blockchain-based application or system.

CO3: Apply appropriate tools and techniques for blockchain development.

CO4: Present and document the project work effectively.

Guidelines for Capstone Project Selection and Execution

1. Students shall identify a real-world problem or application related to Blockchain Technology in consultation with the project guide and industry experts (if available).
2. The project topic should focus on recent trends and applications in blockchain, such as decentralized finance (DeFi), smart contracts, cryptocurrency systems, blockchain security, supply chain management, healthcare data management, or digital identity systems.
3. Students must conduct a literature survey by referring to research papers, technical reports, white papers, and other scholarly resources to understand existing solutions and identify research gaps.
4. The project should involve design, development, implementation, and testing of a blockchain-based system or application.
5. Students are expected to follow systematic project development stages, including problem definition, requirement analysis, system design, implementation, testing, and documentation.
6. Students shall present the progress of the project periodically before the guide or review committee.
7. Students must demonstrate the working prototype or implementation during the final project evaluation.
8. Proper documentation and presentation of the project work is mandatory.

Suggested Timeline for Capstone Project

1. **Week 01–02:** Identification of project problem statement and approval from the guide.
2. **Week 03–04:** Literature survey and study of existing blockchain-based solutions.
3. **Week 05–06:** Requirement analysis and system design for the proposed solution.

4. **Week 07–08:** Development of blockchain architecture / smart contracts / application modules.
5. **Week 09–10:** Implementation and testing of the blockchain system.
6. **Week 11:** Preparation of project report and presentation slides.
7. **Week 12:** Final project demonstration and evaluation.

Guidelines for Capstone Project Report

1. Guidelines for Capstone Project Report
2. Each student shall submit the Capstone Project Report (30–40 pages) as per the prescribed format duly signed by the Guide and Head of the Department / Principal.
3. Contents of the Project Report
    a. Title Page with project title, student name, roll number, guide name, department, institution, year, and university.
    b. Project Approval Certificate
    c. Abstract and Keywords
    d. Acknowledgement
    e. Table of Contents, List of Figures, and List of Tables
    f. Chapters
        i. Introduction and problem statement
        ii. Literature Survey
        iii. System Design and Architecture
        iv. Implementation and Experimental Results
        v. Discussion and Conclusion
        vi. Future Scope
        vii. References (IEEE Format)
        viii. Plagiarism Check Report
4. Students are encouraged to use open-source tools for documentation, referencing, and plagiarism detection.

Guidelines for Assessment

The Capstone Project will be evaluated based on **Term Work (TW) and Oral/Practical (OR)**.

Term Work (TW) – 50 Marks

Criteria	Marks
Project topic relevance and problem definition	05
Literature review and technical depth	05
Project design and implementation progress	05
Project report documentation	05
Attendance and regular interaction with guide	05
Total	**25**

Oral / Practical Examination (OR) – 100 Marks

Criteria	Marks
Demonstration of working blockchain project	40
Understanding of blockchain concepts and implementation	25
Innovation and technical contribution	15
Presentation and communication skills	10
Question and answer session	10
Total	**100**

Marathwada Mitra Mandal’s College of Engineering

Karvenagar, Pune- 52

An Autonomous Institute Affiliated to SPPU

Curriculum Structure and Syllabus

Honours

A.Y. 2026-27

Accredited with A++ grade by NAAC
Recipient of Best College award by SPPU | Accredited by NBA
Recognized under 2(f) and 12(B) of UGC Act 1956
www.mmcoe.edu.in

About Marathwada Mitra Mandal's College of Engineering

MMCOE has been recognised as the BEST COLLEGE by SPPU. MMCOE is one feather in the cap of Marathwada Mitra Mandal's Pune, from the academic year 2006 – 2007.

The Institute offers six undergraduate engineering programmes in Electrical Engineering, Mechanical Engineering, Electronics & Telecommunication Engineering, Information Technology, Computer Engineering, and Artificial Intelligence & Data Science. In addition, it offers six postgraduate programmes M.Tech (Electrical – Power Electronics & Drives), M.Tech (Mechanical – Design Engineering), M.Tech (Electronics & Telecommunication – VLSI & Embedded Systems), M.Tech (Information Technology – Data Science), M.Tech (Computer Engineering), along with a Ph.D. programme in Computer Engineering. It houses 2000+ students and 200 staff members. Along with the focus on core academics, college also takes care of overall personality development of the students.

The college has an active robotics cell, NCC, Training and Placement cell etc. The students are motivated to actively participate in co-curricular and extra-curricular activities. They have brought many laurels to the Institute in the events such as Vinodottam Karandak, Firodiya Karandak, Zest, Dexterity etc.

Institute Vision

To be a globally renowned institution through excellence in engineering education for sustainable and holistic development

Institute Mission

M1: Empower students with cutting-edge technologies and global competencies

M2: Foster culture of research and entrepreneurial mindset

M3: Imbibe social and professional values

M4: Provide an inclusive environment for lifelong learning

Department of Information Technology

Vision and Mission of the Department

- **Vision**

To emerge as a globally competent center in education, research and innovation in Information Technology, fostering sustainable development with a holistic vision.

- **Mission**

M1 : To impart IT education that develops globally competent professionals

M2 : To promote innovation through research and entrepreneurship.

M3 : To encourage lifelong learning and a holistic commitment towards societal development.

- **Program Specific Outcomes (PSOs)**

Information Technology graduates will be able to :

  - Design and develop information systems using principles of computing to solve real-world problems.
  - Integrate emerging IT technologies to innovate and improve technological ecosystems.

Program Outcomes (POs)

PO1: Engineering Knowledge: Apply knowledge of mathematics, natural science, computing, engineering fundamentals and an engineering specialization as specified in WK1 to WK4 respectively to develop to the solution of complex engineering problems.

PO2: Problem Analysis: Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development. (WK1 to WK4)

PO3: Design/Development of Solutions: Design creative solutions for complex engineering problems and design/develop systems/components/processes to meet identified needs with consideration for the public health and safety, whole-life cost, net zero carbon, culture, society and environment as required. (WK5)

PO4: Conduct Investigations of Complex Problems: Conduct investigations of complex engineering problems using research-based knowledge including design of experiments, modelling, analysis & interpretation of data to provide valid conclusions. (WK8).

PO5: Engineering Tool Usage: Create, select and apply appropriate techniques, resources and modern engineering & IT tools, including prediction and modelling recognizing their limitations to solve complex engineering problems. (WK2 and WK6)

PO6: The Engineer and The World: Analyze and evaluate societal and environmental aspects while solving complex engineering problems for its impact on sustainability with reference to economy, health, safety, legal framework, culture and environment. (WK1, WK5, and WK7).

PO7: Ethics: Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national & international laws. (WK9)

PO8: Individual and Collaborative Team work: Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams.

PO9: Communication: Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations considering cultural, language, and learning differences

PO10: Project Management and Finance: Apply knowledge and understanding of engineering management principles and economic decision-making and apply these to one's own work, as a member and leader in a team, and to manage projects and in multidisciplinary environments.

PO11: Life-Long Learning: Recognize the need for, and have the preparation and ability for i) independent and life-long learning ii) adaptability to new and emerging technologies and iii) critical thinking in the broadest context of technological change. (WK8)

Curriculum for B. Tech. in Information Technology with Honours in Internet of Things

Department of Information Technology

General Objectives and Scope:

The Honours Degree Program aims to provide students with advanced knowledge and deeper understanding in their chosen field of study. It focuses on developing critical thinking, analytical abilities, and research skills through specialized courses and practical learning. The program encourages independent learning and innovation, enabling students to gain expertise beyond the regular undergraduate curriculum.

The scope of the honours degree includes opportunities for research, specialization, and skill development. Students may undertake projects or dissertations that enhance their practical and analytical abilities. The program also prepares students for higher education, research careers, and better employment opportunities in industry and academia.

Why Internet of Things (IoT)?

The Honours in Internet of Things (IoT) program aims to provide students with advanced knowledge of connected devices, smart systems, and communication technologies used in modern digital environments. The objective of the program is to develop technical skills in sensor technologies, embedded systems, data communication, and IoT architectures. It also focuses on enhancing problem-solving, innovation, and analytical abilities so that students can design and develop intelligent IoT-based applications for real-world problems.

The scope of IoT honours includes applications in areas such as smart homes, healthcare, agriculture, transportation, and industrial automation. Through advanced coursework, practical projects, and research-oriented learning, students gain expertise in developing and managing IoT solutions. The program also prepares students for higher studies, research opportunities, and careers in emerging technology domains related to IoT and smart systems.

Prerequisites :

1. Digital Electronics
2. Programming Fundamentals (C / Python / Java)
3. Microprocessors and Microcontrollers
4. Embedded Systems
5. Digital Electronics
6. Wireless Communication / Sensor Networks

B. TECH HONORS IN INTERNET OF THINGS (Semester IV, V, VI, VII, VIII)

Curriculum Structure

Course Code	Course Name	Teaching Scheme (Hrs/week)				Examination Scheme						Credits				
		L	P	T	ODL	CIE	ETE	TW	PR	OR	Total	L	P	T	ODL	Total
Semester IV																
IT24HON293R	Foundations of IoT	3	-	-	-	40	60	-	-	-	100	3	-	-	-	3
IT24HON294R	IoT Systems Laboratory	-	4	-	-	-	-	25	-	25	50	-	2	-	-	2
Semester V																
IT24HON343R	IoT Architecture and Design	3	-	-	-	40	60	-	-	-	100	3	-	-	-	3
IT24HON344R	IoT Architecture and Design Laboratory	-	4	-	-	-	-	25	-	25	50	-	2	-	-	2
Semester VI																
IT24HON393R	IoT Communication Protocols	2	-	-	-	40	60	-	-	-	100	2	-	-	-	2
IT24HON394R	IoT Communication Protocols Laboratory	-	2	-	-	-	-	25	-	-	25	-	1	-	-	1
Semester VII																
IT24HON443R	IoT: Sensors, Embedded Interfaces and Secure Platforms	2	-	-	-	40	60	-	-	-	100	2	-	-	-	2
IT24HON444R	IoT: Sensors, Embedded Interfaces and Secure Platforms Laboratory	-	2	-	-	-	-	25	-	-	25	-	1	-	-	1
Semester VIII																
IT24HON492R	Capstone Project in IoT	-	8	-	-	-	-	25	-	100	125	-	4	-		4
Total		10	20	-	-	160	240	125	-	150	675	10	10	-	-	20

L- Lecture	P- Practical	T- Tutorial	CIE-Continuous Internal Evaluation
ETE- End Term Examination	TW- Term work	PR- Practical	OR- Oral
L : 1 Hr.= 1 credit	P: 2 Hr. = 1 Credit	T: 1 Hr. = 1 Credit	ODL-Open Distance Learning

Second Year B.Tech Information Technology														
Semester-IV														
Course Code: IT24HON293R								**Course Name: Foundations of IoT**						
Teaching Scheme (Hours/Week)					**Examination Scheme**						**Credits**			
L	**P**	**T**	**OL**	**ODL**	**CIE**	**ETE**	**TW**	**OR**	**PR**	**TOTAL**	**L**	**P**	**T**	**TOTAL**
3	-	-	-	-	40	60	-	-	-	100	3	-	-	3

Course Objectives:

- To introduce the fundamental concepts and architecture of IoT systems
- To understand IoT devices, sensors, and embedded platforms
- To study IoT communication models and protocols
- To explore data management and cloud integration for IoT
- To analyze security issues, challenges, and future trends in IoT

Course Outcomes:

After learning the course, the students will be able to:

CO1: Express the concepts, architecture, and applications of IoT

CO2: Identify and interface sensors, actuators, and IoT hardware platforms

CO3: Link IoT communication models, networking technologies, and protocols

CO4: Design basic IoT applications using cloud platforms and data analytics

CO5: Evaluate security issues, challenges, and emerging trends in IoT

Unit	Contents	Duration (Hrs.)
1	Definition, evolution, and characteristics of IoT, IoT ecosystem and stakeholders, IoT architecture: three-layer and five-layer models, IoT reference models and functional blocks, IoT enabling technologies, IoT applications	7
2	Sensors and actuators: types, working principles, and interfacing, Embedded systems concepts for IoT, Microcontrollers vs microprocessors Power management in IoT devices, IoT hardware platforms: Arduino, and Raspberry Pi, Device interfacing and firmware basics, Communication interfaces: UART, SPI, I2C.	8
3	IoT communication models, Networking fundamentals for IoT TCP/IP basics and IPv6, 6LoWPAN Wireless communication technologies: Wi-Fi, Bluetooth, Zigbee, LoRa, Cellular IoT IoT protocols: MQTT, CoAP, HTTP, AMQP	8
4	IoT data types and data acquisition,Data preprocessing and real-time data handling, Role of cloud computing in IoT, Edge and fog computing, IoT platforms and services, Device management and data storage, Data analytics and visualization, Application development using APIs.	8
5	IoT security requirements, Threats, vulnerabilities, and attack models Device authentication and authorization, Secure communication and encryption Privacy and ethical issues. Challenges: scalability, interoperability, power, reliability Future trends: AIoT, digital twins, 5G, smart environments	8
	Total Hours	**39**

Text Books

1. Arshdeep Bahga, Vijay Madisetti, "Internet of Things – A hands-on approach", Universities Press, 2015.
2. Olivier Hersent, David Boswarthick, Omar Elloumi, "The Internet of Things: Key Applications and Protocols", 2nd Edition,Wiley Publication, 2012.

Reference Books

1. Rajkumar Buyya, Amir Vahid Dastjerdi, "Internet of Things: Principles and Paradigms", 1st Edition, Elsevier, 2016.
2. David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Robert Barton, Jerome Henry, "IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things", 1st Edition, Cisco Press, 2017.
3. Brian Russell, Drew Van Duren, "Practical Internet of Things Security", 1st Edition, Packt Publishing, 2016.

Online References

1. NPTEL Course: Prof. Sudip Misra, IIT Kharagpur, NOC: "Introduction To Internet Of Things" (LINK).
2. NPTEL Course: Dr. Sanjoy Kumar Parida, Indian Institute of Technology Patna, NOC:"Components And Applications Of Internet Of Things" (LINK).

<table>
<tr><th colspan="15">Second Year B.Tech Information Technology</th></tr>
<tr><td colspan="15">Semester-IV</td></tr>
<tr><td colspan="7">Course Code: IT24HON294R</td><td colspan="8">Course Name: IoT Systems Laboratory</td></tr>
<tr><td colspan="5">Teaching Scheme (Hours/Week)</td><td colspan="6">Examination Scheme</td><td colspan="4">Credits</td></tr>
<tr><td>L</td><td>P</td><td>T</td><td>OL</td><td>ODL</td><td>CIE</td><td>ETE</td><td>TW</td><td>OR</td><td>PR</td><td>TOTAL</td><td>L</td><td>P</td><td>T</td><td>TOTAL</td></tr>
<tr><td>-</td><td>4</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>25</td><td>-</td><td>25</td><td>50</td><td>-</td><td>2</td><td>-</td><td>2</td></tr>
</table>

Course Objectives:

- To introduce the fundamental concepts and architecture of IoT systems
- To understand IoT devices, sensors, and embedded platforms
- To study IoT communication models and protocols
- To explore data management and cloud integration for IoT
- To analyze security issues, challenges, and future trends in IoT

Course Outcomes:

After learning the course, the students will be able to:

CO1: Express the concepts, architecture, and applications of IoT
CO2: Identify and interface sensors, actuators, and IoT hardware platforms
CO3: Link IoT communication models, networking technologies, and protocols
CO4: Design basic IoT applications using cloud platforms and data analytics
CO5: Evaluate security issues, challenges, and emerging trends in IoT

Sr. No.	List of Experiments	Duration (Hrs.)
1	Introduction to IoT Hardware Platform Study of Arduino / ESP32 / Raspberry Pi board Pin configuration, power supply, and IDE installation Uploading and executing a basic program	6
2	Digital Input and Output Operations Interfacing LED and push button Writing programs for digital read and write operations	6
3	Analog Sensor Interfacing temperature or light sensor Reading analog sensor data Calibration and interpretation of sensor values.	6
4	Actuator Control: Interfacing relay module / buzzer / DC motor Controlling actuators based on sensor input Basic automation logic	6
5	Wireless Communication using Wi-Fi / Bluetooth Connecting IoT device to Wi-Fi / BLE Sending and receiving data wirelessly Network configuration basics	6
6	IoT Data Communication using MQTT / HTTP Publishing sensor data using MQTT or HTTP protocol Subscribing to data from broker/server	6
7	Cloud Platform Integration Connecting IoT device to cloud platform Data upload and storage Visualization using dashboards	6

8	Mini Project – End-to-End IoT Application Design and implementation of a simple IoT system Sensor data acquisition, cloud upload, and visualization Optional automation or alert mechanism	10
	Total Hours	**52**

Text Books

1. Arshdeep Bahga, Vijay Madisetti, "Internet of Things – A hands-on approach", Universities Press, 2015.
2. Olivier Hersent, David Boswarthick, Omar Elloumi, "The Internet of Things: Key Applications and Protocols", 2nd Edition,Wiley Publication, 2012.

Reference Books

1. Rajkumar Buyya, Amir Vahid Dastjerdi, "Internet of Things: Principles and Paradigms", 1st Edition, Elsevier, 2016.
2. David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Robert Barton, Jerome Henry, "IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things", 1st Edition, Cisco Press, 2017.
3. Brian Russell, Drew Van Duren, "Practical Internet of Things Security", 1st Edition, Packt Publishing, 2016.

Online References

1. NPTEL Course: Prof. Sudip Misra, IIT Kharagpur, NOC: "Introduction To Internet Of Things" (LINK).
2. NPTEL Course: Dr. Sanjoy Kumar Parida, Indian Institute of Technology Patna, NOC:"Components And Applications Of Internet Of Things" (LINK).

Third Year B. Tech Information Technology

Semester :V

Course Code : IT24HON343R **Course Name: IoT Architecture and Design**

Teaching Scheme (Hours/Week)					Examination Scheme						Credits			
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	TOTAL
3	-	-	-	-	40	60	-	-	-	100	3	-	-	3

Course Objectives:

- To understand the architecture and design principles of Internet of Things systems.
- To study IoT devices, sensors, actuators, and embedded platforms used in IoT environments.
- To analyze communication models and network architecture used in IoT systems.
- To design IoT-based solutions for real-world applications.

Course Outcomes:

After completing the course, the students will be able to:

CO1: Explain the architecture and components of IoT systems.
CO2: Identify and analyze IoT devices, sensors, and embedded platforms.
CO3: Design IoT system architecture for various applications.
CO4: Evaluate IoT frameworks and technologies used in smart environments.

Unit	Contents	Duration (Hrs.)
1	**Introduction to IoT Architecture** Evolution of the Internet of Things, IoT concepts and characteristics, IoT architecture overview, IoT ecosystem and key components (devices, gateways, cloud platforms), enabling technologies for IoT, overview of IoT applications such as smart homes, smart cities, healthcare monitoring, agriculture, and industrial IoT.	7
2	**IoT Devices and Embedded Platforms** IoT hardware components and architecture, sensors and actuators with working principles and applications, microcontrollers and embedded systems in IoT, interfacing sensors with embedded platforms, overview of popular IoT development boards such as Arduino, Raspberry Pi, and ESP32, basic device configuration and data collection.	8
3	**IoT Communication Models** IoT communication models and architecture, device-to-device communication, device-to-cloud communication, device-to-gateway communication, IoT networking basics, wired and wireless communication technologies, introduction to IoT communication protocols and connectivity concepts.	8
4	**IoT System Architecture** Three-layer architecture of IoT (perception, network, application), five-layer architecture of IoT, role of edge computing and fog computing in IoT systems, IoT middleware and service platforms, cloud integration in IoT, data management and system interoperability.	8
5	**IoT Design Principles and Applications** IoT system design methodology, requirement analysis and system modeling, data acquisition and processing in IoT systems, IoT security and privacy considerations, performance and scalability issues, case studies of IoT applications in smart environments and industry.	8
	Total Hours	**39**

Text Books

1. Internet of Things: A Hands-On Approach – Arshdeep Bahga and Vijay Madisetti.
2. Designing the Internet of Things – Adrian McEwen and Hakim Cassimally.
3. Internet of Things Architecture and Design Principles – Raj Kamal.

Reference Books

1. Building the Internet of Things – Maciej Kranz.
2. The Internet of Things: Key Applications and Protocols – Olivier Hersent, David Boswarthick, Omar Elloumi.
3. Architecting the Internet of Things – Dieter Uckelmann, Mark Harrison, Florian Michahelles.

Online References

1. IoT Architecture Reference Model – IEEE
2. Cisco IoT Fundamentals Resources
3. Arduino Official Documentation – https://www.arduino.cc
4. Raspberry Pi Documentation – https://www.raspberrypi.org/documentation

Third Year B. Tech Information Technology														
Semester :V														
Course Code : IT24HON344R					**Course Name: IoT Architecture and Design Laboratory**									
Teaching Scheme (Hours/Week)					**Examination Scheme**						**Credits**			
L	**P**	**T**	**OL**	**ODL**	**CIE**	**ETE**	**TW**	**OR**	**PR**	**TOTAL**	**L**	**P**	**T**	**TOTAL**
-	4	-	-	-	-	-	25	-	25	50	-	2	-	2

Course Objectives:

- To provide hands-on experience with IoT hardware platforms and sensors.
- To understand sensor interfacing and data acquisition in IoT systems.
- To develop IoT applications using embedded platforms.
- To demonstrate data communication between IoT devices and cloud platforms.

Course Outcomes:

After completing the course, the students will be able to:

CO1: Configure IoT development boards and required software tools.

CO2: Interface sensors and actuators with IoT hardware platforms.

CO3: Develop IoT-based data acquisition and communication systems.

CO4: Implement simple IoT applications with cloud connectivity.

Sr. No.	List of Experiments	Duration (Hrs.)
1	Installation and setup of Arduino IDE / IoT development environment.	4
2	Study of **Arduino / ESP32 / Raspberry Pi hardware architecture**.	4
3	Interfacing **LED and push button with Arduino**.	4
4	Interfacing **temperature and humidity sensor (DHT11/DHT22)** with microcontroller.	4
5	Interfacing **light sensor (LDR) and displaying sensor values**.	4
6	Interfacing **ultrasonic sensor for distance measurement**.	4
7	Controlling **actuators such as buzzer or relay modules** using microcontrollers.	4
8	Implementation of **Wi-Fi communication using ESP32 or NodeMCU**.	6
9	Sending sensor data to the IoT **cloud platform (ThingSpeak / Blynk / Firebase)**.	6
10	Development of an **IoT application such as smart lighting or smart temperature monitoring system**	12
	Total Hours	**52**

Text Books

1. Internet of Things: A Hands-On Approach – Arshdeep Bahga and Vijay Madisetti.
2. Designing the Internet of Things – Adrian McEwen and Hakim Cassimally.
3. Internet of Things Architecture and Design Principles – Raj Kamal.

Reference Books

1. Building the Internet of Things – Maciej Kranz.
2. The Internet of Things: Key Applications and Protocols – Olivier Hersent, David Boswarthick, Omar Elloumi.
3. Architecting the Internet of Things – Dieter Uckelmann, Mark Harrison, Florian Michahelles.

Online References

1. IoT Architecture Reference Model – IEEE
2. Cisco IoT Fundamentals Resources
3. Arduino Official Documentation – https://www.arduino.cc
4. Raspberry Pi Documentation – https://www.raspberrypi.org/documentation

Third Year B. Tech Information Technology														
Semester : VI														
Course Code : IT24HON393R							**Course Name: IOT communication and protocol**							
Teaching Scheme (Hours/Week)					**Examination Scheme**						**Credits**			
L	**P**	**T**	**OL**	**ODL**	**CIE**	**ETE**	**TW**	**OR**	**PR**	**TOTAL**	**L**	**P**	**T**	**TOTAL**
2	-	-	-	-	40	60	-	-	-	100	2	-	-	2

Course Objectives:

- Introduce the concepts, characteristics, challenges, and technology trends of the Internet of Things and its business opportunities.
- Develop understanding of IoT system architecture, embedded systems relationships, sensors, actuators, and Wireless Sensor Networks used in IoT applications.
- Familiarize students with IoT networking technologies, communication protocols, and the role of 5G Technology in IoT connectivity.
- Provide knowledge of IoT system management including device management, network management, configuration, monitoring, and firmware updates.
- Introduce architecture, security challenges, and intelligent solutions for Industrial Internet of Things using machine learning and deep learning techniques.

Course Outcomes:

After completing the course, the students will be able to:

CO1:Describe the concepts, characteristics, challenges, and business perspectives of the Internet of Things

CO2: Apply the concepts of sensors, actuators, and Wireless Sensor Networks to understand IoT system design and embedded system integration.

CO3:Analyze IoT architectures, networking technologies, and communication protocols including application, transport, and network layer protocols used in IoT systems.

CO4: Apply IoT management techniques such as device management, configuration, monitoring, and firmware updates for efficient IoT system operation.

CO5:Analyze the architecture, security challenges, and intrusion detection mechanisms in Industrial Internet of Things systems using machine learning and deep learning methods.

Unit	Contents	Duration (Hrs.)
1	Introduction: IoT Technology trends and future opportunities, IoT and Business scope Evolution, Business perspectives, Embedded systems Relationships, Challenges of IoT, Characteristics of IoT, Sensors and Actuators in IoT enabling Industrial Automation, Wireless sensor Networks in IoT, Connecting all the things in Internet of things ,IoT M2M, Software Define Networking. IoT System Management is Essential.	6
2	IoT protocols Introduction IOT life cycle , Physical Design, IOT Conceptual architecture, IOT protocols, Levels of IOT, IOT networking Protocols , Networking standards and technologies in IOT	6
3	Introduction of 5G networks in IoT, IoT Networking consideration and Challenges, Business case for the IoT, Network optimization for IoT devices,Transport Layer protocols, Network Layer Protocols, IoT communication Challenges, Application Protocols for IoT.	6
4	IoT Management Protocols: Device management , Network management in IoT, Configuration and monitoring, Firmware updates and remote control	6

5	Architecture of IIOT Introduction, IIOT layered Architecture ,Three tier IIOT, Security in IIOT, Service based Frameworks, Solutions against Intrusions in IIOT, Machine learning based solutions, Deep Learning based solutions	6
	Total Hours	**30**

Note for Course Coordinator

The End Term Examination (ETE) of 60 marks will be conducted at the department level. The mode, tools, and methodology for conducting the examination will be decided by the Course Coordinator, with due approval from the Chairperson of the Board of Studies (BoS).

Text Books

1. Dr. Vijendra Pratap Singh, Mr. Neeraj Kumar.., "IoT Communication Protocols", ISBN: 978-81-961690-9-1,Deccan International Academic Publishers,2023.
2. Dilip Kumar, K R. Venugopal , "Internet of Things: Architectures, Protocols, and Applications"McgrawHill

Reference Books

1. Bernd Scholz-Reiter, Florian Michahelles, "Architecting the Internet of Things", ISBN 978-3-642-19156-5 e-ISBN 978-3-642-19157-2, Springer, 2016.
2. N. Ida, Sensors, Actuators and Their Interfaces, Scitech Publishers, 2014
3. Simone Cirani, Gianluigi Ferrari, Marco Picone, Luca Veltri, "Internet of Things: Architectures, Protocols and Standards", ISBN: 978-1-119-35967-8 , Wiley

Online References

1. https://onlinecourses.nptel.ac.in/noc19_cs65/preview
2. https://archive.nptel.ac.in/courses/106/105/106105166/
3. https://onlinecourses.nptel.ac.in/noc21_ee85/preview

Third Year B. Tech Information Technology														
Semester : VI														
Course Code : IT24HON394R					**Course Name: IOT communication and protocol Laboratory**									
Teaching Scheme (Hours/Week)					**Examination Scheme**						**Credits**			
L	**P**	**T**	**OL**	**ODL**	**CIE**	**ETE**	**TW**	**OR**	**PR**	**TOTAL**	**L**	**P**	**T**	**TOTAL**
-	2	-	-	-	-	-	25	-	-	25	-	1	-	1

Course Objectives:

- Introduce the concepts, characteristics, challenges, and technology trends of the Internet of Things and its business opportunities.
- Develop understanding of IoT system architecture, embedded systems relationships, sensors, actuators, and Wireless Sensor Networks used in IoT applications.
- Familiarize students with IoT networking technologies, communication protocols, and the role of 5G Technology in IoT connectivity.
- Provide knowledge of IoT system management including device management, network management, configuration, monitoring, and firmware updates.
- Introduce architecture, security challenges, and intelligent solutions for Industrial Internet of Things using machine learning and deep learning techniques.

Course Outcomes:

After completing the course, the students will be able to:

CO1:Describe the concepts, characteristics, challenges, and business perspectives of the Internet of Things

CO2: Apply the concepts of sensors, actuators, and Wireless Sensor Networks to understand IoT system design and embedded system integration.

CO3:Analyze IoT architectures, networking technologies, and communication protocols including application, transport, and network layer protocols used in IoT systems.

CO4: Apply IoT management techniques such as device management, configuration, monitoring, and firmware updates for efficient IoT system operation.

CO5:Analyze the architecture, security challenges, and intrusion detection mechanisms in Industrial Internet of Things systems using machine learning and deep learning methods.

Sr. No.	List of Experiments	Duration (Hrs.)
1	To study various IOT protocols- 6 LowPAN, IPv4/IPv6, Wifi, Bluetooth, MQTT.	2
2	Install and configure the IoT development environment using Arduino IDE and verify the connection between the computer and the IoT development board such as NodeMCU ESP8266 or Arduino Uno by uploading and executing a basic test program.	2
3	Develop and upload a program using Arduino IDE to control an LED connected to Arduino Uno. The LED should blink at a specified interval to demonstrate basic digital output operations	4
4	Interfacing of temperature sensor LM35 with Arduino	2
5	To develop an application to send and receive data with Arduino using HTTP request	2
6	To develop an application that measures the room temperature and posts the temperature value on the cloud platform	2
7	Develop an IoT application where sensor data is transmitted from the IoT device to a web server using the HTTP Protocol and verify the data transfer using a web interface or API response.	4

8	Capture and analyze IoT network traffic generated by IoT devices using Wireshark and identify the communication protocols and data packets used in the IoT network.	4
9	To develop an application based on sensors	4
	Total Hours	**26**

Text Books

1. Dr. Vijendra Pratap Singh, Mr. Neeraj Kumar.., "IoT Communication Protocols", ISBN: 978-81-961690-9-1,Deccan International Academic Publishers,2023.
2. Dilip Kumar, K R. Venugopal , "Internet of Things: Architectures, Protocols, and Applications"McgrawHill

Reference Books

1. Bernd Scholz-Reiter, Florian Michahelles, "Architecting the Internet of Things", ISBN 978-3-642-19156-5 e-ISBN 978-3-642-19157-2, Springer, 2016.
2. N. Ida, Sensors, Actuators and Their Interfaces, Scitech Publishers, 2014
3. Simone Cirani, Gianluigi Ferrari, Marco Picone, Luca Veltri, "Internet of Things: Architectures, Protocols and Standards", ISBN: 978-1-119-35967-8 , Wiley

Online References

1. https://onlinecourses.nptel.ac.in/noc19_cs65/preview
2. https://archive.nptel.ac.in/courses/106/105/106105166/
3. https://onlinecourses.nptel.ac.in/noc21_ee85/preview

Final Year B.Tech Information Technology

Semester-VII

Course Code: IT24HON443R **Course Name: IoT: Sensors, Embedded Interfaces and Secure Platforms**

Teaching Scheme (Hours/Week)					Examination Scheme						Credits			
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	TOTAL
2	-	-	-	-	40	60	-	-	-	100	2	-	-	2

Course Objectives:

- To introduce the architecture and design principles of Internet of Things systems.
- To study various sensors, actuators and embedded interfaces used in IoT devices.
- To understand communication technologies and IoT cloud platforms.
- To analyze security challenges and protection mechanisms in IoT ecosystems.
- To examine real-world IoT applications and case studies in different domains.

Course Outcomes:

After learning the course, the students will be able to:

CO1: Understand the architecture, components and enabling technologies of IoT systems.
CO2: Interface sensors and actuators with embedded platforms for IoT applications.
CO3: Analyze IoT communication protocols and cloud based IoT platforms.
CO4: Evaluate security, privacy and trust issues in IoT deployments.
CO5: Design IoT solutions for real-world applications through case studies.

Unit	Contents	Duration (Hrs.)
1	**Introduction to Internet of Things** Evolution of IoT, IoT ecosystem and architecture, IoT reference models, IoT layered architecture, characteristics of IoT systems, smart devices and IoT nodes, physical design of IoT, identification technologies (RFID, EPC), IoT communication models, overview of IoT application domains such as smart cities, smart homes, healthcare and industrial IoT.	6
2	**Sensors, Actuators and Embedded Interfaces** Fundamentals of sensors and transducers, sensor characteristics and calibration, classification of sensors (temperature, humidity, pressure, motion, light, gas, proximity). Actuators and control mechanisms. Embedded interfaces for IoT – microcontrollers and single board computers, GPIO interfaces, analog and digital interfacing, ADC/DAC concepts, sensor interfacing techniques, introduction to platforms such as Arduino and Raspberry Pi.	6
3	**IoT Communication and Networking** IoT networking concepts, wireless communication technologies (Wi-Fi, Bluetooth Low Energy, ZigBee, LoRaWAN), IPv6 for IoT, 6LoWPAN, IoT communication protocols: MQTT, CoAP, AMQP and HTTP. Edge computing and fog computing concepts, data acquisition and streaming from IoT devices.	6
4	**IoT Platforms and Data Management** Cloud computing for IoT, architecture of IoT platforms, device management and data ingestion. IoT cloud platforms such as AWS IoT, Google Cloud IoT, Microsoft Azure IoT Hub. Data storage, visualization and analytics in IoT systems. Integration of IoT with AI and big data.	6

5	IoT Security and Case Studies Security challenges in IoT, threat models, authentication and authorization mechanisms, encryption techniques, secure communication protocols, privacy issues in IoT networks, blockchain for IoT security. Case studies: Smart home automation, smart agriculture, industrial IoT (Industry 4.0), healthcare monitoring systems and smart transportation.	6
	Total Hours	**30**

Note for Course Coordinator

The End Term Examination (ETE) of 60 marks will be conducted at the department level. The mode, tools, and methodology for conducting the examination will be decided by the Course Coordinator, with due approval from the Chairperson of the Board of Studies (BoS).

Text Books

1. Arshdeep Bahga and Vijay Madisetti, "Internet of Things: A Hands-On Approach", Universities Press, 2015.
2. Raj Kamal, "Internet of Things: Architecture and Design Principles", McGraw Hill Education, 2017.
3. Pethuru Raj and Anupama C. Raman, "The Internet of Things: Enabling Technologies, Platforms and Use Cases", CRC Press, 2017.
4. R. K. Bansal, "Fluid Mechanics & Hydraulic Machines", Laxmi Publication, 2015.
5. Adrian McEwen and Hakim Cassimally, "Designing the Internet of Things", Wiley Publications, 2014.

Reference Books

1. Olivier Hersent, David Boswarthick and Omar Elloumi, "The Internet of Things: Key Applications and Protocols", Wiley Publications, 2012.
2. Samuel Greengard, "The Internet of Things", MIT Press, 2015.
3. David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Rob Barton and Jerome Henry, "IoT Fundamentals: Networking Technologies, Protocols and Use Cases for IoT", Cisco Press, 2017.
4. Vijay Madisetti and Arshdeep Bahga, "Internet of Things: Principles and Paradigms", CRC Press, 2016.

Online References

1. NPTEL Course: Pr NPTEL Course: Prof. Sudip Misra, IIT Kharagpur, "Internet of Things" (Link)
2. NPTEL Course: Prof. Arijit Ukil, IIT Kharagpur, "Embedded Systems Design" (Link)
3. NPTEL Course: Prof. Prabhat Ranjan, IIT Kanpur, "IoT and Applications" (Link)
4. SWAYAM Course: "Internet of Things – Smart Connected Devices" (Link)of. Anand T. N. C., IIT Palakkad, "Thermodynamics" (Link)
5. NPTEL Course: Prof. Suman Chakraborty, IIT Kharagpur, "Introduction To Fluid Mechanics" (Link)
6. NPTEL Course: Prof. Subashisa Dutta, IIT Guwahati, "Fluid Mechanics" (Link)

Final Year B.Tech Information Technology														
Semester-VII														
Course Code: IT24HON444R					**Course Name: IoT: Sensors and Embedded Interfaces Laboratory**									
Teaching Scheme (Hours/Week)					**Examination Scheme**						**Credits**			
L	**P**	**T**	**OL**	**ODL**	**CIE**	**ETE**	**TW**	**OR**	**PR**	**TOTAL**	**L**	**P**	**T**	**TOTAL**
-	2	-	-	-	-	-	25	-	-	25	-	1	-	1

Course Objectives:

- To introduce the architecture and design principles of Internet of Things systems.
- To study various sensors, actuators and embedded interfaces used in IoT devices.
- To understand communication technologies and IoT cloud platforms.
- To analyze security challenges and protection mechanisms in IoT ecosystems.
- To examine real-world IoT applications and case studies in different domains.

Course Outcomes:

After learning the course, the students will be able to:

CO1: Understand the architecture, components and enabling technologies of IoT systems.
CO2: Interface sensors and actuators with embedded platforms for IoT applications.
CO3: Analyze IoT communication protocols and cloud based IoT platforms.
CO4: Evaluate security, privacy and trust issues in IoT deployments.
CO5: Design IoT solutions for real-world applications through case studies.

Sr. No.	Name of the Experiment	Duration (Hrs.)
1	Study of IoT development boards and setup of Arduino / Raspberry Pi environment	2
2	Interfacing temperature and humidity sensor with microcontroller and displaying sensor data	2
3	Interfacing motion or proximity sensor and actuator control using GPIO	2
4	Interfacing analog sensors using ADC and monitoring data	4
5	Implement wireless communication between IoT devices using Wi-Fi / Bluetooth	2
6	Publish and subscribe sensor data using MQTT protocol	2
7	Connecting IoT devices to cloud platform for data logging and visualization	2
8	Implement smart home automation prototype using sensors and actuators	4
9	Study and implementation of IoT security mechanisms (authentication/encryption)	2
10	Mini IoT project: Design and demonstration of a real-world IoT application	4
	Total Hours	26

Text Books

1. P. K. Nag, “Engineering Thermodynamics”, Tata McGraw Hill Publications, 2017.
2. R. K. Rajput, “Engineering Thermodynamics”, EVSS Thermo, Laxmi Publications, 2017.
3. Cengel & Cimbla, “Fluid Mechanics”,TATA McGraw-Hill, 2019.
4. R. K. Bansal, ”Fluid Mechanics & Hydraulic Machines”, Laxmi Publication, 2015.

Reference Books

1. Cengel and Boles, “Thermodynamics an Engineering Approach”, McGraw Hill, 2017.
2. Kundu, Cohen, Dowling, “Fluid Mechanics”, Elsevier India, 2011.
3. Chaim Gutfinger David Pnueli, “Fluid Mechanics” Cambridge University press. 2003.

Online References

1. NPTEL Course: Prof. Anand T. N. C., IIT Palakkad, “Thermodynamics” (Link)
2. NPTEL Course: Prof. Suman Chakraborty, IIT Kharagpur, “Introduction To Fluid Mechanics” (Link)
3. NPTEL Course: Prof. Subashisa Dutta, IIT Guwahati, “Fluid Mechanics” (Link)

Final Year B. Tech Information Technology														
Semester- VIII														
Course Code: IT24HON492R								**Course Name: Capstone Project in IoT**						
Teaching Scheme (Hours/Week)					**Examination Scheme**						**Credits**			
L	**P**	**T**	**OL**	**ODL**	**CIE**	**ETE**	**TW**	**PR**	**OR**	**TOTAL**	**L**	**P**	**T**	**TOTAL**
-	8	-	-	-	-	-	25	-	100	125	-	4	-	4

Course Objectives:

- To assess IoT system architectures for real-world applications.
- To construct innovative IoT solutions for complex engineering problems.

Course Outcomes:

After completing the course, the students will be able to:

CO1: Evaluate suitable IoT architectures for real-world applications.

CO2: Create an innovative IoT based solution for a given problem.

Introductory Information:

Capstone Project in IoT is a practical, industry-oriented project in which students design and develop an end-to-end system by integrating IoT devices for data collection, processing and visualization. The project focuses on real-world problem solving using sensors, embedded controllers and communication protocols. It enables students to apply concepts such as device connectivity, data ingestion, analytics, security and dashboard development in a unified solution. The capstone project also helps students gain hands-on experience in system design, implementation, testing and deployment of smart and connected applications.

Guidelines to Faculty and Students:

1. The Head of the Department / Project Coordinator shall constitute a Project Review Committee for each project group. The project guide shall be a member of the review committee by default, and preferably the same committee shall continue for the entire semester.
2. In case of a sponsored project, an employee from the sponsoring organization may be included as a member of the review committee.
3. The Capstone Project shall be evaluated through three reviews conducted within the single semester by the review committee.
4. The Project Review Committee shall be responsible for evaluating the timely progress of the project.
5. Students shall identify and implement a project of sufficient complexity in IoT and Cloud Computing, having at least four to five major functionalities.

- Students shall apply the concepts learned in Software Engineering and Software Architecture to identify stakeholders, actors and suitable architectural styles, and shall prepare a clear and detailed problem statement for the proposed system.
- The scope of the project shall be finalized by the Review Committee during Review-I.
- If a change in project topic becomes unavoidable, students shall submit a revised synopsis along with a brief review of important related research papers, and the change shall be implemented only after approval of the Review Committee.
- During each review, every student of the project group shall present the individual and group progress before the committee. Each group/student shall give a 10 to 15 minute presentation, followed by a detailed discussion and question–answer session.

- Students shall maintain a record of queries, remarks and suggestions given by the review committee during each review and shall incorporate the necessary changes before the subsequent review.
- A proper project diary / review record containing remarks and suggestions of the review committee shall be maintained and shall be produced during the final evaluation.

- Since the project is of one semester duration, each project group shall prepare and submit one technical report related to the project.

  The report shall include problem definition, literature survey, system design, methodology, implementation, results, analysis and conclusion.

  The report shall be checked for plagiarism using any open plagiarism detection software

6. The final project report shall also be checked for plagiarism before submission.

7 . The students shall be assessed by a panel of examiners.. The assessment shall be based on the work carried out, technical depth of the IoT solution, presentation skills, documentation quality, responses to questions and the final report.

Review Structure (Three Reviews)

Review-I (Problem Definition & Design Review)

1. Finalization of problem statement and objectives
2. Literature survey and identification of gaps
3. System architecture and design
4. Finalization of project scope

Review-II (Implementation & Integration Review)

1. Progress of IoT hardware and device setup
2. Data flow, communication protocols and partial implementation
3. Demonstration of intermediate results

Review-III (Testing & Final Review)

1. Complete system demonstration
2. Testing and validation results
3. Performance and security considerations
4. Final documentation and readiness for evaluation

Evaluation Criteria:

Review	Evaluation Criteria	Marks
Review-I (Problem Definition & Design Review)	Originality and clarity of Problem Statement & Objectives	5
	Depth of understanding of the problem and project scope	5
	Literature Survey and identification of research gaps	5
	System Architecture / Design / Proposed Methodology	5
	Subtotal Review-I	**20**
Review-II (Implementati	Progress of IoT hardware/device setup and integration	5
	Data flow design, communication protocols and implementation progress	5

on & Integration Review)	Demonstration of intermediate results / prototype	5
	Subtotal Review-II	**15**
Review-III (Testing & Final Review)	Complete system demonstration and functionality	5
	Testing, validation and performance analysis	5
	Documentation, presentation and readiness for final evaluation	5
	Subtotal Review-III	**15**
Total		**50**

Marathwada Mitra Mandal's College of Engineering

Karvenagar, Pune- 52

An Autonomous Institute Affiliated to SPPU

"येथे बहुतांचे हित"

Curriculum Structure and Syllabus

Honours

A.Y. 2026-27

Accredited with A++ grade by NAAC
Recipient of Best College award by SPPU | Accredited by NBA
Recognized under 2(f) and 12(B) of UGC Act 1956
www.mmcoe.edu.in

About Marathwada Mitra Mandal's College of Engineering

MMCOE has been recognised as the BEST COLLEGE by SPPU. MMCOE is one feather in the cap of Marathwada Mitra Mandal's Pune, from the academic year 2006 – 2007.

The Institute offers six undergraduate engineering programmes in Electrical Engineering, Mechanical Engineering, Electronics & Telecommunication Engineering, Information Technology, Computer Engineering, and Artificial Intelligence & Data Science. In addition, it offers six postgraduate programmes M.Tech (Electrical – Power Electronics & Drives), M.Tech (Mechanical – Design Engineering), M.Tech (Electronics & Telecommunication – VLSI & Embedded Systems), M.Tech (Information Technology – Data Science), M.Tech (Computer Engineering), along with a Ph.D. programme in Computer Engineering. It houses 2000+ students and 200 staff members. Along with the focus on core academics, college also takes care of overall personality development of the students.

The college has an active robotics cell, NCC, Training and Placement cell etc. The students are motivated to actively participate in co-curricular and extra-curricular activities. They have brought many laurels to the Institute in the events such as Vinodottam Karandak, Firodiya Karandak, Zest, Dexterity etc.

Institute Vision

To be a globally renowned institution through excellence in engineering education for sustainable and holistic development

Institute Mission

M1: Empower students with cutting-edge technologies and global competencies

M2: Foster culture of research and entrepreneurial mindset

M3: Imbibe social and professional values

M4: Provide an inclusive environment for lifelong learning

Department of Information Technology

Vision and Mission of the Department

- **Vision**

  To emerge as a globally competent center in education, research and innovation in Information Technology, fostering sustainable development with a holistic vision.

- **Mission**

  M1 : To impart IT education that develops globally competent professionals

  M2 : To promote innovation through research and entrepreneurship.

  M3 : To encourage lifelong learning and a holistic commitment towards societal development.

- **Program Specific Outcomes (PSOs)**

  Information Technology graduates will be able to :

  - Design and develop information systems using principles of computing to solve real-world problems.
  - Integrate emerging IT technologies to innovate and improve technological ecosystems.

Program Outcomes (POs)

PO1: Engineering Knowledge: Apply knowledge of mathematics, natural science, computing, engineering fundamentals and an engineering specialization as specified in WK1 to WK4 respectively to develop to the solution of complex engineering problems.

PO2: Problem Analysis: Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development. (WK1 to WK4)

PO3: Design/Development of Solutions: Design creative solutions for complex engineering problems and design/develop systems/components/processes to meet identified needs with consideration for the public health and safety, whole-life cost, net zero carbon, culture, society and environment as required. (WK5)

PO4: Conduct Investigations of Complex Problems: Conduct investigations of complex engineering problems using research-based knowledge including design of experiments, modelling, analysis & interpretation of data to provide valid conclusions. (WK8).

PO5: Engineering Tool Usage: Create, select and apply appropriate techniques, resources and modern engineering & IT tools, including prediction and modelling recognizing their limitations to solve complex engineering problems. (WK2 and WK6)

PO6: The Engineer and The World: Analyze and evaluate societal and environmental aspects while solving complex engineering problems for its impact on sustainability with reference to economy, health, safety, legal framework, culture and environment. (WK1, WK5, and WK7).

PO7: Ethics: Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national & international laws. (WK9)

PO8: Individual and Collaborative Team work: Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams.

PO9: Communication: Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations considering cultural, language, and learning differences

PO10: Project Management and Finance: Apply knowledge and understanding of engineering management principles and economic decision-making and apply these to one's own work, as a member and leader in a team, and to manage projects and in multidisciplinary environments.

PO11: Life-Long Learning: Recognize the need for, and have the preparation and ability for i) independent and life-long learning ii) adaptability to new and emerging technologies and iii) critical thinking in the broadest context of technological change. (WK8)

Curriculum for B. Tech. in Information Technology with Honours in Game Design & Development

“ येथे बहुतांचे हित ”

Department of Information Technology

General Objectives and Scope:

The Honours Degree Program aims to provide students with advanced knowledge and deeper understanding in their chosen field of study. It focuses on developing critical thinking, analytical abilities, and research skills through specialized courses and practical learning. The program encourages independent learning and innovation, enabling students to gain expertise beyond the regular undergraduate curriculum.

The scope of the honours degree includes opportunities for research, specialization, and skill development. Students may undertake projects or dissertations that enhance their practical and analytical abilities. The program also prepares students for higher education, research careers, and better employment opportunities in industry and academia.

Why Game Design & Development?

Game Design & Development is a rapidly growing field that combines **creativity, technology, and storytelling** to build interactive digital experiences. With the expansion of the gaming industry across mobile, PC, console, and virtual reality platforms, there is a strong demand for skilled professionals who can design engaging gameplay, graphics, and immersive environments.

Learning Game Design & Development helps students develop skills in **programming, graphics, animation, problem-solving, and user experience design**. It also opens diverse career opportunities such as game developer, game designer, animator, and AR/VR developer, while encouraging innovation and creativity in building interactive applications beyond entertainment, including education and simulation systems.

Prerequisites :

1. Basic knowledge of computer fundamentals
2. Understanding of programming concepts (preferably C, C++, or Python)
3. Familiarity with mathematics and logic (especially algebra and basic geometry)
4. Basic understanding of graphics and animation concepts
5. Interest in gaming, creativity, and storytelling
6. Basic knowledge of software tools and operating systems

B. TECH HONORS IN GAME DESIGN & DEVELOPMENT (Semester V, VI, VII, VIII)

Curriculum Structure

Course Code	Course Name	Teaching Scheme (Hrs/week)				Examination Scheme						Credits				
		L	P	T	ODL	CIE	ETE	TW	PR	OR	Total	L	P	T	ODL	Total
Semester IV																
IT24HON295R	Foundations of Immersive Technologies & Game Design	3	-	-	-	40	60	-	-	-	100	3	-	-	-	**3**
IT24HON296R	Design Lab 1	-	4	-	-	-	-	25	-	25	50	-	2	-	-	**2**
Semester V																
IT24HON345R	Game Design Principles & Mechanics	3	-	-	-	40	60	-	-	-	100	3	-	-	-	**3**
IT24HON346R	Design Lab 2	-	4	-	-	-	-	25	-	25	50	-	2	-	-	**2**
Semester VI																
IT24HON395R	Unity Engine Fundamentals & Visual Scripting	3	-	-	-	40	60	-	-	-	100	3	-	-	-	**3**
IT24HON396R	Design Lab 3	-	2	-	-	-	-	25	-	-	25	-	1	-	-	**1**
Semester VII																
IT24HON445R	Game Programming in Unity with C# & XR Systems	2	-	-	-	40	60	-	-	-	100	2	-	-	-	**2**
IT24HON446R	Design Lab 4	-	2	-	-	-	-	25	-	-	25	-	1	-	-	**1**
Semester VIII																
IT24HON493R	Game Development Studio & Capstone Project	2	-	-	-	40	60	-	-	-	100	2	-	-	-	**2**
IT24HON494R	Design Lab 5	-	2	-	-	-	-	25	-	-	25	-	1	-	-	**1**
Total		**13**	**14**	**-**	**-**	**200**	**300**	**125**	**-**	**50**	**675**	**13**	**7**	**-**	**-**	**20**

L- Lecture	P- Practical	T- Tutorial	CIE-Continuous Internal Evaluation
ETE- End Term Examination	TW- Term work	PR- Practical	OR- Oral
L : 1 Hr.= 1 credit	P: 2 Hr. = 1 Credit	T: 1 Hr. = 1 Credit	ODL-Open Distance Learning

Second Year B.Tech Information Technology															
Semester- IV															
Course Code: IT24HON295R **Course Name: Foundations of Immersive Technologies & Game Design**															
Teaching Scheme (Hours/Week)					**Examination Scheme**						**Credits**				
L	**P**	**T**	**OL**	**ODL**	**CIE**	**ETE**	**TW**	**OR**	**PR**	**TOTAL**	**L**	**P**	**T**	**ODL**	**TOTAL**
3	-	-	-	-	40	60	-	-	-	100	3	-	-	-	3

Course Objectives:

- To introduce students to immersive technologies, games, and simulations as experiential digital systems
- To develop a strong foundation in game design principles and interactive experience design
- To enable students to analyze and design engaging game and immersive systems before implementation.

Course Outcomes:

After learning the course, the students will be able to:

CO1: Explain the fundamentals of immersive technologies, games, and simulation-based systems

CO2: Apply core game design principles such as mechanics, dynamics, and aesthetics

CO3: Analyze player experience, engagement, and interaction in games and immersive systems

CO4: Design conceptual games and immersive experience frameworks using human centered design principles

CO5: Evaluate ethical, accessibility, and societal considerations in immersive and game-based systems

Unit	Contents	Duration (Hrs.)
1	**Introduction to Immersive Technologies** Evolution of games, XR, and simulation systems. Games vs simulations vs interactive experiences. Applications in entertainment, education, and industry. Case studies of interactive systems and experiential technologies.	8
2	**Fundamentals of Game Design** Mechanics–Dynamics–Aesthetics (MDA framework). Core gameplay loops and progression systems. Player motivation and engagement models. Design exercises analyzing gameplay systems.	8
3	**Human-Centred Design & Player Experience** Player personas and player types. Usability, accessibility, and inclusivity in games. Emotional design and experiential storytelling. Analysis of player experience through design case studies.	8
4	**Interaction Design & Spatial Thinking** Interaction models in 2D and 3D environments. Input, feedback, and affordances. Camera systems, navigation, and spatial awareness. Design exercise for conceptual interactive environments.	7
5	**Introduction to Game Engines & Design Documentation** Overview of game development pipelines. Introduction to Unity engine architecture. Game Design Documents (GDD), design communication, and prototype validation methods.	8
	Total Hours	**39**

Note for Course Coordinator
The End Term Examination (ETE) of 60 marks will be conducted at the department level. The mode, tools, and methodology for conducting the examination will be decided by the Course Coordinator, with due approval from the Chairperson of the Board of Studies (BoS).
Text Books
1. Jesse Schell, "The Art of Game Design: A Book of Lenses", CRC Press, 2019. 2. Tracy Fullerton, "Game Design Workshop: A Playcentric Approach to Creating Innovative Games", CRC Press, 2018. 3. Ernest Adams, "Fundamentals of Game Design", New Riders, 2014.
Reference Book
1. Raph Koster, "A Theory of Fun for Game Design", O'Reilly Media, 2013. 2. Katie Salen and Eric Zimmerman, "Rules of Play: Game Design Fundamentals", MIT Press, 2004. 3. Steve Swink, "Game Feel: A Game Designer's Guide to Virtual Sensation", CRC Press, 2009.

Second Year B.Tech Information Technology

Semester-IV

Course Code: IT24HON296R **Course Name: Design Lab 1**

Teaching Scheme (Hours/Week)					Examination Scheme						Credits			
L	**P**	**T**	**OL**	**ODL**	**CIE**	**ETE**	**TW**	**OR**	**PR**	**TOTAL**	**L**	**P**	**T**	**TOTAL**
-	4	-	-	-	-	-	25	25	-	50	-	2	-	2

Course Objectives:

- To analyze digital games and simulations using the MDA framework to understand the relationship between rules and player experience.
- To differentiate between games, simulations, and immersive systems based on realism, goals, and user feedback mechanisms.
- To develop rapid paper prototypes and gameplay loops for interactive experiences targeting real-world social or educational problems.
- To design conceptual 3D environments and XR-based interaction models focusing on navigation, immersion, and player guidance.
- To synthesize design concepts into structured Mini Game Design Documents (GDD) and formal peer-reviewed pitches.

Course Outcomes:

CO1: Deconstruct interactive experiences into mechanics, dynamics, and aesthetics to predict emotional and experiential outcomes.

CO2: Compare and select appropriate interaction models for diverse systems ranging from entertainment games to industrial VR simulations.

CO3: Create functional gameplay loops and paper prototypes tailored to specific player personas and accessibility needs.

CO4: Design spatial layouts and navigation flows for 3D environments that ensure intuitive player guidance and feedback.

CO5: Communicate complex design ideas through professional GDDs, visual mapping, and structured design critiques.

Sr. No.	Name of the Experiment	Duration (Hrs.)
	Part A- Game Analysis Assignments	
1	**Deconstructing a Game Through the MDA Lens** Select any digital game, simulation, or interactive experience you have played or observed. Analyze it using the Mechanics–Dynamics–Aesthetics (MDA) framework. Explore the following: ● What mechanics drive the gameplay? ● How do these mechanics create emergent player dynamics? ● What emotional or experiential outcomes are intended? Reflect on: ● Why the game is engaging (or not). ● How changes in mechanics could produce different experiences. Deliverable: A reflective analysis report with diagrams showing the relationship between mechanics, dynamics, and aesthetics.	4

2	**Games vs Simulations vs Interactive Experiences** Choose three systems: ● One game ● One simulation ● One interactive experiential system (museum installation, VR training, etc.) Compare them in terms of: ● Goals and user motivation ● Level of realism vs abstraction ● Learning or entertainment outcomes ● User interaction and feedback mechanisms Deliverable: Comparative analysis with visual mapping of interaction flow and experience outcomes.	4
3	**Immersive Technology Case Study** Identify a real-world application of immersive technology in fields such as education, healthcare, urban planning, or industry. Investigate: ● What problem the system addresses ● How interaction design and immersive elements improve the experience ● Limitations or challenges of the system ● Your suggestions for improving the experience Deliverable: Analytical report and concept sketches of potential improvements.	4
Part B - Design Exercise / Paper Prototype		
4	**Designing a Core Gameplay Loop** Design a simple interactive experience around a theme such as: ● Climate awareness ● Community problem solving ● Urban mobility ● Health and well-being Your task is to: ● Define the core gameplay loop ● Identify player actions, feedback, and progression ● Create a paper prototype illustrating the interaction cycle Deliverable: Gameplay loop diagram and paper interface sketches.	6
5	**Designing for a Specific Player Persona** Create a player persona (age, interests, motivations, limitations). Then design a game concept tailored to that persona. Consider:	4

	• Accessibility • Player motivations • Interaction simplicity • Emotional experience Deliverable: Persona description + interaction sketches + gameplay concept.	
6	**Conceptual Spatial Experience** Design a conceptual 3D interactive environment such as: • A virtual museum • A learning playground • A disaster response simulation • A historical exploration environment Focus on: • Navigation • Spatial interaction • Player guidance • Camera perspectives Deliverable: Spatial layout sketches and interaction flow diagram.	6
Part C - Mini Game Design Document (GDD) Exercises		
7	**Designing an Educational Immersive Game** Create a Mini Game Design Document (GDD) for a game that teaches a real-world concept (science, history, social awareness, or engineering). Your GDD should include: • Game concept and objective • Target players • Core mechanics and gameplay loop • Player progression system • Learning outcomes Deliverable: A concise 3–4 page Mini GDD with diagrams and illustrations.	6
8	**XR-Based Interactive Experience** Design a VR/AR/MR based immersive experience for a real-world scenario such as: • Medical training • Smart city visualization • Cultural heritage exploration • Industrial training Include in the Mini GDD: • Interaction model	6

	● Player journey ● Environment design ● Feedback and immersion mechanisms Deliverable: Mini GDD with experience flow and interaction sketches.	
	Part D - Design Presentation / Review Exercises	
9	**Game Design Pitch** Students present a 5–7 minute design pitch for their game concept. The pitch should communicate: ● The problem or experience the game addresses ● Target players ● Core gameplay loop ● Unique immersive elements Students must also respond to peer and instructor feedback. Deliverable: Slide presentation and concept explanation.	6
10	**Design Critique & Peer Review** Students participate in a structured design critique session. Each student will: ● Present a prototype or concept ● Critically evaluate two peers' designs ● Suggest improvements based on usability, player experience, and interaction design Deliverable: Short written critique and reflection on the feedback received.	6
	Total Hrs.	**52**

Text Books

1. Jesse Schell, "The Art of Game Design: A Book of Lenses", CRC Press, 3rd Edition, 2019.
2. Tracy Fullerton, "Game Design Workshop: A Playcentric Approach to Creating Innovative Games", CRC Press, 4th Edition, 2018.
3. Scott Rogers, "Level Up! The Guide to Great Video Game Design", Wiley, 2nd Edition, 2014.
4. William R. Sherman and Alan B. Craig, "Understanding Virtual Reality: Interface, Application, and Design", Morgan Kaufmann, 2nd Edition, 2018.
5. Katie Salen and Eric Zimmerman, "Rules of Play: Game Design Fundamentals", MIT Press, 2003.

Reference Books

1. Don Norman, "The Design of Everyday Things", Basic Books, Revised Edition, 2013.
2. Brenda Romero and Ian Schreiber, "Challenges for Game Designers", Charles River Media, 2008.
3. Alan Cooper, Robert Reimann, and David Cronin, "About Face: The Essentials of Interaction Design", Wiley, 4th Edition, 2014.
4. Raph Koster, "A Theory of Fun for Game Design", O'Reilly Media, 2nd Edition, 2013.
5. Jeremy Gibson Bond, "Introduction to Game Design, Prototyping, and Development", Addison-Wesley Professional, 3rd Edition, 2022.

Third Year B.Tech Information Technology															
Semester- V															
Course Code: IT24HON345R							**Course Name: Game Design Principles & Mechanics**								
Teaching Scheme (Hours/Week)					**Examination Scheme**						**Credits**				
L	**P**	**T**	**OL**	**ODL**	**CIE**	**ETE**	**TW**	**OR**	**PR**	**TOTAL**	**L**	**P**	**T**	**ODL**	**TOTAL**
3	-	-	-	-	40	60	-	-	-	100	3	-	-	-	3

Course Objectives:

- To understand advanced concepts of game design, mechanics, and system thinking.
- To analyze gameplay structures, player behavior, and engagement models.
- To design game mechanics, levels, and interactive experiences.
- To develop structured game design documentation and prototype planning skills.

Course Outcomes:

After learning the course, the students will be able to:

CO1:Explain advanced game design principles and mechanics

CO2:Analyze gameplay systems, player interaction, and engagement strategies

CO3: Design game mechanics, levels, and interactive prototypes

CO4: Develop game design documents and plan prototypes for implementation

Unit	Contents	Duration (Hrs.)
1	**Advanced Game Systems & Design Frameworks:** Recap and extension of MDA framework, interconnection of mechanics dynamics and aesthetics, emergent gameplay and systemic design, game economies and resource systems, case studies of gameplay systems	8
2	**Gameplay Mechanics Engineering:** Designing core and secondary mechanics, rule design and constraints, feedback systems visual audio haptic, core gameplay loops and meta progression systems, introduction to state machines and behavior flow	8
3	**Level Design & Spatial Prototyping:** Advanced level design principles, flow pacing and challenge curves, environmental storytelling and spatial narrative, player guidance techniques, 2D and 3D level design, grayboxing and low fidelity prototyping	8
4	**Player Experience Engineering & UX Optimization:** Advanced player psychology, engagement loops, UI UX refinement, accessibility design, measuring player experience and playtesting metrics	7
5	**Game Design Documentation & Production Planning:** Advanced GDD structure, technical design basics, prototype planning and iteration cycles, agile methodology in game development, pre production pipeline concept to validation	8
	Total Hours	**39**

Note for Course Coordinator
The End Term Examination (ETE) of 60 marks will be conducted at the department level. The mode, tools, and methodology for conducting the examination will be decided by the Course Coordinator, with due approval from the Chairperson of the Board of Studies (BoS).

Text Books

1. Jesse Schell, *The Art of Game Design: A Book of Lenses*, CRC Press
2. Tracy Fullerton, *Game Design Workshop: A Playcentric Approach to Creating Innovative Games*, CRC Press

Reference Book

1. Katie Salen and Eric Zimmerman, *Rules of Play: Game Design Fundamentals*, MIT Press
2. Ernest Adams, *Fundamentals of Game Design*, Pearson
3. Scott Rogers, *Level Up! The Guide to Great Video Game Design*, Wiley
4. Tynan Sylvester, *Designing Games: A Guide to Engineering Experiences*, O'Reilly

Third Year B.Tech Information Technology														
Semester-V														
Course Code: IT24HON346R										**Course Name: Design Lab 2**				
Teaching Scheme (Hours/Week)					**Examination Scheme**						**Credits**			
L	**P**	**T**	**OL**	**ODL**	**CIE**	**ETE**	**TW**	**OR**	**PR**	**TOTAL**	**L**	**P**	**T**	**TOTAL**
-	4	-	-	-	-	-	25	25	-	50	-	2	-	2

Course Objectives:

- To understand advanced concepts of game design, mechanics, and system thinking.
- To analyze gameplay structures, player behavior, and engagement models.
- To design game mechanics, levels, and interactive experiences.
- To develop structured game design documentation and prototype planning skills.

Course Outcomes:

After learning the course, the students will be able to:
CO1:Explain advanced game design principles and mechanics
CO2:Analyze gameplay systems, player interaction, and engagement strategies
CO3: Design game mechanics, levels, and interactive prototypes
CO4: Develop game design documents and plan prototypes for implementation

Sr. No.	Name of the Experiment	Duration (Hrs.)
1	**Study and analysis of games using MDA framework**	4
2	Design of core gameplay loop and progression system	4
3	Game mechanics design (rules, constraints, scoring system)	4
4	Preparation of Game Design Document (GDD)	4
5	Paper prototype development of a game concept	4
6	Level design (2D layout / map design / grayboxing)	4
7	UI/UX wireframe design (menus, HUD, navigation)	4
8	Playtesting and feedback analysis	4
9	Game balancing and iteration	6
10	Mini Project: Complete game prototype design and presentation	14
	Total	**52**

Text Books

1. Jesse Schell, The Art of Game Design: A Book of Lenses, CRC Press
2. Tracy Fullerton, Game Design Workshop: A Playcentric Approach to Creating Innovative Games, CRC Press

Reference Books

1. Katie Salen and Eric Zimmerman, *Rules of Play: Game Design Fundamentals*, MIT Press
2. Ernest Adams, *Fundamentals of Game Design*, Pearson

3. Scott Rogers, *Level Up! The Guide to Great Video Game Design*, Wiley
4. Tynan Sylvester, *Designing Games: A Guide to Engineering Experiences*, O'Reilly

Third Year B.Tech Information Technology

Semester- VI

Course Code: IT24HON395R **Course Name: Unity Engine Fundamentals & Visual Scripting**

Teaching Scheme (Hours/Week)					Examination Scheme						Credits				
L	**P**	**T**	**OL**	**ODL**	**CIE**	**ETE**	**TW**	**OR**	**PR**	**TOTAL**	**L**	**P**	**T**	**ODL**	**TOTAL**
3	-	-	-	-	40	60	-	-	-	100	3	-	-	-	3

Course Objectives:
- To introduce Unity game engine architecture and real-time interactive system concepts
- To enable students to implement gameplay logic using Unity Visual Scripting
- To develop system-level thinking for designing interactive and real-time applications

Course Outcomes:
After learning the course, the students will be able to:
CO1: Explain Unity engine architecture and real-time execution flow
CO2: Develop interactive gameplay systems using Unity Visual Scripting
CO3: Apply event-driven and state-based logic in real-time applications
CO4: Analyze gameplay performance and debugging strategies in Unity
CO5: Design structured gameplay logic ready for transition to C# scripting

Unit	Contents	Duration (Hrs.)
1	**Introduction to Unity Engine Architecture** Unity editor workflow and interface. Scenes, GameObjects, and components. Prefabs and hierarchy management. Asset organization and project structure. Practical demonstration of scene setup and environment creation, Prefabs and hierarchy management	8
2	**Event-Driven Systems & Game Loop** Game loop and frame updates. Input systems and event handling. Interaction models and system responses. Designing simple interactive behaviours in real-time systems.	8
3	**Unity Visual Scripting Fundamentals** Introduction to visual scripting concepts. Variables, data flow, and control flow. Event graphs and state-based logic. Implementing interaction logic using visual scripting graphs.	8
4	**Gameplay Systems Using Visual Scripting** Player movement and camera control logic. Interaction systems using triggers and events. UI systems such as score displays and menus. Implementation exercises for gameplay mechanics.	7
5	**Debugging, Performance & Transition to Code** Debugging visual scripting graphs. Performance awareness in real-time applications. Profiling and optimization basics. Mapping visual scripting logic to C# scripts in Unity.	8
	Total Hours	**39**

Note for Course Coordinator

The End Term Examination (ETE) of 60 marks will be conducted at the department level. The mode, tools, and methodology for conducting the examination will be decided by the Course Coordinator, with due approval from the Chairperson of the Board of Studies (BoS).

Text Books

1. Mike Geig, *"Unity Game Development in 24 Hours"*, Sams Publishing, 2018.
2. Alan Thorn, *"Unity 2021 by Example"*, Packt Publishing, 2021.
3. Ryan Henson Creighton, *"Unity Visual Scripting"*, Packt Publishing, 2020.

Reference Book

1. Will Goldstone, *"Unity Game Development Essentials"*, Packt Publishing, 2019.
2. Jesse Freeman, *"Developing Games with Unity"*, O'Reilly Media, 2020.
3. Robert Nystrom, *"Game Programming Patterns"*, Genever Benning, 2014.

<table>
<tr><td colspan="15" align="center">Third Year B.Tech Information Technology</td></tr>
<tr><td colspan="15">Semester-VI</td></tr>
<tr><td colspan="10">Course Code: IT24HON396R</td><td colspan="5">Course Name: Design Lab 3</td></tr>
<tr><td colspan="5">Teaching Scheme (Hours/Week)</td><td colspan="6">Examination Scheme</td><td colspan="4">Credits</td></tr>
<tr><td>L</td><td>P</td><td>T</td><td>OL</td><td>ODL</td><td>CIE</td><td>ETE</td><td>TW</td><td>OR</td><td>PR</td><td>TOTAL</td><td>L</td><td>P</td><td>T</td><td>TOTAL</td></tr>
<tr><td>-</td><td>2</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>25</td><td>-</td><td>-</td><td>25</td><td>-</td><td>1</td><td>-</td><td>1</td></tr>
</table>

Course Objectives:

- To develop proficiency in the Unity Editor, focusing on scene construction, asset management, and the use of Prefabs for scalable 3D environments.
- To understand and implement event-driven programming architectures for real-time interactions using triggers, collisions, and system responses.
- To master graph-based logic (such as Unity Visual Scripting or Bolt) to control object behavior, variables, and data flow without traditional coding.
- To prototype core gameplay systems (scoring, inventory, or puzzles) that demonstrate a functional loop between player input and system feedback.
- To cultivate the ability to document project architecture and logic flows, ensuring technical transparency and project maintainability.

Course Outcomes:

CO1: Construct organized 3D environments in Unity utilizing proper scene hierarchies, nested prefabs, and optimized environmental lighting.

CO2: Implement complex interaction systems using event-based logic to trigger environmental changes and UI feedback based on player actions.

CO3: Develop functional gameplay logic using visual scripting graphs, including the use of conditional statements, state machines, and data variables.

CO4: Prototype a playable interactive experience that integrates player movement, camera control, and a defined goal or exploration mechanic.

CO5: Articulate technical design decisions through annotated logic diagrams and project walkthroughs, identifying areas for optimization and future iteration.

<table>
<tr><td colspan="3" align="center">List of Experiments</td></tr>
<tr><td>Sr. No.</td><td align="center">Name of the Experiment</td><td>Duration (Hrs.)</td></tr>
<tr><td colspan="3" align="center">Part A- Laboratory Exercises / Implementation Tasks</td></tr>
<tr><td>1</td><td>Scene Construction & Object Behaviour Exploration
Create a small interactive 3D environment using the Unity editor. The scene should contain multiple objects with distinct behaviours.
Students should explore:
● Scene hierarchy and GameObject relationships
● Prefabs and reusable components
● Environmental elements such as lighting and spatial layout
● Basic object interaction (movement, trigger zones, or object response)</td><td>2</td></tr>
</table>

	Deliverable: A functional Unity scene demonstrating proper project organization, prefab usage, and scene interaction.	
2	**Event-Driven Interaction System** Design a simple event-driven interactive system where player actions trigger system responses. Examples may include: ● Opening doors when the player approaches ● Activating environmental changes through switches ● Object collection triggering feedback or UI messages Students must implement the logic using Unity's event-based interaction model. Deliverable: Interactive system demonstrating input handling, trigger events, and real-time system responses.	2
3	**Visual Logic Implementation with Graph-Based Scripting** Develop a visual scripting graph that controls the behaviour of an interactive object or system. Students should implement: ● Variables and data flow ● Conditional logic and control flow ● State-based behaviour Example ideas: ● Puzzle logic ● Interactive environmental system ● Dynamic object behaviour Deliverable: A visual scripting graph demonstrating functional interaction logic.	2
	Part B - Gameplay Prototype using Visual Scripting	
4	**Interactive Mini-Game Prototype** Design and implement a simple mini-game prototype using visual scripting. The prototype should include: ● Player interaction mechanics ● Score or progress tracking ● Game feedback (visual, audio, or UI) Students are encouraged to experiment with creative gameplay mechanics rather than focusing only on traditional game genres. Deliverable: Playable prototype demonstrating core gameplay loop and interaction logic.	4
5	**Exploration-Based Interactive Experience**	4

<table>
<tr><td></td><td>Create a free-roaming exploration experience where the player interacts with the environment.
The prototype should include:
● Player movement and camera control
● Interactive objects or triggers
● Environmental storytelling elements
Focus should be on interaction and discovery rather than competition or scoring.
Deliverable:
Interactive exploration environment built using visual scripting logic.</td><td></td></tr>
<tr><td>6</td><td>Gameplay System Prototype
Develop a gameplay system prototype that demonstrates a functional gameplay mechanic.
Examples:
● Simple puzzle mechanic
● Resource collection and scoring system
● Object manipulation system
● Timed challenge system
The emphasis should be on system design and interaction logic, implemented entirely through visual scripting graphs.
Deliverable:
Functional gameplay mechanic prototype with working player interactions.</td><td>4</td></tr>
<tr><td colspan="3">Part C - Technical Documentation Exercises</td></tr>
<tr><td>7</td><td>Unity Project Architecture Documentation
Document the technical structure of your Unity project.
The documentation should describe:
● Scene organization
● Prefab hierarchy
● Asset management strategy
● Visual scripting graph structure
Students should include diagrams illustrating object relationships and interaction flow.
Deliverable:
A structured technical document explaining project architecture and implementation decisions.</td><td>4</td></tr>
<tr><td>8</td><td>Visual Scripting Logic Explanation
Create a technical explanation of a visual scripting system implemented in your project.
Students should describe:
● Variables and data flow</td><td>4</td></tr>
</table>

	● Event triggers and system responses ● Control logic used in the scripting graph ● Possible improvements or optimizations Deliverable: Annotated visual scripting diagrams and technical explanation of the implemented logic.	
	Part D - Demo Presentation Exercises	
9	**Prototype Demonstration & System Walkthrough** Students present a live demonstration of their Unity prototype. The presentation should cover: ● Project objective ● Gameplay or interaction mechanics ● Key visual scripting systems ● Design challenges encountered Students should demonstrate the working system in real time. Deliverable: Short presentation with a live prototype walkthrough.	2
10	**Design Reflection & Improvement Proposal** Students present a critical reflection on their prototype. The presentation should address: ● What worked well in the design ● Technical challenges faced ● Lessons learned from implementing visual scripting ● Proposed improvements for the next iteration Deliverable: Presentation highlighting design evolution and future improvements.	2
	Total Hrs.	**30**

Text Books

1. Jeremy Gibson Bond, "Introduction to Game Design, Prototyping, and Development: From Concept to Playable Game with Unity and C#", Addison-Wesley Professional, 3rd Edition, 2022.
2. Harrison Ferrone, "Learning C# by Developing Games with Unity", Packt Publishing, 8th Edition, 2024.
3. Unity Technologies, "The Unity Game Designer Playbook: Prototyping and Visual Scripting in Unity", Unity Press (Digital), 2025.
4. Nicolas Alejandro Borromeo, "Hands-On Unity 2022 Game Development", Packt Publishing, 3rd Edition, 2023.
5. Mike Geig, "Unity in Action: Multiplatform Game Development with C# and Visual Scripting", Manning Publications, 4th Edition, 2026.

Reference Books

1. **Robert Nystrom**, *"Game Programming Patterns"*, Genever Benning, 2014.
2. **Edward Lavieri**, *"Unity 6 Cookbook: Over 150 Recipes for Real-Time 3D and Immersive Experiences"*, Packt Publishing, 6th Edition, 2025.

3. **Penny de Byl**, *"The Holistic Guide to Game AI with Unity Visual Scripting"*, Focal Press, 2024.
4. **Don Norman**, *"The Design of Everyday Things"*, Basic Books, Revised Edition, 2013.
5. **Unity Technologies**, *"Ultimate Guide to Profiling and Optimization in Unity 6"*, Unity Press (Digital), 2025.

Final Year B.Tech Information Technology															
Semester- VII															
Course Code: **IT24HON445R**							**Course Name:** **Game Programming in Unity with C# & XR Systems**								
Teaching Scheme (Hours/Week)					**Examination Scheme**						**Credits**				
L	**P**	**T**	**OL**	**ODL**	**CIE**	**ETE**	**TW**	**OR**	**PR**	**TOTAL**	**L**	**P**	**T**	**ODL**	**TOTAL**
2	-	-	-	-	40	60	-	-	-	100	2	-	-	-	2

Course Objectives:
- To enable students to develop gameplay systems using C# scripting in Unity
- To introduce real-time programming concepts and performance considerations
- To apply game programming skills to interactive, XR, and simulation-based systems

Course Outcomes:
After learning the course, the students will be able to:
CO1: Explain Unity's programming model and script execution lifecycle
CO2: Develop gameplay mechanics using C# scripts in Unity
CO3: Apply object-oriented and data-driven design principles in game programming
CO4: Implement basic AI, physics-based, and interaction systems
CO5: Design and evaluate interactive systems for games and XR applications

Unit	Contents	Duration (Hrs.)
1	**Unity Programming Model & Script Lifecycle** MonoBehaviour architecture. Script lifecycle (Awake, Start, Update, FixedUpdate). Coroutines and event-based programming. Demonstration of component communication using scripts.	6
2	**C# Scripting for Gameplay Systems** Script structure and modular design. Player movement systems and input handling. Physics interactions and collision detection. Implementing interactive objects using C#.	6
3	**Game Logic, State Management & AI** Finite State Machines (FSM). Behaviour design for interactive characters. Introduction to pathfinding concepts. Implementation of gameplay states and transitions	6
4	**Performance, Optimization & Debugging** Debugging techniques and error handling. Profiling tools and performance optimization. Memory management considerations in real-time applications. Clean code practices for game systems.	6
5	**XR Systems & Interactive Applications in Unity** Introduction to XR development in Unity. XR Interaction Toolkit architecture. Controller and hand interaction systems. Development of simple immersive interaction prototypes.	6
	Total Hours	**30**

Note for Course Coordinator

The End Term Examination (ETE) of 60 marks will be conducted at the department level. The mode, tools, and methodology for conducting the examination will be decided by the Course Coordinator, with due approval from the Chairperson of the Board of Studies (BoS).

Text Books

1. Alan Thorn, *"Unity Game Development in 24 Hours"*, Sams Publishing, 2018.
2. Joe Hocking, *"Unity in Action: Multiplatform Game Development with C#"*, Manning Publications, 2021.
3. Matthew Johnson, *"Learning C# by Developing Games with Unity"*, Packt Publishing, 2020.

Reference Book

1. Robert Nystrom, *"Game Programming Patterns"*, Genever Benning, 2014.
2. Jason Gregory, *"Game Engine Architecture"*, CRC Press, 2018.
3. Eric Lengyel, *"Mathematics for 3D Game Programming and Computer Graphics"*, Cengage Learning, 2012

<table>
<tr><td colspan="15" align="center">Final Year B.Tech Information Technology</td></tr>
<tr><td colspan="15">Semester-VII</td></tr>
<tr><td colspan="9">Course Code: IT24HON446R</td><td colspan="6">Course Name: Design Lab 4</td></tr>
<tr><td colspan="5">Teaching Scheme (Hours/Week)</td><td colspan="6">Examination Scheme</td><td colspan="4">Credits</td></tr>
<tr><td>L</td><td>P</td><td>T</td><td>OL</td><td>ODL</td><td>CIE</td><td>ETE</td><td>TW</td><td>OR</td><td>PR</td><td>TOTAL</td><td>L</td><td>P</td><td>T</td><td>TOTAL</td></tr>
<tr><td>-</td><td>2</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>25</td><td>-</td><td>-</td><td>25</td><td>-</td><td>1</td><td>-</td><td>1</td></tr>
</table>

Course Objectives:

- To gain a deep understanding of the Unity Script Lifecycle and C# fundamentals for high-performance real-time applications.
- To implement decoupled, event-driven architectures that allow independent game objects to communicate efficiently without rigid dependencies.
- To apply structured programming patterns like Finite State Machines (FSM) for managing complex gameplay flows and character AI behaviors.
- To develop natural user interfaces and object interaction models specifically for Virtual and Augmented Reality (XR) environments.
- To cultivate the ability to analyze, debug, and defend technical implementation choices through professional demonstrations and verbal reviews.

Course Outcomes:

CO1: Optimize object behavior by correctly utilizing Unity lifecycle functions (Update vs. FixedUpdate) and Coroutines for time-sensitive tasks.
CO2: Construct robust, event-based communication networks between game objects using C# Delegates, UnityEvents, or the Observer pattern.
CO3: Implement a functional Finite State Machine (FSM) to control player movement, game states (Menu/Play/Pause), and autonomous NPC behaviors.
CO4: Build interactive XR prototypes that support hand-tracking or controller-based grabbing, manipulation, and spatial navigation.
CO5: Critique and troubleshoot project performance, providing technical justifications for script architecture and optimization during a live viva.

<table>
<tr><td>Sr. No.</td><td>Name of the Experiment</td><td>Duration (Hrs.)</td></tr>
<tr><td colspan="3" align="center">Part A- Programming Assignments</td></tr>
<tr><td>1</td><td>Exploring the Unity Script Lifecycle
Develop a Unity script-based interactive behaviour that demonstrates the use of the Unity script lifecycle.
Students should experiment with:
● Script initialization using lifecycle functions
● Frame-based updates and physics-based updates
● Coroutine-based time-dependent behaviours

The goal is to observe how different lifecycle methods influence game object behaviour in real-time systems.

Deliverable:
A Unity scene demonstrating multiple scripted objects with clearly implemented lifecycle behaviour and explanatory comments in the code</td><td>2</td></tr>
</table>

<table>
<tr><td>2</td><td>Interactive Gameplay Objects Using C#

Create a set of interactive gameplay objects using C# scripting.

Students should design objects that:
Respond to player input or environmental triggers
Use physics interactions such as collisions or forces
Communicate with other game objects using scripts
The exercise should emphasize modular and reusable script design.

Deliverable:
A small Unity environment with multiple interactive scripted objects and documented code structure.</td><td>2</td></tr>
<tr><td>3</td><td>Event-Based System with Script Communication

Develop a small interactive system where multiple scripts communicate with each other.

Students should implement:
Event-driven behaviour
Script-to-script communication
Modular gameplay components
Example systems may include:
Interactive puzzle system
Trigger-based event chain
Object interaction network
Deliverable:
A working interactive system demonstrating event-driven programming and component communication</td><td>2</td></tr>
<tr><td colspan="3">Part B - Gameplay System Implementation</td></tr>
<tr><td>4</td><td>Player Controller & Movement System

Design and implement a player movement and control system in Unity using C#.

Students should implement:
Player movement mechanics
Camera behaviour
Input handling
Environmental interaction</td><td>4</td></tr>
</table>

	Students are encouraged to experiment with different movement styles such as exploration, physics-based movement, or platform mechanics. Deliverable: A playable prototype demonstrating a functional player control system.	
5	**Gameplay State Management System** Implement a gameplay state management system using programming logic such as Finite State Machines (FSM). Students should design states such as: ● Menu state ● Gameplay state ● Pause state ● Game-over or success state The goal is to explore how structured state transitions control gameplay flow. Deliverable: Unity prototype with clearly implemented gameplay states and transitions.	4
6	**Interactive Character Behaviour System** Create a basic interactive character behaviour system. Students may implement: ● NPC movement behaviour ● Simple AI logic ● Path-following or patrol behaviour ● Interaction with the player Students should explore how behavioural logic influences player experience. Deliverable: Prototype demonstrating interactive character behaviour using scripted logic.	4
Part C - XR Interaction Prototype		
7	**XR Object Interaction Experience** Design a simple XR interaction prototype using Unity's XR interaction systems. Students should implement: ● Hand/controller-based object interaction ● Object grabbing or manipulation ● Environmental interaction	4

<table>
<tr><td></td><td>The focus should be on creating natural interaction between the user and virtual objects.

Deliverable:
A working XR prototype demonstrating basic immersive interaction mechanics.</td><td></td></tr>
<tr><td>8</td><td>Immersive XR Learning or Exploration Experience

Design an XR-based experiential prototype for a real-world scenario.

Possible themes include:

● Cultural heritage exploration
● Scientific learning environment
● Industrial training simulation
● Virtual museum experience

Students should focus on:

● User interaction design
● Spatial navigation
● Immersive feedback mechanisms

Deliverable:
An XR prototype illustrating interactive immersive experience design.</td><td>4</td></tr>
<tr><td colspan="3">Part D - Final Demonstration & Viva Exercise</td></tr>
<tr><td>9</td><td>Integrated Gameplay System Demonstration

Students present a comprehensive demonstration of their implemented gameplay systems and XR interaction prototype.

The demonstration should include:

● Overview of the gameplay systems implemented
● Explanation of key C# scripts and programming logic
● Demonstration of interaction mechanics and system behaviour
● Reflection on performance optimization and debugging strategies

Students should be prepared to answer conceptual and technical questions related to Unity scripting, gameplay architecture, and immersive interaction systems.

Deliverable:
Live system demonstration with explanation of design decisions, programming structure, and system functionality</td><td>4</td></tr>
<tr><td></td><td>Total Hrs.</td><td>30</td></tr>
</table>

Text Books

1. Harrison Ferrone, "Learning C# by Developing Games with Unity", Packt Publishing, 8th Edition, 2024.
2. Mike Geig, "Unity in Action: Multiplatform Game Development with C#", Manning Publications, 4th Edition, 2026

3. Jonathan Linowes, "Unity 2024 Virtual Reality Projects: Learn Virtual Reality by Developing 10+ Projects", Packt Publishing, 3rd Edition, 2024.

Reference Books

1. **Robert Nystrom**, *"Game Programming Patterns"*, Genever Benning, 2014. (Essential for **Event-Based Systems**).
2. **Unity Technologies**, *"XR Interaction Toolkit (XRI) Documentation & Best Practices"*, Unity Press, 2025.

Final Year B.Tech Information Technology															
Semester- VIII															
Course Code: IT24HON493R							**Course Name: Game Development Studio & Capstone Project**								
Teaching Scheme (Hours/Week)					**Examination Scheme**						**Credits**				
L	**P**	**T**	**OL**	**ODL**	**CIE**	**ETE**	**TW**	**OR**	**PR**	**TOTAL**	**L**	**P**	**T**	**ODL**	**TOTAL**
2	-	-	-	-	40	60	-	-	-	100	2	-	-	-	2

Course Objectives:

- To enable students to apply game development knowledge in a studio-oriented environment
- To develop complete, functional, and industry-relevant Unity-based projects
- To cultivate teamwork, documentation, testing, and presentation skills required in professional game development

Course Outcomes:

After learning the course, the students will be able to:

CO1: Plan and manage a game development project using industry-style workflows
CO2: Design and implement a complete Unity-based game or immersive application
CO3: Apply software engineering practices such as version control, testing, and documentation
CO4: Analyze user feedback, playtesting results, and system performance
CO5: Present and defend a game or immersive system prototype in a professional setting

Unit	Contents	Duration (Hrs.)
1	**Studio Orientation & Project Planning** Overview of game production stages. Roles in game development teams. Pre-production, production, and post-production processes. Case studies of professional game development workflows.	6
2	**Game Production Pipeline & Workflows** Game Design Documents (GDD). Technical design documents. Project scoping and milestone planning. Documentation practices for collaborative development.	6
3	**Implementation, Testing & Iteration** Version control systems (Git). Task management and development pipelines. Iterative development and agile methodologies in game production.	6
4	**Performance, Optimization & Quality Assurance** Methods for testing gameplay systems. Player feedback and usability evaluation. Bug tracking and issue management. Iterative improvement based on testing.	6
5	**Final Delivery, Documentation & Presentation** Preparing project builds for distribution. Documentation and presentation of projects. Creating development portfolios and showcasing projects.	6
	Total Hours	**30**

Note for Course Coordinator

The End Term Examination (ETE) of 60 marks will be conducted at the department level. The mode, tools, and methodology for conducting the examination will be decided by the Course Coordinator, with due approval from the Chairperson of the Board of Studies (BoS).

Text Books

1. Jason Gregory, *"Game Engine Architecture"*, CRC Press, 2018.
2. Clinton Keith, *"Agile Game Development with Scrum"*, Addison-Wesley, 2010.
3. Scott Rogers, *"Level Up! The Guide to Great Video Game Design"*, Wiley, 2014.

Reference Book

1. Ernest Adams, *"Game Design Workshop"*, CRC Press, 2018.
2. Jesse Schell, *"The Art of Game Design: A Book of Lenses"*, CRC Press, 2019.
3. Tracy Fullerton, *"Game Design Workshop"*, CRC Press, 2018.

Final Year B.Tech Information Technology

Semester-VI

Course Code: IT24HON494R **Course Name: Design Lab 5**

Teaching Scheme (Hours/Week)					**Examination Scheme**						**Credits**			
L	**P**	**T**	**OL**	**ODL**	**CIE**	**ETE**	**TW**	**OR**	**PR**	**TOTAL**	**L**	**P**	**T**	**TOTAL**
-	2	-	-	-	-	-	25	-	-	25	-	1	-	1

Course Objectives:

- Help students conceptualize and propose engaging interactive game ideas.
- Instruct students on creating structured Game Design Documents (GDD) and technical architecture plans.
- Introduce industry-standard production planning, task allocation, and milestone scheduling.
- Provide hands-on guidance for building playable prototypes and integrating core gameplay systems.
- Teach testing, debugging, and optimization for the successful delivery of a polished final project.

Course Outcomes:

CO1: Conceptualize engaging interactive game ideas using game design principles.
CO2: Develop structured Game Design Documents (GDD) and technical architecture plans.
CO3: Apply industry-standard production planning, task allocation, and scheduling techniques.
CO4: Build and test playable game prototypes with integrated core gameplay systems.

List of Experiments

Sr. No.	**Name of the Experiment**	**Duration (Hrs.)**
	Phase 1 - Project Ideation & Proposal	
1	**Game Concept Exploration** Students identify a problem domain, learning objective, or experiential theme for their capstone project. They explore different possibilities for creating an interactive game or immersive experience. Activities: ● Brainstorming game concepts ● Identifying target players ● Defining the core experience or problem being addressed Outcome: Short concept note describing the proposed game idea and intended player experience.	2
2	**Project Proposal & Concept Presentation** Students develop a formal project proposal and present their concept to instructors and peers. Activities: ● Defining game objectives and gameplay concept ● Identifying target audience and platform	2

	• Presenting the initial gameplay loop and interaction concept Outcome: Concept presentation and feedback session. Evaluation Component: *Project Proposal & Concept Presentation*	
	Phase 2 - Design & Production Planning	
4	**Game Design Document (GDD) Development** Students prepare a Game Design Document (GDD) describing the overall structure of the game. Activities: • Game mechanics and rules • Player progression system • Level or environment design • Interaction and user experience design Outcome: Structured GDD outlining the core design of the project.	4
5	**Technical Design & System Architecture Planning** Students define the technical architecture of their game system. Activities: • Identifying gameplay systems • Defining system modules • Planning scripting architecture • Selecting required tools and assets Outcome: Technical design document including system diagrams and development strategy.	4
6	**Production Planning & Milestone Scheduling** Students plan the production workflow for their project. Activities: • Dividing the project into development tasks • Defining milestones and sprint goals • Assigning roles within the development team • Creating development timeline Outcome: Project production plan with milestones and task allocation.	4

	Evaluation Component: *Production Planning & Design Documentation*	
	Phase 3 - Development & Implementation	
7	**Prototype Development (Core Gameplay System)** Students implement a functional prototype demonstrating the core gameplay loop. Activities: ● Basic player interaction system ● Core mechanics implementation ● Initial environment setup Outcome: Playable early prototype demonstrating primary gameplay mechanics.	4
8	**Gameplay Systems Integration** Students expand the prototype by integrating multiple gameplay systems. Activities: ● Player control system ● Interactive objects ● Basic UI elements ● Game state management Outcome: Working prototype with integrated gameplay systems.	4
9	**Development Review & Iterative Improvement** Students present their current implementation progress and receive feedback. Activities: ● Code and system architecture review ● Testing gameplay functionality ● Identifying technical challenges Outcome: Improved prototype based on technical review and iterative development. Evaluation Component: *System Architecture & Development Review*	2
	Phase 4- Testing, Optimization & Delivery	
10	**Testing, Debugging & Quality Improvement**	4

	Students conduct structured testing of gameplay systems. Activities: ● Debugging errors ● Performance optimization ● Player testing and usability feedback ● Bug tracking and issue resolution Outcome: Improved system stability and optimized gameplay performance.	
11	**Final Project Demonstration & Portfolio Preparation** Students prepare the final version of their capstone project. Activities: ● Preparing playable builds ● Documenting the development process ● Creating project presentation materials ● Demonstrating the final system Outcome: Final project demonstration and technical explanation. Evaluation Components: ● *Project Demonstration* ● *Technical Viva*	2
	Total Hrs.	**32**

Marathwada Mitra Mandal's College of Engineering

Karvenagar, Pune- 52

An Autonomous Institute Affiliated to SPPU

“ येथे बहुतांचे हित ”

Curriculum Structure and Syllabus

Honours

A.Y. 2026-27

Accredited with A++ grade by NAAC
Recipient of Best College award by SPPU | Accredited by NBA
Recognized under 2(f) and 12(B) of UGC Act 1956
www.mmcoe.edu.in

About Marathwada Mitra Mandal's College of Engineering

MMCOE has been recognised as the BEST COLLEGE by SPPU. MMCOE is one feather in the cap of Marathwada Mitra Mandal's Pune, from the academic year 2006 – 2007.

The Institute offers six undergraduate engineering programmes in Electrical Engineering, Mechanical Engineering, Electronics & Telecommunication Engineering, Information Technology, Computer Engineering, and Artificial Intelligence & Data Science. In addition, it offers six postgraduate programmes M.Tech (Electrical – Power Electronics & Drives), M.Tech (Mechanical – Design Engineering), M.Tech (Electronics & Telecommunication – VLSI & Embedded Systems), M.Tech (Information Technology – Data Science), M.Tech (Computer Engineering), along with a Ph.D. programme in Computer Engineering. It houses 2000+ students and 200 staff members. Along with the focus on core academics, college also takes care of overall personality development of the students.

The college has an active robotics cell, NCC, Training and Placement cell etc. The students are motivated to actively participate in co-curricular and extra-curricular activities. They have brought many laurels to the Institute in the events such as Vinodottam Karandak, Firodiya Karandak, Zest, Dexterity etc.

Institute Vision

To be a globally renowned institution through excellence in engineering education for sustainable and holistic development

Institute Mission

M1: Empower students with cutting-edge technologies and global competencies

M2: Foster culture of research and entrepreneurial mindset

M3: Imbibe social and professional values

M4: Provide an inclusive environment for lifelong learning

Department of Information Technology

Vision and Mission of the Department

- **Vision**

  To emerge as a globally competent center in education, research and innovation in Information Technology, fostering sustainable development with a holistic vision.

- **Mission**

  M1 : To impart IT education that develops globally competent professionals

  M2 : To promote innovation through research and entrepreneurship.

  M3 : To encourage lifelong learning and a holistic commitment towards societal development.

- **Program Specific Outcomes (PSOs)**

  Information Technology graduates will be able to :

  - Design and develop information systems using principles of computing to solve real-world problems.
  - Integrate emerging IT technologies to innovate and improve technological ecosystems.

Program Outcomes (POs)

PO1: Engineering Knowledge: Apply knowledge of mathematics, natural science, computing, engineering fundamentals and an engineering specialization as specified in WK1 to WK4 respectively to develop to the solution of complex engineering problems.

PO2: Problem Analysis: Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development. (WK1 to WK4)

PO3: Design/Development of Solutions: Design creative solutions for complex engineering problems and design/develop systems/components/processes to meet identified needs with consideration for the public health and safety, whole-life cost, net zero carbon, culture, society and environment as required. (WK5)

PO4: Conduct Investigations of Complex Problems: Conduct investigations of complex engineering problems using research-based knowledge including design of experiments, modelling, analysis & interpretation of data to provide valid conclusions. (WK8).

PO5: Engineering Tool Usage: Create, select and apply appropriate techniques, resources and modern engineering & IT tools, including prediction and modelling recognizing their limitations to solve complex engineering problems. (WK2 and WK6)

PO6: The Engineer and The World: Analyze and evaluate societal and environmental aspects while solving complex engineering problems for its impact on sustainability with reference to economy, health, safety, legal framework, culture and environment. (WK1, WK5, and WK7).

PO7: Ethics: Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national & international laws. (WK9)

PO8: Individual and Collaborative Team work: Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams.

PO9: Communication: Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations considering cultural, language, and learning differences

PO10: Project Management and Finance: Apply knowledge and understanding of engineering management principles and economic decision-making and apply these to one's own work, as a member and leader in a team, and to manage projects and in multidisciplinary environments.

PO11: Life-Long Learning: Recognize the need for, and have the preparation and ability for i) independent and life-long learning ii) adaptability to new and emerging technologies and iii) critical thinking in the broadest context of technological change. (WK8)

Curriculum for B. Tech. in Information Technology with Honours in Enterprise Linux Administration

Department of Information Technology

General Objectives and Scope:

The Honours Degree Program aims to provide students with advanced knowledge and deeper understanding in their chosen field of study. It focuses on developing critical thinking, analytical abilities, and research skills through specialized courses and practical learning. The program encourages independent learning and innovation, enabling students to gain expertise beyond the regular undergraduate curriculum.

The scope of the honours degree includes opportunities for research, specialization, and skill development. Students may undertake projects or dissertations that enhance their practical and analytical abilities. The program also prepares students for higher education, research careers, and better employment opportunities in industry and academia.

Why Enterprise Linux Administration ?

Enterprise Linux Administration is essential because Linux is widely used in servers, cloud computing, data centers, and enterprise IT infrastructure. Organizations rely on Linux systems for their stability, security, and performance, creating a strong demand for professionals who can efficiently manage and maintain these environments.

Studying Enterprise Linux Administration equips students with practical skills in system configuration, user management, networking, security, and server administration. It prepares them for roles such as system administrator, DevOps engineer, and cloud engineer, and provides a strong foundation for working with modern technologies like virtualization and cloud platforms.

Prerequisites :

1. Basic knowledge of computer fundamentals
2. Understanding of operating system concepts
3. Familiarity with command line interface (CLI) basics
4. Basic knowledge of computer networks
5. Understanding of file systems and storage concepts
6. Interest in system administration and server management

Curriculum Structure

Course Code	Course Name	Teaching Scheme (Hrs/week)				Examination Scheme						Credits				
		L	P	T	ODL	CIE	ETE	TW	PR	OR	Total	L	P	T	ODL	Total
Semester IV																
IT24HON297R	Linux Fundamentals and Command-Line Administration	3	-	-	-	40	60	-	-	-	100	3	-	-	-	**3**
IT24HON298R	Linux Fundamentals and Command-Line Administration Lab	-	4	-	-	-	-	25	-	25	50	-	2	-	-	**2**
Semester V																
IT24HON347R	Enterprise Linux System Administration	3	-	-	-	40	60	-	-	-	100	3	-	-	-	**3**
IT24HON348R	Enterprise Linux System Administration Lab	-	4	-	-	-	-	25	-	25	50	-	2	-	-	**2**
Semester VI																
IT24HON397R	Linux Networking and Security Administration	2	-	-	-	40	60	-	-	-	100	2	-	-	-	**2**
IT24HON398R	Linux Networking and Security Administration Lab	-	2	-	-	-	-	25	-	-	25	-	1	-	-	**1**
Semester VII																
IT24HON447R	Linux Automation and DevOps with Ansible	2	-	-	-	40	60	-	-	-	100	2	-	-	-	**2**
IT24HON448R	Linux Automation and DevOps with Ansible Lab	-	2	-	-	-	-	25	-	-	25	-	1	-	-	**1**
Semester VIII																
IT24HON495R	Capstone Project	-	8	-	-	-	-	25	-	100	125	-	4	-		4
Total		**10**	**20**	**-**	**-**	**160**	**240**	**125**	**-**	**150**	**675**	**10**	**10**	**-**	**-**	**20**

L- Lecture | P- Practical | T- Tutorial | CIE-Continuous Internal Evaluation
ETE- End Term Examination | TW- Term work | PR- Practical | OR- Oral
L : 1 Hr.= 1 credit | P: 2 Hr. = 1 Credit | T: 1 Hr. = 1 Credit | ODL-Open Distance Learning

Second Year B.Tech Information Technology
Semester- IV

Course Code: IT24HON297R	Course Name: Linux Fundamentals and Command-Line Administration

Teaching Scheme (Hours/Week)					Examination Scheme						Credits				
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	ODL	TOTAL
3	-	-	-	-	40	60	-	-	-	100	3	-	-	-	3

Course Objectives:

- To understand Linux architecture and operating environment
- To use Linux command line efficiently
- To manage files, directories, and permissions
- To install and manage software packages in Linux systems

Course Outcomes:

After learning the course, the students will be able to:

CO1: Explain Linux architecture, distributions, and boot process.
CO2: Use Linux command-line tools for file handling and text processing.
CO3: Manage file systems, permissions, and storage operations.
CO4: Perform software installation and package management in Linux.
CO5: Apply basic system administration tasks in a Linux environment

Unit	Contents	Duration (Hrs.)
1	**Introduction to Linux:** History and evolution of Linux, Open-source philosophy, Linux distributions (Ubuntu, CentOS, RHEL) and enterprise Linux, Linux architecture (kernel, shell, hardware layers), Types of kernels, System calls and kernel modules, Installation methods (GUI/CLI), Disk partitioning basics, Boot process (BIOS/UEFI, GRUB, init/systemd), Runlevels/targets, Introduction to Linux file system structure.	8
2	**Linux Command Line:** Shell environment (bash), Terminal usage and command syntax, File and directory management commands (pwd, ls, cd, mkdir, rm, cp, mv), File viewing and editing tools (cat, less, head, tail, nano/vi), Text processing utilities (grep, cut, sort, wc, awk, sed basics), Wildcards and pattern matching, Input/output redirection (>, >>, <) and pipes (|), Command history and shortcuts, Basic shell scripting concepts.	8
3	**File Systems and Permissions:** Linux file system hierarchy (/, /home, /etc, /var, /usr), File types, Inodes and file metadata, File ownership (user, group), Permission model (read, write, execute), Changing permissions (chmod symbolic and numeric modes), Changing ownership (chown, chgrp), Hard links and soft links, Archiving and compression tools (tar, gzip, bzip2, zip), Disk usage analysis (df, du).	7

4	**Software Management:** Package management concepts, RPM package format, Package management tools (YUM, DNF), Installing, updating, removing packages, Searching and querying packages, Dependency management, Repository configuration and management, Local and remote repositories, Software compilation basics (source code installation), System updates and patch management.	8
5	**Basic System Administration:** User and group management (useradd, usermod, userdel, groupadd), Password management and policies, Process management (ps, top, nice, kill, bg/fg), Job scheduling (cron, at), Disk management (mount, umount, fdisk basics), File system checking (fsck), Basic networking commands (ping, ip, netstat, ss), Introduction to shell scripting (variables, loops, conditional statements), System monitoring and log files.	8
	Total Hours	**39**

Text Books

1. Nemeth, E., Snyder, G., Hein, T., & Whaley, B., *UNIX and Linux System Administration Handbook*, Pearson
2. Sobell, M. G., *A Practical Guide to Linux Commands, Editors, and Shell Programming*, Pearson

Reference Books

1. Shotts, W. E., *The Linux Command Line*, No Starch Press
2. Negus, C., *Linux Bible*, Wiley
3. Ward, B., *How Linux Works*, No Starch Press

Online References

1. NPTEL: *Introduction to Linux / Operating Systems*
2. SWAYAM: Linux and System Administration Courses
3. Coursera: *Linux Command Line Basics*
4. Official Documentation: https://www.kernel.org, https://access.redhat.com/documentation

<table>
<tr><th colspan="16">Second Year B.Tech Information Technology</th></tr>
<tr><td colspan="16">Semester- IV</td></tr>
<tr><td colspan="6">Course Code: IT24HON298R</td><td colspan="10">Course Name: Linux Fundamentals and Command-Line Administration Lab</td></tr>
<tr><th colspan="5">Teaching Scheme (Hours/Week)</th><th colspan="6">Examination Scheme</th><th colspan="5">Credits</th></tr>
<tr><th>L</th><th>P</th><th>T</th><th>OL</th><th>ODL</th><th>CIE</th><th>ETE</th><th>TW</th><th>OR</th><th>PR</th><th>TOTAL</th><th>L</th><th>P</th><th>T</th><th>ODL</th><th>TOTAL</th></tr>
<tr><td>-</td><td>4</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>25</td><td>-</td><td>25</td><td>50</td><td>-</td><td>2</td><td>-</td><td>-</td><td>2</td></tr>
<tr><td colspan="16">Course Objectives:
● To understand Linux architecture and operating environment
● To use Linux command line efficiently
● To manage files, directories, and permissions
● To install and manage software packages in Linux systems</td></tr>
<tr><td colspan="16">Course Outcomes:
After learning the course, the students will be able to:
CO1: Explain Linux architecture, distributions, and boot process.
CO2: Use Linux command-line tools for file handling and text processing.
CO3: Manage file systems, permissions, and storage operations.
CO4: Perform software installation and package management in Linux.
CO5: Apply basic system administration tasks in a Linux environment</td></tr>
</table>

Sr. No	List of Experiments	Duration (Hrs.)
1	Installation of Linux OS (VirtualBox/VMware) and basic system navigation	4
2	Practice basic Linux commands (pwd, ls, cd, mkdir, cp, mv, rm)	4
3	Working with file viewing and editing tools (cat, less, nano/vi)	4
4	File permissions and ownership management (chmod, chown, chgrp)	4
5	Creating and managing hard and soft links	4
6	Use of text processing utilities (grep, cut, sort, wc)	4
7	Input/output redirection and pipes	4
8	Archiving and compression (tar, gzip, zip)	4

9	Package management using YUM/DNF (install, update, remove)	4
10	User and group management with basic process monitoring (ps, top, kill)	4
11	MINI PROJECT on Linux System Administration and Automation Toolkit	12
	Total Hours	**52**

Text Books

1. Nemeth, E., Snyder, G., Hein, T., & Whaley, B., *UNIX and Linux System Administration Handbook*, Pearson
2. Sobell, M. G., *A Practical Guide to Linux Commands, Editors, and Shell Programming*, Pearson

Reference Books

1. Shotts, W. E., The Linux Command Line, No Starch Press
2. Negus, C., Linux Bible, Wiley
3. Ward, B., How Linux Works, No Starch Press

Online References

1. NPTEL: *Introduction to Linux / Operating Systems*
2. SWAYAM: Linux and System Administration Courses
3. Coursera: *Linux Command Line Basics*
4. Official Documentation: https://www.kernel.org, https://access.redhat.com/documentation

Third Year B.Tech Information Technology															
Semester- V															
Course Code: IT24HON347R								**Course Name: Enterprise Linux System Administration**							
Teaching Scheme (Hours/Week)					**Examination Scheme**						**Credits**				
L	**P**	**T**	**OL**	**ODL**	**CIE**	**ETE**	**TW**	**OR**	**PR**	**TOTAL**	**L**	**P**	**T**	**ODL**	**TOTAL**
3	-	-	-	-	40	60	-	-	-	100	3	-	-	-	3

Course Objectives:

- To understand advanced Linux system administration concepts
- To manage users, processes, and system resources efficiently
- To configure and manage storage and file systems
- To administer system services and troubleshoot issues

Course Outcomes:

After learning the course, the students will be able to:

CO1: Manage users, groups, and security policies in Linux systems.

CO2: Monitor and control system processes and performance.

CO3: Configure and manage storage including partitions and LVM.

CO4: Administer system services, boot process, and logging mechanisms.

CO5: Troubleshoot system issues and ensure system reliability.

Unit	**Contents**	**Duration (Hrs.)**
1	**User and Group Management:** Creating, modifying, and deleting users and groups, Password management, Account aging and policies, User environment configuration, Sudo access and security	8
2	**Process and System Monitoring:** Process lifecycle, Process management commands (ps, top, htop, kill, nice), Foreground and background jobs, Job scheduling using cron and at, System monitoring tools and performance analysis	8
3	**Storage Management:** Disk partitioning (fdisk, parted), File systems (ext4, xfs), Logical Volume Management (LVM) concepts and configuration, Mounting and unmounting file systems, Swap space management	7
4	**System Services:** Linux boot process (BIOS/UEFI, GRUB), Managing services using systemd (start, stop, enable, disable), Service configuration, System logging (rsyslog, journalctl), Troubleshooting system services	8

5	**Security and Administration:** File permissions and ACLs, Firewall configuration (firewalld/iptables), SSH configuration, Basic SELinux concepts, Backup and recovery, System updates and patching	8
	Total Hours	**39**

Text Books

1. Nemeth, E., *UNIX and Linux System Administration Handbook*, Pearson
2. Sobell, M. G., *A Practical Guide to Linux Commands, Editors, and Shell Programming*, Pearson

Reference Books

1. Negus, C., *Linux Bible*, Wiley
2. Ward, B., *How Linux Works*, No Starch Press
3. Shotts, W., *The Linux Command Line*, No Starch Press

Online References

1. NPTEL – Linux/System Administration Courses
2. SWAYAM – Linux and Open Source Courses
3. Red Hat Documentation – https://access.redhat.com/documentation
4. Linux Foundation Training – https://training.linuxfoundation.org

Third Year B.Tech Information Technology															
Semester- VI															
Course Code: IT24HON348R Course Name: Enterprise Linux System Administration Lab															
Teaching Scheme (Hours/Week)					**Examination Scheme**						**Credits**				
L	**P**	**T**	**OL**	**ODL**	**CIE**	**ETE**	**TW**	**OR**	**PR**	**TOTAL**	**L**	**P**	**T**	**ODL**	**TOTAL**
-	4	-	-	-	-	-	25	-	25	50	-	2	-	-	2

Course Objectives:

- To understand advanced Linux system administration concepts
- To manage users, processes, and system resources efficiently
- To configure and manage storage and file systems
- To administer system services and troubleshoot issues

Course Outcomes:

After learning the course, the students will be able to:

CO1: Manage users, groups, and security policies in Linux systems.

CO2: Monitor and control system processes and performance.

CO3: Configure and manage storage including partitions and LVM.

CO4: Administer system services, boot process, and logging mechanisms.

CO5: Troubleshoot system issues and ensure system reliability.

Sr. No	List of Experiments	Duration (Hrs.)
1	Create and manage users and groups with permissions	4
2	Configure password policies and sudo access	4
3	Monitor system processes using ps, top, and kill commands	4
4	Schedule tasks using cron and at	4
5	Perform disk partitioning using fdisk/parted	4
6	Configure Logical Volume Management (LVM)	4
7	Mount and manage file systems and swap space	4
8	Manage services using systemctl	4

9	Analyze system logs using journalctl and log files	4
10	Configure basic firewall and SSH security	4
11	MINI PROJECT on Linux Server Administration and Security Automation System	12
	Total Hours	**52**

Text Books

1. Nemeth, E., *UNIX and Linux System Administration Handbook*, Pearson
2. Sobell, M. G., *A Practical Guide to Linux Commands, Editors, and Shell Programming*, Pearson

Reference Books

1. Negus, C., *Linux Bible*, Wiley
2. Ward, B., *How Linux Works*, No Starch Press
3. Shotts, W., *The Linux Command Line*, No Starch Press

Online References

1. NPTEL – Linux/System Administration Courses
2. SWAYAM – Linux and Open Source Courses
3. Red Hat Documentation – https://access.redhat.com/documentation
4. Linux Foundation Training – https://training.linuxfoundation.org

<table>
<tr><th colspan="15">Third Year B.Tech Information Technology</th></tr>
<tr><td colspan="15">Semester- VI</td></tr>
<tr><td colspan="15">Course Code: IT24HON397R Course Name: Linux Networking and Security Administration</td></tr>
<tr><th colspan="5">Teaching Scheme (Hours/Week)</th><th colspan="6">Examination Scheme</th><th colspan="4">Credits</th></tr>
<tr><th>L</th><th>P</th><th>T</th><th>OL</th><th>ODL</th><th>CIE</th><th>ETE</th><th>TW</th><th>OR</th><th>PR</th><th>TOTAL</th><th>L</th><th>P</th><th>T</th><th>ODL</th><th>TOTAL</th></tr>
<tr><td>2</td><td>-</td><td>-</td><td>-</td><td>-</td><td>40</td><td>60</td><td>-</td><td>-</td><td>-</td><td>100</td><td>2</td><td>-</td><td>-</td><td>-</td><td>2</td></tr>
<tr><td colspan="16">Course Objectives:
● To understand Linux networking concepts and configurations
● To configure and manage network services in Linux systems
● To implement security mechanisms and firewall configurations
● To monitor and troubleshoot network and security issues
● To develop skills in secure system administration practices</td></tr>
<tr><td colspan="16">Course Outcomes:
After learning the course, the students will be able to:
CO1: Configure and manage basic Linux networking settings.
CO2: Implement and manage network services in Linux.
CO3: Apply security measures including firewalls and access control.
CO4: Monitor and troubleshoot network and system security issues.
CO5: Ensure secure communication and system hardening in Linux environments.</td></tr>
</table>

Unit	Contents	Duration (Hrs.)
1	**Network Configuration:** IP addressing (IPv4/IPv6) and subnetting basics, Static and dynamic IP configuration, Hostname configuration, Network interface configuration (ifconfig/ip), Network troubleshooting tools (ping, traceroute, netstat, ss), DNS concepts and basic DNS configuration, Editing network configuration files.	6
2	**Remote Access and File Sharing:** SSH configuration and secure remote login, SSH key-based authentication, Secure file transfer (SCP, SFTP), Network File System (NFS) configuration, Samba file sharing setup, File sharing permissions and security, Remote administration best practices.	6
3	**Firewall and Security:** Firewall concepts, Firewall configuration using firewalld (zones, services, ports), Introduction to iptables, SELinux modes (enforcing, permissive, disabled) and basic policies, Access control lists (ACLs), System hardening techniques, User and network security best practices.	6
4	**Network Services:** Web server configuration (Apache HTTP Server), Virtual hosting basics, FTP server setup and configuration, Secure FTP practices, DHCP configuration basics, Service management and troubleshooting using systemctl.	6

5	**Monitoring and Troubleshooting:** Network monitoring tools (tcpdump, wireshark basics), Log analysis (/var/log, journalctl), Performance monitoring, Troubleshooting network and service issues, Intrusion detection basics, Backup and recovery strategies.	6
	Total Hours	**30**

Note for Course Coordinator

The End Term Examination (ETE) of 60 marks will be conducted at the department level. The mode, tools, and methodology for conducting the examination will be decided by the Course Coordinator, with due approval from the Chairperson of the Board of Studies (BoS).

Text Books

1. Nemeth, E., *UNIX and Linux System Administration Handbook*, Pearson
2. Hunt, C., *TCP/IP Network Administration*, O'Reilly

Reference Books

1. Negus, C., *Linux Bible*, Wiley
2. Ward, B., *How Linux Works*, No Starch Press
3. Shotts, W., *The Linux Command Line*, No Starch Press

Online References

1. NPTEL – Computer Networks / Linux Courses
2. SWAYAM – Networking and Security Courses
3. Red Hat Documentation – https://access.redhat.com/documentation
4. Linux Foundation – https://training.linuxfoundation.org

Third Year B.Tech Information Technology															
Semester- VI															
Course Code: IT24HON398R Course Name: Linux Networking and Security Administration Lab															
Teaching Scheme (Hours/Week)					**Examination Scheme**						**Credits**				
L	**P**	**T**	**OL**	**ODL**	**CIE**	**ETE**	**TW**	**OR**	**PR**	**TOTAL**	**L**	**P**	**T**	**ODL**	**TOTAL**
-	2	-	-	-	-	-	25	-	-	25	-	1	-	-	1

Course Objectives:

- To understand Linux networking concepts and configurations
- To configure and manage network services in Linux systems
- To implement security mechanisms and firewall configurations
- To monitor and troubleshoot network and security issues
- To develop skills in secure system administration practices

Course Outcomes:

After learning the course, the students will be able to:
CO1: Configure and manage basic Linux networking settings.
CO2: Implement and manage network services in Linux.
CO3: Apply security measures including firewalls and access control.
CO4: Monitor and troubleshoot network and system security issues.
CO5: Ensure secure communication and system hardening in Linux environments.

Sr. No	List of Experiments	Duration (Hrs.)
1	Configure IP address and hostname using `ip` and `hostnamectl` commands	2
2	Test network connectivity using ping, traceroute, netstat/ss	2
3	Configure and verify DNS settings and name resolution	2
4	Set up SSH server and perform remote login (password & key-based authentication)	4
5	Transfer files securely using SCP and SFTP	4
6	Configure NFS server and client for file sharing	2
7	Set up Samba file sharing between Linux systems	4

8	Configure firewall using firewalld (zones, ports, services)	2
9	Explore and manage SELinux modes and basic policies	4
10	Install and configure Apache Web Server and test web hosting	4
	Total Hours	**30**

Text Books

1. Nemeth, E., *UNIX and Linux System Administration Handbook*, Pearson
2. Hunt, C., *TCP/IP Network Administration*, O'Reilly

Reference Books

1. Negus, C., *Linux Bible*, Wiley
2. Ward, B., *How Linux Works*, No Starch Press
3. Shotts, W., *The Linux Command Line*, No Starch Press

Online References

1. NPTEL – Computer Networks / Linux Courses
2. SWAYAM – Networking and Security Courses
3. Red Hat Documentation – https://access.redhat.com/documentation
4. Linux Foundation – https://training.linuxfoundation.org

Final Year B. Tech Information Technology														
Semester VII														
Course Code: IT24HON447R							**Course Name: Linux Automation and DevOps with Ansible**							
Teaching Scheme (Hours/Week)					**Examination Scheme**						**Credits**			
L	**P**	**T**	**OL**	**ODL**	**CIE**	**ETE**	**TW**	**PR**	**OR**	**TOTAL**	**L**	**P**	**T**	**TOTAL**
2	-	-	-	-	40	60	-	-	-	100	2	-	-	2

Course Objectives:

- To understand automation concepts in Linux environments
- To learn configuration management using Ansible
- To automate system administration and deployment tasks
- To implement DevOps practices for continuous integration and delivery
- To enhance skills in infrastructure automation and orchestration

Course Outcomes:

After learning the course, the students will be able to:
CO1: Explain automation and DevOps concepts in Linux environments.
CO2: Configure and use Ansible for system management tasks.
CO3: Develop Ansible playbooks for automation and deployment.
CO4: Implement configuration management and orchestration using Ansible.
CO5: Apply DevOps practices for efficient system administration and deployment.

Unit	Contents	Hrs.
1	**Introduction to Automation and DevOps:** Concept of system administration and need for automation, Introduction to DevOps culture and practices, Traditional vs DevOps approach, Continuous Integration (CI), Continuous Delivery (CD), Continuous Deployment, DevOps lifecycle, Infrastructure as Code (IaC) concepts, Benefits of automation, Overview of configuration management tools, Introduction to Ansible, Ansible architecture (control node, managed nodes, inventory), Agentless architecture and SSH-based communication.	6
2	**Ansible Basics:** Installation and configuration of Ansible, Setting up control node and managed nodes, Inventory file structure (static and dynamic inventory), Host grouping and variables, Ansible ad-hoc commands, Common modules (ping, command, shell, file, copy, yum/apt), YAML basics and syntax, Running playbooks, Debugging basic Ansible tasks, Introduction to Ansible configuration files.	6
3	**Ansible Playbooks:** Structure of playbooks, Tasks and modules, Variables and facts, Conditionals (when), Loops (with_items), Handlers and notifications, Templates using Jinja2, Roles and directory structure, Reusability and modular design, Error handling in playbooks, Tagging and selective execution, Best practices for writing efficient playbooks.:	6

4	**Configuration Management and Automation:** Managing packages, services, and users using Ansible, File management (copy, template, lineinfile), Service configuration and automation, Deploying applications (e.g., web server setup), Managing configurations across multiple systems, Environment management, Version control integration (Git basics), Automating repetitive administrative tasks, Introduction to Ansible Galaxy	6
5	**DevOps Integration and Advanced Automation:** CI/CD pipeline concepts and workflow, Integration of Ansible with CI/CD tools (Jenkins basics), Automated provisioning and deployment, Infrastructure provisioning basics, Monitoring and logging basics, Security in automation, Secrets management (Ansible Vault), Performance optimization, Real-world DevOps use cases, Best practices in automation and orchestration.	6
	Total Hours	**30**

Note for Course Coordinator

The End Term Examination (ETE) of 60 marks will be conducted at the department level. The mode, tools, and methodology for conducting the examination will be decided by the Course Coordinator, with due approval from the Chairperson of the Board of Studies (BoS).

Text Books

1. Lorin Hochstein, *Ansible: Up and Running*, O'Reilly
2. Jeff Geerling, *Ansible for DevOps*, Leanpub

Reference Book

1. Kim, G., *The Phoenix Project*, IT Revolution Press
2. Morris, K., *Infrastructure as Code*, O'Reilly
3. Turnbull, J., *The Docker Book*, James Turnbull

Online References

1. Ansible Documentation – https://docs.ansible.com
2. NPTEL – DevOps / Cloud Computing Courses
3. SWAYAM – Cloud and DevOps Courses
4. Linux Foundation – https://training.linuxfoundation.org

Final Year B.Tech Information Technology															
Semester- VII															
Course Code: IT24HON448R					**Course Name: Linux Automation and DevOps with Ansible Lab**										
Teaching Scheme (Hours/Week)					**Examination Scheme**						**Credits**				
L	**P**	**T**	**OL**	**ODL**	**CIE**	**ETE**	**TW**	**OR**	**PR**	**TOTAL**	**L**	**P**	**T**	**ODL**	**TOTAL**
-	2	-	-	-	-	-	25	-	-	25	-	1	-	-	1

Course Objectives:

- To understand automation concepts in Linux environments
- To learn configuration management using Ansible
- To automate system administration and deployment tasks
- To implement DevOps practices for continuous integration and delivery
- To enhance skills in infrastructure automation and orchestration

Course Outcomes:

After learning the course, the students will be able to:

CO1: Explain automation and DevOps concepts in Linux environments.

CO2: Configure and use Ansible for system management tasks.

CO3: Develop Ansible playbooks for automation and deployment.

CO4: Implement configuration management and orchestration using Ansible.

CO5: Apply DevOps practices for efficient system administration and deployment.

Sr. No	List of Experiments	Duration (Hrs.)
1	Install and configure Ansible on control node	2
2	Set up inventory file and managed nodes	2
3	Execute Ad-hoc commands for system tasks	2
4	Write simple Ansible playbooks for package installation	2
5	Automate user and service management using playbooks	2
6	Use variables, loops, and conditionals in playbooks	2
7	Implement handlers and roles in Ansible	2
8	Manage files using templates and copy modules	4
9	Automate web server deployment using Ansible	4
10	Integrate Ansible with Git/Jenkins for basic CI/CD workflow	4
	Total Hours	**26**

Text Books

1. Lorin Hochstein, *Ansible: Up and Running*, O'Reilly

2. Jeff Geerling, *Ansible for DevOps*, Leanpub

Reference Book

1. Kim, G., *The Phoenix Project*, IT Revolution Press
2. Morris, K., *Infrastructure as Code*, O'Reilly
3. Turnbull, J., *The Docker Book*, James Turnbull

Online References

1. Ansible Documentation – https://docs.ansible.com
2. NPTEL – DevOps / Cloud Computing Courses
3. SWAYAM – Cloud and DevOps Courses
4. Linux Foundation – https://training.linuxfoundation.org

Final Year B.Tech Information Technology															
Semester- VIII															
Course Code: IT24HON495								**Course Name: Capstone Project**							
Teaching Scheme (Hours/Week)					**Examination Scheme**						**Credits**				
L	**P**	**T**	**OL**	**ODL**	**CIE**	**ETE**	**TW**	**OR**	**PR**	**TOTAL**	**L**	**P**	**T**	**ODL**	**TOTAL**
-	8	-	-	-	-	-	25	100	-	125	-	4	-	-	4

Course Objectives:

- To apply Linux administration concepts in a real-world project.
- To design and configure a Linux-based server environment.
- To implement security, user management, and networking services.
- To develop automation, monitoring, and troubleshooting skills in Linux systems.

Course Outcomes:

After learning the course, the students will be able to:

CO1: Design and configure a Linux-based server environment.
CO2: Implement user management, security, and access control mechanisms.
CO3: Apply shell scripting for automation of administrative tasks.
CO4: Perform system monitoring, troubleshooting, and backup management.

Guidelines for Capstone Project Selection and Execution

1. Select a project based on real-world enterprise Linux administration problems (e.g., server setup, network services, security implementation).
2. Ensure the project covers core Linux administration concepts such as user management, networking, security, and automation.
3. Define clear objectives, scope, and expected outcomes before starting the project.
4. Use appropriate tools and technologies (Linux distributions, shell scripting, server tools).
5. Follow a structured approach: planning → design → implementation → testing → deployment.
6. Maintain proper documentation including system configuration, commands used, and screenshots.
7. Implement security measures such as firewalls, permissions, and access controls.
8. Include testing and troubleshooting to ensure system reliability and performance.
9. Demonstrate backup and recovery mechanisms in the project.
10. Present the project with a clear report and practical demonstration.

Suggested Timeline for Capstone Project

1. **Week 1–2:** Project selection, problem definition, and approval
2. **Week 3–4:** Requirement analysis and project planning
3. **Week 5–7:** System design and initial setup (Linux environment, tools)
4. **Week 8–11:** Implementation and configuration of services
5. **Week 12–13:** Testing, debugging, and performance tuning
6. **Week 14:** Documentation preparation and report writing
7. **Week 15:** Final presentation and project demonstration

Guidelines for Capstone Project Report

1. GInclude a clear title, abstract, and objectives of the project.
2. Provide an introduction and problem statement explaining the need for the project.
3. Describe the system design and architecture with diagrams where necessary.
4. Explain the implementation details including tools, technologies, and configurations used.
5. Include screenshots, commands, and outputs to support the work done.
6. Document testing, results, and performance analysis.
7. Highlight security measures, backup, and recovery mechanisms implemented.
8. Provide conclusion and future scope of the project.
9. Add proper references and bibliography.
10. Ensure the report is well-structured, formatted, and plagiarism-free.

Guidelines for Assessment

The Capstone Project will be evaluated based on **Term Work (TW) and Oral/Practical (OR)**.

Term Work (TW) – 50 Marks

Criteria	Marks
Project topic relevance and problem definition	05
Literature review and technical depth	05
Project design and implementation progress	05
Project report documentation	05
Attendance and regular interaction with guide	05
Total	**25**

Oral / Practical Examination (OR) – 100 Marks

Criteria	Marks
Demonstration of working project	40
Understanding concepts and implementation	25
Innovation and technical contribution	15
Presentation and communication skills	10
Question and answer session	10
Total	**100**

Marathwada Mitra Mandal's College of Engineering

Karvenagar, Pune- 52

An Autonomous Institute Affiliated to SPPU

“येथे बहुतांचे हित”

Curriculum Structure and Syllabus

Honours

A.Y. 2026-27

Accredited with A++ grade by NAAC
Recipient of Best College award by SPPU | Accredited by NBA
Recognized under 2(f) and 12(B) of UGC Act 1956
www.mmcoe.edu.in

About Marathwada Mitra Mandal's College of Engineering

MMCOE has been recognised as the BEST COLLEGE by SPPU. MMCOE is one feather in the cap of Marathwada Mitra Mandal's Pune, from the academic year 2006 – 2007.

The Institute offers six undergraduate engineering programmes in Electrical Engineering, Mechanical Engineering, Electronics & Telecommunication Engineering, Information Technology, Computer Engineering, and Artificial Intelligence & Data Science. In addition, it offers six postgraduate programmes M.Tech (Electrical – Power Electronics & Drives), M.Tech (Mechanical – Design Engineering), M.Tech (Electronics & Telecommunication – VLSI & Embedded Systems), M.Tech (Information Technology – Data Science), M.Tech (Computer Engineering), along with a Ph.D. programme in Computer Engineering. It houses 2000+ students and 200 staff members. Along with the focus on core academics, college also takes care of overall personality development of the students.

The college has an active robotics cell, NCC, Training and Placement cell etc. The students are motivated to actively participate in co-curricular and extra-curricular activities. They have brought many laurels to the Institute in the events such as Vinodottam Karandak, Firodiya Karandak, Zest, Dexterity etc.

Institute Vision

To be a globally renowned institution through excellence in engineering education for sustainable and holistic development

Institute Mission

M1: Empower students with cutting-edge technologies and global competencies

M2: Foster culture of research and entrepreneurial mindset

M3: Imbibe social and professional values

M4: Provide an inclusive environment for lifelong learning

Department of
Computer Engineering

Vision and Mission of the Department

- **Vision**

To excel in computer engineering education for sustainable development.

- **Mission**

M1: To develop globally competent professionals through interdisciplinary learning and Center of Excellence.

M2: To empower research, innovation, and entrepreneurial thinking

M3: To promote ethical values and holistic development

M4: To prepare graduates for lifelong learning and dynamic careers

- **Program Specific Outcomes (PSOs)**

A Graduate of the Computer Engineering Program will be able to

PSO1: Apply problem solving skills to design effective solutions in High Performance Computing, Artificial Intelligence and Cyber Security

PSO2: Develop advanced skill-based solutions using standard Software Engineering practices

- **Program Outcomes (POs)**

PO1: Engineering Knowledge: Apply knowledge of mathematics, natural science, computing, engineering fundamentals and an engineering specialization as specified in WK1 to WK4 respectively to develop to the solution of complex engineering problems.

PO2: Problem Analysis: Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development. (WK1 to WK4)

PO3: Design/Development of Solutions: Design creative solutions for complex engineering problems and design/develop systems/components/processes to meet identified needs with consideration for the public health and safety, whole-life cost, net zero carbon, culture, society and environment as required. (WK5)

PO4: Conduct Investigations of Complex Problems: Conduct investigations of complex engineering problems using research-based knowledge including design of experiments, modelling, analysis & interpretation of data to provide valid conclusions. (WK8).

PO5: Engineering Tool Usage: Create, select and apply appropriate techniques, resources and modern engineering & IT tools, including prediction and modelling recognizing their limitations to solve complex engineering problems. (WK2 and WK6)

PO6: The Engineer and The World: Analyze and evaluate societal and environmental aspects while solving complex engineering problems for its impact on sustainability with reference to economy, health, safety, legal framework, culture and environment. (WK1, WK5, and WK7).

PO7: Ethics: Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national & international laws. (WK9)

PO8: Individual and Collaborative Team work: Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams.

PO9: Communication: Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations considering cultural, language, and learning differences

PO10: Project Management and Finance: Apply knowledge and understanding of engineering management principles and economic decision-making and apply these to one's own work, as a member and leader in a team, and to manage projects and in multidisciplinary environments.

PO11: Life-Long Learning: Recognize the need for, and have the preparation and ability for i) independent and life-long learning ii) adaptability to new and emerging technologies and iii) critical thinking in the broadest context of technological change. (WK8)

Curriculum for B. Tech. in Computer Engineering with Honours in Green Computing Engineering

Department of Computer Engineering

General Objectives and Scope:

Honors degrees offer a more in-depth study of the field in which one is pursuing honors. It might help the student obtain a greater chance at a decent employment than someone with a generic degree. The honors degree entails extensive study that culminates in a research potential in a student.

Completing an honors degree not only prepares one for higher education in a specific field, but also demonstrates to prospective employers a greater capacity for independent learning and research, as well as better problem-solving, critical thinking, independent learning, and communication abilities.

A bachelor's degree with honors will increase employability. This demonstrates to a recruiter that you can work diligently to earn good grades. It demonstrates ambition, self-discipline, and organization, among other excellent characteristics. responsibility.

Why Green Computing Engineering?

Green Computing Engineering is essential because modern computing technologies consume vast amounts of energy and contribute significantly to electronic waste and environmental pollution. By adopting green computing practices, we can reduce carbon footprints, save energy, and extend the lifespan of hardware. It promotes sustainable use of resources through efficient software design, virtualization, and eco-friendly hardware. Additionally, it helps organizations comply with environmental regulations while lowering operational costs. In essence, green computing ensures that technological progress aligns with environmental preservation and a sustainable future.

Prerequisites :

1. Fundamentals of Computer Science
2. Programming and Software Development
3. Knowledge of Energy and Environmental Science
4. Networking and Cloud Computing
5. Data Analysis and Performance Monitoring
6. Awareness of Regulations and Standards

B. TECH HONOURS IN GREEN COMPUTING Engineering (Semester IV, V, VI, VII, VIII)

Curriculum Structure

Course Code	Course Name	Teaching Scheme (Hrs/week)				Examination Scheme						Credits				
		L	P	T	ODL	CIE	ETE	TW	PR	OR	Total	L	P	T	ODL	Total
Semester IV																
CE24HON291R	Foundations of Green Computing Development	3		-	-	40	60	-	-	-	100	3		-	-	3
CE24HON292R	Foundations of Green Computing Development Lab	-	4		-	-	-	50	-	-	50	-	2	-	-	2
Semester V																
CE24HON341R	Green Full-Stack & Carbon-Aware Systems	3	-	-	-	40	60		-	-	100	3	-	-	-	3
CE24HON342R	Green Full-Stack & Carbon-Aware Systems Lab	-	4	-	-	-	-	50	-	-	50	-	2	-	-	2
Semester VI																
CE24HON391R	Sustainable Cloud, DevOps & Data Platforms	3	-		-	40	60*	-	-		100	3	-	-	-	3
CE24HON392R	Sustainable Cloud, DevOps & Data Platforms Lab	-	4	-	-	-	-	50	-	-	50	-	2	-	-	2
Semester VII																
CE24HON441R	Green Computing Industry Practices	-		-	1	-	-	25	-	-	25	-	-	-	1	1
CE24HON442R	Internship on Green Computing Industry Practices	-	4	-	-	-	-	50	-	-	50	-	2	-	-	2
Semester VIII																
CE24HON491R	Green Computing Capstone Project	-	4	-	-	-	-	50	-	-	50	-	2	-	-	2
Total		09	20	–	01	120	180	275	-	-	575	09	10	-	01	20

*** Mode of conduction of ETE will be decided by course coordinator in the supervision of BOS chairperson.**

L- Lecture
P- Practical
T- Tutorial
CIE-Continuous Internal Evaluation
ETE- End Term Examination
TW- Term work
PR- Practical
OR- Oral
L : 1 Hr.= 1 credit
P: 2 Hr. = 1 Credit
T: 1 Hr. = 1 Credit
ODL-Open Distance Learning

Dr. K. S. Thakre
BOS Chairperson

Second Year B.Tech Computer Engineering															
Semester- IV															
Course Code: CE24HON291R					**Course Name: Foundations of Green Computing Development**										
Teaching Scheme (Hours/Week)					**Examination Scheme**						**Credits**				
L	**P**	**T**	**OL**	**ODL**	**CIE**	**ETE**	**TW**	**OR**	**PR**	**TOTAL**	**L**	**P**	**T**	**ODL**	**TOTAL**
3	-	-	-	-	40	60	-	-	-	100	3	-	-	-	3

Course Objectives:

- Introduce the principles of green and sustainable software engineering.
- Develop professional software development skills aligned with sustainability goals.
- Enable students to design and implement energy-efficient and carbon-aware software systems.
- Familiarize students with sustainable practices in full-stack development, cloud computing, and DevOps.
- Promote awareness of ESG (Environmental, Social, and Governance) considerations in software solutions.
- Enhance professional communication, teamwork, and ethical responsibility.

Course Outcomes:

After learning the course, the students will be able to:

CO1: Apply professional and Green Computing principles in software development.

CO2: Design and implement energy-efficient full-stack applications using AI.

CO3: Measure, analyze, and optimize the carbon footprint of software systems using Agentic AI.

CO4: Apply sustainable cloud, DevOps, and data engineering practices.

CO5: Design software solutions aligned with ESG and sustainability goals using Agentic AI.

CO6: Demonstrate professional communication, documentation, and ethical responsibility.

Unit	Contents	Duration (Hrs.)
1	**Introduction to Green Computing Development** Professional Software Development: Difference between amateur and professional developers, Software craftsmanship and engineering ethics. Modern Software Architecture: Overview of 7-layer modern application architecture, Mapping sustainability concerns across layers. Green Computing Development: Definition, principles, and objectives, Role of software in climate change mitigation. Introduction to Green Coding practices.	6
2	**Professional & Green Computing Development Tools** Version control systems: Git fundamentals, GitHub workflows: branching, pull requests, code reviews, Build and dependency management tools, Introduction to software quality and code analysis tools. Carbon measurement tools: Introduction to Code Carbon, Hands-on measurement of software emission.	6

3	**Green Requirements Engineering & Client-Side Development** Software Requirements Specification (SRS): Functional and non-functional requirements, Green Non-Functional Requirements (GNFRs). Carbon footprint fundamentals: Composition of carbon footprint, Sample footprint calculation, Carbon offsetting concepts. Client-side development: Energy-efficient UI/UX design, Page weight reduction techniques, HTML, CSS, JavaScript (Hands-on). Introduction to Low-Code / No-Code platforms.	6
4	**Server-Side & Database Development** Middleware & APIs: RESTful services and API design,API efficiency and sustainability,Postman (Hands-on). Application development models: Cloud-native vs on-premise systems, Energy impact of programming languages. Green Architecture: Design principles, Monitoring and operational sustainability. Database systems: Data foundation concepts, HOT / WARM / COLD storage strategies, Sustainable data management.	10
5	**Cloud Readiness & Professional Skills** Introduction to cloud computing and sustainability, Containers and Docker fundamentals, CI/CD overview using Jenkins, Professional skills: Technical presentation skills, Communicating sustainability impact.	6
6	**Responsible AI for fundamentals** What is Responsible AI - Fairness, Accountability, Transparency, Explainability, Sustainability. AI Carbon Footprint. Training vs inference energy cost. Model size vs environmental impact. Green AI vs Red AI (efficiency-first vs scale-first AI). Ethical AI use in sustainability systems. Regulatory overview (EU AI Act – high-level, non-legal).	6
7	**Responsible AI for Green Computing** Bias in sustainability data. Responsible data collection for climate & energy datasets. Explainable AI (XAI) for energy decisions. Responsible automation vs over-automation. AI lifecycle sustainability (data → model → deployment)	5
	Total Hours	45

Text Books

1. Coral Calero, Mario Piattini, **"Green in Software Engineering"**
2. Robert Cecil Martin, **"Clean Code"**

Reference Books

1. Brunna C Mourao, Leila Karita, Ivan do Carmo Machado, **"Green and Sustainable Software Engineering - a Systematic Mapping Study"**
2. Coral Calero, Mario Piattini, **"Introduction to Green in Software Engineering"**

Online References

1. https://learn.greensoftware.foundation/
2. https://kaspergroesludvigsen.medium.com/the-10-most-energy-efficient-programming-languages-6a4165126670
3. https://greenlab.di.uminho.pt/wp-content/uploads/2017/10/sleFinal.pdf
4. https://www.future-processing.com/blog/everything-you-need-to-know-about-green-and-sustainable-software/
5. https://geekflare.com/green-computing-for-sustainable-future/

Second Year B. Tech Computer Engineering															
Semester IV															
Course Code: CE24HON292R								**Course Name:Foundations of Green Computing Development Lab**							
Teaching Scheme (Hours/Week)					**Examination Scheme**						**Credits**				
L	**P**	**T**	**OL**	**ODL**	**CIE**	**ETE**	**TW**	**PR**	**OR**	**TOTAL**	**L**	**P**	**T**	**ODL**	**TOTAL**
-	4	-	-	-	-	-	50	-	-	50	-	2	-	-	2

Course Objectives:

- Introduce the principles of green and sustainable software engineering.
- Develop professional software development skills aligned with sustainability goals.
- Enable students to design and implement energy-efficient and carbon-aware software systems.
- Familiarize students with sustainable practices in full-stack development, cloud computing, and DevOps.
- Promote awareness of ESG (Environmental, Social, and Governance) considerations in software solutions.
- Enhance professional communication, teamwork, and ethical responsibility.

Course Outcomes:

After completing the course, the students will be able to:

CO1: Apply professional and Green Computing principles in software development.

CO2: Design and implement energy-efficient full-stack applications using AI.

CO3: Measure, analyze, and optimize the carbon footprint of software systems using Agentic AI.

CO4: Apply sustainable cloud, DevOps, and data engineering practices.

CO5: Design software solutions aligned with ESG and sustainability goals using Agentic AI.

CO6: Demonstrate professional communication, documentation, and ethical responsibility.

Week	Contents	Hrs
1	Git & GitHub professional workflows (branching, PRs, reviews)	4
2	Static code analysis and refactoring for energy efficiency	4
3	Integration of Code Carbon for software emission measurement.	4
4	Preparation of SRS with Green Non-Functional Requirements	4
5	Energy-efficient UI development using HTML, CSS, JavaScript	4
6	REST API development and testing using Postman	4
7	Database storage optimization (HOT / WARM / COLD tiers)	4
	Total Hours	**28**

Mini Projects

1. Console & API-based Green E-Commerce Application
2. Console-based green application (E-commerce use case)

AI Experiments

1. Measuring AI Carbon Footprint
2. Responsible Model Selection
3. Explainable AI for Energy Prediction

Third Year B. Tech Computer Engineering														
Semester V														
Course Code: CE24HON341R						Course Name: Green Full-Stack & Carbon -Aware Systems								
Teaching Scheme (Hours/Week)					Examination Scheme						Credits			
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	TOTAL
3	-	-	-	-	40	60	-	-	-	100	3	-	-	3

Course Objectives:

- Introduce the principles of green and sustainable software engineering.
- Develop professional software development skills aligned with sustainability goals.
- Enable students to design and implement energy-efficient and carbon-aware software systems.
- Familiarize students with sustainable practices in full-stack development, cloud computing, and DevOps.
- Promote awareness of ESG (Environmental, Social, and Governance) considerations in software solutions.
- Enhance professional communication, teamwork, and ethical responsibility.

To **Course Outcomes:**

After completing the course, the students will be able to:

CO1: Apply professional and Green Computing principles in software development.

CO2: Design and implement energy-efficient full-stack applications using AI.

CO3: Measure, analyze, and optimize the carbon footprint of software systems using Agentic AI.

CO4: Apply sustainable cloud, DevOps, and data engineering practices.

CO5: Design software solutions aligned with ESG and sustainability goals using Agentic AI.

CO6: Demonstrate professional communication, documentation, and ethical responsibility.

Unit	Contents	Hrs
1	**Sustainable Front-End Engineering** Sustainable UI design principles, Progressive Web Applications (PWA), Performance metrics: Core Web Vitals, Front-end performance and energy audits, Accessibility and sustainability	8
2	**Backend Optimization & API Sustainability** Efficient backend architectures, API payload optimization, Caching strategies and energy efficiency, Asynchronous processing, Backend performance monitoring	8
3	**Green Databases & Data Management** SQL vs NoSQL energy comparison, Query optimization techniques, Indexing vs storage trade-offs, Data lifecycle management, Green data retention strategies	8
4	**Carbon-Aware System Design** Carbon-aware application design principles, Integration of carbon measurement tools, Sustainability dashboards, Metrics and KPIs for Green Computing	7
5	**Agentic AI Concepts** What is Agentic AI?, Difference: Scripted automation vs autonomous agents, Agent lifecycle: Guardrails & sustainability constraint	7

6	**Agentic AI for Green Applications** AI agents for: Energy optimization, Load balancing, Carbon-aware scheduling, Human-in-the-loop systems, Responsible autonomy in green systems	7
	Total Hours	45

Text Books

1. Coral Calero, Mario Piattini, **"Green in Software Engineering"**
2. Robert Cecil Martin, **"Clean Code"**

Reference Book

1. Brunna C Mourao, Leila Karita, Ivan do Carmo Machado, **"Green and Sustainable Software Engineering - a Systematic Mapping Study"**
2. Coral Calero, Mario Piattini, **"Introduction to Green in Software Engineering"**

Online References

1. https://learn.greensoftware.foundation/
2. https://kaspergroesludvigsen.medium.com/the-10-most-energy-efficient-programming-languages-6a4165126670
3. https://greenlab.di.uminho.pt/wp-content/uploads/2017/10/sleFinal.pdf
4. https://www.future-processing.com/blog/everything-you-need-to-know-about-green-and-sustainable-software/
5. https://geekflare.com/green-computing-for-sustainable-future/

Third Year B. Tech Computer Engineering															
Semester V															
Course Code:CE24HON342R						Course Name: Green Full-Stack & Carbon -Aware Systems Lab									
Teaching Scheme (Hours/Week)					Examination Scheme						Credits				
L	P	T	OL	ODL	CIE	ETE	TW	PR	OR	TOTAL	L	P	T	ODL	TOTAL
-	4	-	-	-	-	-	50	-	-	50	-	2	-	-	2

Course Objectives:

- Introduce the principles of green and sustainable software engineering.
- Develop professional software development skills aligned with sustainability goals.
- Enable students to design and implement energy-efficient and carbon-aware software systems.
- Familiarize students with sustainable practices in full-stack development, cloud computing, and DevOps.
- Promote awareness of ESG (Environmental, Social, and Governance) considerations in software solutions.
- Enhance professional communication, teamwork, and ethical responsibility.

Course Outcomes:

After completing the course, the students will be able to:

CO1: Apply professional and Green Computing principles in software development.

CO2: Design and implement energy-efficient full-stack applications using AI.

CO3: Measure, analyze, and optimize the carbon footprint of software systems using Agentic AI.

CO4: Apply sustainable cloud, DevOps, and data engineering practices.

CO5: Design software solutions aligned with ESG and sustainability goals using Agentic AI.

CO6: Demonstrate professional communication, documentation, and ethical responsibility.

Unit	Contents	Hrs
1	Front-end performance audit using Lighthouse	4
2	Page-weight and rendering optimization techniques	6
3	API payload optimization and response compression	4
4	Energy comparison of SQL vs NoSQL databases	4
5	Carbon-aware system design using Code Carbon metrics	6
6	Sustainability dashboard creation	4
	Total Hours	**28**

Mini Projects

1. Full-Stack Carbon-Aware Application

AI Experiments

1. Rule-Based Green AI Agent
2. Carbon-Aware Backend Agent
3. Explainable AI for Energy Prediction

Third Year B. Tech Computer Engineering															
Semester VI															
Course Code: CE24HON391R					Course Name: Sustainable Cloud, DevOps & Data Platforms										
Teaching Scheme (Hours/Week)					Examination Scheme						Credits				
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	ODL	TOTAL
3	-	-	-	-	40	60	-	-	-	100	3	-	-	-	3

Course Objectives:

- Introduce the principles of green and sustainable software engineering.
- Develop professional software development skills aligned with sustainability goals.
- Enable students to design and implement energy-efficient and carbon-aware software systems.
- Familiarize students with sustainable practices in full-stack development, cloud computing, and DevOps.
- Promote awareness of ESG (Environmental, Social, and Governance) considerations in software solutions.
- Enhance professional communication, teamwork, and ethical responsibility.

Course Outcomes:

After completing the course, the students will be able to:

CO1: Apply professional and Green Computing principles in software development.

CO2: Design and implement energy-efficient full-stack applications using AI.

CO3: Measure, analyze, and optimize the carbon footprint of software systems using Agentic AI.

CO4: Apply sustainable cloud, DevOps, and data engineering practices.

CO5: Design software solutions aligned with ESG and sustainability goals using Agentic AI.

CO6: Demonstrate professional communication, documentation, and ethical responsibility.

Unit	Contents	Hrs
1	**Green Cloud Computing** Cloud service models and sustainability, Carbon intensity of cloud regions, Green cloud architecture design, Cost vs energy trade-offs.	8
2	**Containers & Green DevOps** Containerization best practices, Sustainable CI/CD pipelines, DevOps KPIs with sustainability metrics, Automation for energy optimization	8
3	**Observability, Monitoring & Optimization** Monitoring resource utilization, Energy monitoring in production systems, Auto-scaling strategies, Sustainable Site Reliability Engineering (SRE)	8
4	**Sustainable Data Engineering & Security** Energy-efficient data pipelines, ETL vs ELT sustainability, Data minimization principles, Security vs performance vs energy trade-offs	7
5	**Agentic AI in Cloud & DevOps** AI agents vs traditional auto-scaling, Carbon-aware cloud decisions, Multi-objective optimization: Cost, Performance, Emissions, Green MLOps concepts	7
6	**Sustainable Autonomous Systems** Agent failure modes, Safety & rollback mechanisms, Observability for AI agents, Ethical AI in infrastructure automation	7

Total Hours	45

Text Books

1. Coral Calero, Mario Piattini, **"Green in Software Engineering"**
2. Robert Cecil Martin, **"Clean Code"**

Reference Book

1. Brunna C Mourao, Leila Karita, Ivan do Carmo Machado, **"Green and Sustainable Software Engineering - a Systematic Mapping Study"**
2. Coral Calero, Mario Piattini, **"Introduction to Green in Software Engineering"**

Online References

1. https://learn.greensoftware.foundation/
2. https://kaspergroesludvigsen.medium.com/the-10-most-energy-efficient-programming-languages-6a4165126670
3. https://greenlab.di.uminho.pt/wp-content/uploads/2017/10/sleFinal.pdf
4. https://www.future-processing.com/blog/everything-you-need-to-know-about-green-and-sustainable-software/
5. https://geekflare.com/green-computing-for-sustainable-future/

Third Year B. Tech Computer Engineering															
Semester VI															
Course Code:CE24HON392R					Course Name: Sustainable Cloud, DevOps & Data Platforms Lab										
Teaching Scheme (Hours/Week)					Examination Scheme						Credits				
L	P	T	OL	ODL	CIE	ETE	TW	PR	OR	TOTAL	L	P	T	ODL	TOTAL
-	4	-	-	-	-	-	50	-	-	50	-	2	-	-	2

Course Objectives:

- Introduce the principles of green and sustainable software engineering.
- Develop professional software development skills aligned with sustainability goals.
- Enable students to design and implement energy-efficient and carbon-aware software systems.
- Familiarize students with sustainable practices in full-stack development, cloud computing, and DevOps.
- Promote awareness of ESG (Environmental, Social, and Governance) considerations in software solutions.
- Enhance professional communication, teamwork, and ethical responsibility.

Course Outcomes:

After completing the course, the students will be able to:

CO1: Apply professional and Green Computing principles in software development.

CO2: Design and implement energy-efficient full-stack applications using AI.

CO3: Measure, analyze, and optimize the carbon footprint of software systems using Agentic AI.

CO4: Apply sustainable cloud, DevOps, and data engineering practices.

CO5: Design software solutions aligned with ESG and sustainability goals using Agentic AI.

CO6: Demonstrate professional communication, documentation, and ethical responsibility.

Unit	Contents	Hrs
1	Docker container optimization for reduced resource usage	4
2	CI/CD pipeline implementation using Jenkins / GitHub Actions	6
3	Cloud deployment with region-based carbon awareness	4
4	Monitoring energy and resource utilization in production	4
5	Sustainable data pipeline design	6
6	Security vs energy trade-off analysis	4
	Total Hours	**28**

Mini Projects

1. Cloud-Native Green Application

AI Experiments

1. AI Agent for Cloud Resource Optimization
2. Green DevOps Agent

Final Year B. Tech Computer Engineering															
Semester VII															
Course Code:CE24HON441R					Course Name: Green Computing Industry Practices										
Teaching Scheme (Hours/Week)					Examination Scheme						Credits				
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	ODL	TOTAL
-	-	-	-	1	-	-	25	-	-	25	-	-	-	1	1

Course Objectives:

- Introduce the principles of green and sustainable software engineering.
- Develop professional software development skills aligned with sustainability goals.
- Enable students to design and implement energy-efficient and carbon-aware software systems.
- Familiarize students with sustainable practices in full-stack development, cloud computing, and DevOps.
- Promote awareness of ESG (Environmental, Social, and Governance) considerations in software solutions.
- Enhance professional communication, teamwork, and ethical responsibility.

Course Outcomes:

After completing the course, the students will be able to:

CO1: Apply professional and Green Computing principles in software development.

CO2: Design and implement energy-efficient full-stack applications using AI.

CO3: Measure, analyze, and optimize the carbon footprint of software systems using Agentic AI.

CO4: Apply sustainable cloud, DevOps, and data engineering practices.

CO5: Design software solutions aligned with ESG and sustainability goals using Agentic AI.

CO6: Demonstrate professional communication, documentation, and ethical responsibility.

Unit	Contents	Hrs
1	**ESG & Sustainable Technology** ESG fundamentals for engineers, Role of IT in ESG reporting, Sustainable product development, Industry case studies	3
2	**Industry-Scale Green Architectures** Monolith vs microservices (energy perspective), Event-driven architectures, Scalable and sustainable systems, Open-source Green Computing initiatives	3
3	**Capstone Project (Practical Intensive)** Problem identification and sustainability justification, Green system architecture design, Full-stack and cloud implementation, Carbon footprint measurement, ESG-aligned documentation, Final demo, presentation, and viva	3
4	**Advanced Agentic AI** Multi-agent systems, Coordination & negotiation, Ethical autonomy limits, AI governance framework	3
5	**Agentic AI for ESG & Sustainability** Agents for ESG data collection, AI-driven sustainability reporting, Decision traceability for audits	3
	Total Hours	15

Text Books

1. Coral Calero, Mario Piattini, **"Green in Software Engineering"**
2. Robert Cecil Martin, **"Clean Code"**

Reference Book
• Brunna C Mourao, Leila Karita, Ivan do Carmo Machado, **"Green and Sustainable Software Engineering - a Systematic Mapping Study"** • Coral Calero, Mario Piattini, **"Introduction to Green in Software Engineering"**
Online References
1. https://learn.greensoftware.foundation/ 2. https://kaspergroesludvigsen.medium.com/the-10-most-energy-efficient-programming-languages-6a4165126670 3. https://greenlab.di.uminho.pt/wp-content/uploads/2017/10/sleFinal.pdf 4. https://www.future-processing.com/blog/everything-you-need-to-know-about-green-and-sustainable-software/ 5. https://geekflare.com/green-computing-for-sustainable-future/

Final Year B. Tech Computer Engineering															
Semester VII															
Course Code: CE24HON442R					Course Name: Internship on Green Computing Industry Practices										
Teaching Scheme (Hours/Week)					Examination Scheme						Credits				
L	P	T	OL	ODL	CIE	ETE	TW	PR	OR	TOTAL	L	P	T	ODL	TOTAL
-	4	-	-	-	-	-	50	-	-	50	-	2	-	-	2

Course Objectives:

- Introduce the principles of green and sustainable software engineering.
- Develop professional software development skills aligned with sustainability goals.
- Enable students to design and implement energy-efficient and carbon-aware software systems.
- Familiarize students with sustainable practices in full-stack development, cloud computing, and DevOps.
- Promote awareness of ESG (Environmental, Social, and Governance) considerations in software solutions.
- Enhance professional communication, teamwork, and ethical responsibility.

Course Outcomes:

After completing the course, the students will be able to:

CO1: Apply professional and Green Computing principles in software development.

CO2: Design and implement energy-efficient full-stack applications using AI.

CO3: Measure, analyze, and optimize the carbon footprint of software systems using Agentic AI.

CO4: Apply sustainable cloud, DevOps, and data engineering practices.

CO5: Design software solutions aligned with ESG and sustainability goals using Agentic AI.

CO6: Demonstrate professional communication, documentation, and ethical responsibility.

Sr. No.	Industry Practices for Internship
1	ESG & Sustainable Technology
2	Industry-Scale Green Architectures
3	Capstone Project (Practical Intensive)
4	Advanced Agentic AI
5	Agentic AI for ESG & Sustainability
6	ESG & Sustainable Technology
	Total Hours 30

Mandatory Guidelines
The internship is designed to help students understand the role of technology in sustainability, ESG practices, and responsible AI systems. Students are expected to actively explore the given topics, connect theoretical concepts with real-world applications, and develop solutions that promote energy efficiency and sustainable system design. Throughout the internship, students should maintain regular interaction with mentors, seek feedback, and document their learning and progress in a systematic manner. They are encouraged to review research articles, industry case studies, and sustainability reports to gain a broader understanding of how organizations implement ESG practices using technology. While working on different modules, students should adopt a problem-solving and analytical approach, experiment with different ideas, and evaluate the effectiveness of their solutions. They should focus on designing systems that are efficient, scalable, and environmentally responsible. Students are also expected to follow ethical guidelines in AI and data usage, ensuring transparency, fairness, and respect for privacy. Proper version control, testing of developed solutions, and maintaining clear documentation of system design and implementation should be practiced throughout the internship. Students should also demonstrate professional behavior, time management, and teamwork skills when interacting with mentors or collaborators. They are encouraged to think creatively, suggest improvements, and reflect on how their technical work contributes to sustainability and ESG goals. At the end of the internship, students must compile their work into a well-structured report, clearly describing the problem addressed, methodology used, system design, and sustainability impact. The internship will conclude with a final presentation, system demonstration, and viva, where students will share their learning experience and outcomes of the project.

Final Year B. Tech Computer Engineering															
Semester VIII															
Course Code: CE24HON491R								Course Name: Green Computing Capstone Project							
Teaching Scheme (Hours/Week)					Examination Scheme						Credits				
L	P	T	OL	ODL	CIE	ETE	TW	PR	OR	TOTAL	L	P	T	ODL	TOTAL
-	4	-	-	-	-	-	50	-	-	50	-	2	-	-	2

Course Objectives:

- Introduce the principles of green and sustainable software engineering.
- Develop professional software development skills aligned with sustainability goals.
- Enable students to design and implement energy-efficient and carbon-aware software systems.
- Familiarize students with sustainable practices in full-stack development, cloud computing, and DevOps.
- Promote awareness of ESG (Environmental, Social, and Governance) considerations in software solutions.
- Enhance professional communication, teamwork, and ethical responsibility.

Course Outcomes:

After completing the course, the students will be able to:

CO1: Apply professional and Green Computing principles in software development.

CO2: Design and implement energy-efficient full-stack applications using AI.

CO3: Measure, analyze, and optimize the carbon footprint of software systems using Agentic AI.

CO4: Apply sustainable cloud, DevOps, and data engineering practices.

CO5: Design software solutions aligned with ESG and sustainability goals using Agentic AI.

CO6: Demonstrate professional communication, documentation, and ethical responsibility.

Unit	Contents	Hrs
1	Capstone project proposal & sustainability justification	4
2	Green architecture design review	4
3	Full-stack cloud implementation	4
4	Carbon footprint measurement & ESG reporting	4
5	Performance optimization & final deployment	4
6	Technical paper / report preparation	4
7	Final demo, presentation, and viva voce	4
	Total Hours	**28**

Mandatory Capstone AI Requirements

- Group of 3-4 students

Each project must include:

- At least one AI agent
- Responsible AI justification
- Carbon measurement
- ESG alignment report
- Human-in-the-loop design

Marathwada Mitra Mandal's College of Engineering

Karvenagar, Pune- 52

An Autonomous Institute Affiliated to SPPU

" येथे बहुतांचे हित "

Curriculum Structure and Syllabus

Honours

A.Y. 2026-27

Accredited with A++ grade by NAAC
Recipient of Best College award by SPPU | Accredited by NBA
Recognized under 2(f) and 12(B) of UGC Act 1956
www.mmcoe.edu.in

About Marathwada Mitra Mandal's College of Engineering

MMCOE has been recognised as the BEST COLLEGE by SPPU. MMCOE is one feather in the cap of Marathwada Mitra Mandal's Pune, from the academic year 2006 – 2007.

The Institute offers six undergraduate engineering programmes in Electrical Engineering, Mechanical Engineering, Electronics & Telecommunication Engineering, Information Technology, Computer Engineering, and Artificial Intelligence & Data Science. In addition, it offers six postgraduate programmes M.Tech (Electrical – Power Electronics & Drives), M.Tech (Mechanical – Design Engineering), M.Tech (Electronics & Telecommunication – VLSI & Embedded Systems), M.Tech (Information Technology – Data Science), M.Tech (Computer Engineering), along with a Ph.D. programme in Computer Engineering. It houses 2000+ students and 200 staff members. Along with the focus on core academics, college also takes care of overall personality development of the students.

The college has an active robotics cell, NCC, Training and Placement cell etc. The students are motivated to actively participate in co-curricular and extra-curricular activities. They have brought many laurels to the Institute in the events such as Vinodottam Karandak, Firodiya Karandak, Zest, Dexterity etc.

Institute Vision

To be a globally renowned institution through excellence in engineering education for sustainable and holistic development

Institute Mission

M1: Empower students with cutting-edge technologies and global competencies

M2: Foster culture of research and entrepreneurial mindset

M3: Imbibe social and professional values

M4: Provide an inclusive environment for lifelong learning

Department of Computer Engineering

Vision and Mission of the Department

- **Vision**

To excel in computer engineering education for sustainable development.

- **Mission**

M1: To develop globally competent professionals through interdisciplinary learning and Center of Excellence.

M2: To empower research, innovation, and entrepreneurial thinking

M3: To promote ethical values and holistic development

M4: To prepare graduates for lifelong learning and dynamic careers

- **Program Specific Outcomes (PSOs)**

A Graduate of the Computer Engineering Program will be able to

PSO1: Apply problem solving skills to design effective solutions in High Performance Computing, Artificial Intelligence and Cyber Security

PSO2: Develop advanced skill-based solutions using standard Software Engineering practices

- **Program Outcomes (POs)**

PO1: Engineering Knowledge: Apply knowledge of mathematics, natural science, computing, engineering fundamentals and an engineering specialization as specified in WK1 to WK4 respectively to develop to the solution of complex engineering problems.

PO2: Problem Analysis: Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development. (WK1 to WK4)

PO3: Design/Development of Solutions: Design creative solutions for complex engineering problems and design/develop systems/components/processes to meet identified needs with consideration for the public health and safety, whole-life cost, net zero carbon, culture, society and environment as required. (WK5)

PO4: Conduct Investigations of Complex Problems: Conduct investigations of complex engineering problems using research-based knowledge including design of experiments, modelling, analysis & interpretation of data to provide valid conclusions. (WK8).

PO5: Engineering Tool Usage: Create, select and apply appropriate techniques, resources and modern engineering & IT tools, including prediction and modelling recognizing their limitations to solve complex engineering problems. (WK2 and WK6)

PO6: The Engineer and The World: Analyze and evaluate societal and environmental aspects while solving complex engineering problems for its impact on sustainability with reference to economy, health, safety, legal framework, culture and environment. (WK1, WK5, and WK7).

PO7: Ethics: Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national & international laws. (WK9)

PO8: Individual and Collaborative Team work: Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams.

PO9: Communication: Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations considering cultural, language, and learning differences

PO10: Project Management and Finance: Apply knowledge and understanding of engineering management principles and economic decision-making and apply these to one's own work, as a member and leader in a team, and to manage projects and in multidisciplinary environments.

PO11: Life-Long Learning: Recognize the need for, and have the preparation and ability for i) independent and life-long learning ii) adaptability to new and emerging technologies and iii) critical thinking in the broadest context of technological change. (WK8)

Curriculum for B. Tech. in Computer Engineering with Honours in Quantum Computing

Department of Computer Engineering

General Objectives and Scope:

Honors degrees offer a more in-depth study of the field in which one is pursuing honors. It might help the student obtain a greater chance at a decent employment than someone with a generic degree. The honors degree entails extensive study that culminates in a research potential in a student.

Completing an honors degree not only prepares one for higher education in a specific field, but also demonstrates to prospective employers a greater capacity for independent learning and research, as well as better problem-solving, critical thinking, independent learning, and communication abilities.

A bachelor's degree with honors will increase employability. This demonstrates to a recruiter that you can work diligently to earn good grades. It demonstrates ambition, self-discipline, and organization, among other excellent characteristics.

Why Quantum Computing?

Quantum computing has the potential to transform modern computation by enabling solutions to problems that are difficult or impossible for classical computers to solve efficiently. Although current quantum computers are not yet advanced enough to outperform classical systems in many real-world tasks, significant progress is being made in research and development.

As a result, quantum computing is gaining increasing attention among researchers and technology enthusiasts. This emerging technology is considered an advanced form of computing compared to traditional systems. Unlike classical computers that rely on binary bits (0 and 1), quantum computers use principles such as superposition and entanglement to process information.

Advancements in quantum computing are also expected to accelerate developments in artificial intelligence and machine learning, leading to potential breakthroughs in fields such as drug discovery, cybersecurity, cryptography, robotics, and finance. According to a report by McKinsey & Company, the global market value of quantum computing technology could reach US$1 trillion by 2035, highlighting its significant future impact.

Prerequisites :

- Data Structures and Algorithms
- Programming in Python / C#
- Machine Learning and Data Science Basics
- Neural Networks and Deep Learning Basics
- Information Theory and Models of Computation
- Classical Fourier Transform

B. TECH HONOURS IN QUANTUM COMPUTING (Semester IV, V, VI, VII, VIII)

Curriculum Structure

Course Code	Course Name	Teaching Scheme (Hrs/week)				Examination Scheme						Credits				
		L	P	T	ODL	CIE	ETE	TW	PR	OR	Total	L	P	T	ODL	Total
Semester IV																
CE24HON293R	Introduction to Quantum Computing	4	-	-	-	40	60	-	-	-	100	4	-	-	-	4
Semester V																
CE24HON343R	Quantum Programs and Quantum Algorithms	3	-	-	-	40	60	-	-	-	100	3	-	-	-	3
CE24HON344R	Quantum Programs and Quantum Algorithms Lab	-	4	-	-	-	-	50	-	-	50	-	2	-	-	2
Semester VI																
CE24HON393R	Quantum Machine Learning	3	-	-	-	40	60*	-	-	-	100	3	-	-	-	3
CE24HON394R	Quantum Machine Learning Lab	-	4	-	-	-	-	50	-	-	50	-	2	-	-	2
Semester VII																
CE24HON443R	Seminar	-	-	2	-	-	-	50	-	-	50	-	-	2	-	2
Semester VIII																
CE24HON492R	Capstone Project	-	-	4	-	-	-	100	-	-	100	-	-	4	-	4
Total		10	08	06	-	120	180	250	-	-	550	10	04	06	-	20

* **Mode of conduction of ETE will be decided by course coordinator in the supervision of BOS chairperson.**

L- Lecture	P- Practical	T- Tutorial	CIE-Continuous Internal Evaluation
ETE- End Term Examination	TW- Term work	PR- Practical	OR- Oral
L : 1 Hr.= 1 credit	P: 2 Hr. = 1 Credit	T: 1 Hr. = 1 Credit	ODL-Open Distance Learning

Dr. K. S. Thakre
BOS Chairperson

Second Year B. Tech Computer Engineering														
Semester IV														
Course Code: CE24HON293R					Course Name: Introduction to Quantum Computing									
Teaching Scheme (Hours/Week)					Examination Scheme						Credits			
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	TOTAL
4	-	-	-	-	40	60	-	-	-	100	4	-	-	4

Course Objectives:

- Understand the motivation, origin, and industry developments in Quantum Computing including contributions from IBM, Microsoft, Rigetti Computing, and D-Wave Systems.
- Learn the mathematical foundations such as linear algebra, matrices, vector spaces, and operators used in quantum computing.
- Understand basic quantum mechanics concepts like superposition, entanglement, and wave–particle duality.
- Study quantum information basics including qubits, Bloch sphere, quantum gates, and circuits.
- Explore quantum computation models and applications such as quantum algorithms and quantum annealing.

Course Outcomes:

CO1: Explain the fundamentals and evolution of quantum computing.

CO2: Apply linear algebra concepts in representing quantum states.

CO3: Describe key quantum mechanics principles used in quantum computing.

CO4: Represent and analyze qubits, quantum gates, and quantum circuits.

CO5: Compare classical and quantum computing models and applications.

Unit	Contents	Hrs
1	**Motivation for studying Quantum Computing-**Major players in the industry (IBM, Microsoft, Rigetti, D-Wave etc.), Origin of Quantum Computing, Prerequisites of Quantum Computing	7
2	**Mathematical Foundation of Quantum Computing, Essentials in Linear Algebra-** Basics of Matrix Algebra, Basics of Operator Theory,Basics of Vector Spaces Matrix Algebra: basis vectors and orthogonality, inner product and Hilbert spaces, matrices and tensors, unitary operators and projectors, Dirac notation, Eigenvalues and Eigenvectors, Density Matrices and Ensembles	8
3	**Essentials in Computer Science:** Building Blocks of Information Storage and Processing, Software Frameworks, Classical vs Quantum Hardware and Software, Classical vs Quantum Information Theory, Models of Computation in Classical vs Quantum Computing, Computations in the Quantum Circuit Model, Classical vs Quantum Algorithms**,**	7
4	**Essentials in Quantum Mechanics:** The Uncertainty Principle, Wave Particle Duality, Wave Function and Schrodinger Equation, Wavefunctions and Quantum States, Superposition, Entanglement, Interference, Special Quantum States (Bell-State, GHZ State), Quantum Hardware, Quantum Software	7

5	**Overview of major concepts in Quantum Computing:** Qubits and multi-qubits states, Bra-ket notation, Bloch Sphere representation, Representing Qubit States and Operations, Single Qubit Gates and Measurement, Qubit Visualizations (QpiAI Explorer or other QC System), Quantum Circuits and Universal Gates, Quantum Superposition, Quantum Entanglement, Quantum Teleportation Quantum Annealing	7
	Total Hours	36

Text Books

1. Michael A. Nielsen, "Quantum Computation and Quantum Information", Cambridge University Press.
2. David McMahon, "Quantum Computing Explained", Wiley Publication

Reference Books

1. Phillip Kaye, Raymond Laflamme, Michele Mosca An Introduction to Quantum Computing, Oxford University Press.
1. Scott Aaronson Quantum, Computing Since Democritus, Cambridge University Press.
2. Noson S. Yanofsky and Mirco A. Mannucci, Quantum Computing for Computer Scientists, Cambridge University Press.
3. N. David Mermin, Quantum Computer Science: An Introduction, Cambridge University Press.

Online references

- IBM Quantum Learning Platform: https://quantum.ibm.com
- Microsoft Quantum Development Kit Documentation: https://learn.microsoft.com/azure/quantum/
- Qiskit Textbook (IBM): https://qiskit.org/learn/

Third Year B. Tech Computer Engineering														
Semester V														
Course Code: CE24HON343R						Course Name: Quantum Programs and Quantum Algorithms								
Teaching Scheme (Hours/Week)					Examination Scheme						Credits			
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	TOTAL
3	-	-	-	-	40	60	-	-	-	100	3	-	-	3

Course Objectives:

- Understand the architecture and programming model of quantum computing platforms.
- Learn qubit representation, superposition, entanglement, and quantum states used in quantum algorithms.
- Study quantum gates, circuits, and operations used for quantum computation.
- Explore major quantum algorithms and protocols used in quantum computing.
- Gain exposure to quantum programming tools and frameworks such as IBM Quantum Experience, Microsoft Q#, PyQuil, TensorFlow Quantum, and Amazon Braket.

Course Outcomes:

CO1: Explain the architecture and working of quantum computing platforms.

CO2: Represent qubits and multi-qubit systems using superposition and entanglement.

CO3: Design quantum circuits using quantum gates.

CO4: Analyze major quantum algorithms and quantum communication protocols.

CO5: Implement basic quantum programs using quantum computing toolkits.

Unit	Contents	Hrs
1	**Building Blocks of a Quantum Program** **Architecture of a Quantum Computing platform:** Hybrid Approach for Application Development, Details of q-bit system of information representation:Block Sphere, Multi-qubits States, Quantum superposition of qubits (valid and invalid superposition) Quantum Entanglement,Useful states from quantum algorithmic perspective e.g. Bell State, Operation on qubits: Measuring and transforming using gates. Quantum Logic gates and Circuit: Pauli, Hadamard, phase shift, controlled **gates, Ising, Deutsch, swap etc.**	7
2	**Programming model for a Quantum Computing Program**, Steps performed on classical computer, Steps performed on Quantum Computer, Moving data between bits and qubits., Models of Computation used by key players as OEMs in Quantum Computing , Basic techniques exploited by quantum algorithms: Quantum Circuits, Amplitude amplification, Phase Kick-back, Quantum Phase estimation	7
3	**Quantum Walks, Major Quantum Algorithms:** Quantum Fourier Transform, Quantum Factoring: Shor's Algorithm, Quantum Database Search: Grover's Algorithm, Quantum Factoring: Shor's Algorithm, Quantum Database Search: Grover's Algorithm, Deutsch's Algorithm, Deutsch -Jozsa Algorithm, Bernstein-Vazirani Algorithm, Circuit Simulations (on QpiAI Explorer Software or other QC system)	8

4	**Quantum Protocols:** Quantum Teleportation, Superdense Coding, Simulation of QpiAI Explorer Software, Quantum Cryptography and Key Distribution Quantum Communication	7
5	**OSS Toolkits for implementing Quantum program:** IBM quantum experience, Microsoft Q#, Rigetti PyQuil (QPU/QVM) OR Cambridge Quantum Computing, Google's Tensorflow Quantum, Amazon Bracket, D-Wave Frameworks	7
	Total Hours	36

Text Books

4. IBM Quantum Experience: https://quantumexperience,ng,bluemix.net https://quantum-computing.ibm.com/docs/
5. Microsoft Quantum Development Kit https://www.microsoft.com/en-us/quantum/development-kit

Reference Books

1. Forest SDK PyQuil: https://pyquil.readthedocs.io/en/stable/
2. Amazon Bracket Documentation on AWS: https://aws.amazon.com/braket/
6. D-Wave Systems Documentation: https://docs.dwavesys.com/docs/latest/index.html

Online references

- https://learning.quantum.ibm.com/
- Fundamentals of Quantum : Algorithmshttps://quantum.cloud.ibm.com/learning/en/courses/fundamentals-of-quantum-algorithms?utm_source=chatgpt.com

Third Year B. Tech Computer Engineering														
Semester V														
Course Code: CE24HON344R					Course Name: Quantum Programs and Quantum Algorithms Lab									
Teaching Scheme (Hours/Week)					Examination Scheme						Credits			
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	TOTAL
-	4	-	-	-	-	-	50	-	-	50	-	2	-	2

Course Objectives:

- To understand basic quantum computing concepts through simple applications like quantum dice and quantum random number generator.
- To design and simulate simple quantum circuits using quantum gates and measurement operations.
- To implement fundamental quantum algorithms such as Shor's Algorithm and Grover's Algorithm.

Course Outcomes:

CO1: Build basic quantum applications using qubits.

CO2: Design and execute simple quantum circuits and obtain classical outputs.

CO3: Implement and analyze Shor's and Grover's quantum algorithms.

Unit	Contents	Hrs
1	Building Quantum dice	6
2	Building Quantum Random Number Generator	6
3	Implementation of Shor's Algorithms	6
4	Composing simple quantum circuits with q-gates and measuring the output into classical bits.	6
5	Implementation of Grover's Algorithm	6
	Total Hours	30

Text Books

1. IBM Quantum Experience: https://quantumexperience,ng,bluemix.net https://quantum-computing.ibm.com/docs/
2. Microsoft Quantum Development Kit https://www.microsoft.com/en-us/quantum/development-kit

Reference Books

1. Forest SDK PyQuil: https://pyquil.readthedocs.io/en/stable/
2. Amazon Bracket Documentation on AWS: https://aws.amazon.com/braket/
3. D-Wave Systems Documentation: https://docs.dwavesys.com/docs/latest/index.html

Online references

- https://learning.quantum.ibm.com/
- Fundamentals of Quantum : Algorithmshttps://quantum.cloud.ibm.com/learning/en/courses/fundamentals-of-quantum-algorithms?utm_source=chatgpt.com

Third Year B. Tech Computer Engineering															
Semester VI															
Course Code:CE24HON393R								Course Name: : Quantum Machine Learning							
Teaching Scheme (Hours/Week)					Examination Scheme						Credits				
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	ODL	TOTAL
3	-	-	-	-	40	60	-	-	-	100	3	-	-	-	3

Course Objectives

- Understand the fundamentals of Artificial Intelligence, Machine Learning, and Deep Learning required for Quantum Machine Learning.
- Explain NISQ devices, noise models, and hybrid quantum–classical computing architectures.
- Study quantum algorithms used in machine learning such as VQE, QAOA, and HHL.
- Explore Quantum Machine Learning techniques including Quantum Neural Networks and Quantum AI applications.
- Understand quantum hardware technologies including superconducting and semiconducting qubits.

Course Outcomes:

CO1: Explain the basic concepts of AI, ML, and DL relevant to Quantum Machine Learning.
CO2: Describe the characteristics and limitations of NISQ devices and hybrid quantum–classical computing systems.
CO3: Analyze quantum algorithms used in machine learning such as VQE, QAOA, and HHL.
CO4: Apply Quantum Machine Learning techniques such as Quantum Neural Networks to solve computational problems.
CO5: Explain the working principles of quantum hardware technologies, including superconducting and semiconductor qubits.

Unit	Contents	Hrs
1	**Prerequisites for Quantum Machine Learning** Machine Learning, Deep Learning and Artificial Intelligence Basics: Machine Learning Algorithms, Deep Learning AlgorithmsEvolutionary Learning Algorithms **NISQ Devices:** Noise Models, Quantum Error Mitigation, Quantum Volume and Performance Metrics, Hybrid Quantum-Classical Computing	7
2	**Quantum Algorithms Applied :** Variational Quantum Algorithms, Quantum Approximate Optimization, Variational Quantum Eigensolver, HHL Algorithm for Solving Linear Systems, Quantum Machine Learning: Introduction to QNNs	7
3	**Quantum Machine Learning and Quantum AI:** Quantum Neural Networks, Quantum Natural Language Understanding, Quantum Cryptography	7
4	**Application Domains for Quantum Machine Learning:** Chemistry/Material Science, , Space Tech, Finance related Optimisation Problems, Swarm Robotics, Cybersecurity **Quantum Hardware - Superconducting Qubits** Introduction to physical qubits, Circuit Quantum Electrodynamics, Transmon and Coupled Qubits, Control and Readout	8
5	**Quantum Hardware Semiconducting Qubits:** Introduction to physical qubits, Spin Physics and Quantum Dots, Control and Readout, Scalability	7
	Total Hours	36

Text Book
1. Quantum Machine Learning (What Quantum Computing Means to Data Mining) by Peter Wittek, University of Boras, Sweden - Elsevier Publications 2. Principles of Quantum Artificial Intelligence by Andreas Winchert, Instituto Superior Técnico - Universidade de Lisboa, Portugal - World Scientific Publishing, British Library Cataloguing-in-Publication Data

Reference Books
1. Quantum Computation and Quantum Information – Michael A. Nielsen and Isaac L. Chuang. 2. Supervised Learning with Quantum Computers – Maria Schuld and Francesco Petruccione. 3. Machine Learning with Quantum Computers – Maria Schuld and Francesco Petruccione. 4. Quantum Computing for Computer Scientists – Noson S. Yanofsky and Mirco A. Mannucci. 5. Hands-On Quantum Machine Learning with Python – Santanu Ganguly. 6. Quantum Algorithms via Linear Algebra – Richard J. Lipton and Kenneth W. Regan

Online references
● IBM Quantum Learning Portal: https://learning.quantum.ibm.com/ ● Microsoft Quantum Documentation : https://learn.microsoft.com/azure/quantum ● Amazon Braket Developer Guide: https://docs.aws.amazon.com/braket

Third Year B. Tech Computer Engineering														
Semester VI														
Course Code:CE24HON394R							Course Name: Quantum Machine Learning Lab							
Teaching Scheme (Hours/Week)					Examination Scheme						Credits			
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	TOTAL
-	4	-	-	-	-	-	50	-	-	50	-	2	-	2

Course Objectives

- To implement basic quantum algorithms like Deutsch's Algorithm and Deutsch–Jozsa Algorithm.
- To apply quantum computing techniques to problems such as search, factorization, and graph partitioning.
- To explore applications like quantum neural networks, quantum NLP, and quantum software frameworks.

Course Outcomes:

CO1: Implement and analyze Deutsch’s and Deutsch–Jozsa algorithms.
CO2: Apply quantum computing methods to solve optimization and machine learning problems.
CO3: Develop small quantum applications and compare quantum software frameworks.

Unit	Contents	Hrs
1	Implementation of Deutsch’s Algorithm	7
2	Implementation of Deutsch-Jozsa’s Algorithm	7
3	Mini Project such as implementing an API for efficient search using Grover’s Algorithms or Integer factorization using Shor’s Algorithm	8
4	Graph Partitioning using Quantum Machine Learning	7
5	Implementing Quantum Neural Network	8
6	Basics program implementing Quantum Natural Language Understanding Solution	7
7	Comparative study of Quantum Software Frameworks	7
	Total Hours	30

Text Book

1. Quantum Machine Learning (What Quantum Computing Means to Data Mining) by Peter Wittek, University of Boras, Sweden - Elsevier Publications
2. Principles of Quantum Artificial Intelligence by Andreas Winchert, Instituto Superior Técnico - Universidade de Lisboa, Portugal - World Scientific Publishing, British Library Cataloguing-in-Publication Data

Reference Books

1. Quantum Computation and Quantum Information – Michael A. Nielsen and Isaac L. Chuang.
2. Supervised Learning with Quantum Computers – Maria Schuld and Francesco Petruccione.
3. Machine Learning with Quantum Computers – Maria Schuld and Francesco Petruccione.

4. Quantum Computing for Computer Scientists – Noson S. Yanofsky and Mirco A. Mannucci.
5. Hands-On Quantum Machine Learning with Python – Santanu Ganguly.
6. Quantum Algorithms via Linear Algebra – Richard J. Lipton and Kenneth W. Regan

Online references

- IBM Quantum Learning Portal: https://learning.quantum.ibm.com/
- Microsoft Quantum Documentation : https://learn.microsoft.com/azure/quantum
- Amazon Braket Developer Guide: https://docs.aws.amazon.com/braket

Final Year B. Tech Computer Engineering

Semester VII

Course Code:CE24HON443R **Course Name: Seminar**

Teaching Scheme (Hours/Week)					Examination Scheme						Credits			
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	TOTAL
-	-	2	-	-	-	-	50	-	-	50	-	-	2	2

Course Objectives:

- Analyze quantum circuits and basic quantum applications.
- Analyze and implement quantum algorithms such as Grover's Algorithm, Shor's Algorithm, Deutsch's Algorithm, and Deutsch–Jozsa Algorithm.
- Evaluate quantum computing applications and software frameworks.

Course Outcomes:

Students will be able to:

CO1: Analyze and design simple quantum circuits and applications.
CO2: Analyze the implementation of major quantum algorithms.
CO3: Evaluate quantum frameworks for solving computational problems.

This is a subject intended for students to understand and study advanced topics in Quantum Computing. Also to study emerging trends of blockchain To study Quantum Concepts necessary for understanding the Quantum Computing Paradigm and compare the available hardware and software infrastructure and frameworks made available open source by major players in the Industry and Academia.

Suggested Topics

- Building Quantum dice
- Building Quantum Random Number Generator
- Composing simple quantum circuits with q-gates and measuring the output into classical bits.
- Implementation of Shor's Algorithms
- Implementation of Grover's Algorithm
- Implementation of Deutsch's Algorithm
- Implementation of Deutsch-Jozsa's Algorithm
- Mini Project such as implementing an API for efficient search using Grover's Algorithms or Integer factorization using Shor's Algorithm
- Graph Partitioning using Quantum Machine Learning
- Basics program implementing Quantum Natural Language Understanding Solution
- Comparative study of Quantum Software Frameworks

Total Hours **30**

Text Books

1. Michael A. Nielsen, "Quantum Computation and Quantum Information", Cambridge University Press.
2. David McMahon, "Quantum Computing Explained", Wiley
3. Quantum Computation and Quantum Information – Michael A. Nielsen and Isaac L. Chuang.
4. Quantum Machine Learning: What Quantum Computing Means to Data Mining – Peter Wittek.
5. Supervised Learning with Quantum Computers – Maria Schuld and Francesco Petruccione

Reference Books

1. Machine Learning with Quantum Computers – Maria Schuld and Francesco Petruccione.
2. Quantum Algorithms via Linear Algebra – Richard J. Lipton and Kenneth W. Regan.

3. Hands-On Quantum Machine Learning with Python – Santanu Ganguly. 4. Quantum Computing for Computer Scientists – Noson S. Yanofsky and Mirco A. Mannucci
Online references
● IBM Quantum Learning Platform: https://learning.quantum.ibm.com ● Microsoft Quantum Documentation (Q#): https://learn.microsoft.com/azure/quantum ● TensorFlow Quantum Tutorials: https://www.tensorflow.org/quantum ● Amazon Braket Quantum Developer Guide: https://docs.aws.amazon.com/braket

Final Year B. Tech Computer Engineering														
Semester VIII														
Course Code:CE24HON492R								Course Name: Capstone Project						
Teaching Scheme (Hours/Week)					Examination Scheme						Credits			
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	TOTAL
-	-	4	-	-	-	-	100	-	-	100	-	-	4	4

Course Objectives:

- Understand advanced concepts of quantum computing and the quantum computing paradigm.
- Learn to design and implement basic quantum programs and circuits using quantum gates.
- Study and implement important quantum algorithms such as Shor's, Grover's, Deutsch, and Deutsch–Jozsa algorithms.
- Explore applications of quantum computing and quantum machine learning such as QNN and quantum NLP.
- Compare quantum software frameworks and platforms provided by industry and academia.

Course Outcomes:

Students will be able to:

CO1: Analyze and construct quantum circuits for simple applications.
CO2: Evaluate the performance of major quantum algorithms for computational problems.
CO3: Develop quantum-based applications and compare quantum software frameworks.

This subject is intended for students to understand and study advanced topics in Quantum Computing. Also to study emerging trends of blockchain To study Quantum Concepts necessary for understanding the Quantum Computing Paradigm and compare the available hardware and software infrastructure and frameworks made available open source by major players in the Industry and Academia.

Topics

- Building Quantum dice
- Building Quantum Random Number Generator
- Composing simple quantum circuits with q-gates and measuring the output into classical bits.
- Implementation of Shor's Algorithms
- Implementation of Grover's Algorithm
- Implementation of Deutsch's Algorithm
- Implementation of Deutsch-Jozsa's Algorithm
- Mini Project such as implementing an API for efficient search using Grover's Algorithms or Integer factorization using Shor's Algorithm
- Graph Partitioning using Quantum Machine Learning
- Basics program implementing Quantum Natural Language Understanding Solution
- Comparative study of Quantum Software Frameworks

Total Hours	**36**

Text Books

1. Michael A. Nielsen, "Quantum Computation and Quantum Information", Cambridge University Press.
2. David McMahon, "Quantum Computing Explained", Wiley
3. Quantum Computation and Quantum Information – Michael A. Nielsen and Isaac L. Chuang.
4. Quantum Machine Learning: What Quantum Computing Means to Data Mining – Peter Wittek.

5. Supervised Learning with Quantum Computers – Maria Schuld and Francesco Petruccione
Reference Books
1. Machine Learning with Quantum Computers – Maria Schuld and Francesco Petruccione. 2. Quantum Algorithms via Linear Algebra – Richard J. Lipton and Kenneth W. Regan. 3. Hands-On Quantum Machine Learning with Python – Santanu Ganguly. 4. Quantum Computing for Computer Scientists – Noson S. Yanofsky and Mirco A. Mannucci
Online references
● IBM Quantum Learning Platform: https://learning.quantum.ibm.com ● Microsoft Quantum Documentation (Q#): https://learn.microsoft.com/azure/quantum ● TensorFlow Quantum Tutorials: https://www.tensorflow.org/quantum ● Amazon Braket Quantum Developer Guide: https://docs.aws.amazon.com/braket

Marathwada Mitra Mandal's College of Engineering

Karvenagar, Pune- 52

An Autonomous Institute Affiliated to SPPU

"येथे बहुतांचे हित"

Curriculum Structure and Syllabus

Honours

A.Y. 2026-27

Accredited with A++ grade by NAAC
Recipient of Best College award by SPPU | Accredited by NBA
Recognized under 2(f) and 12(B) of UGC Act 1956
www.mmcoe.edu.in

About Marathwada Mitra Mandal's College of Engineering

MMCOE has been recognised as the BEST COLLEGE by SPPU. MMCOE is one feather in the cap of Marathwada Mitra Mandal's Pune, from the academic year 2006 – 2007.

The Institute offers six undergraduate engineering programmes in Electrical Engineering, Mechanical Engineering, Electronics & Telecommunication Engineering, Information Technology, Computer Engineering, and Artificial Intelligence & Data Science. In addition, it offers six postgraduate programmes M.Tech (Electrical – Power Electronics & Drives), M.Tech (Mechanical – Design Engineering), M.Tech (Electronics & Telecommunication – VLSI & Embedded Systems), M.Tech (Information Technology – Data Science), M.Tech (Computer Engineering), along with a Ph.D. programme in Computer Engineering. It houses 2000+ students and 200 staff members. Along with the focus on core academics, college also takes care of overall personality development of the students.

The college has an active robotics cell, NCC, Training and Placement cell etc. The students are motivated to actively participate in co-curricular and extra-curricular activities. They have brought many laurels to the Institute in the events such as Vinodottam Karandak, Firodiya Karandak, Zest, Dexterity etc.

Institute Vision

To be a globally renowned institution through excellence in engineering education for sustainable and holistic development

Institute Mission

M1: Empower students with cutting-edge technologies and global competencies

M2: Foster culture of research and entrepreneurial mindset

M3: Imbibe social and professional values

M4: Provide an inclusive environment for lifelong learning

Department of
Artificial Intelligence and Data Science

“ येथे बहुतांचे हित ”

Vision and Mission of the Department

- **Vision**

  Emerge globally as a leading center for Artificial Intelligence and Data Science education, contributing to sustainable development.

- **Mission**

M1: Equip students with advanced Artificial Intelligence knowledge for industry-ready skills.

M2: Cultivate a culture of research and entrepreneurship to drive innovation and problem-solving.

M3: Inculcate social and professional values to develop sustainable AI solutions.

M4: Foster a lifelong learning that supports upskilling and global engagement.

- **Program Specific Outcomes (PSOs)**

Upon successful completion of the program, graduates will be able to:

PSO1: Demonstrate the ability to understand, apply, and develop fundamental computer programs and AI and Data Science solutions using key concepts such as data analytics, machine learning, deep learning and natural language processing.

PSO2: Develop and implement AI and data-driven models by critically analyzing complex problems, designing effective solutions, and evaluating their impact across diverse real-world applications.

PSO3: Utilize emerging AI and Data Science technologies to innovate, promote entrepreneurial ventures, and develop successful careers in industry and research, while embracing lifelong learning and higher education.

- **Program Outcomes (POs)**

PO1: Engineering Knowledge: Apply knowledge of mathematics, natural science, computing, engineering fundamentals and an engineering specialization as specified in WK1 to WK4 respectively to develop the solution of complex engineering problems.

PO2: Problem Analysis: Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development. (WK1 to WK4)

PO3: Design/Development of Solutions: Design creative solutions for complex engineering problems and design/develop systems/components/processes to meet identified needs with consideration for the public health and safety, whole-life cost, net zero carbon, culture, society and environment as required. (WK5)

PO4: Conduct Investigations of Complex Problems: Conduct investigations of complex engineering problems using research-based knowledge including design of experiments, modelling, analysis & interpretation of data to provide valid conclusions. (WK8).

PO5: Engineering Tool Usage: Create, select and apply appropriate techniques, resources and modern engineering & IT tools, including prediction and modelling recognizing their limitations to solve complex engineering problems. (WK2 and WK6)

PO6: The Engineer and The World: Analyze and evaluate societal and environmental aspects while solving complex engineering problems for its impact on sustainability with reference to economy, health, safety, legal framework, culture and environment. (WK1, WK5, and WK7).

PO7: Ethics: Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national & international laws. (WK9)

PO8: Individual and Collaborative Team work: Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams.

PO9: Communication: Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations considering cultural, language, and learning differences

PO10: Project Management and Finance: Apply knowledge and understanding of engineering management principles and economic decision-making and apply these to one's own work, as a member and leader in a team, and to manage projects and in multidisciplinary environments.

PO11: Life-Long Learning: Recognize the need for, and have the preparation and ability for i) independent and life-long learning ii) adaptability to new and emerging technologies and iii) critical thinking in the broadest context of technological change. (WK8)

Curriculum for B. Tech. in Artificial Intelligence and Data Science with Honours in Cloud Professionals

Department of Artificial Intelligence and Data Science

General Objectives and Scope:

The Honours Degree Program aims to provide students with advanced knowledge and deeper understanding in their chosen field of study. It focuses on developing critical thinking, analytical abilities, and research skills through specialized courses and practical learning. The program encourages independent learning and innovation, enabling students to gain expertise beyond the regular undergraduate curriculum.

The scope of the honours degree includes opportunities for research, specialization, and skill development. Students may undertake projects or dissertations that enhance their practical and analytical abilities. The program also prepares students for higher education, research careers, and better employment opportunities in industry and academia.

Why Cloud Professionals?

Cloud computing is transforming the way organizations store, manage, and process data by shifting from traditional on-premise infrastructure to flexible cloud-based platforms. Cloud professionals design, deploy, and manage scalable systems using services provided by companies such as Amazon Web Services, Microsoft Azure, and Google Cloud. Unlike conventional IT systems that rely on physical servers and local networks, cloud computing uses virtualization, distributed computing, and automated resource management to provide high availability, scalability, and cost efficiency. These technologies enable businesses to quickly develop, test, and deploy applications while maintaining secure and reliable digital infrastructure.

Cloud computing is rapidly becoming a key area of interest for businesses, researchers, and technology enthusiasts. As organizations increasingly adopt cloud solutions, the demand for skilled cloud professionals continues to grow across industries such as healthcare, finance, e-commerce, cybersecurity, and data analytics. Cloud technologies also support advancements in Artificial Intelligence, machine learning, and big data processing.

Prerequisites :

- Computer Fundamentals
- Networking Basics
- Operating Systems

- Linux Basics (Linux)
- Programming Fundamentals (Python / Java)
- Database Fundamentals
- Web Technologies Basics
- Virtualization Basics
- Cybersecurity Fundamentals

B. TECH honours in CLOUD PROFESSIONALS (Semester IV, V, VI, VII, VIII)

Curriculum Structure

Course Code	Course Name	Teaching Scheme (Hrs/week)				Examination Scheme						Credits				
		L	P	T	ODL	CIE	ETE	TW	PR	OR	Total	L	P	T	ODL	Total
Semester IV																
AI24HON291R	Cloud Practitioner Essentials	3		-	-	40	60	-	-	-	100	3		-	-	3
AI24HON292R	Cloud Practitioner Essentials lab	-	4		-	-	-	50	-	-	50	-	2	-	-	2
Semester V																
AI24HON341R	Cloud Architecture and Application Deployment	3		-	-	40	60	-	-	-	100	3		-	-	3
AI24HON342R	Cloud Architecture and Application Deployment lab	-	4		-	-	-	50	-	-	50	-	2	-	-	2
Semester VI																
AI24HON391R	Cloud Security and DevOps	3		-	-	40	60*	-	-	-	100	3		-	-	3
AI24HON392R	Cloud Security and DevOps lab	-	4		-	-	-	50	-	-	50	-	2	-	-	2
Semester VII																
AI24HON441R	Cloud Professionals Industry Practices	-		-	1	-	-	25	-	-	25	-	-	-	1	1
AI24HON442R	Internship on Cloud Professionals Industry Practices	-	4	-	-	-	-	50	-	-	50	-	2	-	-	2
Semester VIII																
AI24HON491R	Cloud Professionals Capstone Project	-	4	-	-	-	-	50	-	-	50	-	2	-	-	2
Total		09	20	–	01	120	180	275	-	-	575	09	10	-	01	20

*** Mode of conduction of ETE will be decided by course coordinator in the supervision of BOS chairperson.**

L- Lecture	P- Practical	T- Tutorial	CIE-Continuous Internal Evaluation
ETE- End Term Examination	TW- Term work	PR- Practical	OR- Oral
L : 1 Hr.= 1 credit	P: 2 Hr. = 1 Credit	T: 1 Hr. = 1 Credit	ODL-Open Distance Learning

Dr. Dhanraj Dhotre
BOS Chairperson

Second Year B.Tech Artificial Intelligence & Data Science

Semester-IV

Course Code: AI24HON291R **Course Name: Cloud Practitioner Essentials**

Teaching Scheme (Hours/Week)				Examination Scheme						Credits				
L	P	T	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	ODL	TOTAL
3		-	-	40	60	-	-	-	100	3	-	-	-	3

Prerequisite: Data Science, Artificial Neural Network

Course Objectives:

- To introduce core cloud computing concepts aligned with AWS.
- To Familiarize learners with AWS services, security, pricing, and architecture.
- To Build skills to deploy and manage applications on AWS.
- To enable learners to make informed cloud adoption decisions.

Course Outcomes:

After completing the course, the students will be able to:

CO1: Explain cloud computing concepts and AWS global infrastructure

CO2: Identify AWS architectural principles and deployment models

CO3: Describe core AWS services and their use cases

CO4: Explain AWS security, identity, and compliance concepts

CO5: Understand AWS pricing models, billing, and support plans.

Unit	Contents	Duration (Hrs.)
1	**Cloud Concepts** Definition of cloud computing, Benefits of cloud computing: Cost optimization, Scalability and elasticity, High availability and reliability, Agility and global reach, Cloud computing vs traditional on-premises IT, Types of cloud computing: Public, Private, Hybrid, Overview of AWS Global Infrastructure: Regions, Availability Zones, Edge Locations.	8
Activity / Case Study	Explain how AWS Regions and Availability Zones improve fault tolerance for an e-commerce website.	
2	**AWS Cloud Architecture & Deployment Models** Cloud service models:IaaS, PaaS, SaaS, AWS Well-Architected Framework – Pillars: Operational Excellence, Security, Reliability, Performance Efficiency, Cost Optimization, Sustainability, Basic cloud architectural principles: High availability, Fault tolerance, Scalability, Migration strategies (6 R's)	8
Activity / Case Study	Identify the AWS service model used by Netflix, Zoom, and Google Drive	
3	**Core AWS Services** Compute Services: Amazon EC2, AWS Lambda, Amazon ECS / EKS, Storage Services: Amazon S3 (object storage), Amazon EBS (block storage), Amazon EFS (file storage), Database Services: Amazon RDS, Amazon DynamoDB, Networking Services: Amazon VPC, Security Groups and NACLs, Amazon Route 53.	8

Activity / Case Study	Explore AWS Management Console and identify compute, storage, and database services	
4	**Security, Identity, and Compliance** AWS Shared Responsibility Model, AWS Identity and Access Management (IAM): Users, Groups, Roles, Policies, Authentication vs Authorization, Data protection: Encryption at rest and in transit, AWS security services: AWS Shield, AWS WAF, Amazon GuardDuty, Compliance and governance: ISO, SOC, GDPR	8
Activity / Case Study	Identify customer vs AWS responsibilities for data security in Amazon S3	
5	**Pricing, Billing, and Support** AWS pricing principles: Pay-as-you-go, Savings Plans, Reserved Instances, AWS Free Tier, AWS Billing and Cost Management, Cost optimization best practices, AWS Support plans: Basic, Developer, Business, Enterprise, AWS Marketplace.	8
Activity / Case Study	Estimate monthly cost for a simple web application using AWS Pricing Calculator	
	Total Hours	40

Text Books

1. Ben Piper & David Clinton, "AWS Certified Cloud Practitioner Study Guide (CLF-C02) ", Orielly, July 2019.
2. Rajesh Daswani, AWS Certified Cloud Practitioner (CLF-C02) Exam Guide, Orielly , Second Edition .

Reference Books

1. Thomas Erl, " Cloud Computing: Concepts, Technology & Architecture", Wileynpt

Online References

1. AWS Cloud Practitioner Essentials — Digital Training (Free), https://aws.amazon.com/blogs/training-and-certification/cloud-practitioner-essentials-digital-course/
2. AWS Cloud Practitioner Essentials on Coursera, https://www.coursera.org/learn/aws-cloud-practitioner-essentials

Second Year B.Tech Artificial Intelligence & Data Science														
Semester- IV														
Course Code: AI24HON292R				Course Name: Cloud Practitioner Essentials Lab										
Teaching Scheme (Hours/Week)				Examination Scheme						Credits				
L	P	T	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	ODL	TOTAL
-	4	-	-	-	-	50	-	-	50	-	2	-	-	2

Prerequisite: : Internet Fundamentals, Familiarity with IT Terminology

Course Objectives:
- Provide a foundational understanding of cloud computing concepts and AWS infrastructure.
- Introduce core cloud services, security principles, and architectural best practices.
- Develop awareness of cloud pricing, cost optimization, and industry use cases.
- Prepare learners for entry-level cloud roles and Cloud Practitioner certification.

Course Outcomes:
After learning the course, the students will be able to:
CO1: Explain fundamental cloud computing concepts and AWS global infrastructure.
CO2: Identify and use appropriate cloud service and deployment models for business scenarios.
CO3: Describe core AWS services related to compute, storage, networking, and databases.
CO4: Explain cloud security, identity management, compliance, and shared responsibility.
CO5: Analyze cloud pricing models and apply basic cost optimization strategies.

List of Experiments

	Name of the Experiment	Duration (Hrs.)
1	A startup currently hosts its application on a single on-premise server and experiences downtime during traffic spikes. The management wants to move to the cloud to improve scalability and availability while minimizing cost. Identify the Cloud Adoption for a Startup, suitable cloud deployment model and the AWS services needed.	04
2	A software company wants to deliver its application globally without managing servers, OS patches, or infrastructure scaling. Identify appropriate cloud service models (IaaS / PaaS / SaaS) and map the model to AWS services.	04
3	An e-commerce website expects heavy traffic during festival sales and wants automatic scaling with minimal manual intervention. Designing Compute Infrastructure for the E-Commerce Platform	04
4	A media company stores large volumes of videos and images and wants secure, highly durable, and cost-effective storage. Identify appropriate AWS storage service and define access control and security measures.	04
5	A FinTech startup needs a database that can handle millions of transactions per day with low latency and high availability. Identify the right Database for a FinTech Application by taking into consideration scalability, consistency, and cost.	04
6	An enterprise has developers, testers, and auditors who need different levels of access to cloud resources. Design an IAM-based	04

	access control strategy for implementing Access Control for an Enterprise Team.Define users, roles, and policies for implementing the Shared Responsibility Model.	
7	A healthcare organization wants to store patient data on AWS while ensuring privacy, compliance, and auditability. Ensure Cloud Security by identifying AWS security services to be used for Healthcare Data.	08
8	A growing company notices increasing cloud bills and wants to optimize costs without affecting performance. Identify cost optimization strategies for a Growing Business.	08
9	Mini Project on cloud storage	12
	Total Hours	52

Text Books

1. Ben Piper & David Clinton, "AWS Certified Cloud Practitioner Study Guide (CLF-C02) ", Orielly, July 2019.
2. Rajesh Daswani, AWS Certified Cloud Practitioner (CLF-C02) Exam Guide, Orielly , Second Edition .

Reference Books

1. Thomas Erl, " Cloud Computing: Concepts, Technology & Architecture", Wileynpt

Online References

1. AWS Cloud Practitioner Essentials — Digital Training (Free), https://aws.amazon.com/blogs/training-and-certification/cloud-practitioner-essentials-digital-course/
2. AWS Cloud Practitioner Essentials on Coursera, https://www.coursera.org/learn/aws-cloud-practitioner-essentials

<table>
<tr><th colspan="15">Third Year B.Tech Artificial Intelligence & Data Science</th></tr>
<tr><th colspan="15">Semester- V</th></tr>
<tr><th colspan="15">Course Code: AI24HON341R Course Name: Cloud Architecture & Application Deployment</th></tr>
<tr><th colspan="4">Teaching Scheme (Hours/Week)</th><th colspan="6">Examination Scheme</th><th colspan="5">Credits</th></tr>
<tr><th>L</th><th>P</th><th>T</th><th>ODL</th><th>CIE</th><th>ETE</th><th>TW</th><th>OR</th><th>PR</th><th>TOTAL</th><th>L</th><th>P</th><th>T</th><th>ODL</th><th>TOTAL</th></tr>
<tr><td>3</td><td></td><td>-</td><td>-</td><td>40</td><td>60</td><td>-</td><td>-</td><td></td><td>100</td><td>3</td><td></td><td>-</td><td>-</td><td>3</td></tr>
<tr><td colspan="15">Prerequisite: Cloud Practitioner / Introduction to Cloud Computing , Basic knowledge of networking, Linux, and web applications</td></tr>
<tr><td colspan="15">Course Objectives:
To design secure, scalable, and resilient cloud architectures on AWS.
To deploy and manage cloud-based applications using AWS services
To implement automation, monitoring, and CI/CD pipelines.
To apply security, governance, and cost-optimization best practices.
To analyze and optimize application performance in the cloud.</td></tr>
<tr><td colspan="15">Course Outcomes:
After learning the course, the students will be able to:
CO1: Design multi-tier and highly available cloud architectures.
CO2: Deploy and manage applications using compute, storage, and networking services.
CO3: Automate infrastructure and application deployment.
CO4: Implement security, monitoring, and fault tolerance
CO5: Optimize cost, performance, and reliability of cloud workloads</td></tr>
</table>

Unit	Contents	Duration (Hrs.)
1	**Deploying Scalable & Resilient Architectures** Multi-tier application architecture ,Elastic Load Balancing (ALB, NLB),Auto Scaling Groups (ASG) & launch templates ,High availability across Availability Zones and fault tolerance patterns,Storage and database choices: RDS, DynamoDB (use cases and trade-offs) ,Backup and disaster recovery strategies **Activity:** Deploy a scalable web application using ALB + ASG, Configure RDS for high availability	8
2	**Application Deployment and Compute Services** Application deployment models: VM-based deployment vs Container-based deployment, AWS Compute options: EC2 vs Elastic Beanstalk ,Introduction to containers and ECS , Serverless architecture overview: Lambda and API Gateway, Application configuration and environment management **Activity:** Deploy a web application using Elastic Beanstalk Create a basic serverless REST API	8
3	**Monitoring, Application Integration & Troubleshooting** Monitoring tools: CloudWatch basics (metrics, dashboards, alarms), Logging fundamentals (CloudWatch Logs),Application integration services: SQS, SNS, EventBridge, Simple workflow with Step Functions (deployment focus) Troubleshooting deployment issues & logs **Activity**: Create CloudWatch dashboard & alarms Build a sample producer-consumer flow using SNS & SQS	8
4	**Performance & Cost Optimization** Cost optimization principles of AWS architecture, Resource right-sizing & purchasing options (Spot, RI, Savings), Performance	8

	optimization techniques, Storage tiering and lifecycle policies ,Using AWS Cost Explorer & Budgets **Activity** : Setup S3 lifecycle policies Use Cost Explorer to analyze sample workload costs	
5	**DevOps, Infrastructure as Code & Deployment Automation** DevOps principles in cloud-native environments, CI vs CD vs CD (Continuous Deployment vs Delivery),Deployment strategies: rolling, blue–green, canary ,DevOps role in scalable and repeatable cloud architectures, Infrastructure as Code (IaC) Concepts, Terraform overview (by HashiCorp): Providers, resources, variables, outputs ,State file concept (local vs remote – conceptual) ,When to choose Terraform in AWS projects	8
	Total Hours	40

Text Books:

1. Amazon Web Services in Action Michael Wittig & Andreas Wittig
2. AWS for Solutions Architects Authors**:** Saurabh Shrivastava, Neelanjali Srivastav, Alberto Artasanchez, Imtiaz Sayed, Siddhartha Choubey

Reference Books

1. AWS Certified Solutions Architect Official Study Guide , Sybex

Online References

1. https://www.udemy.com/course/aws-certified-solutions-architect-associate-saa-c03/
2. https://skillbuilder.aws/learning-plan/2TDWRB1QV8/aws-simulearn-solutions-architect/67BBEU3C7N
3. https://skillbuilder.aws/search?searchText=aws-well-architected-foundations&showRedirectNotFoundBanner=true

Third Year B.Tech Artificial Intelligence & Data Science														
Semester V														
Course Code: AI24HON342R Course Name: Cloud Architecture and application deployment Lab														
Teaching Scheme (Hours/Week)				Examination Scheme						Credits				
L	P	T	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	ODL	TOTAL
-	4	-	-	-	-	50	-	-	50	-	2	-	-	2

Prerequisite: : Cloud Practitioner / Introduction to Cloud Computing , Basic knowledge of networking, Linux, and web applications

Course Objectives:
- Design secure, scalable, and resilient cloud architectures on AWS.
- Deploy and manage cloud-based applications using AWS services
- Implement automation, monitoring, and CI/CD pipelines.
- Apply security, governance, and cost-optimization best practices.
- Analyze and optimize application performance in the cloud.

After learning the course, the students will be able to:
CO1: Design multi-tier and highly available cloud architectures.
CO2: Deploy and manage applications using compute, storage, and networking services.
CO3: Automate infrastructure and application deployment.
CO4: Implement security, monitoring, and fault tolerance
CO5: Optimize cost, performance, and reliability of cloud workloads

List of Experiments

Sr. No.	Name of the Experiment	Duration (Hrs.)
1	Deploy Multi-Tier Scalable Web Application (ALB + ASG) Design and deploy a **highly available multi-tier architecture** using: ● EC2 ● Application Load Balancer (ALB) ● Auto Scaling Group (ASG)	6
2	Deploy and Configure Amazon RDS with High Availability Configure relational database with **Multi-AZ deployment and backup strategy**.	6
3	Application Deployment using Elastic Beanstalk: Deploy a web application using **Elastic Beanstalk**.	6
4	Build Serverless REST API using Lambda & API Gateway	6
5	Monitoring & Application Integration (CloudWatch, SNS, SQS)	6
6	Analyze and optimize cost & performance using Cost Explorer,S3 lifecycle policies, Storage tiering, Right sizing	6
7	Automate AWS infrastructure deployment using Terraform.	6
8	Design and deploy a **complete cloud-native application** with: ● ALB + ASG ● RDS ● Monitoring ● SNS notifications ● Terraform automation Project Submission includes: ● Architecture diagram ● Deployment screenshots	10

	• Cost estimation • Report	
	Total Hours	52

Text Books

1. Amazon Web Services in Action Michael Wittig & Andreas Wittig
2. AWS for Solutions Architects Authors**:** Saurabh Shrivastava, Neelanjali Srivastav, Alberto Artasanchez, Imtiaz Sayed, Siddhartha Choubey

Reference Books

1. AWS Certified Solutions Architect Official Study Guide , Sybex

Online References

1. https://www.udemy.com/course/aws-certified-solutions-architect-associate-saa-c03/
2. https://skillbuilder.aws/learning-plan/2TDWRB1QV8/aws-simulearn-solutions-architect/67BBEU3C7N
3. https://skillbuilder.aws/search?searchText=aws-well-architected-foundations&showRedirectNotFoundBanner=true

<table>
<tr><th colspan="15">Third Year B.Tech Artificial Intelligence & Data Science</th></tr>
<tr><th colspan="15">Semester- VI</th></tr>
<tr><th colspan="15">Course Code: AI24HON391R Course Name: Cloud Security and DevOps</th></tr>
<tr><th colspan="4">Teaching Scheme (Hours/Week)</th><th colspan="6">Examination Scheme</th><th colspan="5">Credits</th></tr>
<tr><th>L</th><th>P</th><th>T</th><th>ODL</th><th>CIE</th><th>ETE</th><th>TW</th><th>OR</th><th>PR</th><th>TOTAL</th><th>L</th><th>P</th><th>T</th><th>ODL</th><th>TOTAL</th></tr>
<tr><td>3</td><td></td><td>-</td><td>-</td><td>40</td><td>60</td><td>-</td><td>-</td><td>-</td><td>100</td><td>3</td><td>-</td><td>-</td><td>-</td><td>3</td></tr>
<tr><td colspan="15">Prerequisite: Cloud Practitioner / Introduction to Cloud Computing , Basic knowledge of networking, Linux, and web applications</td></tr>
</table>

Course Objectives:

- To understand the cloud security for infrastructure and applications.
- Understand the concepts of private and public clouds using modern approaches and tools.
- Explore Identity and Access Management (IAM) mechanisms
- To study authentication and authorization mechanisms in cloud environments
- Apply secure practices in data protection and DevSecOps

Course Outcomes:

After learning the course, the students will be able to:

CO1: Understand the cloud security for infrastructure and applications.
CO2: Understand the concepts of private and public clouds using modern approaches and tools.
CO3: Design security models for various business scenarios.
CO4: Implement and Deploy lightweight Security Services.
CO5: Evaluate container and virtualization security measures.

Unit	Contents	Duration (Hrs.)
1	**Cloud Security Fundamentals** Introduction to Cloud Security, Basic concepts (CIA) - Confidentiality, Integrity and Availability, Understanding Multi tenancy, Multi tenancy concepts, Design considerations for multi-tenant cloud, Security concerns on multi-tenant cloud environments, Security in context of Cloud Deployment and consumption models, Cloud Governance, Risk and Compliance. Case Study : Multi-Tenancy Risk in Public Cloud A SaaS-based HR management system hosts data of multiple organizations on a shared public cloud infrastructure. One tenant reports data leakage due to misconfigured storage permissions.	8
2	**Virtualization and Container Security** Virtualization Security, Vulnerabilities in management, Vulnerabilities in Hypervisor, Vulnerabilities in Virtualization and virtual machines, Platform Hardening to Prevent Threats, Container Security, Network isolation and Protection , Setting up a secure network, Network isolation and segmentation, Load balancer and Ingress services Case Study: Hypervisor Vulnerability Exploit An attacker exploits a hypervisor vulnerability to gain unauthorized access to multiple virtual machines in a data center.	8
3	**Identity and Access Management (IAM)** Identity and Access Management (IAM), IAM Operational Areas, Cloud Identity Management, Identity Provisioning in Cloud, Cloud Authentication, SAML, OAuth2.0, OpenID, Multi Factor authentication, Enterprise SSO and federation, Privileged Access Management (PAM), IBM Cloud App ID	8

	Case Study: IAM Misconfiguration Leading to Data Breach A cloud administrator assigns excessive permissions to developers, resulting in accidental deletion of production databases.	
4	**Data Security** Data Security, protection and IBM key protect in IBM Cloud, Data Integrity in the Cloud, IBM Multi-Cloud Data Encryption, Data in Transit, Protect Data in Transit, Certificate management, Digital certificates Encrypting data in transit, Istio overview to protect microservices Case Study: Unencrypted Data at Rest and Transit A fintech company stores sensitive transaction data without encryption. A man-in-the-middle attack compromises data during transmission.	8
5	DevSecOps Application Security, Secure DevOps, Vulnerability and Patch Management, IBM Cloud Pak for Security Case Study: Security Vulnerability in CI/CD Pipeline A DevOps team deploys applications rapidly but skips security testing. A known vulnerability is exploited in production.	8
	Total Hours	40

Text Books

1. Enterprise Cloud Security and Governance: Efficiently Set Data Protection and Privacy Principles by Zeal Vora

Reference Books

1. Cloud Security: A Comprehensive Guide to Secure Cloud Computing 1st Edition, Kindle Edition
2. Cloud Security Automation: Get to grips with automating your cloud security on AWS and OpenStack

Online References

1. https://www.mygreatlearning.com/academy/premium/cloud-security-practices?utm_source=chatgpt.com
2. https://www.isc2.org/professional-development/express-courses/virtualization-container-technologies-in-cloud-sec?utm_source=chatgpt.com
3. https://www.mygreatlearning.com/academy/premium/cloud-security-practices?utm_source=chatgpt.com

Third Year B.Tech Artificial Intelligence & Data Science

Semester VI

Course Code: AI24HON392R **Course Name:** Cloud Security and DevOps Lab

Teaching Scheme (Hours/Week)				Examination Scheme						Credits				
L	P	T	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	ODL	TOTAL
-	4	-	-	-	-	50	-	-	50	-	2	-	-	2

Prerequisite: : Cloud Practitioner / Introduction to Cloud Computing , Basic knowledge of networking, Linux, and web applications

Course Objectives:

- To understand core cloud security concepts including multi-tenancy, IAM, and governance.
- To implement security controls for virtual machines, containers, and cloud storage.
- To apply data protection techniques such as encryption at rest and in transit using IBM Cloud.
- To configure secure communication using digital certificates and Istio.
- To implement DevSecOps practices by integrating security testing into CI/CD pipelines using Jenkins or GitHub.

After learning the course, the students will be able to:

CO1: Implement secure multi-tenant access control and cloud governance using IAM in Amazon Web Services or Microsoft Azure.

CO2: Apply virtualization and container security techniques including VM hardening and Docker network isolation.

CO3: Configure encryption for data at rest and in transit, including SSL/TLS management in IBM Cloud.

CO4: Implement secure microservices communication using Istio with mTLS.

CO5: Integrate security controls into CI/CD pipelines to enable DevSecOps using Jenkins or GitHub.

List of Experiments

Sr. No.	Name of the Experiment	Duration (Hrs.)
1	Create two cloud user accounts representing different tenants on Amazon Web Services. Configure secure storage permissions and verify that one tenant cannot access the other tenant's data.	04
2	Launch a virtual machine and apply basic security hardening such as disabling unused ports, enabling firewall rules, and configuring secure login methods to protect the VM.	04
3	Run two containerized applications using **Docker**. Configure basic network isolation and test connectivity to ensure containers are securely separated.	04
4	Design and implement basic cloud governance by creating users and roles on **Amazon Web Services** or **Microsoft Azure**. Assign least-privilege permissions to each user and role, and verify access control by reviewing audit logs to ensure compliance with governance policies.	04
5	Configure data encryption at rest and enable secure data transmission using SSL/TLS certificates on **IBM Cloud**. Verify that stored data remains protected from unauthorized access and that data in transit is securely encrypted to prevent interception or tampering.	04

6	Generate and configure digital certificates to enable HTTPS communication for a cloud-hosted application on **IBM Cloud**. Verify secure data transmission and demonstrate how digital certificates protect against man-in-the-middle attacks.	04
7	Deploy two simple microservices and configure secure service-to-service communication using **Istio**. Enable mutual TLS mTLS) to ensure that data exchanged between microservices is encrypted and protected during transit.	08
8	Design and implement a basic CI/CD pipeline using **Jenkins** or **GitHub** and integrate vulnerability scanning tools to detect security issues before deployment. Demonstrate how automated security checks prevent vulnerable applications from being released into production.	08
9	Mini project to Deployment of web Application	12
	Total Hours	52

Text Books

1. Enterprise Cloud Security and Governance: Efficiently Set Data Protection and Privacy Principles by Zeal Vora

Reference Books

1. Cloud Security: A Comprehensive Guide to Secure Cloud Computing 1st Edition, Kindle Edition
2. Cloud Security Automation: Get to grips with automating your cloud security on AWS and OpenStack

Online References

1. https://www.mygreatlearning.com/academy/premium/cloud-security-practices?utm_source=chatgpt.com
2. https://www.isc2.org/professional-development/express-courses/virtualization-container-technologies-in-cloud-sec?utm_source=chatgpt.com
3. https://www.mygreatlearning.com/academy/premium/cloud-security-practices?utm_source=chatgpt.com

Final Year B.Tech Artificial Intelligence & Data Science														
Semester VII														
Course Code: AI24HON441R				Course Name: Cloud Professionals Industry Practices										
Teaching Scheme (Hours/Week)				Examination Scheme						Credits				
L	P	T	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	ODL	TOTAL
-	-	-	1	-	-	25	-	-	25	-	-	-	1	1

Course Objectives:

- To understand the fundamentals of cloud computing and cloud service models.
- To learn the roles and responsibilities of cloud professionals in industry.
- To study cloud architecture, deployment models, and best practices.
- To understand security, compliance, and data protection in cloud environments.
- To explore DevOps, automation, monitoring, and cost optimization techniques in cloud computing.

Course Outcomes:

After completing the course, the students will be able to:

CO1: Explain the basic concepts of cloud computing and cloud service models.

CO2: Understand the roles and practices followed by cloud professionals in industry.

CO3: Apply cloud architecture principles for scalable and reliable systems.

CO4: Implement security, compliance, and data protection strategies in cloud environments.

CO5: Use DevOps tools, automation, and monitoring techniques for cloud operations.

Unit	Contents		Hrs
1	**Introduction to Cloud Computing** Basics of cloud computing, Characteristics of cloud computing, Cloud service models (IaaS, PaaS, SaaS), Cloud deployment models (Public, Private, Hybrid, Multi-cloud), Roles of cloud professionals		6
2	**Cloud Architecture and Design** Cloud architecture components, Scalability and high availability, Microservices architecture, Load balancing, Fault tolerance and performance optimization		6
3	**Cloud Security and Compliance** Cloud security principles, Identity and Access Management (IAM), Data encryption and protection, Network security in cloud, Compliance standards (ISO, GDPR, HIPAA)		6
4	**DevOps and Cloud Automation** DevOps concepts in cloud, Infrastructure as Code (IaC), Continuous Integration and Continuous Deployment (CI/CD), Containerization (Docker, Kubernetes), Monitoring and logging tools		6
5	**Cloud Cost Management and Future Trends** Cloud cost optimization techniques, Resource management and auto-scaling, Backup and disaster recovery, Edge computing, Emerging trends in cloud computing		6
	Total Hours		30

Text Books

1. Rajkumar Buyya, Christian Vecchiola, and S. Thamarai Selvi, Mastering Cloud Computing Foundations and Applications Programming, Tata McGraw-Hill Education.
2. Thomas Erl, Ricardo Puttini, Zaigham Mahmood, Cloud Computing: Concepts, Technology & Architecture, Pearson Education.
3. Barrie Sosinsky, Cloud Computing Bible, Wiley India.

Reference Book

1. Michael J. Kavis, Architecting the Cloud: Design Decisions for Cloud Computing Service Models, Wiley Publications.
2. John Rhoton and Robert Haukioja, Cloud Computing Architected: Solution Design Handbook, Recursive Press.
3. NIST Special Publication, The NIST Definition of Cloud Computing – Peter Mell & Timothy Grance.

Online References

1. Amazon Web Services (AWS) Documentation: https://docs.aws.amazon.com/
2. Microsoft Azure Documentation: https://learn.microsoft.com/azure
3. Google Cloud Documentation: https://cloud.google.com/docs
4. Coursera / edX Cloud Computing Courses: https://www.coursera.org, https://www.edx.org

Final Year B.Tech Artificial Intelligence & Data Science														
Semester VII														
Course Code: AI24HON442R						Course Name: Internship on Cloud Professionals Industry Practices								
Teaching Scheme (Hours/Week)				Examination Scheme						Credits				
L	P	T	ODL	CIE	ETE	TW	PR	OR	TOTAL	L	P	T	ODL	Total
-	4	-	-	-	-	50	-	-	50	-	2	-	-	2

Course Objectives:

- To understand and design secure and scalable cloud infrastructure architectures for modern applications.
- To develop and deploy serverless cloud applications using appropriate cloud services and tools.
- To explore multi-cloud integration strategies for improving flexibility, reliability, and vendor independence.
- To analyze and implement cloud cost optimization and performance monitoring techniques for efficient resource utilization.
- To enhance students' ability to document, present, and demonstrate cloud-based solutions through technical reports, presentations, and final deployment.

To **Course Outcomes:**

After completing the course, the students will be able to:

CO1: Design and evaluate secure cloud infrastructure architectures based on industry best practices.

CO2: Develop, deploy, and manage serverless applications in a cloud environment.

CO3: Implement multi-cloud solutions and integrate services across different cloud platforms.

CO4: Monitor, analyze, and optimize cloud performance and operational costs using appropriate tools.

CO5: Prepare technical documentation, project reports, and presentations, and effectively demonstrate the final cloud solution.

CO6: Demonstrate professional communication, documentation, and ethical responsibility.

CO6: Demonstrate professional communication, documentation, and ethical responsibility.

Sr. No	Industry Practices for Internship
1	Industry Practices for Cloud Computing Internship
2	Cloud Security & Compliance Frameworks
3	Industry-Scale Cloud Architecture Design
4	Advanced Cloud Automation & DevOps Practices
5	Cloud Performance Optimization & Cost Management
Total Hours	52
Mandatory Guidelines	

The internship is designed to help students understand the role of technology in sustainability, ESG practices, and responsible AI systems. Students are expected to actively explore the given topics, connect theoretical concepts with real-world applications, and develop solutions that promote energy efficiency and sustainable system design. Throughout the internship, students should maintain regular interaction with mentors, seek feedback, and document their learning and progress in a systematic manner. They are encouraged to review research articles, industry case studies, and sustainability reports to gain a broader understanding of how organizations implement ESG practices using technology.

While working on different modules, students should adopt a problem-solving and analytical approach, experiment with different ideas, and evaluate the effectiveness of their solutions. They should focus on designing systems that are efficient, scalable, and environmentally responsible. Students are also expected to follow ethical guidelines in AI and data usage, ensuring transparency, fairness, and respect for privacy. Proper version control, testing of developed solutions, and maintaining clear documentation of system design and implementation should be practiced throughout the internship.

Students should also demonstrate professional behavior, time management, and teamwork skills when interacting with mentors or collaborators. They are encouraged to think creatively, suggest improvements, and reflect on how their technical work contributes to sustainability and ESG goals. At the end of the internship, students must compile their work into a well-structured report, clearly describing the problem addressed, methodology used, system design, and sustainability impact. The internship will conclude with a final presentation, system demonstration, and viva, where students will share their learning experience and outcomes of the project.

Final Year B. Tech Artificial Intelligence & Data Science															
Semester VIII															
Course Code: AI24HON491R					Course Name: Cloud Professionals Capstone Project										
Teaching Scheme (Hours/Week)					Examination Scheme						Credits				
L	P	T	OL	ODL	CIE	ETE	TW	PR	OR	TOTAL	L	P	T	ODL	TOTAL
-	4	-	-	-	-	-	50	-	-	50	-	2	-	-	2

Course Objectives:

- To understand and design secure and scalable cloud infrastructure architectures for modern applications.
- To develop and deploy serverless cloud applications using appropriate cloud services and tools.
- To explore multi-cloud integration strategies for improving flexibility, reliability, and vendor independence.
- To analyze and implement cloud cost optimization and performance monitoring techniques for efficient resource utilization.
- To enhance students' ability to document, present, and demonstrate cloud-based solutions through technical reports, presentations, and final deployment.

Course Outcomes:

After completing the course, the students will be able to:

CO1: Design and evaluate secure cloud infrastructure architectures based on industry best practices.

CO2: Develop, deploy, and manage serverless applications in a cloud environment.

CO3: Implement multi-cloud solutions and integrate services across different cloud platforms.

CO4: Monitor, analyze, and optimize cloud performance and operational costs using appropriate tools.

CO5: Prepare technical documentation, project reports, and presentations, and effectively demonstrate the final cloud solution.

CO6: Demonstrate professional communication, documentation, and ethical responsibility.

CO6: Demonstrate professional communication, documentation, and ethical responsibility.

Unit	Contents	Hrs
1	Capstone project proposal	4
2	Cloud Architecture Design & Optimization	8
3	Full-stack cloud implementation	8
4	Cost Efficiency & Resource Management in Cloud	8
5	Performance optimization & final deployment	8
6	Technical paper / report preparation	8
7	Final demo, presentation, and viva voce	8
	Total Hours	52

Mandatory Capstone AI Requirements
Group of 3-4 students

Each project must include:

- At least one AI-powered component or intelligent cloud service integrated into the system
- Secure cloud architecture justification following industry best practices
- Cloud resource utilization and cost optimization analysis
- Performance monitoring and scalability evaluation report
- Deployment documentation with testing and validation results
- Technical report and final presentation with live cloud deployment demonstration