॥ यः क्रियावान् स पण्डितः ॥

SAVITRIBAI PHULE PUNE UNIVERSITY

(Formerly University of Pune)

SYLLABUS

T.E. ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

2024 PATTERN

(With effect from Academic Year 2026-27)

National Education Policy (NEP) - 2020 Compliant Curriculum

www.unipune.ac.in

॥ यः क्रियावान् स पण्डितः ॥

Contents

Nomenclature

AI & DS Artificial Intelligence and Data Science

AICTE All India Council for Technical Education

CCE Comprehensive Continuous Evaluation

ELC Experiential Learning Course

ESE End-Semester Examination

GAPC Graduate Attributes and Professional Competencies

KAP Knowledge and Attitude Profile

MDM Multidisciplinary Minor

NEP National Education Policy

OE Open Elective

OJT On Job Training

PCC Programme Core Course

PEC Programme Elective Course

PEO Programme Educational Objectives

PO Programme Outcomes

PSO Program Specific Outcomes

QPD Question Paper Delivery

SAR Self-Assessment Report

SPPU Savitribai Phule Pune University

UGC University Grants Commission

VSE Vocational and Skills Enhancement Course

WK Knowledge and Attitude Profile

Preface by Board of Studies

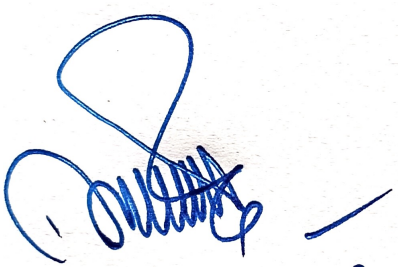
Dear Students and Teachers,

We, the members of Board of Studies Computer Engineering, are very happy to present Third Year Artificial Intelligence and Data Science syllabus effective from the Academic Year 2026-27.

Artificial Intelligence and Data Science have emerged as transformative forces reshaping industries, driving innovation, and impacting our daily lives. Recognizing the growing importance and pervasive nature of these fields, we have designed this comprehensive syllabus to equip students with the foundational knowledge and practical skills. This curriculum is meticulously crafted to provide a holistic learning experience, blending theoretical concepts with hands-on applications. The revised syllabus falls in line with the objectives of NEP-2020, Savitribai Phule Pune University , AICTE, UGC, and various accreditation agencies by keeping an eye on the technological developments, innovations, and industry requirements.

Learners are now getting sufficient time for self learning either through online courses or additional projects for enhancing their knowledge and skill sets. We believe that this well-structured and comprehensive syllabus will serve as a robust foundation for aspiring Computer Engineering and AI professionals, enabling them to contribute significantly to the technological progress and address the challenges of the 21st century.

We would like to place on record our gratefulness to the faculty, students, industry experts and stakeholders for having helped us in the formulation of this syllabus.



Dr. Nilesh Uke
Chairman - Board of Studies (Computer Engineering)
Savitribai Phule Pune University

Members of Board of Studies - Computer Engineering	
Dr. Pramod Patil	Dr. Dipti Patil
Dr. Dhananjay Kshirsagar	Dr. Amol Potgantwar
Dr. Sachin Babar	Dr. Balwant Sonkamble
Dr. Suhasini Itkar	Dr. Sachin Sakhare
Dr. Dipak Patil	Dr. Vandana Dhingra
Dr. Deepali Ujalambkar	Dr. Vaishali Vikhe
Dr. Pradip Jawandhiya	Dr. Sandeep Deshmukh

Program Specific Outcomes (PSO)

- **PSO1:** Demonstrate proficiency in essential concepts of computer science and data science and programming solutions.
- **PSO2:** Formulate robust software design, execution, and testing strategies employing a software paradigms and Artificial Intelligence knowledge to solve real world problems.
- **PSO3:** Apply the techniques of AI and Data Science for forecasting future events in the domain of Healthcare, Education, and Agriculture, Automation , Transport etc

Programme Educational Objectives (PEO)

Program Educational Objectives are broad statements that describe the career and professional accomplishments that the program is preparing graduates to achieve.

PEO	PEO Focus	PEO Statements
PEO1	Core competence	To produce graduates equipped with cutting-edge skills in Artificial Intelligence (AI) and Data Science (DS), with expertise in domains such as Machine Learning (ML), Natural Language Processing (NLP), Generative AI, enabling them to collaborate effectively in interdisciplinary teams to solve real-world industrial and societal challenges.
PEO2	Problem solving skills and Ethics	To empower graduates to think critically, apply mathematical, computational, and ethical frameworks, and design scalable, secure, and fair AI-driven systems
PEO3	Professionalism and Lifelong Learning	To inculcate the ability to adapt to changing technology through continuous learning and contribute to research, innovation, and entrepreneurship in AI and Data Science.

Knowledge and Attitude Profile (WK)

A Knowledge and Attitude Profile (KAP), often represented as WK (Knowledge and Attitude Profile) in some contexts, is a framework or assessment tool used to evaluate an individual's knowledge and attitudes related to a specific area, topic, or domain.

WK1	A systematic, theory-based understanding of the natural sciences applicable to the discipline and awareness of relevant social sciences.
WK2	Conceptually-based mathematics, numerical analysis, data analysis, statistics and formal aspects of computer and information science to support detailed analysis and modelling applicable to the discipline.
WK3	A systematic, theory-based formulation of engineering fundamentals required in the engineering discipline.
WK4	Engineering specialist knowledge that provides theoretical frameworks and bodies of knowledge for the accepted practice areas in the engineering discipline; much is at the forefront of the discipline.
WK5	Knowledge, including efficient resource use, environmental impacts, whole-life cost, re-use of resources, net zero carbon, and similar concepts, that supports engineering design and operations in a practice area.
WK6	Knowledge of engineering practice (technology) in the practice areas in the engineering discipline.
WK7	Knowledge of the role of engineering in society and identified issues in engineering practice in the discipline, such as the professional responsibility of an engineer to public safety and sustainable development.
WK8	Engagement with selected knowledge in the current research literature of the discipline, awareness of the power of critical thinking and creative approaches to evaluate emerging issues.
WK9	Ethics, inclusive behavior and conduct. Knowledge of professional ethics, responsibilities, and norms of engineering practice. Awareness of the need for diversity by reason of ethnicity, gender, age, physical ability etc. with mutual understanding and respect, and of inclusive attitudes.

Reference: Self-Assessment Report (SAR) Format Undergraduate Engineering Programs Graduate Attributes and Professional Competencies Version 4.0 (GAPC V4.0) - (August 2024) Page 55.

Programme Outcomes (PO)

Program Outcomes are statements that describe what students are expected to know and be able to do upon graduating from the program. These relate to the skills, knowledge, attitude and behaviour that students acquire through the program. On successful completion of B.E. in Artificial Intelligence and Data Science, graduating students/graduates will be able to:

PO1	Engineering knowledge	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	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/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)

Reference: Self-Assessment Report (SAR) Format Undergraduate Engineering Programs Graduate Attributes and Professional Competencies Version 4.0 (GAPC V4.0) - (August 2024) Page 56.

General Rules and Guidelines

- **Course Outcomes (CO):** Course Outcomes are narrower statements that describe what students are expected to know, and are able to do at the end of each course. These relate to the skills, knowledge and behaviour that students acquire in their progress through the course.
- **Assessment:** Assessment is one or more processes, carried out by the institution, that identify, collect, and prepare data to evaluate the achievement of Program Educational Objectives and Program Outcomes.
- **Evaluation:** Evaluation is one or more processes, done by the Evaluation Team, for interpreting the data and evidence accumulated through assessment practices. Evaluation determines the extent to which Program Educational Objectives or Program Outcomes are being achieved, and results in decisions and actions to improve the program

Guidelines for Examination Scheme

Theory Examination: The theory examination shall be conducted in two different parts Comprehensive Continuous Evaluation (CCE) and End-Semester Examination (ESE).

Comprehensive Continuous Evaluation (CCE) :

1. CCE of 30 marks based on all the Units of course syllabus to be scheduled and conducted at institute level.
2. Case studies included under each unit are intended to support applied learning and are part of Comprehensive Continuous Evaluation
3. These case studies will be assessed through internal assessment components such as presentations, assignments, or group discussions. They shall not be included in the End-Semester Theory Examination.
4. To design a Comprehensive Continuous Evaluation scheme for a theory subject of 30 marks with the specified parameters, the allocation of marks and the structure can be detailed as follows:

Sr.	Parameters	Marks	Coverage of Units
1	Unit Test	12 Marks	Units 1 & Unit 2 (6 Marks/Unit)
2	Assignments / Case Study	12 Marks	Units 3 & Unit 4 (6 Marks/Unit)
3	Seminar Presentation / Open Book Test/ Quiz	06 Marks	Unit 5

5. CCE of 15 marks based on all the Units of course syllabus to be scheduled and conducted at institute level. To design a CCE scheme for a theory subject of 15 marks with the specified parameters, the allocation of marks and the structure can be detailed as follows:

Sr.	Parameters	Marks	Coverage of Units
1	Unit Test	10 Marks	Units 1 & Unit 2 (5 Marks/Unit)
2	Seminar Presentation / Open Book Test/ Assignments/Case Studies	05 Marks	Units 3 & Unit 4

- **Unit Test**
  - **Format :** Questions designed as per Bloom's Taxonomy guidelines to assess various cognitive levels (Remember, Understand, Apply, Analyze, Evaluate, Create).
  - **Implementation**: Schedule the test after completing Units 1 and 2. Ensure the question paper is balanced and covers key concepts and applications.
- **Sample Question Distribution**
  - Remembering (2 Marks): Define key terms related to [Topic from Units 1 and 2].
  - Understanding (2 Marks): Explain the principle of [Concept] in [Context].
  - Applying (2 Marks): Demonstrate how [Concept] can be used in [Scenario].
  - Analyzing (3 Marks): Compare & contrast [Two related concepts] from Units 1 and 2.
  - Evaluating (3 Marks): Evaluate the effectiveness of [Theory/Model] in [Situation].
- **Assignments / Case Study** : Students should submit one assignment or one Case Study Report based on Unit 3 and one assignment or one Case Study Report based on Unit 4.
  - **Format:** Problem-solving tasks, theoretical questions, practical exercises, or case studies that require in-depth analysis and application of concepts.
  - **Implementation:** Distribute the assignments or case study after covering Units 3 and 4. Provide clear guidelines and a rubric for evaluation.
- **Seminar Presentation:**
  - **Format:** Oral presentation on a topic from Unit 5, followed by a Q&A session.
  - **Deliverables:** Presentation slides, a summary report in 2 to 3 pages, and performance during the presentation.
  - **Implementation:** Schedule the seminar presentations towards the end of the course. Provide students with ample time to prepare and offer guidance on presentation skills.
- **Open Book Test:**
  - **Format:** Analytical and application-based questions to assess depth of understanding.
  - **Implementation:** Schedule the open book test towards the end of the course, ensuring it covers critical aspects of Unit 5.
- **Quiz :**
  - **Format:** Quizzes can help your students practice existing knowledge while stimulating interest in learning about new topic in that course. You can set your quizzes to be completed individually or in small groups.
  - **Implementation:** Online tools and software can be used create quiz. Each quiz is made up of a variety of question types including multiple choice, missing words, true or false etc
- **Example Timeline for conducting CCE:**
  - Weeks 1-4 : Cover Units 1 and 2
  - Week 5 : Conduct Unit Test (12 marks)
  - Weeks 6-8 : Cover Units 3 and 4

- Week 9 : Distribute and collect Assignments / Case Study (12 marks)
- Weeks 10-12 : Cover Unit 5
- Week 13 : Conduct Seminar Presentations or Open Book Test or Quiz (6 marks)

- **Evaluation and Feedback:**
  - **Unit Test:** Evaluate promptly and provide constructive feedback on strengths and areas for improvement.
  - **Assignments / Case Study:** Assess the quality of submissions based on the provided rubric. Offer feedback to help students understand their performance.
  - **Seminar Presentation:** Evaluate based on content, delivery, and engagement during the Q&A session. Provide feedback on presentation skills and comprehension of the topic.
  - **Open Book Test**: Evaluate based on the depth of analysis and application of concepts. Provide feedback on critical thinking and problem-solving skills.

End-Semester Examination (ESE)

End-Semester Examination (ESE) of 70 marks written theory examination based on all the unit of course syllabus scheduled by university. Question papers will be sent by the University through QPD (Question Paper Delivery). University will schedule and conduct ESE at the end of the semester.

- **Format and Implementation :**
  - **Question Paper Design** : Below structure is to be followed to design an End-Semester Examination (ESE) for a theory subject of 70 marks on all 5 units of the syllabus with questions set as per Bloom's Taxonomy guidelines and 14 marks allocated per unit.
  - **Balanced Coverage**: Ensure balanced coverage of all units with questions that assess different cognitive levels of Bloom's Taxonomy: Remember, Understand, Apply, Analyze, Evaluate, and Create. The questions should be structured to cover:
    - Remembering: Basic recall of facts and concepts.
    - Understanding: Explanation of ideas or concepts.
    - Applying: Use of information in new situations.
    - Analyzing: Drawing connections among ideas.
    - Evaluating: Justifying a decision or course of action.
    - Creating: Producing new or original work (if applicable).
  - **Detailed Scheme for 70 Marks :** Unit-Wise Allocation (14 Marks per Unit): Each unit will have a combination of questions designed to assess different cognitive levels. By following this scheme, you can ensure a comprehensive and fair assessment of students' understanding and application of the course material, adhering to Bloom's Taxonomy guidelines for cognitive skills evaluation.
  - **Detailed Scheme for 35 Marks :** Unit-Wise Allocation (08 Marks for Unit 1 , 09 Marks for Unit 2, Unit 3 and Unit 4) : Each unit will have a combination of questions designed to assess different cognitive levels. By following this scheme, you can ensure a comprehensive and fair assessment of students' understanding and application of the course material, adhering to Bloom's Taxonomy guidelines for cognitive skills evaluation.

***

Curriculum Structure - Semester V

Third Year Engineering (2024 Pattern) – Artificial Intelligence and Data Science

Course Code	Course Name	Course Type	Teaching Scheme			Examination Scheme						Credits			
			Theory	Tut	Practical	CCE	ESE	Term Work	Practical	Oral	Total	Theory	Tut	Practical	Total
PCC301AID	Networks and Security	PCC	3	-	-	30	70	-	-	-	100	3	-	-	3
PCC302AID	Machine Learning	PCC	3	-	-	30	70	-	-	-	100	3	-	-	3
PCC303AID	Natural Language Processing	PCC	3	-	-	30	70	-	-	-	100	3	-	-	3
PCC304AID	Natural Language Processing Lab	PCC		-	2	-	-	25	-	25	50	-	-	1	1
PCC305AID	Machine Learning Lab	PCC		-	4	-	-	50	25	-	75	-	-	2	2
PEC321AID	Elective I	PEC	3	-	-	30	70	-	-	-	100	3	-	-	3
PEC322AID	Elective I Lab	PEC		-	2	-	-	50	-	-	50	-		1	1
MDM331AID	Internet of Things	MDM		2	2	-	-	25		25	50	-	2	1	3
	Open Elective*	OE*	2	-	-	15	35		-	-	50	2	-	-	2
ELC342AID	Technical Seminar	ELC		-	2	-	-	-	-	25	25	-	-	1	1
Total			**14**	**2**	**12**	**135**	**315**	**150**	**25**	**75**	**700**	**14**	**2**	**6**	**22**

***Note:** Students can opt for Open Electives offered by different discipline/faculty like Arts, Science, Commerce, Management, Humanities or Inter-Disciplinary studies. Example – Open Elective I - IPR and Cyber Laws, Sustainability Development, Digital Personal Data Protection, The Constitution of India etc.

Programme Elective I	
PEC321AAID	Blockchain
PEC321BAID	Cloud Computing
PEC321CAID	Computer Vision
PEC321DAID	Gaming and Animation

Curriculum Structure - Semester - VI

Third Year Engineering (2024 Pattern) – Artificial Intelligence and Data Science

Course Code	Course Name	Course Type	Teaching Scheme			Examination Scheme						Credits			
			Theory	Tut	Practical	CCE	ESE	Term Work	Practical	Oral	Total	Theory	Tut	Practical	Total
PCC351AID	Software Engineering	PCC	3	-	-	30	70	-	-	-	100	3	-	-	3
PCC352AID	Deep Learning	PCC	3	-	-	30	70	-	-	-	100	3	-	-	3
PCC353AID	Deep Learning Lab	PCC	-	-	4	-	-	25	25	-	50	-	-	2	2
PCC354AID	Software Engineering Lab	PCC	-	-	2	-	-	25	-	25	50	-	-	1	1
PEC361AID	Elective II	PEC	2	-	-	30	70	-	-	-	100	2	-	-	2
PEC362AID	Elective III	PEC	3	-	-	30	70	-	-	-	100	3	-	-	3
PEC363AID	Elective II- Lab	PEC		-	2	-	-	25		25	50	-	-	1	1
MDM371AID	Robotics and Automation	MDM		1	2	-	-	50	-	-	50	-	1	1	2
VSE372AID	Solar Technology and Development	VSE	-	-	2	-	-	50	-	-	50	-	-	1	1
ELC381AID	Internship/OJT	ELC	-	-	8	-	-	25		25	50	-	-	4	4
Total			**11**	**1**	**20**	**120**	**280**	**200**	**25**	**75**	**700**	**11**	**1**	**10**	**22**

Programme Elective II	
PEC361AAID	Cyber Security and Data Privacy
PEC361BAID	Big Data Engineering
PEC361CAID	High Performance Computing
PEC361DAID	AR/VR

Programme Elective III	
PEC362AAID	Ethical Hacking
PEC362BAID	Quantum AI
PEC362CAID	Reinforcement Learning
PEC362DAID	UI/UX Design

Savitribai Phule Pune University, Pune

Maharashtra, India

TE - Artificial Intelligence and Data Science

Semester - V

Savitribai Phule Pune University **Third Year - Artificial Intelligence and Data Science (2024 Pattern)**		
PEC301AID: Network and Security		
Teaching Scheme	**Credits**	**Examination Scheme**
Theory : 03 Hours/Week	02	CCE : 30 Marks End-Semester : 70 Marks

Prerequisite Courses: Computer Organization and Operating Systems. , Data Communication and Digital Communication

Course Objectives: The course aims to:

1. Explain the fundamental concepts of computer networks, architectures, protocols, and networking technologies.
2. Apply Data Link and Network layer protocols, addressing schemes, and routing concepts in network communication.
3. Analyze transport and application layer protocols and their role in end-to-end network communication.
4. Analyze network security threats, attacks, and defence mechanisms used to protect network infrastructures.
5. Apply cryptographic techniques and access control models to enhance network security.

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1 :** Describe the fundamental concepts of computer networks, including architectures, protocols, and networking technologies.
- **CO2 :** Apply Data Link and Network layer protocols, addressing, and routing concepts to design and simulate computer networks.
- **CO3:** Analyze transport and application layer protocols for reliable data communication and network services
- **CO4:** Identify and analyze network security threats, attack methodologies, and defense mechanisms to secure network infrastructures.
- **CO5:** Apply cryptographic techniques and appropriate access control models to ensure secure data communication and resource protection in information systems.

Course Contents

Unit I - Introduction to Computer Networks (09 Hours)

Definition, Types of Networks: Local area networks (LAN), Metropolitan area networks (MAN), Wide area networks (WAN), Wireless networks, Networks Protocol, The OSI Reference Model, TCP/IP Model, Network Topologies, Types of Transmission Medium. Network Architectures: Client-Server, Peer-to-Peer, Hybrid. Network Devices: Bridge, Switch, Router, Gateway, Access Point, Design issues of layers, AI powered networks, application of artificial intelligence in computer networks

Case Study: Design case study for Network Architecture Used in Modern Banking Systems or Case Study on AI-Powered Network

Unit II - Data Link Layer and Network Layer Protocols (09 Hours)

Data Link Layer: Functions and services to Network Layer; framing techniques; error detection and correction (Parity, Hamming Code, CRC); ARQ strategies (Stop-and-Wait, Go-Back-N, Selective

Repeat); flow control (Sliding Window); MAC protocols (ALOHA, CSMA/CD, CSMA/CA, Binary Exponential Backoff); overview of IEEE 802.3 and IEEE 802.11 standards; WAN protocols (PPP, HDLC).

Network Layer: Functions and switching techniques (Circuit, Message, Packet Switching); IP addressing (IPv4, IPv6, subnetting, CIDR); Network Address Translation (NAT); Network layer protocols (ARP, ICMP); routing concepts (Static and Dynamic Routing, Distance Vector, Link State); routing protocols (RIP, OSPF, BGP – introductory level).

Security Integration: Logical network segmentation, controlled access through subnetting/NAT, and secure routing fundamentals.

Case Study: Design and simulate a Smart Campus network using Cisco Packet Tracer by implementing IPv4 addressing with subnetting/CIDR, NAT, static and dynamic routing (RIP/OSPF), and analyzing Data Link Layer error control and Network Layer packet forwarding through routing tables and simulation results.

Unit III Transport and Application Layer Protocols (09 Hours)

Transport Layer: Fundamentals, Functions of Transport Layer, Process-to-process communication and port addressing, Segmentation and reassembly, Multiplexing and de-multiplexing; Protocols - TCP - Features, TCP segment structure, connection establishment (three-way handshake), flow control, congestion control; UDP - Characteristics, UDP segment structure, comparison of TCP and UDP;

Port Numbers and Socket Programming Concepts: Well-known, registered and dynamic port numbers, Socket concept and client-server communication model;

Application Layer: Fundamentals, Functions of Application Layer, Client-server architecture and service models; Protocols: HTTP and HTTPS – web communication basics, DNS – domain name resolution process, FTP – file transfer mechanisms, SMTP, POP3, IMAP – electronic mail protocols overview

Case Study: Develop a basic secure Live TV Streaming application to demonstrate client-server communication.

Unit IV Network Security Threats and Attacks (09 Hours)

Network Security Fundamentals: CIA triad, threat types, attack lifecycle, attack surface. Reconnaissance and Discovery Attacks: Foot printing, OSINT, scanning, enumeration, vulnerability identification;

Packet and MITM Attacks: Sniffing, ARP spoofing, MAC flooding, session hijacking, and MITM; Denial of Service and Botnet Attacks: DoS, DDoS, botnets, flooding, amplification attacks; Spoofing and Protocol Attacks: IP spoofing, DNS spoofing, email spoofing, and routing threats overview.

Wireless and Malware-based Threats: Rogue AP, evil twin attack overview, worm propagation, ransomware spread. Network Defense Mechanisms: Firewall, IDS/IPS, segmentation, defense-in-depth.

Case Study: Develop a secure Internet Banking application and analyze the impact of ARP Spoofing and DDoS attacks on banking services, along with suitable prevention and mitigation techniques.

Unit V Cryptography and Access Control Models (09 Hours)

Introduction to Cryptography: Need for Cryptography in Network Security, Security services: Confidentiality, Integrity, Authentication, Non-repudiation, Basic terminology, Types of cryptographic attacks: Brute force, Cryptanalysis.

Types: Symmetric Key Cryptography: DES, AES; Asymmetric Key Cryptography: RSA. Hash Functions and Digital Signatures.

Access Control Models: Discretionary Access Control (DAC), Mandatory Access Control (MAC), Role-Based Access Control (RBAC), Attribute-Based Access Control (ABAC)

Case Study: Securing an AI-Based Healthcare Data Platform

Learning Resources

Text Books:

1. James F. Kurose, Keith W. Ross, “Computer Networking: A Top-Down Approach”, 8th Edition, Pearson, 2021.

2. Wichert, Andreas, “Quantum Artificial Intelligence with Qiskit”, Chapman & Hall / CRC Press, 1st Edition, 2025, ISBN: 978-1032448978.

3. Andrew S. Tanenbaum, Nick Feamster, David J. Wetherall, “Computer Networks”, 6th Edition, Pearson Education, 2021.

4. Behrouz A. Forouzan, “Data Communications and Networking”,6th Edition, McGraw-Hill Education, 2022.

5. William Stallings, “Cryptography and Network Security: Principles and Practice”, 7th Edition, Pearson Education, 2017.

Reference Books

1. William Stallings, “Data and Computer Communications”, 10th Edition, Pearson Education, 2014.

2. Larry L. Peterson, Bruce S. Davie, “Computer Networks: A Systems Approach”, 5th Edition, Morgan Kaufmann, 2012.

3. Douglas E. Comer, “Computer Network and Internet”, 6th Edition, Pearson Education, 2015.

4. William Stallings, “Network Security Essentials: Applications and Standards”, 6th Edition, Pearson Education, 2017.

5. Matt Bishop, “Computer Security: Art and Science”, 2nd Edition, Addison-Wesley, 2018.

6. V. K. Pachghare, “Cryptography and Information Security”, 3rd Edition, PHI Learning.

MOOC / NPTEL/YouTube Links

1. Computer Networks and Network Security, IBM Skills Network Team – By Coursera

2. Quantum Computing by Prof .Debabrata Goswami, IIT Kanpur - https://nptel.ac.in/courses/104104082

3. Network Security, by Prof. Gaurav S. Kasbekar, IIT Bombay https://onlinecourses.nptel.ac.in/noc25_ee5

E-Books

1. https://www.youtube.com/watch?v=KsL_uwa0ekY

2. https://www.youtube.com/@JimKurose

3. https://www.youtube.com/@learningacademy1006

Savitribai Phule Pune University **Third Year Artificial Intelligence and Data Science- 2024 Pattern**		
PCC302AID : Machine Learning		
Teaching Scheme	**Credits**	**Examination Scheme**
Theory : 03 Hours/Week	03	CCE : 30 Marks End-Semester : 70 Marks

Prerequisite Courses: Probability and Statistics, Data Science, Python Programming
Course Objectives: The course aims to:

1. Provide the fundamental concepts of Machine Learning.
2. Develop an understanding of regression concepts, techniques, and evaluation metrics used for predictive modeling.
3. Imbibe knowledge of classification models and algorithms for solving real-world classification problems.
4. Familiarize students with clustering algorithms and ensemble learning techniques.
5. Give insight into reinforcement learning concepts and their use in sequential decision-making problems.

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1:** Apply fundamental Machine Learning concepts in various learning paradigms and real-world engineering applications.
- **CO2:** Make use of various types of regression models for predictive modeling and data analysis.
- **CO3:** Identify different types of classification problems, including binary, multiclass, balanced, and imbalanced classification.
- **CO4:** Analyze clustering algorithms for grouping similar data points and ensemble learning techniques for improving model performance.
- **CO5:** Distinguish reinforcement learning from supervised and unsupervised learning approaches.

Course Contents

Unit I - Fundamentals of Machine Learning (09 Hours)

Introduction to machine learning, scope of machine learning, AI vs ML vs Data Science, traditional programming vs ML paradigm, and real-world engineering applications. Types of Learning: Supervised, unsupervised, semi-supervised, and reinforcement learning. Models of Machine Learning: Geometric model, probabilistic models, logical models, grouping and grading models, parametric and non-parametric models. Introduction to Feature Engineering. Feature Transformation: Dimensionality reduction techniques- Principal Component Analysis (PCA); Linear Discriminant Analysis (LDA).

Case Study: Machine Learning Based Student Performance Prediction and Feature Engineering Analysis

Unit II - Supervised Learning-Regression (09 Hours)

Introduction to regression, need of regression, regression vs correlation. Types of regression: Univariate vs Multivariate, Linear vs Nonlinear, Simple vs Multiple, Bias–Variance Tradeoff, Overfitting and Underfitting. **Regression Techniques**: Simple and Multiple Linear Regression; Polynomial Regression; Decision Tree Regression, Random Forest Regression, Support Vector Regression. Regularization

Techniques: Ridge Regression (L2); Lasso Regression (L1).
Evaluation Metrics: Mean Squared Error (MSE); Mean Absolute Error (MAE); Root Mean Squared Error (RMSE); R-squared (R^2).

Case Study: Comparative Study of Regression Techniques for House Price Prediction

Unit III - Supervised Learning-Classification (09 Hours)

Introduction to classification, need of classification. Types of Classification: Binary and Multiclass, Binary vs. Multiclass Classification, Balanced and Imbalanced Classification Problems. Binary Classification: Linear classification model, decision boundary. Performance Evaluation: Confusion Matrix, Accuracy, Precision, Recall, F1-Score. **Multiclass Classification:** One-vs-One and One-vs-All classification techniques, multiclass confusion matrix; Per-Class Precision and Per-Class Recall; Macro, Micro and Weighted Averaging Methods.

Classification Algorithms: K-Nearest Neighbors (KNN), Linear Support Vector Machine (SVM), Soft Margin SVM. **Kernel Functions in SVM:** Radial Basis Function (RBF/Gaussian) Kernel, Polynomial Kernel, Sigmoid Kernel.

Case Study: Comparative Study of Classification Algorithms for Email Spam Detection.

Unit IV - Unsupervised Learning and Ensemble Learning -(09 Hours)

Introduction to clustering, need for clustering, types of clustering, Hierarchical Clustering – Agglomerative and Divisive methods, Partitioning Methods: K-Means clustering algorithm, advantages and limitations, Elbow method, Silhouette method; K-Medoids, Density-Based Clustering: DBSCAN algorithm, working mechanism, advantages and limitations. **Distribution-Based Clustering:** Gaussian Mixture Model. Applications, introduction to Ensemble Learning, homogeneous and heterogeneous ensemble methods, advantages and limitations. Basic Ensemble Techniques: Voting (Max Voting, Averaging, Weighted Averaging). Advanced Ensemble Techniques: Bagging and Random Forest. Boosting: AdaBoost, Gradient Boosting , Stacking.

Case Study: Customer Segmentation and Sales Prediction

Unit V - Reinforcement Learning - (09 Hours)

Introduction, need of reinforcement learning, components of reinforcement learning, comparison with supervised and unsupervised learning, applications and challenges of reinforcement learning. Markov Decision Process: Markov property, elements of MDP, episodic and continuing tasks. **Reinforcement Learning Framework:** Policy, state value function, action value function, Bellman equation, optimal policy. **Reinforcement Learning Algorithms:** Exploration vs Exploitation, ε-greedy strategy, dynamic programming, Q-Learning algorithm and update rule, simple reinforcement learning for game playing- Tic-Tac-Toe.

Case Study: Smart Traffic Signal Control using Q-Learning.

Learning Resources

Text Books:

1. Alpaydin, Ethem, “Introduction to Machine Learning”, 2nd Edition, MIT Press, 2014, ISBN: 978-0262028189.
2. Müller, Andreas C. and Guido, Sarah, “Introduction to Machine Learning with Python: A Guide for Data Scientists”, 1st Edition, O’Reilly Media, 2016, ISBN: 978-1-449-36941-5.
3. Mitchell, Tom M., “Machine Learning”, 1st Edition, McGraw-Hill Education, 1997, ISBN: 978-0070428072.

Reference Books

1. Flach, Peter, “Machine Learning: The Art and Science of Algorithms that Make Sense of Data,” 1st Edition, Cambridge University Press, 2012, ISBN: 978-1107422223.

2. Murphy, Kevin P., “Machine Learning: A Probabilistic Perspective”, 1st Edition, MIT Press, 2012, ISBN: 978-0262018029.

3. Shalev-Shwartz, S., & Ben-David, S., “Understanding Machine Learning: From Theory to Algorithms”, 1st Edition, Cambridge University Press, 2014, ISBN: 978-1107057135.

MOOC / NPTEL/YouTube Links

1. NPTEL Course: Introduction to Machine Learning, by Prof.Balaraman Ravindran , IIT Madras https://nptel.ac.in/courses/106106139

2. NPTEL Course: Introduction to Machine Learning, by Prof.Sudeshna Sarkar.IIT kharagpur https://nptel.a

3. Fundamentals of Machine Learning: https://www.youtube.com/watch?v=65BlnPBLbXM

4. Machine Learning Basics: https://www.youtube.com/watch?v=bytGpKPWvIY

E-Books

1. Hastie, Trevor, Tibshirani, Robert, & Friedman, Jerome, “The Elements of Statistical Learning: Data Mining, Inference, and Prediction”, 2nd Edition, Springer, 2009, ISBN: 978-0387848570. https://hastie.su.domains/ElemStatLearn/

2. Sutton, Richard S., & Barto, Andrew G., “Reinforcement Learning: An Introduction”, 2nd Edition, MIT Press, 2018, ISBN: 978-0262039246. https://www.andrew.cmu.edu/course/10-703/textbook/Bart

Savitribai Phule Pune University **Third Year Artificial Intelligence and Data Science- 2024 Pattern**		
PCC303AID - Natural Language Processing		
Teaching Scheme	**Credits**	**Examination Scheme**
Theory : 03 Hours/Week	03	CCE : 30 Marks End-Semester : 70 Marks

Prerequisite Courses: Data Structures , Artificial Intelligence , Probability & Statistics

Course Objectives: The course aims to:

1. Introduce students to the fundamental concepts of natural language processing and linguistic structures.
2. Enable students to apply text preprocessing techniques, corpus handling methods, and feature extraction approaches.
3. Facilitate understanding of statistical models and parsing techniques used in NLP tasks.
4. Help students use NLP models to design simple text-processing applications.
5. Enable students to understand modern approaches in neural NLP, including deep learning and transformer architectures.

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1:** Examine NLP fundamentals, levels of linguistic analysis, and grammatical formalisms.
- **CO2:** Utilize text preprocessing, corpus handling, and feature extraction techniques.
- **CO3:** Interpret statistical NLP techniques such as N-grams, smoothing, and probabilistic parsing.
- **CO4:** Demonstrate NLP tasks such as POS tagging, NER, and sentiment analysis.
- **CO5:** Analyze modern NLP concepts, including attention mechanisms, encoder–decoder models, and BERT architecture.

Course Contents

Unit I - Linguistic Foundations and Formal Language Models (09 Hours)

Introduction to Natural Language Processing: History of NLP, Generic NLP System, Levels of NLP, Knowledge in Language Processing, Ambiguity in Natural Language, Stages, Challenges, and Applications of NLP

Formal Language Concepts: Alphabets, strings, languages, regular expressions, finite automata (DFA/NFA), and limitations of regular languages for natural language.

Context-Free Grammars : CFG components, Derivations, parse trees, Ambiguity

Linguistic Levels: Morphology (roots, affixes, morphological parsing), Finite-State Transducers (FSTs) for morphology, Syntax, semantics, pragmatics,

Edit Distance and Spelling Correction: Minimum Edit Distance (MED), Levenshtein distance, Noisy channel model

Approaches to NLP: Rule-based, Data-based, and Knowledge-based Approaches.

Python-based NLP libraries: Natural Language Toolkit (NLTK), spaCy, TextBlob, and use cases.

Case Study: Design a simple rule-based system to validate and analyze basic English sentences such as “The cat eats food” using regular expressions and Context-Free Grammar (CFG). The system should preprocess the text, define grammar rules for sentence structure (Subject–Verb–Object), generate a parse tree for valid sentences, and explain why regular expressions alone cannot fully capture nested or hierarchical sentence structures.

Unit II - Text Representation & Feature Engineering (09 Hours)

Corpus: Types of corpora, Annotation and formats, POS and NER annotation schemes, Annotation tools

Text Preprocessing: Tokenization, Sentence segmentation, Normalization, true casing, Stemming and lemmatization, Text Cleaning, stemming (Porter Stemmer algorithm), Stopword Removal, Handling OOV (Out-of-vocabulary) **Words:** Unknown tokens, subword techniques, and challenges with NLP systems.

Word Representation: One-hot, distributional semantics, co-occurrence matrices, and information extraction. Sparse vs Dense Representations: Limitations of sparse vectors (high dimensionality, sparsity) and advantages of dense embeddings, Lexical Resources: WordNet, synsets, semantic relations.

Feature Engineering: Concept of feature engineering, types of features, and K-gram models: Character-level and word-level k-grams and their applications in text representation, spelling correction, and information retrieval.

Feature Extraction: Bag of Words, Document-Term Matrix, TF-IDF;

Feature selection: Chi-square, Mutual Information

Word embeddings: Word2Vec (CBOW, Skip-gram), GloVe, Embedding intuition

Case Study: Sentiment Analysis of Product Reviews, Spam Email Detection System

Unit III - Statistical NLP Models and Language Modeling (09 Hours)

Probability Basics: Conditional Probability concept and applications in NLP; Maximum Likelihood Estimation (MLE)—parameter estimation techniques for languages and statistical models.

N-gram Language Models: Unigram, bigram, trigram, Training, perplexity, Backoff and interpolation

Classification Models: Naive Bayes for text, Logistic Regression, Evaluation for text classification

Tagging Models: Hidden Markov Models (HMMs), Viterbi algorithm, HMM-based POS tagging

Parsing with Probabilities: PCFG, Probabilistic parsing, Top-down and bottom-up parsing

Case Study: Develop and analyze a Part-of-Speech (POS) tagging system using a probabilistic sequence model.

Unit IV - NLP Tasks (9 Hours)

Sequence Labeling Tasks: POS Tagging (HMM, Brill tagger), Named Entity Recognition (rule-based + statistical view) Parsing: Dependency parsing (transition-based and graph-based concepts), constituency vs. dependency, Parsing errors and evaluation Evaluation Metrics: Confusion matrix, Precision, Recall, F1, evaluation for tagging and parsing, token-level accuracy, entity-level F1 (NER), Parse accuracy, Attachment score (basic idea), BLEU, ROUGE. Core **NLP Applications:** Sentiment analysis, Text classification pipeline, document similarity, Recommendation System, Information retrieval: Vector Space Model, Information Extraction using sequence labelling, sentiment analysis, Word **Sense Disambiguation (WSD):** concept, Lesk algorithm, and applications. Speech Processing: Mel-Frequency Cepstral Coefficients (MFCC), Automatic Speech Recognition (ASR) pipeline Model Deployment: API-based NLP Services, Introduction to Model Deployment and Inference, REST APIs, Request–Response Architecture, Overview of Hugging Face Inference APIs, Workflow of NLP Deployment, Introduction to Cloud Platforms, Basic Concepts of Containerization, and Performance Monitoring with Ethical Considerations.

Case Study: BERT-based text classification for customer support queries

Unit V - Modern Neural NLP and Transformers (9 Hours)

Neural NLP Overview: Motivation, limits of statistical models, Dense vector representations

Attention Mechanism: Intuition and motivation, Self-attention, Scaled dot-product attention, Advantages over traditional models

Encoder–Decoder Models: Sequence-to-sequence architecture, applications, machine translation methods (rule-based machine translation (RBMT), Statistical Machine Translation (SMT), Neural Machine Translation (NMT)), Text summarization

Transformer Architecture: Multi-head attention, Positional embeddings, Feedforward layers, Advantages over RNN

Contextual Embeddings: BERT architecture, Token embeddings, segment embeddings. Downstream tasks: QA, text classification

Modern Applications: Chatbots, Sentiment analysis, Document classification, Retrieval-Augmented Generation (RAG)

Case Study: AI-Powered Academic Chatbot System. Retrieval Augmented Question Answering System. Legal Document Summarization. Voice enabled regional language assistant.

Learning Resources

Text Books:

1. Jurafsky, D., & Martin, J. H., Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition, 3rd ed., Pearson / Online, 2023–Present.

2. Manning, C. D., & Schütze, H., Foundations of Statistical Natural Language Processing, MIT Press, 1999.

Reference Books

1. T V Geetha, "Understanding Natural Language Processing – Machine Learning and Deep Learning Perspectives", Pearson, 2024.

2. Steven Bird,Ewan Klein and Edword Loper," NLP with Python: Analyzing text with the Natural Language Toolkit", O'Reilly Media,Inc

3. Nitin Indurkhya and Fred J. Damerau, "Handbook of Natural Language Processing", 2nd ed. CRC press.

MOOC / NPTEL/YouTube Links

1. https://nptel.ac.in/courses/106101007

2. https://onlinecourses.nptel.ac.in/noc26_cs45/preview

E-Books

1. Yoav Goldberg. A primer on neural network models for natural language processing, 2015. URL http://u.cs.biu.ac.il/˜yogo/nnlp.pdf

Savitribai Phule Pune University **Third Year Artificial Intelligence and Data Science- 2024 Pattern**		
PCC304AID - Natural Language Processing Laboratory		
Teaching Scheme	**Credits**	**Examination Scheme**
Practical : 02 Hours/Week	01	Term Work: 25 Marks Oral: 25 Marks

Prerequisite Courses : Probability and Statistics, Data Science, Python Programming

Companion Course: Machine Learning

Course Objectives:

1. Apply text preprocessing and feature extraction techniques for transforming raw textual data into structured representations.
2. Implement classical and neural machine learning models for natural language processing tasks.
3. Analyze and evaluate NLP models using appropriate performance metrics.
4. Design and develop NLP-based applications using modern tools and transformer models.
5. Introduce students to real-world NLP systems such as chatbots, semantic search, and summarization

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1:** Implement text preprocessing and linguistic analysis techniques using standard NLP libraries.
- **CO2:** Extract and analyze statistical and embedding-based text representations.
- **CO3:** Develop and evaluate machine learning models for text classification tasks.
- **CO4:** Implement transformer-based applications such as summarization, semantic search, and question answering.
- **CO5:** Design and demonstrate an end-to-end NLP system integrating preprocessing, modeling, and evaluation.

Guidelines for Instructor's Manual

The instructor‘s manual is to be developed as a reference and hands-on resource. It should include prologue (about University/program/ institute/ department/foreword/ preface), curriculum of the course, conduction and Assessment guidelines, topics under consideration, concept, objectives, outcomes, set of typical applications/assignments/ guidelines, and references.

Guidelines for Student's Laboratory Journal

The laboratory assignments are to be submitted by student in the form of journal. Journal consists of Certificate, table of contents, and handwritten write-up of each assignment (Title, Date of Completion, Objectives, Problem Statement, Software and Hardware requirements, Assessment grade/marks and assessor's sign, Theory- Concept in brief, algorithm, flowchart, test cases, Test Data Set(if applicable), mathematical model (if applicable), conclusion/analysis). Program codes with sample output of all performed assignments are to be submitted as softcopy. As a conscious effort and little contribution towards Green IT and environment awareness, attaching printed papers as part of write-ups and program listing to journal must be avoided. Use of DVD containing students programs maintained by Laboratory In-charge is highly encouraged. For reference one or two journals may be maintained with program prints in the Laboratory.

Guidelines for Laboratory /Term Work Assessment

Continuous assessment of laboratory work should be based on overall performance of Laboratory assignments by a student. Each Laboratory assignment assessment will assign grade/marks based on parameters, such as timely completion, performance, innovation, efficient codes, and punctuality.

Guidelines for Laboratory Conduction

The NLP Laboratory shall be conducted with an emphasis on hands-on implementation of theoretical concepts using standard NLP libraries and tools. Each laboratory session should begin with a brief discussion of the underlying concepts, algorithms, and expected outcomes before implementation. Students are required to perform experiments individually for Part A and Part B assignments and collaboratively for the mini-project. Proper documentation of problem statements, methodology, datasets used, code implementation, evaluation metrics, and result analysis must be maintained in the laboratory journal. Emphasis should be given to experimentation, comparison of models, performance evaluation, and use of modern tools such as NLTK, spaCy, Scikit-learn, and Transformer frameworks. Continuous assessment should consider timely submission, correctness of implementation, innovation, clarity of analysis, and viva performance.

Suggested List of Laboratory Experiments/Assignments

Group A (Any 4)

1. NLP Pipeline and Linguistic Analysis: Develop a Python-based NLP system to apply and analyze the basic stages of Natural Language Processing on a given text corpus. The system should:
   - Perform preprocessing:
   - Tokenization
   - Stop-word removal
   - Lemmatization/Stemming
   - POS tagging
   - Identify morphological components (root, suffix)
   - Interpret ambiguity in sentences and classify type (lexical/syntactic).
   - Compare outputs using two NLP libraries (NLTK and spaCy).
2. Regular Expressions and Finite Automata: Design and implement a system using Regular Expressions to apply and evaluate pattern matching techniques for structured text processing. The system should Construct regular expressions to validate: Email IDs, Phone numbers, Dates, Extract patterns, URLs,
   (a) Hashtags Design and analyze a Finite Automaton (DFA/NFA) for Binary strings ending with "01"
   (b) Evaluate limitations of regular languages in representing nested sentence structures.
3. Edit Distance and Spelling Correction: Develop a system to apply and analyze string similarity techniques for spelling correction. The system should:
   - Implement Minimum Edit Distance (MED) / Levenshtein distance.
   - Compute and compare distances between Misspelled and correct words
   - Apply the Noisy Channel Model to suggest the most probable correction.
   - Evaluate performance of the system on a dataset.
4. Text Preprocessing and OOV Handling: Design and implement a preprocessing pipeline for textual data. The system should perform:
   - Tokenization

- Sentence segmentation
- Stopword removal
- Stemming and Lemmatization
- Text normalization and true casing
- Identify and handle Out-of-Vocabulary (OOV) words using:
- Unknown token replacement
- Analyze and compare text before and after preprocessing.
- Dataset: IMDB Movie Reviews Dataset , Twitter dataset (noisy text)

5. Feature Extraction and Text Representation: Develop a system to convert text into numerical representations. The system should
   (a) Implement: One-hot encoding , Bag of Words (BoW) , Document-Term Matrix , TF-IDF
   (b) Compare and analyze: Sparse vs dense representations , Feature dimensionality and sparsity ,
   (c) Interpret and visualize feature vectors. Dataset: Custom dataset (10–20 documents) , 20 Newsgroups dataset (subset)

Group B (Any 4)

1. Text Classification using Naïve Bayes and Logistic Regression: Develop a text classification system to classify documents into predefined categories. The system should:
   (a) Preprocess the text data (tokenization, stopword removal).
   (b) Convert text into numerical features using TF-IDF.
   (c) Implement: Naïve Bayes classifier , Logistic Regression model
   (d) Evaluate model performance using: Accuracy , Precision, Recall, F1-score ,Confusion Matrix
   (e) Compare the performance of both models and justify the results. (Dataset: IMDB Movie Reviews Dataset , SMS Spam Dataset)

2. N-gram Language Model and Perplexity Analysis: Develop a language model to predict word sequences using N-gram techniques. The system should:
   (a) Build : Unigram , Bigram , Trigram models
   (b) Estimate probabilities using Maximum Likelihood Estimation (MLE).
   (c) Apply smoothing techniques (Laplace or Good-Turing).
   (d) Compute perplexity for each model.
   (e) Compare model performance and analyze the effect of smoothing. (Dataset: Brown Corpus , Gutenberg Corpus)

3. POS Tagging using Hidden Markov Model (HMM): Develop a Part-of-Speech (POS) tagging system using Hidden Markov Models. The system should:
   (a) Model POS tagging using HMM (states, observations).
   (b) Estimate transition and emission probabilities.
   (c) Implement the Viterbi algorithm to find the most probable tag sequence.
   (d) Evaluate tagging accuracy on test sentences.
   (e) Analyze errors and discuss limitations of HMM. (Dataset: Penn Treebank Dataset , NLTK tagged corpus)

4. Named Entity Recognition (NER) and Evaluation: Develop a Named Entity Recognition (NER) system to identify entities such as persons, locations, and organizations. The system should:
   (a) Preprocess input text data.
   (b) Apply NER using:
      i. Rule-based approach (pattern matching)
      ii. Pre-trained statistical model (spaCy)
   (c) Extract and classify named entities.
   (d) Evaluate model performance using: Precision , Recall , F1-score
   (e) Compare rule-based and statistical approaches and justify the results. (Dataset:CoNLL-2003 dataset , Custom news articles)
5. Dependency Parsing and Sentence Analysis: Develop a system to analyze grammatical relationships in sentences using dependency parsing. The system should:
   (a) Parse sentences using a dependency parser.
   (b) Identify syntactic relationships (subject, object, modifiers).
   (c) Compare: Dependency parsing ,Constituency parsing (conceptual comparison)
   (d) Visualize parse trees/graphs.
   (e) Analyze parsing errors and discuss limitations. (Dataset:Custom sentence dataset)
6. Sentiment Analysis and Text Classification Pipeline: Develop a sentiment analysis system using a complete NLP pipeline. The system should:
   (a) Perform preprocessing and feature extraction (TF-IDF).
   (b) Implement a classification model (Logistic Regression / Naïve Bayes).
   (c) Classify text into positive/negative sentiments.
   (d) Evaluate performance using: Accuracy , Precision, Recall, F1-score , Confusion Matrix
   (e) Analyze model performance and suggest improvements.(Dataset:IMDB Movie Reviews Dataset , Twitter Sentiment Dataset)
7. Text Classification using BERT : Develop a text classification system using a pre-trained BERT model. The system should:
   (a) Load a pre-trained BERT model using a transformer library.
   (b) Preprocess input text using tokenization and encoding.
   (c) Fine-tune BERT for classification (e.g., sentiment analysis).
   (d) Evaluate model performance using: Accuracy , Precision, Recall, F1-score,
   (e) Dataset:(IMDB Movie Reviews Dataset , SST-2 Dataset)

Group C (Any 1)

1. Text Summarization using Transformer Models : Develop a text summarization system using a transformer-based model. The system should:
   (a) Use a pre-trained model (BART/T5) for summarization.
   (b) Input long text documents and generate summaries.
   (c) Evaluate summaries using: ROUGE-1, ROUGE-2, ROUGE-L
   (d) Compare extractive vs abstractive summarization results.

(e) Dataset: (CNN/Daily Mail dataset, News article dataset)

2. Chatbot using LLM with Prompt Engineering: Develop a chatbot using a large language model. The system should:

   (a) Design prompts for different tasks (Q&A, summarization, classification).

   (b) Generate responses using an LLM API.

   (c) Evaluate response quality and consistency.

   (d) Improve performance using prompt engineering techniques.

   (e) Dataset:(Custom queries , FAQ dataset)

Savitribai Phule Pune University **Third Year Artificial Intelligence and Data Science- 2024 Pattern**		
PCC353AID- Machine Learning Lab		
Teaching Scheme	**Credits**	**Examination Scheme**
Practical : 04 Hours/Week	02	Term Work: 50 Marks Practical: 25 Marks

Prerequisite Courses: Probability and Statistics, Data Science, Python Programming

Companion Course : Machine Learning

Course Objectives:

1. Enable students to apply feature engineering techniques, implement PCA and LDA for feature reduction, and use a multiple linear regression model with suitable open-access datasets.
2. Provide knowledge of polynomial regression, regularization techniques, and logistic regression using suitable open-access datasets.
3. Introduce Support Vector Machine classifiers, K-Means clustering, and DBSCAN using suitable open-access datasets.
4. Familiarize with ensemble learning techniques and reinforcement learning using suitable datasets and simple decision-making problems.

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1:** Apply feature engineering, dimensionality reduction techniques, and multiple linear regression to prepare datasets for predictive modeling.
- **CO2:** Demonstrate the use of polynomial regression, regularization techniques, and logistic regression for regression and classification modeling using suitable datasets.
- **CO3:** Make use of Support Vector Machine classifiers, K-Means clustering, and DBSCAN to perform classification and clustering on suitable datasets.
- **CO4:** Evaluate predictive modeling and simple decision-making tasks using ensemble learning techniques and the Q-Learning algorithm.

Guidelines for Instructor's Manual

The instructor's manual is to be developed as a reference and hands-on resource. It should include prologue (about University/program/ institute/ department/foreword/ preface), curriculum of the course, conduction and Assessment guidelines, topics under consideration, concept, objectives, outcomes, set of typical applications/assignments/ guidelines, and references.

Guidelines for Student's Laboratory Journal

The laboratory assignments are to be submitted by student in the form of journal. Journal consists of Certificate, table of contents, and handwritten write-up of each assignment (Title, Date of Completion, Objectives, Problem Statement, Software and Hardware requirements, Assessment grade/marks and assessor's sign, Theory- Concept in brief, algorithm, flowchart, test cases, Test Data Set(if applicable), mathematical model (if applicable), conclusion/analysis). Program codes with sample output of all performed assignments are to be submitted as softcopy. As a conscious effort and little contribution towards Green IT and environment awareness, attaching printed papers as part of write-ups and program listing to journal must be avoided. Use of DVD containing students programs maintained by Laboratory In-charge is highly encouraged. For reference one or two journals may be maintained with program prints in the Laboratory.

Guidelines for Laboratory /Term Work Assessment

Continuous assessment of laboratory work should be based on overall performance of Laboratory assignments by a student. Each Laboratory assignment assessment will assign grade/marks based on parameters, such as timely completion, performance, innovation, efficient codes, and punctuality.

Guidelines for Practical Examination

Problem statements must be decided jointly by the internal examiner and external examiner. During practical assessment, maximum weightage should be given to satisfactory implementation of the problem statement. Relevant questions may be asked at the time of evaluation to test the student's understanding of the fundamentals, effective and efficient implementation. This will encourage, transparent evaluation and fair approach, and hence will not create any uncertainty or doubt in the minds of the students. So, adhering to these principles will consummate our team efforts to the promising start of student's academics.

Guidelines for Laboratory Conduction

The instructor is expected to frame the assignments by understanding the prerequisites, technological aspects, utility and recent trends related to the topic. The assignment framing policy needs to address the average students and inclusive of an element to attract and promote the intelligent students. Use of open source software is encouraged. Based on the concepts learned, Instructors may also set one assignment or mini-project that is suitable to AI & DS branch beyond the scope of the syllabus.

Operating System recommended: - 64-bit Open Source Linux or its derivatives, or Windows OS.

Programming tools recommended: - Open Source Python, Programming tools like Jupyter Notebook, Pycharm, Spyder etc.

Suggested List of Laboratory Experiments/Assignments

1. Apply basic feature engineering techniques such as handling missing values, encoding categorical variables, and scaling numerical attributes on the suitable open access dataset to prepare a clean and model ready dataset for machine learning analysis.
2. Make use of dimensionality reduction process using PCA and LDA on a suitable open-access high-dimensional dataset and observe the influence of feature reduction on model performance and computational efficiency.
3. Develop a multiple linear regression model using a suitable real-world open-access dataset and assess its predictive capability using regression evaluation metrics such as MSE, RMSE, and R^2.
4. Demonstrate the use of polynomial regression with varying polynomial degrees on a suitable open-access dataset and observe the effect of model complexity on prediction accuracy in terms of the bias–variance trade-off.
5. Classify data using regularization techniques such as Ridge and Lasso regression on a suitable multivariate open-access dataset and observe the influence on model overfitting and multi-collinearity.
6. Illustrate the performance of a binary classification model using logistic regression on a suitable open-access dataset through a confusion matrix, precision, recall, and F1-score.
7. Utilize Support Vector Machine classifiers with linear and nonlinear kernels on a suitable open-access dataset and compare classification performance using appropriate evaluation metrics.
8. Apply the K-Means clustering algorithm on a suitable open-access dataset to obtain the optimal number of clusters using the Elbow or Silhouette method.
9. Employ a density-based clustering algorithm such as DBSCAN using a suitable open-access dataset to identify the clustering structure and noise points.

10. Analyze the performance of ensemble learning techniques such as Random Forest or Boosting on a suitable open-access dataset in comparison with a baseline machine learning model.

11. Evaluate the performance of reinforcement learning using the Q-Learning algorithm for a simple game or decision-making problem (e.g., Tic-Tac-Toe or Grid World), focusing on exploration–exploitation strategies and the optimal policy.

Mini Project: Design and develop a Machine Learning application for a real-world problem using suitable ML techniques, including data preprocessing, model training, testing, and performance evaluation. Students may implement classification, regression, clustering, recommendation, or predictive analytics–based solutions and present meaningful insights through the developed application.

Learning Resources

Virtual Laboratory (links)

- https://vlab.spit.ac.in/ai/
- https://ml-iitr.vlabs.ac.in/
- https://teachablemachine.withgoogle.com/
- https://www.openml.org/

MOOC Courses (Web Links)

- NPTEL Course: Introduction to Machine Learning, by Prof.Balaraman Ravindran , IIT Madras https://nptel.ac.in/courses/106106139
- NPTEL Course: Introduction to Machine Learning, by Prof.Sudeshna Sarkar.IIT kharagpur https://nptel.a

Savitribai Phule Pune University **Third Year Artificial Intelligence and Data Science- 2024 Pattern**		
PEC321AAID : Blockchain		
Teaching Scheme	**Credits**	**Examination Scheme**
Theory : 03 Hours/Week	03	CCE : 30 Marks End-Semester: 70 Marks

Prerequisite Courses: Data Structures, Network and Security, Machine Learning

Course Objectives: The course aims to:

1. Identify the fundamentals of blockchain technology and its applications in AI and data science
2. Examine cryptocurrency, smart contracts, and decentralized data systems
3. Analyze consensus algorithms and their role in distributed ledger systems
4. Construct smart contracts using Ethereum and Solidity for data-driven applications
5. Evaluate blockchain solutions for real-world AI and data science challenges

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1:** Describe the fundamentals of blockchain technology
- **CO2:** Demonstrate the use of cryptocurrency wallets to perform blockchain transactions.
- **CO3:** Analyze appropriate consensus mechanisms for distributed systems
- **CO4:** Construct smart contracts using Ethereum and Solidity for data-driven applications
- **CO5:** Develop blockchain-based solutions for applications in AI and data science

Course Contents

Unit I - Introduction to Blockchain and Cryptographic Foundations (09 Hours)

Cryptography: Symmetric Key Cryptography and Asymmetric Key Cryptography, Elliptic Curve Cryptography (ECC), Cryptographic Hash Functions: SHA256, Digital Signature Algorithm (DSA), Merkle Trees **Blockchain Basics:** History, Limitation of Centralized System, Decentralized Systems, Layers in Blockchain **Types of Blockchain:** Public, Private and Consortium

Case Study: Hash-based integrity verification

Unit II - Cryptocurrency and Bitcoin (09 Hours)

Introduction: Bitcoin and the Cryptocurrency, Cryptocurrency, Bitcoin Transactions and Scripts, Wallets and Keys, Mining Process, Blockchain Security Mechanisms, Cryptocurrency Economics **Types of Cryptocurrency:** Cryptocurrency Usage, Cryptowallets: Metamask, Coinbase, Binance **Types of Blockchain Platforms**: Bitcoin, Ethereum, Hyperledger, IoTA, Corda, and R3.

Case Study: Create your own wallet for cryptocurrency using any of the blockchain platforms.

Unit III Consensus Mechanism (09 Hours)

Blockchain Architecture: Layers Of Blockchain: Application Layer, Execution Layer, Semantic Layer, Propagation Layer, Consensus Layer **Consensus Algorithms:** Proof of Work, Byzantine General Problem, Proof of Stake, Proof of Elapsed Time, Proof of Activity, Proof of Burn, Proof of Authority, Scalability and Performance

Case Study: Select an appropriate consensus mechanism (such as PBFT or PoA) for a blockchain-based decentralized energy trading system

Unit IV - Ethereum and Smart Contracts (09 Hours)

Ethereum Platform Architecture: Need of Ethereum, Type of Ethereum Platforms, Ethereum Virtual Machine (EVM), Gas and Transaction Fees, Smart Contract Fundamentals, Solidity Programming Language, Deploying And Interacting with Smart Contracts using Remix, Decentralized Applications (Dapps), Decentralized Storage (IPFS) Web3.Js Basics

Case Study: Deploy a smart contract for the supply chain

Unit V - Blockchain Applications (09 Hours)

Blockchain For Data Provenance and Integrity, Decentralized Data Marketplaces, Blockchain In Healthcare (Medical Records, Drug Traceability), Supply Chain Transparency And Tracking, Financial Services And Defi, Identity Management Systems, Voting Systems, NFTs And Digital Assets, Integration of Blockchain with AI/ML Models

Case Study: Secure model sharing using blockchain

Learning Resources

Text Books:

1. Imran Bashir, "Mastering Blockchain," 2nd Edition, Packt Publishing, 2018
2. Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller, and Steven Goldfeder, "Bitcoin and Cryptocurrency Technologies", Princeton University Press, ISBN:9780691171692 1st Edition, 2016

Reference Books

1. Andreas M. Antonopoulos, Mastering Bitcoin: Programming the Open Blockchain, 2nd Edition, O'Reilly Media, Sebastopol, California, USA, 2017
2. Melanie Swan, Blockchain: Blueprint for a New Economy, O'Reilly Media, Sebastopol, California, USA, 2015.
3. Daniel Drescher, Blockchain Basics: A Non-Technical Introduction in 25 Steps, Apress, New York, USA, 2017.
4. William Mougayar, The Business Blockchain: Promise, Practice, and Application of the Next Internet Technology, Wiley, Hoboken, New Jersey, USA, 2016.
5. Arshdeep Bahga and Vijay K. Madisetti, Blockchain Applications: A Hands-On Approach, VPT Publications, Blacksburg, Virginia, USA, 2017.

MOOC / NPTEL/YouTube Links

1. NPTEL Course on "Introduction to Blockchain Technology & Applications" Link: https://nptel.ac.in/cours
2. NPTEL Course on "Blockchain and its Applications"Link: https://nptel.ac.in/courses/106/105/1061051
3. Coursera: "Blockchain Basics" by University at Buffalo
4. edX: "Blockchain for Business" by The Linux Foundation Quantum Computing by Prof .Debabrata Goswami, IIT Kanpur Link: https://nptel.ac.in/courses/104104082

E-Books

1. https://www.asau.ru/files/pdf/1789486.pdf

2. https://users.cs.fiu.edu/~prabakar/cen5079/Common/textbooks/Mastering_Blockchain_2nd _Edition.pdf

3. https://www.lopp.net/pdf/princeton_bitcoin_book.pdf

4. https://www.blockchainexpert.uk/book/blockchain-book.pdf

Savitribai Phule Pune University **Third Year Artificial Intelligence and Data Science- 2024 Pattern**		
PEC321BAID - Cloud Computing		
Teaching Scheme	**Credits**	**Examination Scheme**
Theory : 03 Hours/Week	03	CCE : 30 Marks End-Semester: 70 Marks

Prerequisite Courses: Computer Networks

Companion Course: Programming Elective Course Lab (PEC322-COM)

Course Objectives: The course aims to:

1. To understand the basic concepts of cloud computing and virtualization
2. To understand the implementation of virtualization in cloud computing
3. To learn the application and security on cloud
4. To study risk management in cloud computing
5. To comprehend the modern cloud environment and emerging technologies

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1:** Comprehend basic concepts of cloud computing environment
- **CO2:** Analyze Virtualization for cloud and install Virtualization software
- **CO3:** Deploy applications on Cloud
- **CO4:** Apply security in cloud applications
- **CO5:** Analyze emerging technologies in modern cloud computing

Course Contents

Unit I - Introduction to Cloud Computing (09 Hours)

Cloud Fundamentals: Definition, Importance of cloud computing, Advantages and Disadvantages of Cloud Computing, Characteristics, Categories of Clouds: Private clouds, Public clouds **Cloud Service Models**: SaaS, PaaS, IaaS, Cloud Architecture, Cloud Storage: Distributed Data Storage, Data management, Cloud Deployment Models

Case Study: Cloud Computing Model of Amazon

Unit II Virtualization in Cloud Computing (09 Hours)

Virtualization: What's virtualization, Benefits of Virtualization, Types of Virtualization: Processor virtualization, Memory virtualization, Full virtualization, Para virtualization, and Device virtualization, Virtual Clustering, Virtualization Architecture, Containerization and orchestration, Understanding importance of Hypervisors, Virtualization Applications, Issues with Virtualization, Virtualization and Cloud Computing: Virtualizations in Cloud, Virtual Infrastructure, CPU Virtualization, Network and Storage Virtualization

Case Study of VMware: Full virtualization, Xen: Para Virtualization, Microsoft HyperV

Unit III Cloud Platforms and Applications - (09 Hours)

Industrial Cloud Platforms: Amazon Web Services (AWS)- AWS infrastructure, Components, Amazon Simple DB, Elastic Cloud Computing (EC2), Amazon Storage System, Amazon Database Services. Microsoft Azure: Azure core concepts, SQL Azure, and Application Services for managed runtimes. Open Source Platforms: Overview of OpenStack, CloudStack, and Eucalyptus for private cloud deployment. Cloud Applications

Data-Intensive & Emerging Applications: Smart Cities & IoT: Integrating sensor data from traffic, waste management, and power grids into a centralized cloud dashboard. AI/ML in the Cloud: Case study on Google Photos (image recognition) or Alexa (Natural Language Processing) using cloud-based TPU/GPU instances. Healthcare & Biology: Gene sequencing, protein folding and ECG analysis in the cloud. Geoscience: Satellite image processing and seismic data analysis using cloud clusters.

Case Study: The Google Case Study Data Processing: The evolution from MapReduce to Dremel and BigQuery. Storage Innovation: Understanding the Google File System (GFS) and BigTable as the backbone of global search.

Unit IV - Security in Cloud Computing - (09 Hours)

Risks in Cloud Computing: Risk Management, Enterprise-Wide Risk Management, Types of Risks in Cloud Computing. Data Security in Cloud: Security Issues, Challenges, advantages, Disadvantages, Cloud Digital persona and Data security, Content Level Security. Cloud Security Services: Confidentiality, Integrity and Availability, Security Authorization Challenges in the Cloud, Secure Cloud Software Requirements, Secure Cloud Software Testing

Case Study: Cloud Security Tool: Acunetix

Unit V - Modern Cloud Environment & Emerging Technologies (09 Hours)

Future Trends in cloud Computing, Mobile Cloud, Comet Cloud, Multimedia Cloud: IPTV, Energy Aware Cloud Computing, Distributed Cloud Computing Vs. Edge Computing, Containers, Dockers, Kubernets, Pod Management

Green Cloud & Sustainability: Sustainable Cloud Architecture, Energy-efficient data centre design and carbon footprint tracking.

Case Studies on DevOps: DocuSign, Forter, Gengo

Learning Resources

Text Books:

1. A. Srinivasan, J. Suresh, “Cloud Computing: A Practical Approach for Learning and Implementation”, Pearson, ISBN: 978-81-317-7651-3
2. Rajkumar Buyya, Christian Vecchiola, S. Thamarai Selvi, “Mastering Cloud Computing”, McGraw Hill Education, ISBN-13:978-1-25-902995-0

Reference Books

1. James Bond ,“The Enterprise Cloud”, O’Reilly Media, Inc. ISBN: 9781491907627
2. Dr. Kris Jamsa, “Cloud Computing: SaaS, PaaS, IaaS, Virtualization and more”, Wiley Publications, ISBN: 978-0-470-97389-9
3. Anthony T. Velte Toby J. Velte, Robert Elsenpeter, “Cloud Computing: A Practical Approach”, 2010, The McGraw-Hill.
4. Gautam Shrof, “ENTERPRISE CLOUD COMPUTING Technology Architecture, Applications”, Cambridge University Press, ISBN: 9780511778476
5. Tim Mather, Subra K, Shahid L.,”Cloud Security and Privacy”, Oreilly, ISBN-13 978-81-8404-815-5

6. Ronald L. Krutz, Russell Dean Vines, “Cloud Security: A Comprehensive Guide to Secure Cloud Computing”, Wiley- India,2010

7. Rajkumar Buyya, James Broberg, Andrzej M. Goscinski, Cloud Computing: Principles and Paradigms, Editors: Wile, 2011

E-Books

1. http://dhoto.lecturer.pens.ac.id/lecture_notes/internet_of_things/CLOUD%20COMPUTING%20Princip
2. https://www.lpude.in/SLMs/Master%20of%20Computer%20Applications/Sem_2/DECAP470_CLOUD_
3. https://studytm.wordpress.com/wp-content/uploads/2014/03/hand-book-of-cloud-computing.pdf
4. https://arpitapatel.files.wordpress.com/2014/10/cloud-computing-bible1.pdf
5. https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.500-291r2.pdf

MOOC Courses:

1. https://onlinecourses.nptel.ac.in/noc26_cs55/preview
2. http://www.ndl.gov.in/he_document/nptel/nptel/N_C_S_A_E_C_C_A_D_S_N_I_T_C_C_536752663
3. https://onlinecourses.nptel.ac.in/noc26_cs29/preview?
4. https://onlinecourses.nptel.ac.in/noc21_cs15/preview?

Savitribai Phule Pune University **Third Year Artificial Intelligence and Data Science- 2024 Pattern**		
PEC321CAID: Computer Vision		
Teaching Scheme	**Credits**	**Examination Scheme**
Theory : 03 Hours/Week	03	CCE : 30 Marks End-Semester: 70 Marks

Course Objectives: The course aims to:

1. To introduce the fundamental principles of image formation, image representation, and basic image processing techniques used in computer vision systems.
2. To develop understanding of spatial and frequency domain image processing techniques for image enhancement and filtering.
3. To explore frequency domain processing and edge detection
4. To implement feature extraction and image segmentation techniques used to interpret visual information.
5. To understand advanced computer vision applications including object detection, tracking, and real-world AI vision systems.

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1:** Discuss image representation, camera models, and basic image processing concepts.
- **CO2:** Apply spatial domain filtering techniques for given image.
- **CO3:** Analyze different frequency techniques for a given application.
- **CO4:** Solve feature extraction and image segmentation tasks.
- **CO5:** Analyze modern computer vision applications such as object detection, tracking, and autonomous vision systems

Course Contents

Unit I - Fundamental Concepts of Image Processing and Computer Vision (09 Hours)

Introduction to Computer Vision: definition, scope and challenges; relationship between image processing, pattern recognition and computer vision; Human visual system vs. machine vision: gaps and opportunities, The CV pipeline: Acquisition $\rightarrow$ Preprocessing $\rightarrow$ Feature Extraction $\rightarrow$ Understanding $\rightarrow$ Actuation human visual perception vs machine vision.

Digital Image Fundamentals: digital image representation, pixels and resolution, sampling and quantization, spatial resolution, gray level resolution.

Color Image Processing: color perception, RGB color model, CMY and CMYK models, HSV/HSI color models, color transformations and color spaces.

Image Operations: point operations, intensity transformations, gray level transformations, contrast stretching, image negatives, logarithmic and power law transformations.

Histogram Processing: image histogram, histogram equalization, histogram specification.

Applications of Computer Vision: medical imaging, surveillance systems, biometrics, industrial inspection, robotics, autonomous vehicles.

Case Study: Automatic Number Plate Recognition (ANPR), An automated system captures vehicle images and extracts number plates using basic image representation and preprocessing techniques. It highlights the importance of resolution, grayscale conversion, and image acquisition in real-world vision systems.

Unit II - Image Enhancement and Spatial Domain Filtering (09 Hours)

Image Enhancement Basics: Point operations & intensity transformations, Histogram equalization & specification.

Spatial Domain Filtering: Convolution concept, Smoothing filters: Mean, Gaussian, Median, Bilateral Sharpening filters

Noise models and basic noise removal: Types of Noise Models, Gaussian Noise, Salt-and-Pepper Noise, Poisson Noise (Shot Noise) , Speckle Noise – multiplicative noise (common in medical/ultrasound images)

Case Study: Medical Image Enhancement for X-Ray Analysis

X-ray images are enhanced using filtering and histogram techniques to improve visibility of fractures and abnormalities.

This demonstrates how noise removal and contrast enhancement support accurate medical diagnosis.

Unit III - Frequency Domain Processing and Edge Detection (09 Hours)

Frequency Domain Concepts: Fourier Transform, DFT, FFT , Frequency spectrum (magnitude & phase)

Frequency filtering: Low-pass, High-pass, Band-pass

Inverse transformation: Concept of Inverse Transformation , Inverse Discrete Fourier Transform (IDFT).

Edge Detection: Sobel, Prewitt, Roberts Laplacian, LoG , Canny Edge Detector , the 'gold standard'- Gaussian smoothing, gradient, NMS, double thresholding, hysteresis, Comparison of edge detectors: performance vs. noise sensitivity

Practical application: Lane boundary detection, document scanning.

Case Study: Lane Detection in Autonomous Driving, Frequency filtering and edge detection techniques are used to identify lane boundaries from road images. It shows how edge operators and transforms help in real-time navigation and driver assistance systems.

Unit IV - Feature Extraction and Image Segmentation (09 Hours)

Feature extraction concepts: Edge-based features, Corner detection (Harris, Shi-Tomasi)

Texture analysis: Types of Texture, Statistical Texture Features, Texture Representation Methods, Frequency-Based Texture Analysis (Conceptual), Applications of Texture Analysis.

Segmentation techniques: Thresholding (global, adaptive, Otsu) , Region-based methods, Clustering Morphological operations

Shape representation & descriptors : Boundary Descriptors Region Descriptors Moment Invariants Fourier Descriptors, Shape Matching Techniques

Case Study: Panorama Image Stitching in Mobile Cameras , Multiple images are stitched by detecting and matching key features across overlapping regions. This illustrates the role of feature extraction and segmentation in creating seamless panoramic views.

Unit V - Advanced Computer Vision Applications (09 Hours)

Object Detection Techniques: sliding window approach, region proposal methods. Modern Object Detection Algorithms, Object Tracking: tracking fundamentals, Kalman filter based tracking.

Motion Analysis: optical flow, motion detection. Motion Analysis and Optical Flow.

3D Vision Basics: stereo vision, depth estimation. Human Activity Recognition. Face Detection and Recognition. Vision Systems in Robotics and Autonomous Vehicles.

Applications of Computer Vision: smart surveillance, healthcare diagnostics, agricultural monitoring, augmented reality, industrial automation.

Case Study: Face Detection and Smart Attendance System, A vision system detects and recognizes faces to automate attendance using ML/DL techniques. It demonstrates real-time detection, recognition, and ethical considerations in surveillance systems.

Learning Resources

Text Books:

1. Digital Image Processing-Rafael C. Gonzalez and Richard E. Woods, 4th Edition, Pearson Education, 2018.
2. Computer Vision: Algorithms and Applications-Richard Szeliski, Springer, 2022.
3. Computer Vision: A Modern Approach-David A. Forsyth and Jean Ponce, 2nd Edition, Pearson, 2011.
4. Learning OpenCV-Gary Bradski and Adrian Kaehler, O'Reilly Media, 2016.

Reference Books

1. Deep Learning-Ian Goodfellow, Yoshua Bengio, Aaron Courville, MIT Press, 2016.
2. Pattern Recognition and Machine Learning-Christopher M. Bishop, Springer, 2006.
3. Multiple View Geometry in Computer Vision-Richard Hartley and Andrew Zisserman, Cambridge University Press, 2004.
4. Deep Learning for Computer Vision-Rajalingappaa Shanmugamani, Packt Publishing, 2018.

MOOC / NPTEL/YouTube Links

1. Instructor: Prof. Jayanta Mukhopadhyay. IIT Kharagpur , Link: https://nptel.ac.in/courses/106105216
2. https://www.youtube.com/watch?v=1EJ84QqkxWc
3. https://www.youtube.com/watch?v=wVE8SFMSBJ0

E-Books

1. https://visionbook.mit.edu/nerf.html

Savitribai Phule Pune University **Third Year Artificial Intelligence and Data Science- 2024 Pattern**		
PEC321DAID: Gaming and Animation		
Teaching Scheme	**Credits**	**Examination Scheme**
Theory : 03 Hours/Week	03	CCE : 30 Marks End-Semester: 70 Marks

Prerequisite Courses: Mathematics, Programming
Course Objectives: The course aims to:

1. Provide foundational knowledge of computer graphics concepts relevant to animation and gaming applications.
2. Introduce fundamental techniques of 2D and 3D animation used in digital media.
3. Familiarize students with the stages and workflow of the animation production pipeline.
4. Develop an understanding of game design principles, mechanics, and development tools.
5. Encourage creative thinking and logical problem-solving in designing basic animated and gaming applications.

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1:** Explain and analyze computer graphics concepts used in animation and gaming
- **CO2:** Apply animation techniques for 2D/3D content creation.
- **CO3:** Analyze and describe animation production workflows.
- **CO4:** Design structured game mechanics and gameplay systems.
- **CO5:** Develop basic game prototypes integrating animation and gaming concepts.

Course Contents

Unit I - Fundamentals of Computer Graphics (09 Hours)

Introduction to Computer Graphics, Transformations: 2D and 3D transformations (Translation, Rotation, Scaling), Projection: Orthographic vs. Perspective projections, Cohen-Sutherland Line Clipping, Lighting Basics: Ambient, Diffuse, and Specular reflection models; Shading Algorithms: Phong Shading and Gauraud shading. Color Models (RGB, CMY, HSV)

Case Study: Graphics Transformations in Angry Birds -Study the use of 2D transformations such as translation, rotation, and scaling in character movement and object interaction. Discuss basic projection and collision representation used in simple 2D game environments.

Unit II - Principles and Techniques of Animation (09 Hours)

Definition and History of Animation, 12 Principles of Animation, Types of Animation: Traditional, Stop Motion, Computer Animation,

2D Animation Techniques: Keyframes and Tweening (Motion paths, Interpolation Techniques, Timing charts), Layer based Animation,

3D Animation Techniques: Modeling, Rigging, Skinning, Motion Capture and Simulation,

Case Study: 12 Principles of Animation in Frozen - Analyze the application of squash and stretch, anticipation, follow-through, and timing in character animation. Identify how keyframes and rigging techniques enhance realism in animated sequences.

Unit III - Animation Production Pipeline (09 Hours)

Pre-Production: Storyboarding, Scriptwriting, Concept Art, Production: Modeling, Texturing, Lighting, Rendering, Post-Production: Compositing, Editing, Sound Integration, Visual Effects (VFX) and Motion Graphics, Rendering Engines and File Formats, Role of Physics and AI in Modern Animation,

Case Study: Case Studies: Short Animation Film Workflow, Workflow of Big Buck Bunny -Examine the stages of animation production including storyboarding, modeling, lighting, rendering, and post-production. Understand how structured workflow ensures quality output.

Unit IV - Fundamentals of Game Design (09 Hours)

Introduction to Gaming: History and Genres, Game Design Elements: Story, Character, Level, and Interface Design, Gameplay Mechanics and Game Balancing, Game Flow, Rules, and Objectives, Game Art and Audio Design, Introduction to Game Engines: Unity, Unreal Engine, Godot, Prototyping and Testing of Games

Case Study: Game Mechanics in Tetris -Study core gameplay mechanics, scoring system, difficulty progression, and player engagement model. Analyze how simple design principles create addictive gameplay.

Unit V - Game Development and Integration with Animation (09 Hours)

2D and 3D Game Development Process, Physics: Motion, Gravity, Collision Detection, and User Interactions, Integration of Animated Assets in Games, Game AI Basics (Pathfinding, Decision Making), Optimization and Performance in Games, Introduction to XR (AR/VR/MR) in Gaming

Case Study: NPC Behavior in Pac-Man -Analyze ghost movement patterns as an example of basic rule-based AI behavior. Discuss state-based logic and simple decision-making in early game development.

Learning Resources

Text Books:

1. D. Hearn, M. Baker, “Computer Graphics with OpenGL”, 4th Edition, Pearson Education, 2014, ISBN 978-93-325-1871-1
2. Andrew Hogue, Rick Parent, "Computer Animation Algorithms and Techniques", 4th Edition, Morgan Kaufmann Publishers, 2022
3. Jason Gregory, “Game Engine Architecture”, 4th Edition, A K Peters/CRC Press, 2023.

Reference Books

1. K. L. Murdock, Autodesk Maya 2024 Basics Guide. Boston: SDC Publications, 2023.
2. C. Briggs, An Essential Introduction to Maya Character Rigging, 2nd ed. Abingdon: Routledge, 2021.
3. “Understanding 3D Animation Using Maya” by John Edgar Park
4. “Animated Storytelling: Simple Steps For Creating Animation and Motion Graphics” by Liz Blazer
5. “Gpu Gems 2: Programming Techniques for High – Performance Graphics and General – Purpose Computation” by Matt Pharr and Randima Fernando (Series Editor)
6. “Creating 3-D Animation: The Aardman Book of Filmmaking” by Peter Lord and Brian Sibley
7. “3D Animation Essentials (Essentials (John Wiley))” by Andy Beane
8. “Exploring 3D Animation with Maya 6 (Design Exploration)” by Peter Young and Patricia Beckmann

9. “Blender 3D by Example” by Romain Caudron and Pierre-Armand Nicq

MOOC / NPTEL/YouTube Links

1. A N I M A T I O N s By Dr. Abhishek Kumar & Dr. Achintya Singhal https://onlinecourses.swayam2.ac.in/ learning/preview/cec26_ge03
2. https://youtu.be/4ZSWwHk4AOQ?si=N_6CFC5RmOh0T3DO
3. https://www.youtube.com/watch?v=6XBq8_iNlxY

Savitribai Phule Pune University **Third Year Artificial Intelligence and Data Science- 2024 Pattern**		
PEC322AID - Elective - I Laboratory		
Teaching Scheme	**Credits**	**Examination Scheme**
Practical : 02 Hours/Week	01	Term Work: 50 Marks

Guidelines for Instructor's Manual

The instructor's manual is to be developed as a reference and hands-on resource. It should include prologue (about University/program/ institute/ department/foreword/ preface), curriculum of the course, conduction and Assessment guidelines, topics under consideration, concept, objectives, outcomes, set of typical applications/assignments/ guidelines, and references.

Guidelines for Student's Laboratory Journal

The laboratory assignments are to be submitted by student in the form of journal. Journal consists of Certificate, table of contents, and handwritten write-up of each assignment (Title, Date of Completion, Objectives, Problem Statement, Software and Hardware requirements, Assessment grade/marks and assessor's sign, Theory- Concept in brief, algorithm, flowchart, test cases, Test Data Set(if applicable), mathematical model (if applicable), conclusion/analysis). Program codes with sample output of all performed assignments are to be submitted as softcopy. As a conscious effort and little contribution towards Green IT and environment awareness, attaching printed papers as part of write-ups and program listing to journal must be avoided. Use of DVD containing students programs maintained by Laboratory In-charge is highly encouraged. For reference one or two journals may be maintained with program prints in the Laboratory.

Guidelines for Laboratory /Term Work Assessment

Continuous assessment of laboratory work should be based on overall performance of Laboratory assignments by a student. Each Laboratory assignment assessment will assign grade/marks based on parameters, such as timely completion, performance, innovation, efficient codes, and punctuality.

Guidelines for Laboratory Conduction - Blockchain

List of Assignment - Group A

1. Design and implement a Python program to demonstrate the working of the SHA-256 hashing algorithm and analyze the avalanche effect.
2. Implement Elliptic Curve Cryptography for generating and verifying digital signatures
3. Implement and simulate a basic blockchain data structure using Python in order to understand the concept of block creation, SHA-256 hashing, and linking of blocks to form a secure and immutable blockchain.
4. Design and implement a blockchain-based system for tracking different versions of an Artificial Intelligence (AI) model in order to ensure secure, transparent, and tamper-resistant model updates.
5. Implement a blockchain system with a Proof of Work consensus mechanism to understand the process of mining and block validation.
6. Installation of MetaMask, create your own wallet using MetaMask for crypto transactions and study spending Ether per transaction
7. Write a smart contract on a test network for the Bank account of a customer for following operations.
   (a) Deposit Money

(b) Withdraw Money

(c) Show Balance

8. Write a program in solidity to create Student data. Use the following constructs:

   (a) Structures

   (b) Arrays

   (c) Fallback

   Deploy this as a smart contract on Ethereum and Observe the transaction fee and Gas value.

9. Develop a Blockchain based application dApp (de- centralized app) for e-voting system.

Mini Project - in the group of 3 to 4 students - Title: Blockchain-Based Decentralized Application

Concepts to be Implemented: Students must incorporate at least four of the following:

1. SHA-256 or cryptographic hashing for data integrity,
2. Smart contract development using Solidity deployed on Ethereum testnet,
3. Consensus mechanism simulation (PoW, PoS, or Snow/Avalanche),
4. Wallet creation and transaction management using MetaMask,
5. Decentralized data storage using IPFS
6. Integration of blockchain with an AI/ML model for secure data provenance or model versioning.
7. Sample Application Domains (not limited to): Decentralized voting system, Supply chain product tracking, Blockchain-based academic certificate verification, Healthcare record management, NFT-based digital asset marketplace.

Deliverables: Problem statement, system design, source code, deployed contract address on testnet, demonstration, and a brief report. Tools: Solidity, Remix IDE / Hardhat / Truffle, MetaMask, Python, Web3.js, IPFS (optional)

Savitribai Phule Pune University **Third Year Artificial Intelligence and Data Science- 2024 Pattern**		
PEC322AID - Elective - I Laboratory (Cloud Computing)		
Teaching Scheme	**Credits**	**Examination Scheme**
Practical : 02 Hours/Week	01	Term Work: 50 Marks

Guidelines for Laboratory Conduction - Cloud Computing

List of Assignment - Group A (Any THREE from 1 to 5, 6th and 7th is mandatory)

1. Study fundamental concepts of cloud, understand components of Cloud Architecture and Deployment models for Cloud.
2. Study the concept of Storage as a Service (SaaS) and implement cloud storage using a cloud platform.
3. Install Virtualbox/VMware Workstation with different flavors of linux or windows OS on top of windows 7 or 8.
4. Install a C compiler in the virtual machine created using virtual box and execute Simple Programs.
5. Implement procedure to transfer the files from one virtual machine to another virtual machine
6. Google App Engine (GAE) Setup: Install Google App Engine. Create hello world app or any other simple web applications using python/java.
7. Case Study on PaaS (Google App Engine). Use GAE launcher to launch the web applications.

List of Assignment - Group B (Any THREE from 1 to 5, 6th and 7th is mandatory)

1. Simulate a cloud scenario using CloudSim and run a scheduling algorithm that is not present in CloudSim.
2. Creating an application in Salesforce.com using Apex Programming Language
3. Study creating a warehouse application in Salesforce.com
4. Study installation and Configuration of Hadoop.
5. Install Hadoop single node cluster and run simple application like wordcount.
6. Study the Cloud Computing Security Issue
7. Case study on Amazon or Microsoft Azure Cloud

Group C (Mini-Project) - Any ONE (in the group of 3 to 4 students)

1. Setup your own Cloud for Software as a Service (SaaS) over the existing LAN in your laboratory. Write your own code for Cloud Controller using Open Source Technologies to implement with HDFS.
2. Implement the basic Operations such as divide the file in segments /blocks and download file from cloud in encrypted form. Host a portfolio or documentation site using Amazon S3 (AWS), Azure Blob Storage, or Google Cloud Storage. Integrate a Content Delivery Network (CDN) like AWS CloudFront to learn about global content distribution and HTTPS.

Savitribai Phule Pune University **Third Year Artificial Intelligence and Data Science- 2024 Pattern**		
PEC322AID - Elective - I Laboratory (Computer Vision)		
Teaching Scheme	**Credits**	**Examination Scheme**
Practical : 02 Hours/Week	01	Term Work: 50 Marks

Guidelines for Laboratory Conduction - Computer Vision

Prerequisite: Introduction to OpenCV and Image Basics

- Install Python, OpenCV, and required libraries
- Read and display an image
- Convert image to grayscale
- Save and resize image
- Understand image shape, pixel values

Outcome: Students learn basic image I/O and representation.

List of Assignment

1. Implement following Image Transformations : Resize, rotate, crop, and flip images, Translate and scale images, Convert between color spaces (BGR to RGB, BGR to Gray, BGR to HSV) - Outcome: Students understand geometric and color transformations.
2. Implement Image Filtering and Noise Removal: Add noise to an image, Apply mean, Gaussian, and median filters, Compare smoothing results - Outcome: Students learn preprocessing and noise reduction techniques.
3. Implement Image Enhancement Techniques: Contrast stretching, Histogram calculation and histogram equalization, Visual comparison of original vs enhanced image - Outcome: Students learn to improve image quality.
4. Implement Edge Detection and analyze each technique : Implement Sobel, Prewitt (using kernels), and Canny edge detection, Compare outputs for different thresholds, Outcome: Students learn boundary detection techniques
5. Implement Morphological Operations : Implement erosion, dilation, opening, and closing, Apply on binary and grayscale images, Observe effects on shapes and noise, Outcome: Students understand shape-based image processing.
6. Implement Image Segmentation: Simple thresholding (global and adaptive), K-means clustering for image segmentation, Display segmented results, Outcome: Students learn basic segmentation methods.
7. Implement Feature Detection and Matching, Detect keypoints using ORB / SIFT (if available), Compute descriptors, Match features between two images using BFMatcher, Visualize matches, Outcome: Students learn feature extraction and matching.
8. Implement Object Detection using Haar Cascade, Use pre-trained Haar cascade for face detection, Detect faces in images and webcam video, Draw bounding boxes around detected faces, Outcome: Students learn classical object detection methods.

Mini Project – Simple Computer Vision Application (Group Task - of 3 to 4 students)

Develop a real world application demonstrating the computer vision concepts. Below are the sample applications, students are free to choose appropriate computer vision application to implement.

- Real-time face detection system
- Motion detection using background subtraction
- Document scanner using edge detection and contour detection
- Image stitching (basic)

Outcome: Students integrate multiple concepts into a working application.

Guidelines for Laboratory Conduction - Gaming and Animation

Group A

1. Implement basic 2D geometric transformations (translation, rotation, scaling) using openGL.
2. Implement Cohen Sutherland Line clipping using openGL.
3. Create a short storyboard representing an animated sequence.
4. Design a 2D animation using keyframes and tweening techniques.
5. Model and rig a simple 3D object or character
6. Develop a short animation sequence demonstrating lighting, textures, and rendering.
7. Prepare a Game Design Document (GDD) for a small game concept.
8. Develop a simple 2D interactive game prototype using Unity or Godot.
9. Integrate animated 3D assets within a game scene and control them via scripts.

Group B - Mini Project (in the group of 3 to 4 students)

1. Design and develop a short interactive game/animated scene combining animation and gameplay logic.

Savitribai Phule Pune University **Third Year Artificial Intelligence and Data Science- 2024 Pattern**		
MDM331AID : Internet of Things		
Teaching Scheme	**Credits**	**Examination Scheme**
Tutorial: 02 Hour/Week **Practical**: 02 Hours/Week	02 01	Term Work : 25 Marks Oral: 25 Marks

Prerequisite Courses: Digital Electronics, Programming and Problem Solving, Embedded Systems

Course Objectives: The course aims to:

1. Explore the fundamentals, history, and evolution of Raspberry Pi, Arduino, and other microcontrollers.
2. Interface sensors (IR, temperature, gas) with Arduino/Raspberry Pi and develop real-time embedded applications.
3. Implement machine learning models for prediction, classification, and anomaly detection using IoT sensor data.
4. Design smart IoT-based systems using sensors.
5. Develop secure IoT systems for secured data transmission.

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1:** Elaborate the architecture and working principles of Raspberry Pi, Arduino, and microcontrollers.
- **CO2:** Build a real-time IoT applications using various sensors.
- **CO3:** Develop predictive models using machine learning techniques.
- **CO4:** Implement classification and anomaly detection models for IoT applications.
- **CO5:** Design secure IoT solutions ensuring authentication, privacy, and secure data communication.

Course Contents for Tutorials

PART A

Introduction to IoT and System Architecture (02 Hours)

Introduction to IoT: IoT Components, IoT Design, Role of cloud in IoT, Applications of IoT, Physical & Logical design of IoT, IoT setup using Arduino Uno or Raspberry Pi.

IoT Architecture: M2M & Simplified IoT Architecture, IoT protocol stack, Introduction to M2M.

IoT Communication, Networking and Protocols (03 Hours)

IoT Communication: various IoT Communication Models, Communication Patterns: D2D,D2C,D2G Back-end Data Sharing.

IoT Networks: TCP/IP Model ,6LoWPAN, IPv4 vs IPv6, Short-Range vs long range Communication Protocols: RFID, LoRa & LoRaWAN, Cellular IoT, Wi-Fi (802.11 variants), AI-enabled Networking.

IoT Data Acquisition, Cloud and Edge Platforms (03 Hours)

IoT Data Acquisition: Types of sensors and actuators, hardware and protocols -ADC, DAC, GPIO, UART, SPI, I2C, Real-time data streaming and buffering, Challenges in IoT data acquisition.

Cloud Platforms for IoT: IoT cloud platforms (AWS IoT Core, Microsoft Azure IoT Hub, Google Cloud), Cloud services for IoT. IoT Data Management and Compute Stack: Cloud IoT;

Edge Computing in IoT- Concept of edge computing and fog computing, Edge devices and gateways, optimization for latency, bandwidth optimization, enhanced privacy.

PART B

Intelligent IoT Applications (03 Hours)

AI in IoT- Artificial Intelligence and IoT integration (AIoT), Need for intelligent analytics in IoT systems, Various IoT Applications. Challenges: noisy, incomplete and heterogeneous data. Predictive Analytics using machine learning.

Security and Privacy in IoT (03 Hours)

IoT Security: IoT Threats, Vulnerabilities & Risk Model, DTLS, authentication & Authorization, Encryption Techniques.

Protocols: TLS/SSL , HTTP and HTTPS in IoT, MQTT, Publish/Subscribe model, CoAP, RESTful communication, AMQP Protocol ,XMPP.

Privacy in IoT Systems: Types of Data Collected by IoT Devices, Privacy Risks and Challenges in IoT, Privacy Preservation Techniques in IoT.

Guidelines for Instructor's Manual

1. The instructor 's manual is to be developed as a reference and hands-on resource.
2. It should include prologue (about University/program/ institute/ department/foreword/ preface), curriculum of the course, conduction and Assessment guidelines, topics under consideration, concept, objectives, outcomes, set of typical applications/assignments/ guidelines, and references.

Guidelines for Student's Lab Journal

The laboratory assignments are to be submitted by student in the form of journal. Journal consists of Certificate, table of contents, and handwritten write-up of each assignment (Title, Date of Completion, Objectives, Problem Statement, Software and Hardware requirements, Assessment grade/marks and assessor's sign, Theory- Concept in brief, algorithm, flowchart, test cases, Test Data Set(if applicable), mathematical model (if applicable), conclusion/analysis. Program codes with sample output of all performed assignments are to be submitted as softcopy. As a conscious effort and little contribution towards Green IT and environment awareness, attaching printed papers as part of write-ups and program listing to journal must be avoided. Use of DVD containing students programs maintained by Laboratory In-charge is highly encouraged. For reference one or two journals may be maintained with program prints in the Laboratory.

Guidelines for Lab /TW Assessment

Continuous assessment of laboratory work should be based on overall performance of Laboratory assignments by a student. Each Laboratory assignment assessment will assign grade/marks based on parameters, such as timely completion, performance, innovation, efficient codes, punctuality and performance on time submission, oral Q & A

Guidelines for Laboratory Conduction

List of laboratory assignments is provided below for reference. The instructor is expected to frame the assignments by understanding the prerequisites, technological aspects, utility and recent trends related to the topic. It is appreciated if the assignments are based on real world problems/applications. Encourage students for appropriate use of coding style, proper indentation and comments. Use of open-source software and recent versions is to be encouraged. In addition to this, instructors may

assign one real life application in the form of a mini-project. Based on the concepts learned. Instructors may also set one assignment or mini project.

Suggested List of Laboratory Experiments/Assignments - Part A (Any Five)

1. Develop an application for the connectivity of the Arduino UNO/Raspberry Pi circuit with sensors.
2. Implement an application to detect obstacles and notify users using LEDs to understand the connectivity of Raspberry-Pi / Arduino with IR sensor.
3. Implement a simple IoT application using an Arduino board to control an LED through GPIO pins in order to understand the physical design of IoT devices and device actuation.
4. Develop a program to detect the gas leakage in the surrounding environment
5. Configure & implement an IoT device (Arduino UNO IoT / ESP module / Raspberry Pi) to connect to a Wi-Fi network and transmit sensor data to a remote computer using TCP/IP communication.
6. Implement an edge computing model where sensor data is processed locally on an edge device (Raspberry Pi) before being transmitted to the cloud in order to reduce latency and bandwidth usage.
7. Design an RFID based identification system that reads tag information and displays the ID on a monitoring system to demonstrate short-range communication technologies in IoT.

Suggested List of Laboratory Experiments/Assignments - Part B ((Any Five)

1. Develop an IoT application that uses the communication protocol to transmit sensor data between an IoT device and a server.
2. Perform data cleaning, normalization, and visualization on IoT sensor dataset(any dataset can be considered)
3. Implement an IoT application to collect real-time temperature data using an IoT sensor and apply a machine learning model to predict future temperature values.(Any Machine Learning model)
4. To design and implement a password-protected smart sensor system that allows access to sensor data only after successful authentication, demonstrating basic IoT security concepts. (Any security Algorithm can be used
5. To implement secure data transmission in an IoT system using a sensor and any communication security protocol
6. Develop & Implement a machine learning model to perform classification (e.g., activity recognition or device state detection) and anomaly detection (e.g., fault detection in industrial IoT sensors).
7. Case study for Data Minimization Using Smart Threshold Alert by providing Privacy-by-Design

Part C - Mini project

Develop a Privacy-Aware Intelligent Smart IoT application. : Design and Implementation of an AIoT-Based Smart Sensor System with Cloud-Based Data Preprocessing and Intelligent Decision-Making using Machine Learning techniques, Incorporating Secure Communication Protocols with Built-in Security and Privacy Mechanisms, with Complete Deliverables including Working Demonstration and Technical Report.

Learning Resources

Text Books:

1. Ovidiu Vermesan, Peter Friess, “Internet of Things: Converging Technologies for Smart Environments and Integrated Ecosystems”, River Publishers, ISBN: 978-87-92982-73-5 (Print)

2. Volker Ziemann, “A Hands-On Course in Sensors Using the Arduino and Raspberry Pi”, 2018, 1st Edition, CRC Press, United States. ISBN: 978-1-032-37748-3 (hbk)ISBN: 978-1-032-37619-6 (pbk).

3. Nathan Ida, “Sensors, Actuators and their Interfaces - A Multidisciplinary Introduction”, 2020, 2 nd Edition, IET, United Kingdom. ISBN: 978-1-78561-835-2, e-ISBN: 978-1-78561-836-9

Reference Books

1. Internet of Things: A Hands-on Approach by Arshdeep Bahga and Vijay Madisetti, ISBN 978 81 7371 954 7

2. Olivier Hersent, David Boswarthick, Omar Elloumi, “The Internet of Things – Key Applications and Protocols”, ISBN: 978-1-119-99435-0, January 2012, 376 pages Wiley.

MOOC / NPTEL/YouTube Links

1. Arduino Official Documentation (Reference for microcontroller interfacing and code) https://www.ardui
2. All About Circuits - Textbooks (Fundamentals of electronics, circuits, op-amps) https://www.allaboutcirc
3. SparkFun/Adafruit Sensor & Actuator Tutorials (Practical interfacing guides for common components) https://learn.sparkfun.com/tutorials
4. https://www.youtube.com/watch?v=KsL_uwa0ekY
5. https://onlinecourses.nptel.ac.in/noc22_cs53/preview
6. https://onlinecourses.swayam2.ac.in/ntr25_ed21/preview

E-Books

1. https://pg.its.edu.in/sites/default/files/KCA043%20Internet%20of%20things%20-IoT%20by%20Raj%
2. https://jcer.in/jcer-docs/E-Learning/Digital%20Library%20/E-Books/Internet-of-things-a-hands-on-approach-%20Arshadeep.pdf

Savitribai Phule Pune University		
Open Electives		
Teaching Scheme	**Credits**	**Examination Scheme**
Theory : 02 Hours/Week	02	CCE : 15 Marks End-Semester: 35 Marks

Open Electives (OE) are multidisciplinary courses allowing students to study subjects outside their core discipline to foster holistic development and skill enhancement. Students pick subjects outside their core specialization from the following list to broaden their knowledge base.

Sr.	Open Elective Course Name	Offering Discipline
1	IPR and Cyber Laws	Law / Faculty of Humanities
2	Agri Business Management: Banking Operation and finance	Commerce & Management
3	Product Costing for Mechanical Engineering	Commerce & Management
4	Sustainability Development	Commerce & Management
5	Material and Logistics	Management
6	The Constitution of India	Law / Faculty of Humanities
7	Digital Personal Data Protection	Law / Faculty of Humanities
8	Environmental Law	Law / Faculty of Humanities
9	Construction Law and Contracts	Law / Faculty of Humanities
10	Human Resource Management	Management
11	Statistics and Computer Applications	Commerce & Management
12	Business Administration	Commerce & Management
13	Business Marketing	Commerce & Management
14	Entrepreneurship Development	Commerce & Management
15	Banking ,Finance & Insurance	Commerce & Management
16	Cost & Works Accounting	Commerce & Management

Savitribai Phule Pune University **Third Year - Computer Science & Design Engineering (2024 Pattern)**		
ELC342CSD: Technical Seminar		
Teaching Scheme	**Credits**	**Examination Scheme**
Practical : 02 Hours/Week	01	Oral/Presentation : 25 Marks

Course Objectives: The course aims to:

1. To develop research orientation and technical communication skills in emerging Computer Engineering and Artificial Intelligence domains.
2. To enable students to critically review, analyze, and synthesize contemporary research papers, white papers, patents, and technical standards.
3. To promote interdisciplinary thinking aligned with NEP-2020 multidisciplinary philosophy.
4. To inculcate ethical awareness, sustainability perspective, and societal impact analysis of AI systems.
5. To prepare students for industry, higher education, entrepreneurship, and innovation ecosystems.

Course Outcomes: Upon successful completion of this course, students will be able to:

- CO1: Identify and select emerging and relevant technical topics through literature survey.
- CO2: Analyze and synthesize information from research papers, journals, and credible sources.
- CO3: Demonstrate effective technical communication skills through oral presentation.
- CO4: Prepare a structured technical report following academic writing standards.
- CO5: Use modern tools (presentation software, plagiarism checkers, referencing tools) for seminar preparation.

Guidelines for Conduct of Technical Seminar

The Technical Seminar shall be research-oriented and domain-specific, focusing strictly on recent development in Computer Engineering.

Topic Selection Guidelines

- Topic must be from emerging Computer Engineering and Artificial Intelligence domains (last 3–5 years).
- Must involve a minimum of 5 recent research papers (IEEE, ACM, Elsevier, Springer etc). They should summarizing paper – Reading abstracts and finding ideas, conclusion, Advantages of Their approach, and the drawbacks of the papers. Generalize results from a research paper to related research problems. Comparing the approach - Identify weaknesses and strengths in recent research articles in the subject. Practical sessions on how to read, analyze and summarize research papers.
- Should not be a basic textbook topic.
- Must include: Problem statement, State-of-the-art analysis, Comparative study, Ethical & societal impact, Interdisciplinary themes aligned with NEP encouraged.
- Topic approval by a faculty panel

Seminar Process

- **Stage 1: Orientation & Topic Finalization (Week 1–2)**
  - Conduct an orientation session explaining: Objectives of the technical seminar, Evaluation criteria and expected outcomes
  - Each student must submit: Title of the seminar, Problem statement, Relevance to current technology trends, Approval by guide is mandatory before proceeding
- **Literature Survey & Problem Understanding (Week 2–4)**
  - Students must: Refer minimum 5–8 recent research papers (IEEE, Springer, Elsevier, ACM etc.)
  - Use scholarly databases like IEEE Xplore, Google Scholar, ScienceDirect
  - Prepare a literature survey matrix, including:
    * Author/year
    * Methodology used
    * Key findings
    * Limitations
  - Identify: Research gaps and Challenges in existing approaches
- **Synopsis Preparation & Presentation (Week 4–5)**
  - Submit a 2–3 page synopsis including: Introduction, Literature insights, Objectives, Proposed seminar scope
  - Conduct a Synopsis Presentation (5–7 minutes): Evaluate clarity of understanding, Receive feedback for improvement
  - Approval required before proceeding to full report
- **In-depth Study & Content Development (Week 5–8)**
  - Students should: Deeply analyze concepts, models, architectures, or case studies, Include diagrams, flowcharts, and comparative tables
  - Weekly review meetings with guide: Track progress, Ensure conceptual clarity,
  - Emphasis on: Critical analysis (not just description), Real-world applications
- **Draft Report Submission & Review (Week 8–10)**
  - Submit first draft of the report
  - Guide provides feedback on: Technical content quality, Structure and coherence, Referencing and plagiarism,
  - Students must revise based on suggestions by the guide
- **Pre-Seminar Presentation (Mock Evaluation) (Week 10–11)**
  - Conduct a mock presentation simulating final evaluation
  - Focus on: Presentation skills, Time management, Handling questions
  - Peer and faculty feedback should be incorporated
- **Final Report Submission (Week 11–12)**
  - Submit: Final hard copy (if required), Soft copy (PDF format)
  - Ensure:Proper formatting, Plagiarism compliance (<20%), Correct referencing using reference managers like Zotero and Mendeley Desktop

- **Final Seminar Presentation & Viva Voce (Week 12–13)**
  - Presentation duration: 10–15 minutes, Followed by Q&A session (5–10 minutes)
  - Evaluation based on: Depth of understanding, Analytical ability, Communication skills

Method of Evaluation

- During the seminar session each student is expected to prepare and present a topic on engineering/ technology, for duration of about 12to 15 minutes.
- Each student is expected to present atleast twice during the semester and the student is evaluated based on that.
- At the end of the semester, he / she can submit a report on his / her topic of seminar and marks are given based on the report.
- A Faculty guide is to be allotted and he / she will guide and monitor the progress of the student and maintain attendance.

Savitribai Phule Pune University, Pune

Maharashtra, India

TE - Artificial Intelligence and Data Science - 2024 Pattern

Semester VI

Savitribai Phule Pune University **Third Year Artificial Intelligence and Data Science- 2024 Pattern**		
PCC351AID- Software Engineering		
Teaching Scheme	**Credits**	**Examination Scheme**
Theory : 03 Hours/Week	03	CCE : 30 Marks End-Semester: 70 Marks

Prerequisite Courses: Programming Fundamentals, Data Structures, Object-Oriented Programming Concepts

Companion Description: This course provides a comprehensive understanding of software engineering principles and software modelling techniques required for developing reliable, scalable, and maintainable software systems. The course is structured into five units: two units focusing on core software engineering concepts, two units dedicated to software modelling using UML, and one unit covering latest and advanced trends in software engineering

Course Objectives: The course aims to:

1. To understand fundamental principles and challenges of software engineering.
2. To study software process models and requirement engineering techniques.
3. To apply software modelling concepts using UML diagrams.
4. To understand project management, testing concepts.
5. To introduce students to modern and advanced software engineering practices.

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1:** Explain software engineering principles and software development life cycle models.
- **CO2:** Analyze and document software requirements using standard practices.
- **CO3:** Design software systems using UML-based software modelling techniques.
- **CO4:** Apply project management, testing, and quality assurance techniques.
- **CO5:** Understand and evaluate modern and advanced trends in software engineering.

Course Contents

Unit I - Introduction to Software Engineering (09 Hours)

Software Engineering: Definition, Importance, Software Crisis, Software Myths & Reality, The Software Process: Software Process Framework, SDLC, Prescriptive Process Models: Waterfall, V-Model, Incremental, Evolutionary Process Models: Prototyping, Spiral, Rapid Application Development (RAD), Agile Development Models: What Is Agility? Agility and the Cost of Change, Extreme Programming, Other Agile Process Models, Overview of Scrum and Kanban.

Case Study: Development of an Online Examination System to demonstrate various Software Life Cycle Models and Software Process Structures.

Unit II - Requirements Engineering (09 Hours)

Requirements Engineering, Establishing the Groundwork: Identifying Stakeholders, Recognizing Multiple Viewpoints, working toward Collaboration, Asking the First Questions, Non-functional Requirements, Traceability Eliciting Requirements: Collaborative Requirements Gathering, Usage Scenarios, Elicitation Work Products, Developing Use Cases, Building the Analysis Model, Elements of the Analysis Model, Negotiating Requirements: Requirement Validation and Management. Software Requirements Specification (SRS)

Case Study: Requirements Engineering and SRS development for a College ERP System covering stakeholder identification, requirement elicitation, use case modelling, non-functional requirements, validation and traceability.

Unit III Software Modelling (09 Hours)

Introduction to Software Architecture, Architectural design principles and patterns, Types of architectures: Layered, Client-Server, Micro services, MVC, Architectural styles and quality attributes, Architecture

documentation (4+1 view model), Object Modelling Using UML Basic Object-Orientation Concepts, Unified Modelling Language (UML), UML Diagrams, Use Case Model, Class Diagrams, Interaction Diagrams, State Chart Diagram, Activity Diagrams, Package Diagrams, Component Diagrams, Deployment Diagram.

Case Study: Design and UML modelling of a Library Management System to demonstrate abstraction, modularity, object-oriented design principles, and complete UML modelling including structural and behavioural diagrams.

Unit IV - Software Project Management and Testing (09 Hours)

Software Project Management Complexities, Project Planning, Metrics for Project Size Estimation, Project Estimation Techniques, Empirical Estimation Techniques, COCOMO—A Heuristic Estimation Technique, Scheduling, Organisation and Team Structures, Staffing, Risk Management, Software Configuration Management.

Software Testing, A Strategic Approach to Software Testing, Test Strategies for Conventional Software, Test Strategies for Object-Oriented Software, Test Strategies for WebApps, Validation Testing, System Testing, Debugging, Defect Life Cycle, Manual and Automated testing

Case Study: Software Project Planning, Estimation, Scheduling, Risk and Configuration Management for an E-Governance Citizen Service Portal using COCOMO and empirical estimation techniques.

Unit V - Advanced Concepts in Software Engineering (09 Hours)

Agile and DevOps practices, CI/CD pipelines, DevSecOps and Secure Software Development, Software Architecture Patterns (Event-Driven, Serverless, Reactive Systems), AI in Software Engineering (AI- assisted development, automated testing), Low-Code / No-Code Platforms, Software Sustainability and Green Computing.

Case Study: Design and implementation of a Smart City Service Portal using Agile practices, DevOps, CI/CD pipelines, Microservices architecture, Cloud-native deployment, and reusable software components.

Learning Resources

Text Books:

1. Roger S. Pressman, Software Engineering: A Practitioner's Approach, Mcgraw-Hill.
2. Ian Sommerville, Software Engineering, Pearson Education.

Reference Books

1. Rajib Mall, Fundamentals Of Software Engineering, Fifth Edition, ISBN-978-93-88028-02-8
2. Pankaj Jalote, An Integrated Approach to Software Engineering, Springer.
3. Booch, Rumbaugh, Jacobson, The UML User Guide, Addison-Wesley.

Savitribai Phule Pune University **Third Year Artificial Intelligence and Data Science- 2024 Pattern**		
PEC352AID: Deep Learning		
Teaching Scheme	**Credits**	**Examination Scheme**
Theory : 03 Hours/Week	03	CCE : 30 Marks End-Semester: 70 Marks

Prerequisite Courses: Linear Algebra, Probability and Statistics, Machine Learning
Course Objectives: The course aims to:

1. Introduce the fundamental concepts of artificial neural networks
2. Familiarize students with the fundamentals of deep learning.
3. Provide knowledge of Convolutional Neural Networks architectures.
4. Acquaint with the need for sequence models in processing sequential data.
5. Enable students to gain knowledge of the fundamentals of deep generative models.

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1:** Apply neural network concepts for solving real-world problems.
- **CO2:** Make use of appropriate activation and loss functions in model design.
- **CO3:** Utilize Convolutional Neural Networks techniques for solving real-world problems.
- **CO4:** Analyze the need for sequence models in handling sequential data problems.
- **CO5:** Distinguish between different deep generative models such as GANs and autoencoders.

Course Contents

Unit I - Fundamentals of Artificial Neural Networks (09 Hours)

Introduction to Neural Network, Biological neuron vs artificial neuron. Artificial Neuron Model: Structure, mathematical representation. Introduction to activation functions. Neural Network Architectures: Single-layer feedforward networks, Multi-layer feedforward networks, network representation. Learning in Artificial Neural Network: Supervised learning, Unsupervised learning, training vs testing, loss (error) function. Perceptron Model: Single-layer perceptron, perceptron learning rule, limitations. Multilayer Perceptron: Need for hidden layers. Overview of forward propagation and backpropagation, advantages over perceptron, applications of Artificial Neural Neywork.

Case Study: Handwritten Digit Recognition using Artificial Neural Network

Unit II - Deep Learning and Training Mechanisms (09 Hours)

Introduction, difference between Machine Learning and Deep Learning. Deep Neural Networks (DNN): Architecture and representation. Activation functions: ReLU, Sigmoid, Tanh, Softmax. Forward propagation. Loss functions: MSE, Cross-Entropy. Backpropagation. Gradient Descent: Batch, Stochastic, Mini-batch. Optimization Algorithms: SGD, AdaGrad, RMSProp, Adam, applications of deep learning. Challenges in training deep networks: Vanishing and exploding gradients, weight initialization techniques, Overfitting and underfitting. Regularization techniques: L1/L2 regularization, Dropout.

Case Study: Predict house prices based on features like area, location, and number of rooms using a Deep Neural Network.

Unit III - Convolutional Neural Networks (09 Hours)

Introduction, convolution operation and feature maps, kernel and convolution, filter size calculations, padding and stride, pooling layers, CNN architecture design. CNN architectures: LeNet, AlexNet, VGG, ResNet. Transfer learning and fine-tuning. Object detection technique in deep learning: YOLO (You Only Look Once). Applications.

Case Study: Traffic sign recognition system using CNN-based transfer learning for automated detection and classification of road signs.

Unit IV - Recurrent Neural Networks (RNN) (09 Hours)

Need for sequence models, Recurrent Neural Networks (RNN) architecture, Bidirectional RNNs, Deep Recurrent Networks, Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), introduction to Transformers. Text embeddings: Word2Vec, GloVe. Applications of sequence models: Natural language processing, speech recognition, time series forecasting, video analysis, and healthcare signal processing.

Case Study: Build a sequence model using GRU/LSTM to predict future stock prices or weather conditions based on historical time-series data.

Unit V - Deep Generative Models (09 Hours)

Introduction, Generative Adversarial Networks. Autoencoders: Traditional autoencoders, variational autoencoders. Model compression and pruning, model deployment using REST APIs, TensorFlow Lite, Edge AI, ethical considerations in deep learning, and energy efficiency and sustainability in deep learning models.

Case Study: Develop a Generative Adversarial Network (GAN) based system for classifying images as real or fake, focusing on detecting synthetic or manipulated images.

Learning Resources

Text Books:

1. Goodfellow, Ian, Bengio, Yoshua, and Courville, Aaron, "Deep Learning," 1st Edition, MIT Press, 2016, ISBN 978-0262035613.
2. Chollet, François, "Deep Learning with Python," 2nd Edition, Manning Publications, 2021, ISBN 978-1617296864.
3. Géron, Aurélien, "Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow," 3rd Edition, O'Reilly Media, 2022, ISBN 978-109812597.

Reference Books

1. Aggarwal, Charu C., "Neural Networks and Deep Learning: A Textbook," 1st Edition, Springer, 2006, ISBN 978-3319944623.
2. Bishop, Christopher M., "Pattern Recognition and Machine Learning," 1st Edition, Springer, 2006, ISBN 978-0387310732.
3. Zhang, Aston, Lipton, Zachary C., Li, Mu, and Smola, Alexander J., "Dive into Deep Learning," 1st Edition, Cambridge University Press, 2023, ISBN 978-1009387422.

MOOC / NPTEL/YouTube Links

1. NPTEL course: "Deep Learning" by Prof. Mitesh M. Khapra, IIT Madras / IIT Ropar. https://onlinecourse
2. NPTEL course: "Deep Learning" by Prof. P. K. Biswas, IIT Kharagpur. https://onlinecourses.nptel.ac.in/n
3. Dive into Deep Learning (Interactive): Link: https://d2l.ai

E-Books

1. Dive into Deep Learning: https://arxiv.org/pdf/2106.11342 https://d2l.ai/?utm_source=chatgpt.com
2. Deep Learning: An Introduction for Applied Mathematicians, Catherine F. Higham, Desmond J. Higham, 2018. https://arxiv.org/pdf/1801.05894

Savitribai Phule Pune University **Third Year Artificial Intelligence and Data Science- 2024 Pattern**		
PCC353AID - Deep Learning Laboratory		
Teaching Scheme	**Credits**	**Examination Scheme**
Practical : 02 Hours/Week	01	Term Work: 25 Marks Practical: 25 Marks

Prerequisite Courses : Artificial intelligence, Natural Language Processing, Machine Learning

Companion Course: Deep Learning

Course Objectives:

1. To develop the ability to set up a Python-based deep learning environment and work with a single-layer perceptron.
2. To enable students to implement an MLP using backpropagation and understand activation and loss functions.
3. To provide knowledge of optimization algorithms and regularization techniques for improving neural network performance.
4. To familiarize students with CNN design and transfer learning using pretrained models.
5. To familiarize students with object detection and generative models using YOLO, autoencoders, and GANs.

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1:** Demonstrate tensor operations, matrix manipulations, and the functioning of a single-layer perceptron for binary classification on linearly separable data.
- **CO2:** Apply an MLP on a standard dataset to observe the effect of activation and loss functions on model performance.
- **CO3:** Make use of gradient descent variants and regularization techniques to improve model convergence and performance.
- **CO4:** Analyze CNN architectures and transfer learning approaches by comparing pretrained models with custom CNNs for image data.
- **CO5:** Examine object detection and generative models by assessing detection performance and output quality.

Guidelines for Instructor's Manual

The instructor‘s manual is to be developed as a reference and hands-on resource. It should include prologue (about University/program/ institute/ department/foreword/ preface), curriculum of the course, conduction and Assessment guidelines, topics under consideration, concept, objectives, outcomes, set of typical applications/assignments/ guidelines, and references.

Guidelines for Student's Laboratory Journal

The laboratory assignments are to be submitted by student in the form of journal. Journal consists of Certificate, table of contents, and handwritten write-up of each assignment (Title, Date of Completion, Objectives, Problem Statement, Software and Hardware requirements, Assessment grade/marks and assessor's sign, Theory- Concept in brief, algorithm, flowchart, test cases, Test Data Set(if applicable), mathematical model (if applicable), conclusion/analysis). Program codes with sample output of all performed assignments are to be submitted as softcopy. As a conscious effort and little contribution towards Green IT and environment awareness, attaching printed papers as part of write-ups and

program listing to journal must be avoided. Use of DVD containing students programs maintained by Laboratory In-charge is highly encouraged. For reference one or two journals may be maintained with program prints in the Laboratory.

Guidelines for Laboratory /Term Work Assessment

Continuous assessment of laboratory work should be based on overall performance of Laboratory assignments by a student. Each Laboratory assignment assessment will assign grade/marks based on parameters, such as timely completion, performance, innovation, efficient codes, and punctuality.

Guidelines for Practical Examination

Problem statements must be decided jointly by the internal examiner and external examiner. During practical assessment, maximum weightage should be given to satisfactory implementation of the problem statement. Relevant questions may be asked at the time of evaluation to test the student's understanding of the fundamentals, effective and efficient implementation. This will encourage, transparent evaluation and fair approach, and hence will not create any uncertainty or doubt in the minds of the students. So, adhering to these principles will consummate our team efforts to the promising start of student's academics.

Guidelines for Laboratory Conduction

A list of laboratory assignments is provided below for reference. The instructor is expected to frame the assignments by understanding the prerequisites, technological aspects, utility and recent trends related to the topic. The assignment-framing policy should address the average student and include an element to attract and promote intelligent students. The instructor may set multiple assignment sets and distribute them across student batches. It is appreciated if the assignments are based on real-world problems/applications. Encourage students to use coding style, proper indentation and comments appropriately.

Use of open-source software and recent versions is to be encouraged. Python is preferably used.

In addition, instructors may assign a real-world application as a mini-project. based on the concepts learned.

Suggested List of Laboratory Experiments/Assignments

1. Environment Setup and Tensor Operations: Demonstrate the setup of a Python-based deep learning environment using TensorFlow or PyTorch, and perform basic tensor operations, matrix manipulations, and data visualization
2. Implementation of Perceptron: Demonstrate a single-layer perceptron for binary classification on linearly separable data and measure its performance.
3. Multilayer Perceptron with Backpropagation: Build a Multi-Layer Perceptron (MLP) using backpropagation for classification on a standard dataset (e.g., Iris/MNIST).
4. Activation Functions and Loss Analysis: Identify the impact of different activation functions (ReLU, Sigmoid, Tanh) and loss functions on model convergence and accuracy.
5. Optimization Algorithms Comparison: Develop a neural network using different Gradient Descent variants (SGD, Mini-batch, Adam, RMSProp) and observe convergence behavior.
6. Regularization Techniques: Apply Dropout, Batch Normalization, and Early Stopping to control overfitting and measure model performance.
7. Convolutional Neural Network (CNN) Design and Implementation: Design a Convolutional Neural Network for image data (e.g., MNIST/CIFAR-10) and visualize feature maps.
8. Transfer Learning using Pretrained CNN: Examine a pretrained model (e.g., ResNet/VGG) for image classification and compare its performance with a custom CNN.

9. Object Detection using YOLO: Analyze object detection using a pretrained YOLO model on images/videos and assess detection performance.
10. Autoencoder / GAN Implementation: Design an autoencoder or GAN for image reconstruction or generation and assess output quality.

Mini Project: Students may select any one topic for the mini-project from the following list (not limited to these).

- Disaster Detection using Satellite Images
- Sentiment Analysis for Social Awareness
- Crop Disease Detection System
- Air Quality Prediction using Deep Learning

Learning Resources

Virtual Laboratory (links):

1. https://vlab.spit.ac.in/ai/
2. https://scte-iitkgp.vlabs.ac.in/

MOOC Courses (Web Links):

1. NPTEL course: "Deep Learning" by Prof. Mitesh M. Khapra, IIT Madras / IIT Ropar. https://onlinecourse
2. NPTEL course: "Deep Learning" by Prof. P. K. Biswas, IIT Kharagpur. https://onlinecourses.nptel.ac.in/n

Savitribai Phule Pune University **Third Year Artificial Intelligence and Data Science- 2024 Pattern**		
PCC354COM- Software Engineering Laboratory		
Teaching Scheme	**Credits**	**Examination Scheme**
Practical : 02 Hours/Week	01	Term Work: 25 Marks Oral : 25 Marks

Prerequisite Courses: Object Oriented Programming, Data Structures & Database Management Systems

Companion Course : Software Engineering and Modeling

Course Objectives:

1. To apply software engineering principles to real-world problem statements.
2. To develop Software Requirement Specification (SRS) documents.
3. To design and model software systems using UML tools.
4. To implement testing strategies and quality assurance techniques.
5. To explore Agile practices, version control, and DevOps fundamentals.

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1:** Perform feasibility analysis and project estimation for software systems.
- **CO2:** Develop SRS document using standard IEEE format.
- **CO3:** Design UML diagrams using modeling tools.
- **CO4:** Prepare and execute software test cases and quality plans.
- **CO5:** Apply Agile methodology and version control tools in software development.
- **CO6:** Develop a mini-project using complete SDLC approach.

Guidelines for Instructor's Manual

The instructor‘s manual is to be developed as a reference and hands-on resource. It should include prologue (about University/program/ institute/ department/foreword/ preface), curriculum of the course, conduction and Assessment guidelines, topics under consideration, concept, objectives, outcomes, set of typical applications/assignments/ guidelines, and references.

Guidelines for Student's Laboratory Journal

The laboratory assignments are to be submitted by student in the form of journal. Journal consists of Certificate, table of contents, and handwritten write-up of each assignment (Title, Date of Completion, Objectives, Problem Statement, Software and Hardware requirements, Assessment grade/marks and assessor's sign, Theory- Concept in brief, algorithm, flowchart, test cases, Test Data Set(if applicable), mathematical model (if applicable), conclusion/analysis). Program codes with sample output of all performed assignments are to be submitted as softcopy. As a conscious effort and little contribution towards Green IT and environment awareness, attaching printed papers as part of write-ups and program listing to journal must be avoided. Use of DVD containing students programs maintained by Laboratory In-charge is highly encouraged. For reference one or two journals may be maintained with program prints in the Laboratory.

Guidelines for Laboratory /Term Work Assessment

Term work evaluation shall be based on:

- Timely Completion – 5 Marks
- Quality of Documentation – 5 Marks
- Tool Usage & Implementation – 5 Marks
- Innovation & Problem Solving – 5 Marks
- Mini Project Performance – 5 Marks

Guidelines for Practical Examination

Problem statement jointly decided by internal and external examiners. Evaluation criteria:

- Implementation – 40%
- Understanding – 30%
- Documentation – 20%
- Viva – 10%

Guidelines for Laboratory Conduction

- Emphasis on experiential learning and skill development. Problem-based and case-based learning approach.
- Collaborative mini-project in teams of 3–4 students.
- Use of industry-relevant tools (StarUML, Visual Paradigm, GitHub). Continuous evaluation based on innovation and implementation.

List of Assignments (Part A) - Any FIVE , 6th and 7th Mandatory

1. Study and compare Waterfall Model, V-Model, Spiral Model, and Incremental Model. Select suitable life cycle model. Identify deliverables and justify model selection.
2. Identify the stakeholders involved in the selected software system, Prepare questionnaire and Develop usage scenarios
3. Study the concepts and practices of Scrum, Kanban, and Extreme Programming used in Agile software development. Analyze how these methodologies help in managing project tasks, improving collaboration, and delivering software incrementally. Based on a selected software system, write sample user stories to capture system requirements. Further, create a sample sprint plan and track tasks using the project management tool Jira to understand Agile workflow management.
4. Prepare IEEE-format SRS document. Include Functional and Non-functional Requirements.
5. Requirements Traceability & Validation. Prepare RTM and perform requirement validation.
6. Risk identification and mitigation plan preparation.
7. Estimate using LOC/FP and apply COCOMO. Prepare Gantt chart.
8. Agile Project Planning and Tracking using Jira Study Scrum, Kanban and Extreme Programming. Prepare sample sprint in jira. To understand and implement Agile project management practices using Jira tool.

List of Assignments (Part B) Any 6 from Part-B (any 3 from 1 to 5, and 6,7,8)

1. Study and understand UML modeling and draw Use case diagram. Use the Unified Modeling Language (UML) to model selected system such as a Library Management System or Online Shopping System. Analyze the system requirements and represent them using Use case diagram.Also Draw Data Flow Diagram (DFD), Entity Relationship Diagram (ER Diagram).
2. UML Structural Modelling Draw Class, Package and Component Diagrams.
3. UML Behavioral Modelling Draw Sequence, Activity and State Diagrams.
4. Architectural Design Design Layered, Client-Server and Microservices architecture using 4+1 view model.
5. Software Testing – Test Case Design Write test cases to perform Black Box and White Box testing.
6. Manual and Automated Testing Perform manual testing and automated testing using tools (Selenium) Implement a simple Selenium WebDriver code in Java/Python to open Google Chrome, navigate to Google Search, and enter text in the search box.
7. Study DevOps and CI/CD Implementation Study the architecture and core concepts of DevOps and understand how CI/CD improve software delivery. Implement a simple CI/CD pipeline by creating a project repository using Git. Configure the pipeline to automatically build and test the application whenever code changes are pushed to the repository, demonstrating the basic workflow of automated software integration and deployment.
8. AI in Software Engineering / Low-Code Platform Use AI tool for code/testing OR develop app using low-code platform.

List of Assignments (Part C) - Mini-Project (In Team of 3-4 Students)

Problem Identification, Feasibility Study, SRS Preparation, UML Modeling, Design Implementation (optional prototype), Testing Plan, Agile Sprint Documentation, Final Report and Demonstration. Suggested from the list, but not limited to

1. Library Management System
2. Online Food Ordering System
3. Hospital Management System
4. Learning Management System

Savitribai Phule Pune University **Third Year Artificial Intelligence and Data Science- 2024 Pattern**		
PEC361AAID: Cyber Security and Data Privacy		
Teaching Scheme	**Credits**	**Examination Scheme**
Theory : 02 Hours/Week	02	CCE : 30 Marks End-Semester: 70 Marks

Prerequisite Courses: Operating Systems, and basic programming (C or Python)
Course Objectives: The course aims to:

1. Introduce security fundamentals, threats, basic risks, and ethical practices.
2. Apply cryptographic concepts for confidentiality, integrity, and authentication
3. Examine security controls for systems, applications in real-world scenarios.
4. Equip students with the technical knowledge and skills needed to protect and defend against cyber threats.
5. Develop skills in incident response lifecycle, and compliance with CERT-In and DPDP regulatory frameworks

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1 :** Identify common threats and select suitable security controls for a given scenario.
- **CO2 :** Explain encryption, hashing, digital signatures, and PKI for protecting data.
- **CO3 :** Analyze common web and API vulnerabilities
- **CO4 :** Apply secure SDLC and vulnerability management.
- **CO5 :** Describe data protection techniques, and interpret the regulatory framework under the DPDP

Course Contents

Unit I - Introduction to Cyber Security & Data Privacy (06 Hours)

Cybersecurity – definition, significance, objectives; Cybersecurity vs. Information Security; Cybersecurity in AI and Data Science; CIA Triad; AAA Model; Defense-in-Depth; Least Privilege; Threats; Vulnerabilities; Risk Assessment; Control Measures; Data Privacy – definition, significance; Personally Identifiable Information (PII); Cyber Ethics; Responsible Disclosure; Data Lifecycle Management; Privacy-by-Design. **Case Study:** Data Exposure in a Student Mobile Application

Unit II Data Encryption Techniques (06 Hours)

Cryptography – applications and need in cybersecurity; Cryptographic algorithms and classification; Data in transit, at rest, and in use; Symmetric and asymmetric encryption – AES, RSA, DES; Hashing techniques – bcrypt, SHA-256, integrity mechanisms; Secure password hashing – salting, slow hashes (Argon2id); Digital signatures – principles, applications, benefits; PKI, digital certificates,

Case Study: Analyzing Password Breach (like the 2012 LinkedIn)

Unit III Secure Engineering and Vulnerability Management (06 Hours)

Secure SDLC: requirements, design review, Secure configuration & hardening: baseline builds, CIS benchmark, least functionality Patch & vulnerability management: identification, triage, remediation, verification, CVSS Secrets management: API keys, credentials, environment variables, rotation

Software supply chain: dependency risks, SBOM-Software Bill of Materials, signing, update hygiene Security testing: SAST, DAST, dependency scanning, misconfiguration checks

Case Study: Secure SDLC review for a student portal

Unit IV - Security and Threats Handling (06 Hours)

API Security : Authentication Mechanisms and Patterns, Rate Limiting, Logging, Monitoring ,Classification of Cyber Threats and Attack Methodologies, Malware Typologies: Viruses, Worms, Trojans, Ransomware, Spyware Rootkits ,Social Engineering and Phishing Attacks: Phishing, Spear Phishing, Vishing,

AI-Powered Social Engineering Attacks: deepfakes (voice/video), Cloud Security Threats: misconfigurations, insecure APIs, identity and access management.

Case Study: Web Portal Defacement through Input Injection

Unit V - Incident Response and regulatory frameworks (DPDP) (06 Hours)

Incident Response Lifecycle: preparation, detection, containment, eradication, recovery, Incident Response Playbooks Development, AI-generated phishing Incident Response for AI Threats: detection, analysis, mitigation, Security Testing Strategies: identification, prioritization, remediation of vulnerabilities CERT-In Reporting & DPDP Breach Reporting: overview, compliance requirements; DPDP Act 2023 ;DPDP Rules 2025: key provisions, incident response considerations ,Incident Response under Indian Regulatory Frameworks

Case Study: Ransomware incident response plan with evidence collection and recovery steps.

Learning Resources

Text Books:

1. Nina Godbole and Sunit Belapure, Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives, Wiley India.
2. William Stallings, Cryptography and Network Secuity, Pearson India.
3. Tim Mather, Subra Kumaraswamy, Shahed Latif, Cloud Security and Privacy, O'Reilly (India editions via Shroff).

Reference Books

1. Charles P. Pfleeger and Shari Lawrence Pfleeger, Security in Computing, Pearson India.
2. Michael Howard and David LeBlanc, Writing Secure Code, Microsoft Press.
3. ISO/IEC 27001 and ISO/IEC 27701 (latest editions).
4. Gordon Lyon, Nmap Network Scanning.

MOOC / NPTEL/YouTube Links

1. OWASP Foundation videos.
2. ZAP Getting Started Guide.
3. Wireshark User Guide.
4. Greenbone Community documentation.

Official and Open Resources :

1. DPDP Act 2023 and DPDP Rules 2025 (MeitY).

2. CERT-In Cyber Security Directions and FAQs; CERT-In incident reporting form.
3. OWASP Top 10 and OWASP API Security Top 10; OWASP ASVS.
4. NIST SP 800-61 (Incident Response).
5. Cloud Security Alliance Security Guidance (selected domains).
6. AWS Well-Architected Security Pillar (IAM, logging, shared responsibility).
7. NIST SP 800-218 (Secure Software Development Framework, SSDF).

Savitribai Phule Pune University **Third Year Artificial Intelligence and Data Science- 2024 Pattern**		
PEC361BAID : Big Data Engineering		
Teaching Scheme	**Credits**	**Examination Scheme**
Theory : 02 Hours/Week	02	CCE : 30 Marks End-Semester: 70 Marks

Prerequisite Courses: Basic understanding of Data Structures, Database Management Systems (DBMS), Operating Systems, and programming knowledge in Java or Python is essential.

Course Objectives: The course aims to:

1. Interpret key Big Data principles and the challenges encountered during implementation.
2. Describe the architecture of Hadoop, including HDFS and the MapReduce programming model.
3. Utilize Big Data processing tools and analytics frameworks.
4. Implement Big Data techniques for real-world applications such as web and social network analytics.
5. Assess ethical issues and promote responsible use of Big Data.

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1:** Analyze the key characteristics of Big Data along with its architecture and distributed computing concepts.
- **CO2:**Operate within the Hadoop ecosystem, including HDFS and MapReduce..
- **CO3:**Apply Hadoop tools and processing frameworks for data analytics tasks.
- **CO4:**Design and develop Big Data analytics solutions for practical applications.
- **CO5:**Develop secure solutions that uphold data privacy and meet compliance requirements

Course Contents

Unit I - Introduction to Big Data & Its Ecosystem (06 Hours)

Introduction to Big Data, Evolution of Big Data, Characteristics of Big Data (Volume, Velocity, Variety, Veracity, Value), Types of Data – Structured Data, Semi-Structured Data, Unstructured Data, Limitations of Traditional Data Processing Systems, Distributed Computing Concepts, Parallel Processing, Big Data Architecture Layers, Overview of Hadoop Ecosystem, Applications of Big Data.

Case Study: A college generates large amounts of data such as attendance, exam results, LMS logs, and feedback forms. The traditional database system becomes slow during result processing. How can Big Data concepts help in handling Volume, Velocity, and Variety of campus data

Unit II Hadoop and HDFS - (06 Hours)

Hadoop Architecture, Core Components of Hadoop, Hadoop Distributed File System (HDFS), NameNode, DataNode, Secondary NameNode, Block Storage Mechanism, Data Replication Strategy, Fault Tolerance in Hadoop, MapReduce Programming Model, Map Phase, Reduce Phase, Combiner Function, YARN Architecture, Resource Management in Hadoop.

Case Study: Design a solution using the Hadoop Distributed File System (HDFS) to resolve storage capacity, scalability constraints, and fault tolerance issues for an e-commerce company storing millions of daily transaction records on a single server.

Unit III -Apache Spark and Data Processing (06 Hours)

Basics of Apache Spark, Spark Architecture – Driver, Executor, Cluster Manager, Resilient Distributed Datasets (RDD), RDD Operations – Transformations and Actions, DataFrames, Spark SQL, Spark Streaming Basics, Comparison between Hadoop MapReduce and Spark, In-Memory Processing Concept.

Case Study: Enable a social media platform to efficiently manage unstructured data in various formats (user posts, images, comments, likes) by leveraging NoSQL databases like MongoDB for flexible schema design and high scalability.

Unit IV - Big Data Tools and Applications (06 Hours)

Data Ingestion Techniques, Apache Flume, Apache Sqoop, Batch Processing, Real-Time Processing, Data Pipeline Architecture, Cloud-Based Big Data Platforms, Big Data Security Basics, Data Governance Concepts, Industry Applications – Healthcare, Banking, E-commerce, Smart Cities.

Case Study: Leverage Big Data analytics techniques to enable a bank to detect fraudulent transactions in real time, analyze customer spending behavior, and support predictive decision-making for fraud prevention.

Unit V - Big Data Security & Governance (06 Hours)

Security: Security challenges in Big Data, authentication and authorization, data encryption (at rest and in transit), access control mechanisms (RBAC and ABAC), audit and compliance monitoring, risk management and disaster recovery

Governance: Data privacy and compliance, data governance and lifecycle, data anonymization and masking, ethical considerations

Case Study: An organization handling sensitive data requires Big Data security and governance mechanisms like encryption, RBAC/ABAC access control, and data masking to ensure privacy, compliance, and secure data management.

Learning Resources

Text Books:

1. Diego Rodrigues, "Fundamentals of Big Data: With Hadoop and Spark", Self/Academic Press, 2024 Edition, ISBN: 978-XXXXXXX (latest 2024 integrated Hadoop–Spark framework text covering modern big data systems).
2. Jules S. Damji, Brooke Wenig, Tathagata Das, and Denny Lee, "Learning Spark: Lightning-Fast Big Data Analysis", O'Reilly Media, 2nd Edition, 2020, ISBN: 9781492050049.

Reference Books

1. Ambrish Kumar Sharma, J Jegan Amarnath, Dr. G. Vadivel, and S. Suganya, "A Brief Guide to Big Data & Hadoop", AG Publishing House, 2022, ISBN: 978-93-90593.
2. Mayank Bhushan, "Big Data Analytics: Introduction to Hadoop, Spark and Machine Learning", BPB Publications, 2024 Edition, ISBN: 978-93-5551

MOOC / NPTEL/YouTube Links

1. https://www.coursera.org/specializations/packt-big-data-foundations-with-hadoop-and-spark?- Includes Hadoop, Spark, Scala, and Kafka concepts with hands-on labs.
2. https://www.educative.io/courses/introduction-to-big-data-and-hadoop?- beginner-friendly Big Data fundamentals and Hadoop basics.
3. YouTube/Video Links: 1. https://www.youtube.com/watch?v=YHiS441Bk1E

E-Books

1. https://www.overdrive.com/media/3001853/big-data-analytics?utm_source=chatgpt.com

Savitribai Phule Pune University **Third Year Artificial Intelligence and Data Science- 2024 Pattern**		
PEC261CAID - High Performance Computing		
Teaching Scheme	**Credits**	**Examination Scheme**
Theory : 03 Hours/Week	03	CCE : 30 Marks End-Semester: 70 Marks

Course Objectives: The course aims to:

1. To understand the fundamentals of High Performance Computing.
2. To learn parallel computing concepts and performance evaluation techniques.
3. To apply basic parallel programming models.
4. To understand GPU and accelerator-based computing.
5. To explore cloud-based HPC for AI applications.

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1 :** Explain fundamental HPC concepts and performance metrics.
- **CO2 :** Apply parallel programming using shared and distributed memory models.
- **CO3 :** Describe GPU and accelerator-based computing for AI workloads.
- **CO4 :** Analyze HPC solutions for AI and data-intensive applications.
- **CO5 :** Evaluate cloud-based HPC systems for scalable computing.

Course Contents

Unit I - Fundamentals of Modern HPC (09 Hours)

Introduction to HPC: Evolution from sequential to parallel computing, need for HPC in modern workloads, limitations of sequential systems. Parallel vs Distributed Computing. Shared vs Distributed Memory. Basics of Parallel Computing: Concurrency vs Parallelism, Data Parallelism, Task Parallelism, Flynn's Classification (SISD, SIMD, MIMD). Performance Metrics: Execution time, Speedup, Efficiency, Throughput, Scalability (Strong and Weak Scaling). Performance Laws: Amdahl's Law and Gustafson's Law. Applications of HPC in AI, ML, Big Data and Scientific Computing.

Case Study: Speedup Analysis of Deep Learning Model Training using Multi-core Processors

Unit II - Parallel Programming Models (09 Hours)

Introduction to Parallel Programming and "Thinking Parallel". Shared Memory Programming using OpenMP (parallel for, reduction, tasks – basic concepts). Distributed Memory Programming using MPI (send, receive, broadcast – basic communication). Comparison of OpenMP and MPI. Introduction to Hybrid Programming. Synchronization concepts: barriers and basic race conditions.

Case Study: Parallel Matrix Multiplication using OpenMP vs MPI.

Unit III - Accelerators and GPU Computing (09 Hours)

Need for hardware accelerators in modern HPC and limitations of CPU-centric systems for AI workloads. Overview of accelerators such as GPUs, FPGAs and TPUs. Fundamentals of GPU architecture: streaming multiprocessors, thread hierarchy (threads, blocks, grids), SIMT execution model, and GPU memory hierarchy. Basic GPU execution workflow. Introduction to CUDA programming model: kernel launch, host-device interaction, and basic memory management concepts. Role of GPUs in AI and deep learning: matrix operations, training and inference acceleration. Overview of optimized libraries such

as cuBLAS and cuDNN. Basic performance considerations: memory bandwidth, latency, and idea of multi-GPU systems with CPU–GPU coordination.

Case Study: Performance measurement, bottleneck identification, optimisation (kernel fusion, Tensor Cores, batch size tuning), Comparison with CPU-only baseline & with FPGA-based inference (optional)

Unit IV - HPC for AI and Data-Intensive Applications (09 Hours)

Introduction to HPC for AI : Why AI needs HPC , computational challenges in deep learning , Key aspect of HPC in AI, Architecture, Overview of AI workloads: training vs.inference, Data Parallelism vs Model Parallelism, Examples of AI framworks and their HPC integration Distributed Deep Learning, Application of HPC for AI. Introduction to Data-Intensive Applications : Big Data Architectures, Distributed File Systems, Data Locality, MapReduce Paradigm, Spark-based Distributed Processing. Performance challenges in large-scale AI systems: communication overhead and scalability issues.

Case Study: Distributed Training of a Deep Learning Model using Data Parallelism.

Unit V - Cloud and Modern HPC Systems (09 Hours)

Introduction to Cloud-based HPC and virtual clusters. On-premise HPC vs Cloud HPC.

Infrastructure as a Service (IaaS) for HPC workloads. Containers (Docker – basic concept) for portable HPC applications. Introduction to Kubernetes for workload orchestration. Elastic scaling and auto-scaling in cloud environments. Cost-aware computing and use of spot instances. Overview of Energy efficiency and green computing in HPC. Future trends: Exascale computing, AI-driven HPC optimization.

Case Study: Deployment of AI Model Training on Cloud Cluster – Cost and Scalability Analysis.

Learning Resources

Text Books:

1. Grama, Ananth, Gupta, Anshul, Karypis, George, and Kumar, Vipin, Introduction to Parallel Computing, Pearson Education, 2nd Edition, 2003, ISBN: 978-0201648652.
2. Quinn, Michael J., Parallel Programming in C with MPI and OpenMP, McGraw-Hill, 2004, ISBN: 978-0072822564.
3. Hwang, Kai, Dongarra, Jack, and Fox, Geoffrey, Distributed and Cloud Computing: From Parallel Processing to the Internet of Things, Morgan Kaufmann, 1st Edition, 2012, ISBN: 978-0123858801.

Reference Books

1. Kirk, David B., and Hwu, Wen-mei W., Programming Massively Parallel Processors: A Hands-on Approach, Morgan Kaufmann, 3rd Edition, 2016, ISBN: 978-0128119860.
2. Rauber, Thomas and Rünger, Gudula, Parallel Programming for Multicore and Cluster Systems, Springer, 2nd Edition, 2013, ISBN: 978-3642378010.

MOOC / NPTEL/YouTube Links

1. High Performance Computing by Prof. S. Gopalakrishnan, IIT Madras https://nptel.ac.in/courses/10610
2. Parallel Computer Architecture and Programming – IIT Kanpur (NPTEL) https://nptel.ac.in
3. Introduction to High Performance Computing https://www.youtube.com/watch?v=KsL_uwa0ekY

E-Books

1. https://dl.acm.org/doi/pdf/10.5555/3455710 NVIDIA CUDA Programming Guide (Official Documentation)
2. https://docs.nvidia.com/cuda/ OpenMP API Specification (Official Documentation)
3. https://www.openmp.org/specifications/ MPI Standard Documentation
4. https://www.mpi-forum.org/docs/

Savitribai Phule Pune University **Third Year Artificial Intelligence and Data Science- 2024 Pattern**		
PEC361DAID: Augmented Reality and Virtual Reality		
Teaching Scheme	**Credits**	**Examination Scheme**
Theory : 03 Hours/Week	03	CCE : 30 Marks End-Semester: 70 Marks

Prerequisite Courses: Linear Algebra, Probability and Statistics, Machine Learning

Course Objectives: The course aims to:

1. To understand the fundamentals and evolution of Augmented Reality (AR) and Virtual Reality (VR) technologies.
2. To learn basic 3D graphics and scene representation concepts required for immersive environments.
3. To study interaction techniques and user interface design principles used in immersive systems.
4. To understand augmented reality tracking, spatial mapping and mobile AR frameworks.
5. To analyze applications, challenges and ethical considerations of AR/VR systems.

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1:** Explain the concepts, components and applications of AR/VR systems. (L2 – Understand)
- **CO2:** Apply basic 3D scene representation and transformation concepts in immersive environments. (L3 – Apply)
- **CO3:** Describe interaction techniques and user interface design principles in virtual environments. (L2/L3 – Understand/Apply)
- **CO4:** Explain tracking, registration and spatial mapping techniques used in augmented reality systems. (L2 – Understand)
- **CO5:** Analyze real-world applications, performance issues and ethical considerations in AR/VR systems. (L4 – Analyze)

Course Contents

Unit I - Introduction to AR and VR (09 Hours)

Introduction and evolution of immersive technologies, definitions of AR, VR, MR and XR, components of AR/VR systems, hardware devices such as head-mounted displays (HMDs), controllers and sensors, human perception, immersion and presence, field of view and frame rate considerations, motion sickness issues in VR, applications of AR/VR in gaming, healthcare, education and industry, challenges and future trends.

Case Study: Virtual Reality in Medical Training, Study of VR-based surgical simulation systems used in medical education to train surgeons in a safe and controlled virtual environment.

Unit II - Representing the Virtual World (8 Hours)

Basics of 3D graphics for immersive environments, representation of virtual environments, 3D geometric objects and scene structures, coordinate systems (world, local, camera), geometric transformations (translation, rotation, scaling), lighting and materials in virtual scenes, texture mapping concepts, spatial audio and 3D sound, haptic feedback concepts, avatars and virtual objects.

Case Study: Design of Virtual Environments in Game Engines, Study of how game engines such as Unity create immersive 3D environments using models, lighting, textures and spatial audio.

Unit III Interaction and Navigation in Virtual Environments (8 Hours)

Interaction techniques in virtual environments, navigation and locomotion methods, gesture-based interaction, controller-based interaction, natural user interfaces, VR user interface design principles, immersive UX design, cybersickness reduction techniques, performance and latency considerations

Case Study: Interaction Design in VR Gaming, Analysis of interaction mechanisms in the VR rhythm game Beat Saber focusing on gesture-based interaction and user engagement.

Unit IV Augmented Reality Systems and Tracking (8 Hours)

Principles of augmented reality, marker-based tracking techniques, marker-less tracking approaches, concept of simultaneous localization and mapping (SLAM), overview of sensor fusion, spatial mapping and plane detection, AR content registration and anchoring, mobile AR frameworks such as ARCore and ARKit.

Case Study: Augmented Reality in Retail Visualization, Study of AR applications used by furniture retailers that allow customers to visualize products in their homes using mobile devices and marker-less AR.

Unit V Advanced Immersive Systems and Applications (8 Hours)

Mixed reality systems, WebXR and browser-based immersive applications, collaborative virtual environments, industrial and healthcare AR/VR applications, ethical, privacy and security issues in immersive systems, future trends in spatial computing.

Case Study: Virtual Reality for Industrial Training, Analysis of VR-based safety and equipment training used in manufacturing industries to simulate real-world operations.

Learning Resources

Text Books:

1. Understanding Virtual Reality: Interface, Application and Design — William R. Sherman and Alan B. Craig, Morgan Kaufmann.
2. Augmented Reality: Principles and Practice — Dieter Schmalstieg and Tobias Hollerer, Addison-Wesley.

Reference Books

1. Learning Virtual Reality — Tony Parisi, O'Reilly Media.
2. Augmented Reality for Developers — Jonathan Linowes, Packt Publishing.
3. Virtual Reality Technology — Grigore Burdea and Philippe Coiffet, Wiley.
4. 3D User Interfaces: Theory and Practice — Doug Bowman, Addison-Wesley.

MOOC / NPTEL/YouTube Links

1. Foundation for Virtual and Augmented Reality Systems, IIT Guwahati : https://onlinecourses.nptel.ac.in/e-learning/preview/noc26_cs03
2. Foundations of Virtual Reality, IIT Madras, Prof. M. Manivannan : https://onlinecourses.nptel.ac.in/e-learning/preview/noc25_cs87
3. Mobile Virtual Reality and Artificial Intelligence : https://onlinecourses.nptel.ac.in/e-learning/preview/

Savitribai Phule Pune University **Third Year Artificial Intelligence and Data Science- 2024 Pattern**		
PEC362AAID: Ethical Hacking		
Teaching Scheme	**Credits**	**Examination Scheme**
Theory : 03 Hours/Week	03	CCE : 30 Marks End-Semester: 70 Marks

Prerequisite Courses: Computer Networks, Operating Systems, Cyber Security Fundamentals

Course Objectives: The course aims to:

1. Equip students with core knowledge of ethical hacking, its application scope, relevant legal frameworks, and the evolving cyber threat landscape
2. Develop understanding of reconnaissance techniques and attack surface analysis using OSINT and exposure assessment methods.
3. Build knowledge of common vulnerabilities, exploitation concepts, and attack techniques used in real-world scenarios.
4. Analyze identity-based attacks, social engineering methods, and post-compromise activities in modern systems.
5. Explore security challenges in modern environments including web, API, cloud, and AI-driven systems along with ethical responsibilities

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1 :** Apply knowledge of ethical hacking frameworks, lifecycle stages, and professional guidelines to assess modern threat landscapes.
- **CO2 :** Utilize reconnaissance methodologies and tools to identify exposed assets and map attack surfaces effectively
- **CO3 :** Identify common vulnerabilities and demonstrate basic exploitation concepts in controlled environments.
- **CO4 :** Analyze identity-based attacks, credential misuse, and post-compromise activities to understand attacker behavior.
- **CO5 :** Evaluate security risks in web, API, cloud, and AI systems and understand ethical responsibilities in cybersecurity practices.

Course Contents

Unit I - Ethical Hacking Approach & Modern Threat Landscape (09 Hours)

Ethical Hacking – definition, purpose, and scope; Difference between penetration testing, vulnerability assessment, and red teaming; Ethical hacking lifecycle – reconnaissance, scanning, exploitation, post-compromise, reporting; Rules of Engagement – authorization, scope, ethical boundaries; Attacker mindset – targeting weakest link, identity-focused attacks, attack chaining; Overview of MITRE ATT&CK – understanding attacker tactics.

Modern Threat Landscape: Modern attack trends – identity-based attacks, cloud misconfigurations, API abuse, supply chain attacks, Phishing evolution – AiTM phishing, session/token theft, Zero Trust Security Model and identity-centric security principles ,AI-augmented attacks – LLM prompt injection, adversarial ML, deepfake-enabled social engineering.

Case Study: Modern identity-based breach involving credential misuse, AiTM phishing, and unauthorized cloud access (e.g., analysis of a real-world Microsoft 365 compromise scenario).

Unit II Reconnaissance & Vulnerability Discovery (09 Hours)

Reconnaissance – definition, importance, passive vs active; OSINT – public sources; Digital footprint and attack surface; DNS and domain intelligence; Service exposure; Vulnerability scanning concept; WHOIS, ASN lookups, BGP peering analysis, Google Dorking and advanced search operators, GitHub/GitLab OSINT – exposed secrets, API keys, credentials in public repos

Scanning & Enumeration: Network scanning – Nmap techniques: SYN, UDP, version detection, OS fingerprinting ,Service enumeration :banner grabbing, version identification, Vulnerability scanning concept – Nessus, OpenVAS, Cloud attack surface :exposed S3 buckets, Azure blobs, GCP storage misconfigurations ,Internet-wide exposure awareness : Shodan queries, Censys facets, exposed IoT devices ,Attack surface management (ASM) platforms .

Case Study: Data exposure due to misconfigured cloud storage: full reconnaissance chain from OSINT to bucket discovery, impact analysis, and responsible disclosure process.

Unit III Vulnerabilities & Exploitation Concepts (09 Hours)

Vulnerabilities : types; Difference between vulnerability and exploit; CVE, CVSS scoring, NVD, Exploit-DB, VulnHub, Metasploit Framework : architecture, modules Web vulnerabilities – OWASP Top 10 (2021 & 2025 updates) – overview of each category ,broken access control, authentication weaknesses; Common attack concepts – SQL Injection, XSS; Password attacks; Credential attacks; Exploitation concept.

Network & System Attacks: Password attacks – brute force, dictionary, credential stuffing, rainbow tables

Network sniffing :ARP poisoning, passive capture (Wireshark),Wireless attacks : WPA2 cracking, Evil Twin, Buffer overflow concept – stack vs. heap

Emerging: IoT & OT Attack Surface: IoT attack vectors ,OT/ICS/SCADA security fundamentals and attack scenarios, Embedded device exploitation awareness (UART, JTAG)

Case Study: Web application compromise due to chained vulnerabilities – SQL injection to authentication bypass, privilege escalation, and data exfiltration in a simulated e-commerce portal.

Unit IV : Identity Attacks & Post-Compromise Overview (09 Hours)

Identity as Attack Surface: Identity-centric security ,Active Directory attacks ,Azure AD / Entra ID attack paths Credential-based attacks ,MFA bypass techniques.

Social Engineering: Social engineering , Phishing campaigns , AiTM (Adversary-in-the-Middle) phishing Business Email Compromise (BEC) , AI-enhanced social engineering – voice cloning, deepfake video calls, Physical security – badge cloning, social engineering for physical access.

Post-Compromise Activities: Post-exploitation objectives , Session/token abuse Data exfiltration methods and C2 (Command & Control) frameworks overview, Covering tracks .

Detection Awareness: SIEM alerts and EDR detection for attacker techniques, Attacker OPSEC – avoiding detection in red team engagements ,Threat hunting basics from attacker perspective.

Case Study : Identity-based attack chain: spear phishing $\rightarrow$ AiTM credential capture $\rightarrow$ MFA bypass $\rightarrow$ lateral movement via Pass-the-Hash $\rightarrow$ data exfiltration from SharePoint. Analysis includes detection opportunities at each stage.

Unit V - Modern Attack Domains: Web, API, Cloud & AI Threats (09 Hours)

Web Penetration Testing: Web pentesting methodology , Advanced web attacks – HTTP request smuggling, web cache poisoning, clickjacking ,

Tools: Burp Suite Pro (intruder, repeater, active scan), OWASP ZAP,

API Security Testing: REST, GraphQL, gRPC API attack techniques , API fuzzing, rate limit bypass, parameter pollution , JWT attacks ,

Tools: Postman, Insomnia

Cloud Penetration Testing: Cloud pentesting scope – AWS, Azure, GCP specific attack paths,, Cloud metadata exploitation (SSRF to IMDS), Container security – Docker escape, Kubernetes RBAC misconfigurations

Tools: Pacu (AWS), ScoutSuite, Prowler, CloudSploit

AI/LLM Security (Emerging Area): AI attack surface – LLM prompt injection, jailbreaking, adversarial inputs Poisoning attacks on ML training data, Securing AI/ML pipelines – model hardening, input validation, output filtering.

Ethical & Legal Framework: IT Act 2000 & Amendments – Sections 43, 66, 66C, 66F relevant to hacking

Case Study: Cloud/API breach simulation: SSRF to cloud metadata service → IAM token extraction → privilege escalation → S3 data exfiltration. Full penetration test report writing exercise.

Learning Resources

Text Books:

1. Patrick Engebretson – The Basics of Hacking and Penetration Testing, 3rd Ed., Syngress/Elsevier, (ISBN number 978-0443438868). March 2026
2. Rafay Baloch – Ethical Hacking and Penetration Testing Guide, Auerbach/CRC Press, 2014 (ISBN: 978-1482208917).
3. Matt Walker – CEH Certified Ethical Hacker All-in-One Exam Guide, 5th Ed., McGraw-Hill, 2022 (ISBN: 978-1264274567).
4. Georgia Weidman – Penetration Testing: A Hands-On Introduction to Hacking, No Starch Press, 2014 (ISBN: 978-1593275648

Reference Books

1. Jon Erickson – Hacking: The Art of Exploitation, 2nd Ed., No Starch Press, 2008 (ISBN: 978-1593271442).
2. 2Peter Kim – The Hacker Playbook 3: Practical Guide to Penetration Testing, Secure Planet, 2018 (ISBN: 978-1980901754). [Red teaming, advanced TTPs]
3. Christopher Hadnagy – Social Engineering: The Science of Human Hacking, 2nd Ed., Wiley, 2018 (ISBN: 978-1119433385).
4. Chris Anley et al. – The Shellcoder's Handbook: Discovering and Exploiting Security Holes, 2nd Ed., Wiley, 2007 (ISBN: 978-0470080238).
5. Michael Sikorski & Andrew Honig – Practical Malware Analysis, No Starch Press, 2012 (ISBN: 978-1593272906).
6. Daniel Miessler – The Real Internet of Things, Leanpub

MOOC / NPTEL Courses:

1. Ethical Hacking by Prof. Indranil Sen Gupta, IIT Kharagpur (NPTEL)
2. Cyber Security and Privacy by Prof. Saji K Mathew, IIT Madras (NPTEL)
3. Introduction to Cyber Security – Dr. Jeetendra Pande, Uttarakhand Open University
4. Practical Ethical Hacking – TCM Security (Heath Adams) – Udemy/TCM Academy
5. Web Application Hacking & Bug Bounty – TryHackMe and HackTheBox Academy

6. CEH v12 Preparation – EC-Council Official (paid) / free preview modules

YouTube Links:

1. The Cyber Mentor (TCM Security) – Full Ethical Hacking Course
2. NetworkChuck – Ethical Hacking Playlist (Beginner-friendly)
3. IppSec – HackTheBox Walkthroughs (Advanced Pentesting)
4. John Hammond – CTF & Malware Analysis
5. David Bombal – Network Hacking & Ethical Hacking
6. LiveOverflow – Binary Exploitation & Web Hacking
7. STOK – Bug Bounty Hunting & Web App Security

E-Books

1. Alana Maurushat – Ethical Hacking, University of Ottawa Press (Open Access, free PDF)
2. PCI Security Standards Council – Penetration Testing Guidance (Free PDF from pcisecuritystandards.org)
3. OWASP Testing Guide v4.2 (Free PDF from owasp.org) – comprehensive web pentesting methodology
4. NIST SP 800-115 – Technical Guide to Information Security Testing (free from nist.gov)
5. Zaid Sabih – Learn Ethical Hacking from Scratch (Packt eBook / Udemy course)
6. CREST – Penetration Testing & Security Assessment Guides (free PDFs from crest-approved.org)
7. HackTricks Book (free, gitbook.io/hacktricks) – continuously updated red team techniques

Practice Platforms & Lab Resources

1. TryHackMe (tryhackme.com) – Guided learning paths for ethical hacking
2. HackTheBox Academy (academy.hackthebox.com) – Professional penetration testing labs
3. PentesterLab (pentesterlab.com) – Web application security exercises
4. PortSwigger Web Security Academy (portswigger.net/web-security) – Free, world-class web hacking labs
5. VulnHub (vulnhub.com) – Downloadable vulnerable VMs for offline practice
6. DVWA, WebGoat, Metasploitable – Local lab environments for classroom use

Savitribai Phule Pune University **Third Year Artificial Intelligence and Data Science- 2024 Pattern**		
PEC362BAID: Quantum Artificial Intelligence		
Teaching Scheme	**Credits**	**Examination Scheme**
Theory : 03 Hours/Week	03	CCE : 30 Marks End-Semester: 70 Marks

Prerequisite Courses: Engineering Mathematics, Engineering Physics

Course Objectives: The course aims to:

1. Introduce the fundamental concepts and motivation behind quantum computing.
2. Develop the ability to use vector spaces, Hilbert spaces, operators, and tensor products in quantum mechanics.
3. Enable students to evaluate the advantages and limitations of quantum algorithms in solving computational problems.
4. Provide insight into quantum supervised and unsupervised learning techniques.
5. Familiarize with Quantum AI approaches for decision-making and problem-solving in diverse domains.

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1 :** Apply principles of superposition, entanglement, and quantum interference to represent and manipulate qubits.
- **CO2 :** Utilize the postulates of quantum mechanics to describe quantum measurements, expectation values, and time evolution of quantum systems.
- **CO3 :** Make use of fundamental quantum algorithms to address computational problems efficiently.
- **CO4 :** Analyze quantum learning techniques and quantum circuits for solving complex problems.
- **CO5 :** Examine Quantum AI techniques for decision-making and problem-solving in diverse domains.

Course Contents

Unit I - Fundamentals of Quantum Computing (09 Hours)

Overview of Quantum Computing: Motivation and evolution of quantum computing. Limitations of Classical Computing: Computational complexity, parallelism limits, and the need for quantum advantage. Quantum Mechanics for Computing: Superposition, entanglement, and quantum interference. Qubits vs Classical Bits: Representation of qubits and Bloch sphere visualization. Quantum Gates: Pauli gates (X, Y, Z), Hadamard gate, Controlled-NOT gate (CNOT). Quantum Circuits and Measurement: Quantum circuit model and measurement in quantum systems.

Case Study: Classical Bit vs Quantum Bit: Superposition Experiment

Unit II Mathematical Foundations of Quantum Mechanics (09 Hours)

Linear Algebra Framework for Quantum Mechanics: Vector spaces, complex Hilbert spaces, inner products, orthogonality, linear operators, eigenvalues and eigenvectors, Hermitian and unitary operators, tensor products. Postulates of Quantum Mechanics: Quantum states and state space, observables as operators, measurement postulate and Born rule, expectation values, time evolution of quantum

states, quantum states in Hilbert space, qubits and Bloch sphere representation, quantum measurement theory, density operator formalism, composite quantum systems. Fundamentals of Quantum Principle: No-cloning theorem and implications for quantum information and quantum AI.

Case Study: Detection of Eavesdropping in a Quantum AI Communication System using the No-Cloning Principle.

Unit III - Quantum Algorithms (09 Hours)

Overview of quantum algorithms and quantum speedup, Phil's algorithm, David Deutsch's algorithm, and David Deutsch–Jozsa algorithm, Daniel Simon's algorithm, Lov Grover's Search algorithm, Quantum Fourier Transform (QFT), Peter Shor's algorithm for integer factorization, introduction to quantum walk and search application.

Case Study: Classical vs Quantum Search: Understanding Grover's Algorithm.

Unit IV : Quantum Machine Learning (09 Hours)

Introduction to quantum-classical hybrid learning, data encoding and quantum feature maps, vibrational quantum circuits (VQC) and parametrized quantum circuits, quantum optimization and the parameter shift rule. Quantum Supervised Learning: Quantum support vector machine (QSVM) and quantum neural networks. Quantum Unsupervised Learning: Quantum K-Means and Quantum Principal Component Analysis (QPCA), quantum generative models, and the barren plateau problem.

Case Study: Conceptual Study of Hybrid Quantum-Classical Machine Learning Models

Unit V : Applications of Quantum Artificial Intelligence (09 Hours)

Quantum AI for Optimization Problems: Quantum-enhanced combinatorial optimization, logistics and supply chain optimization, resource allocation and scheduling, smart grid, and traffic optimization.

Quantum AI in Finance and Business Analytics: Portfolio optimization, risk analysis and modeling, fraud detection, and financial forecasting.

Quantum AI in Healthcare and Life Sciences: Drug discovery optimization, molecular modeling, genomic data analysis.

Quantum AI in Cybersecurity: Quantum-enhanced anomaly detection, intrusion detection systems, AI-assisted post-quantum cryptanalysis.

Quantum AI in Intelligent Systems: Quantum machine learning applications, intelligent decision systems, and advanced AI architectures.

Case Study: Quantum AI-Based Smart Traffic Signal Optimization System.

Learning Resources

Text Books:

1. Wichert, Andreas, "Quantum Artificial Intelligence with Qiskit," 1st Edition, Chapman & Hall / CRC Press, 2025, ISBN 978-1032448978.

2. Bernhardt, Chris, "Quantum Computing for Everyone," 1st Edition, The MIT Press, 2019, ISBN 978-0262039253.

3. Kaye, P., Laflamme, R., and Mosca, M., "An Introduction to Quantum Computing," 1st Edition, OUP Oxford, 2006, ISBN 978-0191524615.

4. Schuld, Maria, Petruccione, Francesco, "Supervised Learning with Quantum Computers", 1st Edition, Springer International Publishing, 2018, ISBN 978-3319964249.

Reference Books

1. Nielsen, M. A., Chuang, I. L., "Quantum Computation and Quantum Information", 10th Edition, Cambridge University Press, 2010, ISBN 978-1107002173.

2. Wittek, Peter, “Quantum Machine Learning: What Quantum Computing Means to Data Mining”, 1st Edition, Elsevier Science, 2014, ISBN 978-0128010990.

3. Yanofsky, Noson S., Mannucci, Mirco A., “Quantum Computing for Computer Scientists”, 1st Edition, Cambridge University Press, ISBN 978-0-521-87996-5.

MOOC / NPTEL/YouTube Links

1. NPTEL course: “Quantum Computing” by Prof. Debabrata Goswami, IIT Kanpur https://www.youtube.co Gm0yRYwTj7Fs6jyzYa83HErSrpXgPQ

2. NPTEL course: “Introduction to Quantum Computing: Quantum Algorithms and Qiskit” by Prof. Prabha Mandayam, Prof. Anupama Ray, Prof. Sheshashayee Raghunathan, IIT Madras & IBM Research.

   (a) https://onlinecourses.nptel.ac.in/noc26_cs89/preview

3. Quantum Computation Fundamentals: https://www.youtube.com/watch?v=KsL_uwa0ekY

4. Quantum machine learning: https://qiskit.org/learn/course/machine-learning-course/

5. Center for Excellence in Quantum Technology: https://research.ibm.com/blog/next-wave-quantum-centric-supercomputing

E-Books

1. Mermin, N. D., “Quantum Computer Science: An Introduction,” Cambridge University Press, 2007, ISBN 978-0521876572

Savitribai Phule Pune University **Third Year Artificial Intelligence and Data Science- 2024 Pattern**		
PEC362CAID - Reinforcement Learning		
Teaching Scheme	**Credits**	**Examination Scheme**
Theory : 03 Hours/Week	03	CCE : 30 Marks End-Semester: 70 Marks

Prerequisite Courses: Artificial Intelligence, Machine Learning
Course Objectives: The course aims to:

1. Understand fundamentals and analyze core elements of reinforcement learning
2. Develop the ability to model decision-making problems using Markov Decision Processes
3. Apply Monte Carlo and Temporal Difference learning methods for prediction and control in model-free reinforcement learning environments.
4. Analyze and implement advanced reinforcement learning algorithms
5. Understand reinforcement learning with human feedback, multi-agent systems, and ethical considerations for deploying RL in real-world applications.

Course Outcomes: Upon successful completion of this course, students will be able to:

- Explain the fundamentals and need of Reinforcement Learning
- Determine optimal policies for sequential decision-making problems
- Apply Monte Carlo methods for prediction and control in model-free environments.
- Apply deep reinforcement learning techniques to complex real-world problems.
- Understand multi-agent reinforcement learning and interaction strategies..

Course Contents

Unit I - Introduction to Reinforcement Learning (09 Hours)

Introduction to Reinforcement Learning(RL) and Need of RL, Comparison of Supervised, Unsupervised and Reinforcement Learning, Agent–Environment Interaction Framework, Rewards, Returns and Discount Factor, Episodic and Continuing Tasks, Types of Reinforcement Learning (Model-based, Model-free, On-policy, Off-policy), Elements of Reinforcement Learning (Policy, Value Function, Model, Reward Signal), Challenges in Reinforcement Learning (Exploration–Exploitation, tradeoff, Sample inefficiency, Delayed rewards, Curse of dimensionality), Multi-Armed Bandit Problem and its relationship to Reinforcement Learning, Exploration Strategies: ε-greedy, Optimistic Initialization, Upper Confidence Bound (UCB), Introduction to Thompson Sampling, Practical considerations in reward design and reward shaping.

Case Study: Tic-Tac-Toe

Unit II Markov Decision Process (09 Hours)

Markov property, Markov chains and Markov Reward Processes. Definition and components of Markov Decision Process (state space, action space, transition probability, reward function, discount factor). Policies, state-value and action-value functions. Bellman expectation and Bellman optimality equations. Optimal policy, principle of optimality and convergence properties. Applications of MDP in sequential decision-making problems, Advantage Function and its interpretation, Relationship between Value Functions and Policy Optimization

Case Study: : Warehouse Robot Navigation using Markov Decision Process

Unit III - Monte Carlo Methods for Model Free Prediction and Control (09 Hours)

Introduction, Monte Carlo Prediction, Monte Carlo Estimation of Action Values, Monte Carlo Control, Monte Carlo Control without Exploring Starts, Off-policy Prediction via Importance Sampling (hints), Incremental Implementation of Monte Carlo Prediction, Off-Policy Monte Carlo Control, Temporal Difference Learning (TD) Concept and Update Rule, Bootstrapping vs Monte Carlo – Bias-Variance Tradeoff, SARSA (On-Policy TD Control), Comparison between SARSA and Q-Learning, Practical stability considerations in TD methods.

Case Study: : Analyze and implement Monte Carlo methods for solving an episodic reinforcement learning problem without prior knowledge of environment dynamics.

Unit IV - Q-Learning and Deep Q-Networks (09 Hours)

Deep Q Network, Double Deep Q Network, Two estimators, Dueling Networks and Architecture, Recurrent DQN, Comparison with Baseline DRQN, Hierarchical DQN, Policy Gradient Methods – REINFORCE Algorithm, Actor-Critic Architecture, Advantage Actor-Critic (A2C/A3C) – Conceptual Overview+ Proximal Policy Optimization (PPO) – Clipped Objective and Stability Intuition, Comparison: Value-Based vs Policy-Based Methods, Continuous Action Spaces and their Challenges, Industrial RL frameworks (Gymnasium, Stable-Baselines3, RL lib)

Case Study: FPS shooter games, Object detection as a RL problem

Unit V - RL with Human Feedback (09 Hours)

Reinforcement Learning from Human Feedback (RLHF), Reward Modeling, Preference Learning, Policy Optimization with Human Feedback, Alignment and Safety Considerations, Role of PPO in RLHF, Multi-Agent Reinforcement Learning (MARL), Cooperative vs Competitive Agents, Self-Play, Centralized vs Decentralized Training, Offline Reinforcement Learning (Batch RL) – Conceptual Overview, Safety, Ethics and Deployment Challenges in Real-World RL Systems, RL in Agentic AI Architectures (LLM + RL + Planning Integration)

Case Study: RLHF case study in language model alignment

Learning Resources

Text Books:

1. Reinforcement Learning with TensorFlow: A beginner's guide to designing self-learning systems with TensorFlow and OpenAI Gym, Sayon Dutta, Packt Publishing (24 April 2018)https://web.stanford.e
2. Reinforcement Learning: An Introduction, second edition Richard S. Sutton and Andrew G. Barto, The MIT Press Cambridge, Massachusettshttps://incompleteideas.net/book/the-book-2nd.html?u

Reference Books

1. Richard S. Sutton and Andrew G. Barto, "Reinforcement Learning: An Introduction" https://rlhfbook.com chap1-3/?utm_source
2. Sudharsan Ravichandiran, "Hands-On Reinforcement Learning with Python: Master Reinforcement and Deep Reinforcement Learning Using OpenAI Gym and TensorFlow" https://www.packtpub.com in/product/hands-on-reinforcement-learning-with-python-9781788836522?utm_source
3. Maxim Lapan, "Deep Reinforcement Learning Hands-On: Apply Modern RL Methods, with Deep Q-networks, Value Iteration, Policy Gradients, TRPO, AlphaGo Zero and More" https://www.packtpub.co in/product/deep-reinforcement-learning-hands-on-9781788834245?utm_source
4. Sergey Levine – Deep Reinforcement Learning Course Notes (Policy Gradient & Actor-Critic)
5. John Schulman et al. – Proximal Policy Optimization Algorithms

6. Christiano et al. – Deep Reinforcement Learning from Human Preferences (RLHF)

7. Busoniu et al. – Multi-Agent Reinforcement Learning: A Survey https://www.researchgate.net/publicatio agent_Reinforcement_Learning_A_Survey?utm_source

MOOC / NPTEL/YouTube Links

1. Quantum Computing by Prof .Debabrata Goswami, IIT Kanpur https://nptel.ac.in/courses/104104082

2. Reinforcement Learning Specialization https://www.coursera.org/specializations/reinforcement learning

3. Reinforcement Learning Lecture Series 2021 https://www.deepmind.com/learning resources/reinforcem learning-lecture-series-2021

4. Introduction to Reinforcement Learning with David Silver https://www.deepmind.com/learning resources/introduction-to-reinforcement-learning-with-david- silver

5. Reinforcement Learning Specialization https://www.coursera.org/specializations/reinforcement learning

6. Reinforcement Learning Lecture Series 2021: https://www.deepmind.com/learning resources/reinforce learning-lecture-series-2021

7. Introduction to Reinforcement Learning with David Silver: https://www.deepmind.com/learning resources/introduction-to-reinforcement-learning-with-david-silver

E-Books

1. https://www.cs.toronto.edu/∼urtasun/courses/CSC411_Fall16/19_rl.pdf

2. Jason Gauci, “Horizon: Facebook’s Open Source Applied Reinforcement Learning Platform”, https://doi.org/10.48550/arXiv.1811.00260

Savitribai Phule Pune University **Third Year Artificial Intelligence and Data Science- 2024 Pattern**		
PEC362DAID - User Interface and User Experience Design		
Teaching Scheme	**Credits**	**Examination Scheme**
Theory : 03 Hours/Week	03	CCE : 30 Marks End-Semester: 70 Marks

Course Objectives: The course aims to

1. To understand principles of user-centered interface design.
2. To apply visual and interaction design techniques for digital interfaces.
3. To analyze user requirements using UX research methods.
4. To develop wireframes and prototypes for applications.
5. To explore modern UI/UX tools and industry practices.

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1:** Explain fundamental concepts of UI/UX and user-centered design principles.
- **CO2:** Apply human-computer interaction and visual design principles in interface design.
- **CO3:** Analyze user needs using UX research techniques and information architecture.
- **CO4:** Develop wireframes and interactive prototypes using modern design tools.
- **CO5:** Evaluate modern UI/UX trends, tools, and industry practices for designing effective digital products.

Course Contents

Unit I -Introduction to UI/UX Design - (09 Hours)

What is UI/UX Design: Importance of User-Centric Design, Goals of User Interface Design , Design Thinking Process, Core Principles, Role of UX in Product Development Lifecycle, mental and conceptual model.

Usability Design Principles: 4 Design Principles, Schneiderman's Golden Rules, Gestalt Principles of Design, Visual Design Principles, Form versus Function, Metaphors, Idioms and Affordances in UI design, User Interface Elements: Input Controls, Navigation Components, Information Components, Containers User Research: Qualitative and Quantitative User Research, Behavioral and Attitudinal User Research, Use of Personas, User Stories and Scenarios, Affinity Mapping.

Case Study: A good and a bad User Interface Design

Unit II - Usability Engineering, Evaluation and Testing (09 Hours)

Usability Engineering: Concept of usability, usability principles, benefits of usability for users and organizations, internationalization and localization, human errors and their impact on usability. Usability Evaluation: Human information processing and memory, Fitts's Law and Hick's Law, usability inspection methods such as Heuristic Evaluation and Cognitive Walkthrough, user studies and field studies. Usability Testing: Planning and conducting usability testing, Think-Aloud testing, A/B testing, use of heatmaps, and basic UX metrics for evaluating user experience. **UX Design Foundations:** Ideation and research in UX design, content and interaction mapping, paper prototyping, introduction to wireframes and interface layout, applying Nielsen's usability heuristics, refining UI based on user feedback.

Case study: Mobile Banking App UX Analysis (Example Apps Google Pay, PhonePe, Paytm)

Unit III - WEB DESIGN: STRATEGIES AND INFORMATION ARCHITECTURE (09 Hours)

User-centric design : The UX Phases - Waterfall vs. Agile - Web vs. App. Determining Strategy: User Research - Inspiration - Analytics - User Needs and Client Needs - Target Audience - What is in and What is Out: Outlining Scope - Content and Functionality. The Sitemap: Introduction to Sitemaps - Information Architecture - Sitemap Concerns - annotated process - Elements - Treejack Introduction - Treejack Analysis.

Case study: Analyze product search, checkout flow, recommendation systems, and visual hierarchy (Amazon, Flipkart, Myntra)

Unit IV - Wireframing Fundamentals - (09 Hours)

Introduction to wireframes, purpose and importance in UX design, low-fidelity vs high-fidelity wireframes, sketching interfaces, layout grids, design hierarchy, and responsive layouts. Prototyping Concepts: Introduction to prototypes, types of prototypes (paper prototypes, digital prototypes, interactive prototypes), fidelity levels, advantages of prototyping in UX design, iterative design process. **Interaction Design:** Principles of interaction design, micro-interactions, feedback mechanisms, navigation flows, usability considerations in interaction design, designing intuitive interfaces. UI Design Tools: Introduction to modern UI/UX design tools such as Figma, Adobe XD, Sketch, and InVision. Creating interactive prototypes, collaborative design workflows, design handoff to developers. **Design Documentation:** User flow diagrams, storyboards, design specifications, and design system basics.

Case study: Study location-based UX and real-time interaction design (Swiggy, Zomato)

Unit V - Designing UX for Tomorrow - (09 Hours)

Emerging technologies in UX Design: Voice UI, Touchless gesture control, Intelligent UX, Conversational UX, Immersive Media and Fluid UX Designing for Web and Mobile Interfaces, IoT applications, Industry Specific UX Design: FinTech, Education, Health Care, E-commerce and Industrial Websites, designing for Wearable Devices, designing for Augmented Reality, Virtual Reality and Mixed Reality, Tomorrow's challenges in UX design.

Case Study: ATM Interface Redesign for Senior Citizens with the objectives to improve usability for elderly and first-time users.

Learning Resources

Text Books:

1. Ben Shneiderman, Catherine Plaisant, Maxine Cohen, Steven Jacobs, Niklas Elmqvist, Nicholas Diakopoulos – Designing the User Interface: Strategies for Effective Human-Computer Interaction, Pearson.
2. Don Norman – The Design of Everyday Things, Basic Books.
3. Russ Unger and Carolyn Chandler – A Project Guide to UX Design, New Riders.

Reference Books

1. Steve Krug – Don't Make Me Think: A Common Sense Approach to Web Usability, New Riders.
2. Alan Cooper, Robert Reimann, David Cronin – About Face: The Essentials of Interaction Design, Wiley.
3. Jesse James Garrett – The Elements of User Experience, New Riders.
4. Jeff Gothelf & Josh Seiden – Lean UX, O'Reilly.

MOOC / NPTEL/YouTube Links

1. Google UX Design Professional Certificate – Coursera
2. Interaction Design Specialization – Coursera (UC San Diego)
3. User Experience Design Fundamentals – Udemy
4. Introduction to User Experience Design – Georgia Tech (Coursera)
5. Figma UI/UX Design Essentials – Udemy

Savitribai Phule Pune University **Third Year Artificial Intelligence and Data Science- 2024 Pattern**		
PEC363AID - Elective II Laboratory		
Teaching Scheme	**Credits**	**Examination Scheme**
Practical : 02 Hours/Week	01	Term Work: 25 Marks Oral : 25 Marks

Guidelines for Instructor's Manual

The instructor's manual is to be developed as a reference and hands-on resource. It should include prologue (about University/program/ institute/ department/foreword/ preface), curriculum of the course, conduction and Assessment guidelines, topics under consideration, concept, objectives, outcomes, set of typical applications/assignments/ guidelines, and references.

Guidelines for Student's Laboratory Journal

The laboratory assignments are to be submitted by student in the form of journal. Journal consists of Certificate, table of contents, and handwritten write-up of each assignment (Title, Date of Completion, Objectives, Problem Statement, Software and Hardware requirements, Assessment grade/marks and assessor's sign, Theory- Concept in brief, algorithm, flowchart, test cases, Test Data Set(if applicable), mathematical model (if applicable), conclusion/analysis). Program codes with sample output of all performed assignments are to be submitted as softcopy. As a conscious effort and little contribution towards Green IT and environment awareness, attaching printed papers as part of write-ups and program listing to journal must be avoided. Use of DVD containing students programs maintained by Laboratory In-charge is highly encouraged. For reference one or two journals may be maintained with program prints in the Laboratory.

Guidelines for Laboratory /Term Work Assessment

Continuous assessment of laboratory work should be based on overall performance of Laboratory assignments by a student. Each Laboratory assignment assessment will assign grade/marks based on parameters, such as timely completion, performance, innovation, efficient codes, and punctuality.

Guidelines for Laboratory Conduction - Cyber Security and Data Privacy Lab

List of Assignment - Group A

1. A data science team is designing an AI based student analytics portal. Demonstrate the application of cybersecurity and data privacy principles by explaining the CIA
2. A legacy banking system still relies on DES for certain secure transactions. Implement a program to perform encryption and decryption using DES to simulate secure data exchange.
3. TriadA secure messaging app requires text encryption using classical ciphers. Develop a program to encrypt and decrypt user messages using the Playfair Cipher to demonstrate secure communication basics.
4. A system administrator needs to restrict access to a server. Configure basic firewall rules using iptables to allow HTTP traffic while denying SSH connections.
5. A software vendor wants to ensure authenticity of updates. Implement digital signature generation and verification using RSA to validate sender identity and message integrity.
6. An online examination system must ensure that uploaded answer sheets are not tampered with; generate MD5 and SHA 1 hash values to verify data integrity.
7. A social media platform detects multiple fake accounts spreading malicious links. Simulate a social engineering attack using fake profiles and demonstrate preventive measures such as user awareness training, link filtering etc

8. A banking application needs to establish a secure communication channel between client and server; simulate Diffie Hellman key exchange to generate a shared secret key over an insecure network

9. A college student receives threatening messages on social media. To seek justice, the student must use official cybercrime reporting platforms such as the National Cyber Crime Reporting Portal. Demonstrate how to access the platform, file a complaint, and track the case status to ensure timely action

10. A college admission portal collects personal details of students such as name, address, and academic records. To comply with the DPDP Act, the portal must ensure that sensitive data is securely stored, transmitted only with consent, and protected from unauthorized access.

Mini project: Implement any one of the following

1. Secure Chat Application: Build a messaging app that integrates Diffie Hellman for key exchange, hashing for integrity, RSA signatures for authentication, and firewall rules for traffic control.

2. Digital Document Vault: Create a secure repository where files are hashed, signed, and verified before storage or retrieval, with firewall protection for access control.

3. IoT Device Security Framework: Design a lightweight security protocol for IoT devices using key exchange, hashing, and signatures, plus firewall rules to isolate devices from external threats.

4. Incident Response Simulator: Develop a simulation tool where students respond to cyber incidents by applying cryptography, hashing, signatures, and firewall defense in real time scenarios.

Savitribai Phule Pune University **Third Year - Artificial Intelligence and Data Science (2024 Pattern)**		
PEC363AID - Elective II Laboratory (Big Data Engineering)		
Teaching Scheme	**Credits**	**Examination Scheme**
Practical : 02 Hours/Week	01	Term Work: 25 Marks Oral : 25 Marks

Guidelines for Laboratory Conduction - Big Data Engineering

List of Assignment - Group A

1. Big Data Characteristics Analysis :Design a system that collects college data such as attendance, results, LMS logs, and feedback and analyzes it using Big Data tools to handle Volume, Velocity, and Variety of data efficiently.
2. Large-Scale Social Media Data Analyzer :Develop a system that processes structured, semi-structured, and unstructured social media data and classifies posts based on topics or sentiment using distributed processing.
3. Distributed File Storage System using HDFS :Implement a Hadoop-based distributed storage system to store large datasets and demonstrate data replication, fault tolerance, and block storage mechanism.
4. E-Commerce Transaction Analysis using MapReduce :Build a MapReduce application that processes large e-commerce transaction datasets to generate insights such as top-selling products, customer purchase patterns, and sales statistics.

List of Assignment - Group B (Any TWO)

1. Real-Time Log Analysis using Apache Spark :Create a Spark-based system that analyzes server or application logs in real time to detect errors, anomalies, and system performance issues.
2. Big Data Movie Recommendation System using Spark :Develop a Spark-based recommendation engine that analyzes large movie rating datasets and suggests movies based on user preferences.
3. Real-Time Fraud Detection System for Banking :A food delivery app receives thousands of live orders per minute. Design a Kafka-based data ingestion pipeline to handle high-velocity streaming data.
4. Smart City Data Analytics Platform :Develop a system that integrates transport, energy, and environmental datasets and uses Big Data analytics to provide insights for smart city planning and resource management.
5. Design and implement a secure Big Data storage system where sensitive data is stored in encrypted format and accessed through controlled authorization mechanisms. Apply authentication and authorization (RBAC/ABAC)
6. Develop a Big Data governance model that ensures data privacy, compliance, and ethical usage of data.

List of Assignment - Group C

1. Large-Scale Data Deduplication System:Design a Big Data application that processes large datasets stored in HDFS and identifies duplicate records efficiently using distributed processing frameworks such as MapReduce or Spark.

2. Distributed Data Processing for Weather Data Analysis:Develop a system that collects and analyzes large historical weather datasets to identify climate trends and patterns using Hadoop and Spark analytics tools.

3. Big Data Based Online Advertisement Analysis:Create a system that processes large-scale online advertisement clickstream data to analyze user engagement, click-through rates, and advertising performance.

4. Energy Consumption Analytics using Big Data:Design a platform that collects and processes energy usage data from multiple smart meters and analyzes consumption patterns to support efficient energy management.

5. Big Data Pipeline for Data Integration and Processing:Develop a data pipeline architecture that collects data from multiple sources, performs ingestion using tools such as Flume or Sqoop, stores it in HDFS, and processes it using Apache Spark.

Mini Project (Any One)

1. Real-Time Social Media Sentiment Analysis:Develop a Big Data application that collects social media data (tweets or posts) and analyzes user sentiment (positive, negative, neutral) using Apache Spark and streaming tools.

2. E-Commerce Product Recommendation System:Design a scalable system that analyzes large customer purchase datasets and recommends products based on user behavior and purchase history using Spark analytics.

Savitribai Phule Pune University **Third Year - Artificial Intelligence and Data Science (2024 Pattern)**		
PEC363AID - Elective II Laboratory (High Performance Computing)		
Teaching Scheme	**Credits**	**Examination Scheme**
Practical : 02 Hours/Week	01	Term Work: 25 Marks Oral : 25 Marks

Guidelines for Laboratory Conduction - High Performance Computing

1. Laboratory sessions should be conducted in alignment with the approved HPC theory syllabus.
2. Required software tools (OpenMP, MPI, CUDA, Spark, etc.) must be properly installed and tested before conducting practicals.
3. Each experiment should include brief explanation of theory followed by hands-on implementation.
4. Students should compare sequential and parallel versions of programs wherever applicable to analyze performance.
5. Emphasis should be given to performance metrics such as execution time, speedup, and efficiency.
6. Continuous assessment should be carried out based on implementation, understanding, and timely submission of assignments

List of Assignment

1. Install and configure OpenMP/MPI environment. Write a simple C/Python program to measure execution time of a sequential program and analyze performance.
2. Implement a CPU-based parallel program (e.g., array sum or matrix addition using OpenMP). Calculate speedup and efficiency for different number of threads.
3. Implement parallel matrix multiplication or sorting using OpenMP directives. Compare sequential vs parallel execution time.
4. Implement parallel matrix addition or multiplication using MPI (send, receive, broadcast). Analyze communication overhead.
5. Implement a computational problem (e.g., sorting, numerical computation, or array processing) using both OpenMP and MPI. Analyze and compare execution time, speedup, and scalability of the two programming models.
6. Write a simple CUDA program (vector addition or matrix addition). Compare CPU vs GPU execution time.
7. Implement a simple deep learning operation (matrix multiplication or convolution using PyTorch/TensorFlow with GPU). Measure training time with and without GPU.
8. Implement a basic MapReduce task (word count or log analysis) using Apache Spark.Analyze performance for different data sizes.
9. Mini Project: HPC-based AI or Data-Intensive Application : Develop and implement a small project using HPC concepts such as parallel programming, GPU computing, or distributed processing.

Savitribai Phule Pune University **Third Year - Artificial Intelligence and Data Science (2024 Pattern)**		
PEC363AID - Elective II Laboratory (AR/VR)		
Teaching Scheme	**Credits**	**Examination Scheme**
Practical : 02 Hours/Week	01	Term Work: 25 Marks Oral : 25 Marks

Guidelines for Laboratory Conduction - AR/VR

Group A

1. To understand the working of immersive hardware devices (Study HMDs, VR controllers, depth sensors, motion trackers.)
2. Creating a Basic 3D Scene in Unity :
   - Install Unity and required SDK
   - Create a 3D virtual room/environment.
   - Add geometric objects, lighting and textures.
   - Apply camera controls.
3. Geometric Transformations and Camera Navigation - Implement translation, rotation and scaling of objects.
4. Virtual Object Interaction using Controllers - To implement interaction techniques in VR.
5. Gesture-Based Interaction System - To design natural user interfaces (Capture hand gestures using webcam/sensors)
6. Marker-Based Augmented Reality Application - To implement basic AR tracking

Savitribai Phule Pune University **Third Year Artificial Intelligence and Data Science- 2024 Pattern**		
MDM331COM- Robotics and Automation		
Teaching Scheme	**Credits**	**Examination Scheme**
Tutorial : 01 Hour/Week **Practical**: 02 Hours/Week	01 01	Term Work : 50 Marks

Prerequisite Courses: Engineering Physics, Engineering Mathematics, Basics of Electrical Engineering, Basics of Electronics Engineering & Engineering Graphics.

Course Objectives: The course aims to:

1. To introduce various types of Robots and the functional elements of Robotics.
2. To impart knowledge of robot drive systems & educate on various sensors used in Robotic automation.
3. To introduce various types the end effectors.
4. To impart knowledge of basics of Robot Programming and robotic Applications

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1:** UNDERSTAND basic concepts of robotics.
- **CO2:** SELECT appropriate drive & sensors for Robotic applications.
- **CO3:** To COMPARE and SELECT robot and end effectors as per application.
- **CO4:** To know about the fundamentals of robot programming and applications.

Course Contents

Unit I - Fundamentals of Robotics (03 Hours)

Definitions of Industrial Robot, Type and Classification of Robots, Asimov‘s laws of robotics, Robot configurations, Robot Components, Robot Degrees of Freedom, Work volume and work envelope, Robot Joints and symbols, Robot Coordinates

Unit II - Sensors (03 Hours)

Transducers and sensors, Sensors in robotics, Principles and applications of the following types of sensors- Proximity Sensors, Photoelectric Sensors, Position sensors – Piezoelectric Sensor, LVDT, Resolvers, Encoders – Absolute and Incremental: - Optical, Magnetic, Capacitive, pneumatic Position Sensors, Range Sensors- Range Finders, Laser Range Meters, Touch Sensors, Force and torque sensors

Unit III - Industrial Automation and AI in Robotics (03 Hours)

Introduction to industrial automation
PLC basics and SCADA overview
IoT in automation systems
Basics of AI in robotics (computer vision, ML concepts)
Case studies: smart factories, autonomous robots

Unit IV Autonomous Navigation & Path Planning (03 Hours)

Environment Representation: Occupancy grids, topological maps, and configuration spaces.

Path Planning Algorithms: Dijkstra’s Algorithm, A* (A-star) search, and RRT (Rapidly-exploring Random Trees).

Localization: Odometry, sensor fusion (Kalman Filters), and particle filters.

Obstacle Avoidance: Potential field methods and reactive control loops.

Unit V Robotic Middleware & Industry 4.0 (03 Hours)

Robot Operating System (ROS): Architecture (Nodes, Topics, Services, Messages) and Workspace management.

Simulation Environments: Working with Gazebo and RViz for testing and validation.

Industrial Automation: Introduction to PLC (Programmable Logic Controllers) and SCADA systems.

Industry 4.0: Impact of IoT, Big Data, and AI in smart manufacturing and automation.

Learning Resources

Text Books:

1. Industrial Robotics – Mikell P. Groover, McGraw Hill
2. Robotics: Control, Sensing, Vision and Intelligence – McGraw Hill
3. Introduction to Robotics – Tata McGraw Hill
4. Robotics Engineering – Prentice Hall
5. Programmable Logic Controllers – McGraw Hill

Reference Books

Guidelines for Laboratory Conduction

List of Assignment - Group A (Any SIX)

1. Study of different robotic components, joints, links, and end effectors
2. Interfacing and control of DC motor using Arduino
3. Servo motor position control using PWM signals
4. Obstacle detection robot using ultrasonic sensor
5. Design and implementation of line follower robot
6. Interfacing IR sensors and proximity sensors with microcontroller
7. Stepper motor control for robotic arm movement
8. Basic PLC programming using ladder logic for industrial automation
9. IoT-based automation system using sensors and cloud monitoring

Group B - Mini-project: Design of a simple autonomous robotic or automation application - Suggested List

1. Smart warehouse robot
2. Automated parking system
3. IoT-based industrial monitoring system
4. Pick-and-place robotic arm
5. Smart conveyor automation
6. Home automation using sensors and actuators

Savitribai Phule Pune University **Third Year - Artificial Intelligence and Data Science (2024 Pattern)**		
VSE372COM - Solar Technology and Maintenance		
Teaching Scheme	**Credits**	**Examination Scheme**
Practical : 02 Hours/Week	01	Term Work: 50 Marks

Prerequisites: Basic knowledge of Physics (especially topics like electricity, magnetism semiconductors,light/energy concepts, Basic Electrical Engineering or Basic Electronics, Engineering Mechanics. Heat and
energy concepts

Course Objectives: The course aims to:

1. Apply Safely install, wire, and commission basic solar PV systems while measuring key performance parameters.
2. Analyze Break down the impact of environmental and operational factors on solar system efficiency and diagnose common faults.
3. Evaluate Judge the effectiveness of maintenance and troubleshooting procedures for solar PV components and systems.
4. Create Develop simple practical solutions or documentation for improving solar system performance in mini-projects.

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1:** Apply safe installation, wiring, commissioning, and performance measurement of basic solar PV systems.
- **CO2:** Analyze the impact of environmental/operational factors on solar PV efficiency and diagnose common faults.
- **CO3:** Evaluate the effectiveness of maintenance and troubleshooting procedures for solar PV components and systems.
- **CO4:** Create simple practical solutions or documentation for improving solar system performance via mini-projects.

Practical Assignments

1. Experiment No 1, 2 and 10 are compulsory.
2. Perform any 2 Experiments from 3 to 5
3. Perform any 3 Experiments from 6 to 9

1. Measurement of solar irradiance using pyrometer/lux meter at different times/angles. Real-World Assignment: Survey irradiance on your college rooftop for one day. Calculate daily energy generation for a 100W panel and suggest best installation time/angle for maximum output.
2. Plot I-V and P-V characteristics of solar PV module under varying light & temperature. Real-World Assignment: Simulate cloudy/rainy day conditions. Calculate module efficiency and estimate annual energy loss in Pune climate.
3. Survey and Comparative Analysis of Solar PV Installation Systems: Grid-Tied, Hybrid, and Off-Grid Configurations. Real-World Assignment: Survey 2–3 real solar installations (e.g., college rooftop, nearby home/business, or online/virtual

4. Series and parallel connection of PV modules, observe mismatch issues. Real-World Assignment: Design a small array for 12V/24V system (e.g., for laptop charging or lab fan). Calculate total power and suggest fuse/ diode protection for mismatch in a multi-panel rooftop installation.
5. Installation and wiring of standalone solar PV system (PV $\rightarrow$ Charge controller $\rightarrow$ Battery $\rightarrow$ Load/Inverter) Real-World Assignment: Prepare a complete wiring diagram and BOM for a 100W system to power a college water cooler or hostel room. Include safety earthing and cable sizing as per real IEC standards.
6. Preventive maintenance: Cleaning, visual inspection, corrosion/loose connection check. Real-World Assignment: Inspect any existing solar panel in college/hostel. Prepare a 6-month maintenance schedule with cost estimation (dust cleaning, tightening)
7. Grid-Related Maintenance Checks for Grid-Tied Solar PV Systems: Inverter Health, Performance Monitoring, and Fault Diagnosis. Real-World Assignment: Survey a real grid-tied installation, Prepare a maintenance schedule:Monthly inverter check, quarterly visual, annual professional inspection.
8. Mounting structure assembly: Rooftop/ground mount, tilt adjustment, stability check Real-World Assignment: Design a simple mounting frame for windy Pune conditions. Calculate wind load and suggest material/cost for a 5kW residential installation.
9. IoT-Based Real-Time Solar PV System Monitoring and Performance Dashboard.
10. Industrial Visit to Solar Energy Facility in Pune Region: Hands-On Learning of Solar PV System Operations and Maintenance

Test Books:

1. S.P. Sukhatme, Solar Energy
2. C.S. Solanki, Solar Photovoltaics
3. D.P. Kothari et al., Renewable Energy Sources
4. G.D. Rai, Non-Conventional Energy Sources
5. H.P. Garg, Solar Energy Utilization

Reference Books:

1. Solar Photovoltaic Technology and Systems: A Manual for Technicians, Trainers and EngineersAuthor: Chetan Singh Solanki.
2. Solar PV System: Design, Installation, Operation and MaintenanceAuthors: L. Ashok Kumar and K. Mohana Sundaram.
3. Solar Engineering of Thermal Processes, Photovoltaics and Wind (5th Edition) Authors: John A. Duffie, William A. Beckman (updated with Nathan Blair).
4. Principles of Solar Engineering (3rd Edition) Authors: D. Yogi Goswami, Frank Kreith, Jan F. Kreider

NPTEL Course:

1. Solar Photovoltaics: Fundamentals, Technology and Applications: https://onlinecourses.nptel.ac.in/noc
2. SkillCat or Other Free Solar Training (Installation Focus). https://www.skillcatapp.com/solar-installation-training

Savitribai Phule Pune University **Third Year - Artificial Intelligence and Data Sceince (2024 Pattern)**		
ELC381AID- Internship/On Job Training		
Teaching Scheme	**Credits**	**Examination Scheme**
Theory : 08 Hours/Week	04	Oral : 50 Marks

Internships are educational and career development opportunities, providing practical experience in a field or discipline. Internships are far more important as the employers are looking for employees who are properly skilled and having awareness about industry environment, practices and culture. Internship is structured, short-term, supervised training often focused around particular tasks or projects with defined time scales. Core objective is to expose technical students to the industrial environment, which cannot be simulated/experienced in the classroom and hence creating competent professionals in the industry and to understand the social, economic and administrative considerations that influence the working environment of industrial organizations. Engineering internships are intended to provide students with an opportunity to apply theoretical knowledge from academics to the realities of the field work/training.

Course Objectives: The course aims to:

1. To expose students to real-world industry practices.
2. To bridge the gap between academic learning and practical implementation.
3. Develop professional competency, ethics, communication, and teamwork skills.
4. To encourage self-learning and problem-solving abilities.
5. Encourage innovation, entrepreneurship, and research aptitude.

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1:** Apply theoretical knowledge to solve real-world engineering problems.
- **CO2:** Demonstrate technical competency in tools/technologies used in industry.
- **CO3:** Exhibit professional ethics and teamwork.
- **CO4:** Prepare technical reports and deliver effective presentations on industrial training experience.
- **CO5:** Analyze industrial processes and suggest feasible improvements or innovations.

Guidelines

1. Students should opt for a internship/JOT that would provide them to gain ample field knowledge in the relevant field of engineering such that theoretical knowledge gained in the class can be applied to solve the practical/ field problem.
2. Students must have to opt for technical internship after VI semester and before VII semester, preferably during summer break.
3. **Undergoing a training programme / Course at a particular organization for specified duration is NOT considered as summer internship**
4. However student can attend such programs mentioned in above to learn new tools for short duration that would help for solving the problem undertaken in the internship
5. Students should take a challenging task, may be a small portion, and apply the knowledge gained to solve it.

6. Internship can also involve data collection from different sources, including generating experimental data, collection of data from field etc. The data may be analyzed later on.
7. Different central and state government organizations, CSIR labs, premier institutions like IITs and IIMs, DRDO,public sector undertaking organizations, top IT companies may be considered for internships.
8. Student need to submit Synopsis, Permission letter and offer letter to Internship coordinator before proceeding to internship.
9. Internship completion will be considered only after submission of valid documents at the end of internship like Completion certificate, Report and presentation of work done, feedback from industry etc.
10. Student will appear for term work evaluation where he/she will present the work done before mentor(s) at the end of internship.

Suggested Internship Activities

- Students are expected to perform the following activities during internship:
- Phase I – Orientation and Requirement Study
  - Understanding organization structure
  - Study of workflow and operational processes
  - Requirement analysis and project allocation
  - Understanding tools and technologies used
- Phase II – Technical Learning and Development
  - Coding and implementation
  - Database design and integration
  - Software testing and debugging
  - API integration and deployment
  - Use of version control systems
  - Documentation practices
- Phase III – Project Execution
  - Module development
  - Testing and validation
  - Performance optimization
  - Client interaction (if applicable)
  - Team collaboration
- Phase IV – Documentation and Presentation
  - Preparation of internship report
  - Preparation of project demonstration
  - Final presentation and viva voce

Deliverables

- Internship Joining Report
- Weekly Logbook
- Mid-term Progress Report
- Supervisor Feedback (Initial)

Internship Structure

The internship may be carried out in any one of the following domains:

- Software Development
- Artificial Intelligence and Machine Learning
- Data Science and Analytics
- Cloud Computing and DevOps
- Cyber Security
- Web and Mobile Application Development
- IoT and Embedded Systems
- Networking and System Administration
- Automation and Robotics Software
- Research and Development
- Entrepreneurship and Startup Projects
- Government/NGO Technical Projects

Nature of Internship

Students shall undergo internship/training in one of the following:

- Registered companies / startups
- Government organizations
- Research institutions
- Recognized industry-academic collaborative projects
- Internships may be conducted in offline, online, or hybrid mode, subject to proper approval and verification.

Guidelines for Internship Report Writing

1. Preliminary Pages
    - Cover Page

- Certificate from Organization
- Certificate from Department
- Acknowledgement
- Abstract
- Table of Contents

2. Chapter 1 – Organization Profile
    - Company overview
    - Vision and mission
    - Products/services
    - Organizational structure
3. Chapter 2 – Problem Statement and Objectives
    - Project title
    - Need of project
    - Objectives
    - Scope
4. Chapter 3 – Technologies and Methodology
    - Software/hardware tools used
    - Development methodology
    - System architecture
    - Database design
5. Chapter 4 – Work Carried Out
    - Tasks completed
    - Screenshots/results
    - Challenges faced
    - Solutions implemented
6. Chapter 5 – Learning Outcomes
    - Technical learning
    - Professional skills acquired
    - Industry exposure
    - Future scope
7. Chapter 6 – Conclusion
    - Summary of work
    - Achievements
    - Suggestions

References : IEEE format references preferred
Appendices

- Source code snippets
- Certificates
- Additional screenshots

Learning Resources

Text Books:

1. W. J. King and James G. Skakoon , The Unwritten Laws of Engineering , ASME Press
2. Stuart Walesh, Engineering Your Future: The Professional Practice of Engineering
3. Eliyahu M. Goldratt, The Goal: A Process of Ongoing Improvement
4. AICTE Internship policy : AICTE Internship Policy: Guidelines & Procedures
5. AICTE Internship Portal : https://internship.aicte-india.org

Savitribai Phule Pune University, Pune

Maharashtra, India

Task Force for Curriculum Design and Development Third Year Artificial Intelligence and Data Science- 2024 Pattern

Programme Coordinator

Dr. Dipti D. Patil - Member, Board of Studies - Computer Engineering

Core Committee Members

Dr. Shraddha Pandit	Modern COE,Pune
Dr. Kalpana Metre	ITMBU,Vadodara,Gujrat
Dr. Vinay Nalawade	S B Patil College of Engineering,Indapur
Dr. Dikshendra Sarpate	ISBM College Of Engineering Pune
Dr. Girish Potdar	PICT,Pune

Compiled By

Dr. Vinay Nalawade , S B Patil College of Engineering,Indapur

Team Members for Course Design

Networks & Security	
Dr. Vishal Puranik	Ajeenkya DY Patil School of Engineering
Prof. Ankita Tidake	Ajeenkya DY Patil School of Engineering
Dr Mohammed Baig	Maulana Mukhtar Ahmad Nadvi technical Campus
Prof.Chandan Wagh	Dr D.Y.Patil College Of Engineering and Innovation, Pune
Prof.Pooja Kajale	DVVP COE Ahilyanagar
Prof. Dipali Shinde	PVG COE & SSD IOM Nashik
Mr.Tushar B. Kute	MITU Research, Pune

Natural Language Processing	
Dr. Vinay S. Nalawade	S B Patil College of Engineering,Indapur
Dr. Rama Gaikwad	Anantrao Pawar college of Engineering and Research
Dr. Shwetal Patil	MMIT
Dr. Sujeet More	Trinity College of Engineering and Research
Prof. Abha Pathak	Dr. D. Y. Patil College of Engineering and Innovation Varale
Prof.Anjali Phaltane	Dr.Vithalrao Vikhe Patil College of Engineering, Ahilyanagar
Mr. Anshuman Jadhav	Independent Consultant

Internet of Things	
Dr. Kalpana Metre	ITMBU,Vadodara,Gujrat
Dr Rohini Hanchate	Nutan Maharashtra Institute of Engineering & Technology
Prof. Charushila Patil	GGS College of Engineering And Research Centre, Nashik
Dr.Pramod Dhamdhere	MMIT Lohegaon Pune
Prof. Manasi D. Karajgar	DYPCOE, AKURDI, PUNE
Dr.Saloni N.Shah	SPCOET,Someshwar
Mr.Piyush Sonewar	Pixaflip Technologies Pvt.Ltd.,Pune

Deep Learning	
Dr. Minakshi Atre	PVG,Pune
Dr. Shraddha Pandit	Modern COE,Pune
Dr.Jitendra Musale	Anantrao Pawar college of Engineering and Research
Dr. Dipti D. Patil	Cummins College of Engineering for Women,Pune
Mr.Kedar Kulkarni	Apple Inc.,Austin, Texas

Cyber Security	
Dr. Chhaya Gosavi	Cummins COEW,Pune
Dr. Bhagyashree Dhakulkar	Ajeenkya DY Patil School of Engineering, Lohegaon Pune
Dr. Bajirao Shirole	Loknete Gopinathji Munde Institute of Engineering Education and Research, Nashik
Prof.Pradip Shendage	Vidya Pratishthan's Kamalnayan Bajaj Institute of Engineering,Baramati
Mr.Venkatta Gutulla	Independent Consultant

Computer Vision	
Dr. Vinay S. Nalawade	S B Patil College of Engineering,Indapur
Dr Aarti Dandavate	G.H.Raisoni College of Engineering & Management,Pune
Dr. Neeta Maitre	Cummins College of Engineering for Women,Pune
Dr Mrs Suvarna Pawar	Trinity Academy of Engineering,Pune
Prof.D.S.Bhadane	Ajinkya D Y Patil University,Pune
Prof.S.D.Pawar	Vidya Pratishthan's Kamalnayan Bajaj Institute of Engineering,Baramati
Mr.Pradip D.Kulkarni	Visdin Solutions

Gaming & Animation	
Dr. Nuzhat Faiz Shaikh	Wadia COE
Prof. Harsha Sonune	Cummins College of Engineering for Women,Pune
Dr.Shabnam Sayyad	AISSMS COE,Pune
Dr.Poonam Railkar	SKNCOE,Pune
Prof.Mahesh Shinde	VIT,Pune
Mr.Yogesh Surwase	FirebirdVR,Pune

Blockchain	
Dr. Kalpana Metre	ITMBU,Vadodara,Gujrat
Mrs Geetanjali Mohole	JES Institute of Technology Management and Research, Nashik
Dr Pravin Latane	Dr D Y Patil College of Engineering and Innovation
Dr Shivaji R. Lahane	GES R. H. Sapat CoE MSR, Nashik
Prof. Varsha Mhaske	Shivnagar Vidya Prasarak Mandal's College of Engineering, Malegaon
Dr. Nitin Shivale	JSPM 's Bhivarabai Sawan institute of technology and research, Pune
Mr.Shriyash Pande	Avalabs

Big Data Engineering	
Dr. Dikshendra Sarpate	ISBM College Of Engineering Pune
Dr.Renuka Kajale	Nutan Maharashtra Institute of Engineering and Technology,Pune
Dr. Shraddha Khonde	Wadia College of Engineering,Pune
Prof. Mangal Kale	Dr. Vithalrao Vikhe Patil COE, Ahmednagar
Prof. Vivek Badgujar	JES Institute of Technology Management & Research, Nashik
Prof. Alka Kumbhar	GSMCOE balewadi
Dr.Javed Shaikh	Capital Numbers Infotech PVT LTD

High Performance Computing	
Dr. Girish Potdar	PICT,Pune
Dr.Abhinav Thorat	Amrutvahini College of Engineering, Sangamner
Dr. Dewanand Meshram	RMD Sinhgad school of Engineering,Pune
Prof.Anand Gharu	MET's Institute of Engineering, Bhujbal Knowledge City, Nashik
Prof. Prasad Lahare	PVG,s COE & SSD IOM,Nashik
Prof. Sandeep Jadhav	Sanghavi College of Engineering, Nashik
Mr. Trambak Pawar	Telus Digital

Ethical Hacking	
Dr. Dikshendra Sarpate	ISBM College Of Engineering, Pune
Dr Prakash Gadekar	Marathwada Mitra Mandals College of Engineering,Pune
Prof.Suchitra S.Pakale	Cummins College of Engineering for Women,Pune
Prof. Sagar Dhanake	Dr. D Y Patil College of Engineering and Innovation
Prof. Vishal Naykwadi	VIIT Pune
Prof Meenal Kamlakar	Cummins College of Engineering for Women,Pune
Mr.Sachin Kulkarni	flydubai

Quantum AI	
Dr. Shraddha Pandit	Modern COE,Pune
Prof. Prashant Koli	R. H. Sapat College of Engineering,Nashik
Dr.Archana Chaudhari	Symbiosis Institute of Technology
Dr.Ritu Dudhamal	MIT,Alandi
Dr. Darshan Medhane	Kakasaheb Wagh Institute of Engineering,Nashik
Dr.Manjusha Tatiya	Indira College of Engineering and Management,Pune

Reinforcement Learning	
Dr.Minakshi Pradeep Atre	PVG,Pune
Prof.Geeta M Kodabagi	Ajeenkya DY Patil school of engineering Lohegaon pune
Dr.Prajwal Gaikwad	AISSMS IOIT
Dr.Ravindra Apare	PVG's COETM,Pune
Prof. Hitendra S. Khairnar	Cummins College of Engineering for Women,Pune
Prof.Dipak Pawar	ISBM College of Engineering Pune
Mr.Sushrut Guruji	Wipro Technology Pune

Machine Learning	
Dr. Shraddha Pandit	Modern COE,Pune
Prof. Pradip Ghorpade	VP's Kamalnayan Bajaj Institute of Engineering & Technology
Dr. Sulochana Sonkamble	JSPM Narhe Technical Campus, Pune
Dr. Girisha Bombale	SNDCOE & RC,Yeola
Prof. Pradnya Bachhav	GGS College of Engineering & Research Centre, Nashik
Prof.Varsha P.Chavan	Anantrao Pawar college of Engineering and Research
Mr.Nachiket Kulkarni	AutoDesk, Pune

Chairman

Dr. Nilesh Uke - Board of Studies Computer Engineering

Savitribai Phule Pune University, Pune

Dean

Dr. Raosaheb Latpate - Dean – Science and Technology
Savitribai Phule Pune University, Pune

***