॥ यः क्रियावान् स पण्डितः ॥

SAVITRIBAI PHULE PUNE UNIVERSITY

(Formerly University of Pune)

DATA SCIENCE

SYLLABUS

T. E. COMPUTER SCIENCE AND ENGINEERING (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

PSO	Program Specific Outcomes
ELC	Experiential Learning Course
ELC	Experimental Learning
MDM	Multidisciplinary Minor
OE	Open Elective
PCC	Programme Core Course
PEC	Programme Elective Course
PEO	Programme Educational Objectives
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 Computer Science and Engineering (Data Science) syllabus effective from the Academic Year 2025-26. The present curriculum will be implemented for Third Year of Engineering from the academic year 2025-26. Subsequently this will be carried forward for BE in AY 2027-28.
Computer Engineering is a dynamic discipline that lies at the intersection of electrical engineering and computer science. It provides the foundation for the design, development, and application of computer systems and other computing devices. This curriculum is designed to provide students with a comprehensive understanding of the fundamental principles, theories, and practices of computer engineering, while also preparing them for the ever-evolving technological landscape.

The revised syllabus falls in line with the objectives of NEP-2020, Savitribai Phule Pune University, AICTE New Delhi, UGC, and various accreditation agencies by keeping an eye on the technological developments, innovations, and industry requirements.Wherever possible additional resource links of platforms such as NPTEL, Swayam are appropriately provided at the end of each course. Learners are now getting sufficient time for self learning either through online courses or additional projects for enhancing their knowledge and skill sets. Learners can be advised to take up online courses, on successful completion they are required to submit certification for the same. This will definitely help learners to facilitate their enhanced learning based on their interest.

This curriculum is the result of extensive consultation with academic experts, industry professionals, and alumni to ensure relevance and excellence. It is designed not only to meet the current industry standards but also to prepare students for higher studies and research in the field of computer engineering.

We hope that this curriculum will inspire students to become competent professionals, responsible citizens, and contributors to the technological advancement of society.

Dr. Nilesh Uke
Chairman
Board of Studies-Computer Engineering

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:** Computing and Analytical Proficiency : Graduates will be proficient in designing efficient algorithms, developing intelligent software systems, and utilizing modern computing tools and technologies using Algorithms, System Software, Machine Learning, Artificial Intelligence, Web Applications, Big Data Analytics and Networking for large scale data processing and analytical tasks.
- **PSO2:** Data-Driven Solution Development - Graduates will apply principles of data science, including statistical analysis and machine learning to extract actionable insights and develop innovative solutions for complex engineering and societal challenges.
- **PSO3:** Successful Career and Entrepreneurship- Graduates will exhibit the ability to conduct research, foster innovation, and engage in lifelong learning in data science, contributing to advancements in artificial intelligence, big data, cloud computing, and interdisciplinary fields.

Programme Educational Objectives (PEO)

Program Educational Objectives (PEOs): 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	Attainment of key principles and practices of computation, mathematics and basic principles of engineering to ensure that graduates are able to apply their software development skills in design and implementation of practical systems consisting of software and/or hardware components.
PEO2	Problem solving skills and Ethics	Analyze real-life problems and impart science-based engineering education to develop professional skills that will prepare the students for immediate employment in the industry.
PEO3	Professionalism and Lifelong Learning	Imbibe lifelong learning, professional and ethical attitude for embracing global challenges and make positive impact on environment and society.

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.

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)

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 (CCE) 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 Comprehensive Continuous Evaluation (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 Study	05 Marks	Units 3 & Unit 4 (5 Marks/Unit)

Format and Implementation of Comprehensive Continuous Evaluation (CCE)

- **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:** 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.

Curriculum Structure - Semester V

Third Year Engineering (2024 Pattern) – Computer Science and Engineering (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
PCC301CDS	Artificial Intelligence	PCC	3	-	-	30	70	-	-	-	100	3	-	-	3
PCC302CDS	Computer Networks	PCC	3	-	-	30	70	-	-	-	100	3	-	-	3
PCC303CDS	Theory of Computation	PCC	3	-	-	30	70	-	-	-	100	3	-	-	3
PCC304CDS	Artificial Intelligence Laboratory	PCC		-	2	-	-	25	25	-	50	-	-	1	1
PCC305CDS	Computer Networks Laboratory	PCC		-	4	-	-	25		25	50	-	-	2	2
PEC321CDS	Elective I	PEC	3	-	-	30	70	-	-	-	100	3	-	-	3
PEC322CDS	Elective I Lab	PEC		-	2	-	-	25	-	-	25	-	-	1	1
MDM331CDS	Next Generation IoT	MDM		2	2	-	-	25	-	25	50	-	2	1	3
	Open Elective	OE	2	-	-	15	35		-	-	100	2	-	-	2
ELC342CDS	Technical Seminar	ELC		-	2	-	-	-	-	25	25	-	-	1	1
Total			**14**	2	**12**	**135**	**315**	**100**	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

Program Elective I	
PEC321CDSA	Natural Language Processing
PEC321CDSB	Cloud Computing
PEC321CDSC	Soft Computing
PEC321CDSD	Design and Analysis of Algorithm

Curriculum Structure - Semester - VI

Third Year Engineering (2024 Pattern) – Computer Science and Engineering (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
PCC351CDS	Machine Learning	PCC	2	-	-	30	70	-	-	-	100	3	-	-	2
PCC352CDS	Data Science and Big Data Analytics	PCC	3	-	-	30	70	-	-	-	100	3	-	-	3
PCC353CDS	Machine Learning Laboratory	PCC	-	-	4	-	-	25		25	50	-	-	2	2
PCC354CDS	Data Science and Big Data Analytics Laboratory	PCC	-	-	2	-	-	25	25	-	50	-	-	1	1
PCC361CDS	Elective II	PEC	3	-	-	30	100	-	-	-	100	3	-	-	3
PEC362CDS	Elective III	PEC	3	-	-	30	70	-	-	-	100	3	-	-	3
PEC363CDS	Elective II Laboratory	PEC		-	2	-	-	25	-	25	50	-	-	1	1
MDM371CDS	Robotics and Automation	MDM		1	2	-	-	50	-	-	50	-	1	2	2
VSE372CDS	Solar Technology and Maintenance	VSE	-	-	2	-	-	50	-	-	50	-	-	1	1
ELC381CDS	Internship/OJT	ELC	-	-	8	-	-	-	50	-	50	-	-	4	4
Total			**11**	1	20	**120**	310	**175**	75	50	**700**	**12**	1	11	**22**

Program Elective II	
PEC361ACDS	Cybersecurity and Data Privacy
PEC361BCDS	Blockchain Technology
PEC361CCDS	Augumented Reality and Virtual Reality
PEC361DCDS	Quantum Artificial Intelligence

Program Elective III	
PEC362ACDS	Generative AI
PEC362BCDS	Ethical Hacking
PEC362CCDS	User Interface and User Experience Design
PEC362DCDS	Computer Vision

Savitribai Phule Pune University, Pune

Maharashtra, India

TE - Computer Science and Engineering (Data Science)

2024 Pattern

Semester V

With effect from Academic Year 2026-27

Savitribai Phule Pune University Third Year_Computer Science and Engineering(Data Science)		
PCC301CDS- Artificial Intelligence		
Teaching Scheme	**Credits**	**Examination Scheme**
Theory : 03 Hours/Week	03	CCE : 30 Marks End-Semester: 70 Marks

Prerequisite Courses : Data Structures

Companion Course: Artificial Intelligence Lab

Course Objectives:

- To introduce fundamental concepts of Artificial Intelligence, intelligent agents, and ethical considerations in AI.
- To develop the ability to model and solve problems using state-space search and heuristic algorithms.
- To enable students to design solutions for adversarial and constraint satisfaction problems using appropriate AI techniques.
- To impart knowledge representation skills using propositional and first-order logic with inference mechanisms.
- To equip students with AI planning techniques and relate them to real-world industrial applications.

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

- CO1: Explain fundamental concepts of Artificial Intelligence, Agents and ethical implications of AI in real-world scenarios.
- CO2: To formulate real-world problems as state-space search models and implement appropriate heuristic, local, and online search algorithms
- CO3: To model real-world competitive and constraint-based problems and implement adversarial search algorithms and CSP techniques
- CO4: To represent real-world knowledge using propositional and first-order logic and apply Inference techniques and derive logical conclusions in knowledge-based AI systems.
- CO5: To apply AI planning techniques for solving classical planning problems and Relate them to real-world industrial AI applications.

Course Contents

Unit I - Introduction to AI and Intelligent Agents (7 hours)

Introduction to Artificial Intelligence, Foundations of Artificial Intelligence, History of Artificial, Intelligence, Limits of AI, Ethics of AI, Future of AI, AI Components, AI Architectures, Intelligent Agents, Agents and Environments, Good Behavior: Concept of Rationality, Types of Agents, Nature of Environments, Structure of Agents.

Case Study: Autonomous Taxi Agent – Waymo One, AI in Healthcare – IBM Watson for Oncology

Unit–II: Problem Solving : State Space Approach and Search Strategies (9 hours)

State Space Search: Tower of Hanoi. Informed (Heuristic) Search Strategies: Introduction to Greedy BFS, A* Search, Iterative-deepening, Heuristic Functions. Local Search and Optimization Problems: Hill-climbing search, Simulated annealing, Local beam search. Online Search Agents and Unknown Environments: Online search problems.

Case Study: Warehouse robots (Amazon Kiva) and self-driving cars, Logistics and Routing: Traveling Salesman Problem, Google DeepMind – AI for Energy Efficiency in Data Centers

Unit–III: Adversarial Search, and Game theory (8 hours)

Optimal Decisions in Games, Heuristic Alpha–Beta Tree Search, Monte Carlo Tree Search, Stochastic Games, Partially Observable Games, Limitations of Game Search Algorithms, Constraint Satisfaction Problems (CSP), Constraint Propagation: Inference in CSPs, Backtracking Search for CSPs

Case Study: AlphaGo – AI in Strategic Board Games, Strategic Decision-Making in Imperfect-Information Games-- Libratus, Adversarial Search and Constraint Reasoning in Computer Chess - IBM deep blue

Unit-IV : Knowledge representation using Logical Formalisms, Propositional and First order Predicate Calculus (8 hours)

Introduction to Logical Formalisms: Role of logic in Artificial Intelligence, Knowledge-based agents Syntax and semantics of logical systems

Propositional Logic – Basics and Inference: Propositional symbols and well-formed formulas, Logical connectives, Inference rules: Modus Ponens, Modus Tollens, Resolution

First Order Predicate Logic – Fundamentals: Motivation for First Order Logic, Quantifiers (∀, ∃), Well-formed formulas, Translating natural language into FOL

Inference in First Order Predicate Logic – Fundamentals: Motivation for First Order Logic, Quantifiers (∀, ∃), Well-formed formulas, Translating natural language into FOL

Case Study: Medical Expert System – MYCIN, Knowledge-Based Reasoning in Intelligent Search Systems:- AI-Based Rule Engine in Google Search

Unit V : Planning and Industrial Applications of AI (8 hours)

Planning: Overview, An example Domain The Blocks world, The components of planning system, Goal stack planning, Nonlinear planning using constraint posting, Hierarchical planning

Industrial Application of AI : AI in Healthcare, AI in Finance, AI in Retail, AI in Agriculture, AI in Education, AI in Transportation, AI in Experimentation and Multi-disciplinary research

Case Study: Case Study — AI-Driven Supply Chain & Production Planning (Manufacturing)

Learning Resources

Text Books:

1. Stuart Russell and Peter Norvig, "Artificial Intelligence: A Modern Approach", Third edition, Pearson, 2003, ISBN :10: 0136042597
2. Elaine Rich,Kevin Knight and Nair,"Artificial Intelligence",TMH,ISBN-978-0-07-008770-5
3. Saptarsi Goswami, Amit Kumar Das and Amlan Chakrabarti, "AI for Everyone – A Beginner's Handbook for Artificial Intelligence", Pearson, 2024
4. Deepak Khemani, "A First Course in Artificial Intelligence", McGraw Hill Education(India), 2013, ISBN : 978-1-25-902998-1

Reference Books

1. Patrick Henry Winston, "Artificial Intelligence", Addison-Wesley Publishing Company, ISBN: 0-201-53377-4
2. Dr.Nilakshi Jain,"Artificial Intelligence,As per AICTE: Making a System Intelligent",Wiley publication, ISBN: 9788126579945
3. Nilsson Nils J , "Artificial Intelligence: A new Synthesis", Morgan Kaufmann Publishers Inc. San Francisco, CA, ISBN:978-1-55-860467-4
4. Dr. Lavika Goel, "Artificial Intelligence: Concepts and Applications", Wiley publication, ISBN:97881265

e-Books:

1. http://repo.darmajaya.ac.id/5094/1/Lecture-AI.pdf
2. https://www.freebookcentre.net/ComputerScience-Books-Download/Digital-notes-on-Artificial-Intelligence.html

3. https://www.kdnuggets.com/10-free-artificial-intelligence-books-for-2025

MOOC :

- Artificial Intelligence: Knowledge Representation And Reasoning By Prof. Deepak Khemani, IIT Madras https://onlinecourse
- An Introduction to Artificial Intelligence By Prof. Mausam, IIT Delhi https://nptel.ac.in/courses/106102220

Savitribai Phule Pune University Third Year - Computer Science and Engineering(Data Science) (2024 Pattern)		
PCC302CDS- Computer Networks		
Teaching Scheme	**Credits**	**Examination Scheme**
Theory : 03 Hours/Week	03	CCE : 30 Marks End-Semester: 70 Marks

Prerequisite Courses : Digital Electronics and Logic Design, Fundamentals of Internet of Things

Companion Course: Computer Network and Security Laboratory

Course Objectives:

1. Introduce fundamental concepts of networking, hardware, software, and reference models.
2. Develop knowledge of physical, data link, network, transport, and application layers with emphasis on design issues, services, and protocols.
3. Equip students with the ability to analyze and compare routing, error control, congestion control, and quality of service mechanisms.
4. Familiarize students with widely used protocols such as TCP, UDP, IP, HTTP, DNS, SMTP, and emerging technologies in multimedia and wireless networks.
5. Strengthen problem-solving skills through case studies and exemplars that connect theoretical concepts to real-world applications.

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

- CO1: Explain the fundamental concepts of computer networks, layered architecture, and physical transmission media.
- CO2: Apply error detection, correction, and reliable data transfer techniques at the data link layer.
- CO3: Analyze routing algorithms, addressing schemes, and Internet protocols at the network layer.
- CO4: Compare and evaluate transport layer protocols (TCP, UDP, SCTP, RTP) and mechanisms for congestion control and QoS.
- CO5: Demonstrate understanding of application layer protocols (HTTP, DNS, Email, FTP, TELNET, DHCP, SNMP) and emerging network applications with security.

Course Contents

Unit I - : Introduction to Computer Networks and Physical Layer (09 Hours)

Introduction to computer networks; uses of computer networks – business applications, home applications, mobile users ; network hardware – PAN, LAN, MAN, WAN and internetworks; network software – protocol hierarchies, design issues for layers, connection-oriented and connection-less services, service primitives, relationship between services and protocols; reference models – OSI and TCP/IP models. Physical Layer: guided transmission media; wireless transmission; telephone system; narrowband and broadband communication systems. Introduction to Network Security – Security goals: Confidentiality, Integrity, Availability (CIA triad); basic threats and vulnerabilities in communication systems.

Case Study: Comparison of OSI vs TCP/IP in real-world networks, case study on broadband vs narrowband communication in India.

Unit–II: Data Link Layer (9 Hours)

Data Link Layer – services provided to the network layer, framing and addressing; design issues of the data link layer – error control, flow control and reliable data transfer; error detection and correction techniques – parity, checksum, CRC

and basic error correction concepts; data link layer protocols – elementary protocols and Stop-and-Wait protocol; sliding window protocols – pipelining, Go-Back-N and Selective Repeat protocols; example data link layer technologies – packet over SONET. Security at Data Link Layer – Error detection vs data integrity, introduction to secure framing concepts; link-layer security basics.

Case Study: Case study on CRC error detection in Ethernet, SONET backbone deployment in telecom networks.

Unit–III: Network Layer (9 Hours)

Network Layer – Services & Design Issues: connectionless service, connection-oriented service, QoS support, error control, flow control, store-and-forward switching, congestion control, reliability, internetworking challenges; Routing Algorithms: shortest path routing, flooding, distance vector routing, link state routing; Internet Architecture & Protocols: IP addressing (IPv4 classes, CIDR, subnetting), IPv4 vs IPv6, IP datagram, ICMP, ARP, RIP, OSPF. Network Layer Security – IP Security (IPsec): Authentication Header (AH), Encapsulating Security Payload (ESP); secure routing concepts; basics of firewalls and packet filtering.

Case Study: Case study on IPv6 adoption in ISPs, routing comparison between RIP and OSPF in enterprise networks.
Case Study: IPv6 adoption with IPsec support, firewall deployment in enterprise networks.

Unit-IV: Transport Layer Protocols (9 Hours)

Process to Process Delivery, Services, Socket Programming. Elements of Transport Layer Protocols: Addressing, Connection establishment, Connection release, Flow control and buffering, Multiplexing, Congestion Control. Transport Layer Protocols: TCP and UDP, SCTP, RTP, Congestion control and Quality of Service (QoS), Differentiated services, TCP and UDP for Wireless networks. Transport Layer Security – SSL/TLS protocols, secure socket communication, security issues in TCP/UDP, congestion control in secure environments.

Case Study: Case study on TCP congestion control in 4G/5G networks, RTP in video conferencing applications.

Unit V: Application Layer (9 Hours)

Introduction – principles of application layer, client-server and peer-to-peer models; Web and HTTP – request/response, persistent vs. non-persistent connections, cookies, caching, performance; DNS – hierarchy, resource records, name resolution, caching, security issues; Email – SMTP, MIME, POP3, IMAP, webmail, message format, security; FTP – basics, file transfer process, legacy relevance; TELNET – remote login, limitations, legacy use; DHCP – dynamic host configuration, IP allocation, management; SNMP – network management, monitoring, MIBs, security considerations; Emerging Topics – multimedia applications, RTP for streaming, peer-to-peer applications, cloud-based services. Application Layer Security – HTTPS, Secure Email (PGP, S/MIME), DNS Security (DNSSEC); Network Attacks – DoS/DDoS, Phishing, Spoofing, Man-in-the-Middle; Introduction to Firewalls, Intrusion Detection Systems (IDS), Intrusion Prevention Systems (IPS); Security in cloud and multimedia applications.

Case Study: Case study on DNS security attacks (DNS spoofing), HTTP caching in CDNs, SMTP in enterprise email systems.

Learning Resources

Text Books:

1. Andrew S. Tanenbaum, David J. Wetherall – Computer Networks, 5th Edition, Pearson Education.
2. Behrouz A. Forouzan – Data Communications and Networking, 5th Edition, McGraw Hill.
3. William Stallings – Data and Computer Communications, 10th Edition, Pearson Education.

Reference Books

1. Kurose, James F., Ross, Keith W. – Computer Networking: A Top-Down Approach, 8th Edition, Pearson.
2. Peterson, Larry L., Davie, Bruce S. – Computer Networks: A Systems Approach, 5th Edition, Morgan Kaufmann.

e-Books:

1. https://people.cs.clemson.edu/~jmarty/courses/kurose/KuroseCh1-2.pdf
2. http://eti2506.elimu.net/Introduction/Books/Data Communications and Networking By Behrouz A.Forouzan.pdf
3. http://intronetworks.cs.luc.edu/current/ComputerNetworks.pdf
4. https://www.tutorialspoint.com/data_communication_computer_network/data_communicati on_computer_network_tutor

MOOC :

- nptel.ac.in/courses/106/105/106105183
- nptel.ac.in/courses/106/105/106105080
- nptel.ac.in/courses/106/105/106105081
- nptel.ac.in/courses/106/106/106106091
- nptel.ac.in/courses/106/105/106105031
- https://www.mooc-list.com/tags/computer-networking
- https://www.coursera.org/courses?query=computer%20network

Savitribai Phule Pune University Third Year_Computer Science and Engineering(Data Science) (2025 Pattern)		
PCC303CDS- Theory Of Computations		
Teaching Scheme	**Credits**	**Examination Scheme**
Theory : 03 Hours/Week	03	CCE : 30 Marks End-Semester: 70 Marks

Prerequisite Courses : Discrete Mathematics

Course Objectives:

1. Recall and understand the basics of mathematical concepts and machines
2. To design deterministic/nondeterministic automata and interconversion.
3. Design regular expressions and prove non-regular languages using the Pumping Lemma and Myhill–Nerode Theorem.
4. Design CFGs, PDAs, and Turing Machines for given languages.
5. Understand the basic concepts of computability theory.

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

- CO1: Apply the knowledge of basics of mathematics and logic for designing Finite Automata and its variants.
- CO2: Construct regular expression to present regular language and understand pumping lemma and Myhill–Nerode Theorem for RE
- CO3: Design Context Free Grammars and learn to simplify the grammar.
- CO4: Construct appropriate computational models to solve given problems using Pushdown Automaton model
- CO5: Able to design Turing Machines for various computational problems and understand different classes of problems, classify and analyze them

Course Contents

Unit I - : Introduction to Formal Languages and Finite Automata (08 Hours)

Basic Concepts: Finite and infinite set, Symbols, Strings : Empty String, Substring of a string, Concatenation of strings, Language :Formal Language Definition, Finite representation of languages , Operations on languages: Union, Concatenation, Kleene star and Kleene plus, Concept of Basic Machine.

FA without output: Finite State Machines(FSM), Deterministic and Nondeterministic FA(DFA & NFA), epsilon NFA, Conversion of NFA with epsilon moves to NFA, Conversion of NFA to DFA, and Conversion of NFA with epsilon moves to DFA, Minimization of DFAs.

FA with output: Moore and Mealy machines - Definition, Construction, Inter- Conversion.

Case Study: FSM for vending Machine, Spell checker, Finite Automata in ATM PIN Validation System

Unit–II: Regular Expressions and Languages (6 Hours)

Introduction, Operators of RE, Precedence of operators, Algebraic laws for RE, Language to Regular Expressions, Equivalence of two REs, Kleene's theorem. Conversions: RE to NFA, DFA, DFA to RE using Arden's theorem, Pumping Lemma for Regular languages, Closure(union, intersection, complementation, concatenation, Kleene closure) and Decision properties of Regular languages(Membership, Emptiness,Finiteness and Infiniteness) The Myhill–Nerode Theorem

Case Study: RE for variable name validation, RE to match a specific word from given string ,Regular Expressions in Email Validation System

Unit–III: Context Free Grammars (CFG) and Languages (09 Hours)

Formal Definition of Context Free Grammar, Sentential form, Derivation and Derivation Tree/ Parse Tree, Context Free Language (CFL), Ambiguous Grammar, writing grammar for language, Simplification of CFG: Eliminating ϵ-productions, unit productions, useless production, useless symbols, and Normal Forms- Chomsky normal form, Greibach normal form, Pumping Lemma for CFG, Closure properties of CFL, Chomsky Hierarchy, Applications of CFG:-Palindromes, Parenthesis Match ,Parser, Markup languages, XML and Document Type Definitions.

Case Study: Grammar Design for Arithmetic Expressions, Designing a Grammar for a Simple Programming Language Construct

Unit-IV: Push Down Automata (09 Hours)

Introduction, Formal definition of PDA, Equivalence of Acceptance by Final State & Empty stack, Non-deterministic PDA (NPDA), PDA & Context Free Language, Equivalence of PDA and CFG, PDA vs CFLs. Applications of PDA,Introduction to Post Machine,

Case Study: Applying PDA for Top-Down Parsing, Bottom-up Parsing Pushdown Automata (PDA) in Compiler Syntax Checking,XML / HTML Tag Validation

Unit V: Turing Machine and Computability Theory (08 Hours)

Turing machine (TMs): Basic model, definition, and representation, TM Instantaneous Description, Transition Function, Language accepted TM, Deterministic Turing Machines (DTM), and Construction of DTM. Universal Turing Machine (UTM), Church-Turing hypothesis, Comparison between FA, PDA and TM. Turing Machine Halting Problem.

Decidable Problems and Undecidable Problems, Church-Turing Thesis.

Reducibility: Undecidable Problems that are recursively enumerable, A Simple Undecidable.

Complexity Classes: Time and Space Measures, The Class P, Examples of problems in P, The Class NP, Examples of problems in NP, P Problem Versus NP Problem, NP-completeness and hard Problems.

Case Study: Application of Turing Machine for Language Recognition, Comparative Study of Variants of Turing Machines, Analysis of the Halting Problem

Learning Resources

Text Books:

1. Hopcroft J., Motwani R., Ullman J., "Introduction to Automata Theory, Languages andComputations", Third edition, 2008, Pearson Education Asia. ISBN: 9788131720479.
2. Michael Sipser, ”Introduction to The Theory of Computation”, Third edition, 2017 Thomson Course Technology, ISBN: 9781131525296.

Reference Books

1. Daniel Cohen., “Introduction to Computer Theory”, Second edition, 2011, Wiley Publications (India) ISBN: 9788126513345.
2. H.R. Lewis, C. H. Papadimitriou, ”Elements of the Theory of Computation”, Second edition, 2006, Prentice Hall Inc. ISBN: 8131703878.
3. John C Martin. "Introduction to Language and Theory of Computation", Third edition, 2012, Tata McGraw- Hill, ISBN: 978007660489.
4. K. L. P Mishra, N. Chandrashekaran (2003), Theory of Computer Science-Automata Languages and Computation, 2nd edition, Prentice Hall of India, India.
5. Vivek Kulkarni, “Theory of Computation”, Oxford University Press, ISBN 0-19-808458

e-Books:

1. https://cglab.ca/~michiel/TheoryOfComputation/TheoryOfComputation.pdf 2.
2. https://www.cs.virginia.edu/~robins/Sipser_2006_Second_Edition_Problems.pdf

MOOC :

- Theory of Computation - IIT Kanpur whttps://nptel.ac.in/courses/106104148

Savitribai Phule Pune University Third Year - Computer Science and Engineering (Data Science) (2025 Pattern)		
PCC304CDS- Artificial Intelligence Laboratory		
Teaching Scheme	**Credits**	**Examination Scheme**
Practical : 02 Hours/Week	01	Term Work: 25 Marks Practical: 25 Marks

Course Objectives:

1. To provide a comprehensive understanding of the fundamental concepts, principles, and methodologies of Artificial Intelligence.
2. To equip students with the ability to design and implement core AI algorithms for problem-solving, search, reasoning, and learning.
3. To develop analytical and practical skills for modeling and solving real-world problems using appropriate AI techniques.

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

- CO1: Explain and differentiate fundamental concepts, components, and techniques of Artificial Intelligence, including intelligent agents and AI methodologies.
- CO2: Design and implement AI algorithms for state-space search, heuristic search, reasoning, and basic learning tasks using programming tools.
- CO3: Model real-world problems as AI problems and apply appropriate AI techniques to obtain optimal or near-optimal solutions.
- CO4: To apply AI planning techniques for solving solve classical planning problems and relate them to real-world industrial AI applications.

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

List of Assignments (Part A) - Any FIVE	
1	To design and implement a simple reflex agent for the Vacuum Cleaner world and analyze its rational behavior in a given environment.
2	To implement a model-based intelligent agent capable of navigating a grid environment with obstacles using internal state representation.
3	Implement A* Search for 8 puzzle problem
4	Implement Adversarial Search with Alpha-Beta pruning
5	Implement Tower of Hanoi by State Space Search
6	To implement a Sudoku solver using backtracking and constraint propagation techniques.
7	Implement Local Beam Search for Traveling Salesman Problem.

List of Assignments (Part B) - 8,9,10 mandatory and Any SIX from 11 to 15	
8	Use BFS to solve a planning problem. Examples can be 8 puzzle problem, Robot path planning, Blocks world etc.
9	Use DFS for problem solving. Examples can be water jug problem , Missionaries and Cannibals ,maze solving, blocks world etc.
10	To implement Backtracking search to solve the 8-Queens constraint satisfaction problem.
11	To develop a simple chatbot using predefined rules and pattern matching.
12	To design an intelligent agent that guesses a number using feedback (higher/lower).
13	To implement Goal Stack Planning for solving the Blocks World problem and achieve a specified goal configuration.
14	To develop a game-playing agent using the Minimax algorithm for optimal decision-making.
15	To optimize adversarial search by implementing Alpha-Beta pruning and compare its efficiency with Minimax.

List of Assignments (Part C) - Mini-Project (In Team of 3-4 Students) - Any ONE	
16	Build unbeatable Tic-Tac-Toe AI using basic minimax to depth 9. Play 10 games vs random opponent
17	Medical Diagnosis System (Rule-Based Expert System) using forward and backward chaining
18	To develop a simple rule-based recommendation system (e.g., movie or book suggestion system).

Savitribai Phule Pune University Third Year - Computer Science and Engineering(Data Science)(2025 Pattern)		
PCC305CDS- Computer Networks Laboratory		
Teaching Scheme	**Credits**	**Examination Scheme**
Practical : 02 Hours/Week	01	Term Work: 25 Marks Oral: 25 Marks

Prerequisite Courses : Digital Electronics and Logic Design, Discrete Mathematics, Computer Organization & Microprocessor, Internet of Things

Companion Course: Computer Network Laboratory

Course Objectives:

1. Provide hands-on experience with computer networking concepts.
2. Familiarize students with network models, protocols, and tools.
3. Develop skills in analyzing and simulating network behavior.
4. Encourage practical implementation of routing, addressing, and error detection techniques.
5. Build competence in using tools like Packet Tracer and Wireshark for network configuration and analysis.

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

- CO1: Compare and analyze OSI and TCP/IP reference models, topologies, transmission media, and LAN/WAN configurations using simulation tools.
- CO2: Apply error detection and correction techniques, subnetting, and IP addressing schemes to design efficient and reliable networks.
- CO3: Simulate and implement transport layer protocols (Go-Back-N, Selective Repeat, TCP/UDP sockets) and client-server applications such as file transfer and DNS lookup.
- CO4: Configure and evaluate routing protocols and services while capturing and analyzing traffic
- CO5: Work collaboratively on a mini-project to design, implement, and analyze a networking solution, demonstrating innovation, teamwork, and professional skills in laboratory practice.

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

List of Assignments (Part A) - 1, 2 mandatory and Any THREE from 3 to 7	
1	Case study on comparison of OSI and TCP/IP reference models with respect to layers, functions, protocols, and real-world implementation. (Compulsory)
2	Demonstrate the different types of topologies and types of transmission media by using a packet tracer tool. (Compulsory)
3	Setup a WAN which contains wired as well as wireless LAN by using a packet tracer tool. Demonstrate transfer of a packet from LAN 1 (wired LAN) to LAN2 (Wireless LAN).
4	Capture and analyze Ethernet frames to study physical addressing using Wireshark Packet Analyzer Tool.
5	Write a program for error detection and correction for 7/8 bits ASCII codes using Hamming Codes or CRC.
6	Write a program to simulate Go back N and Selective Repeat Modes of Sliding Window Protocol in Peer-to-Peer mode.
7	Write a program to implement link state /Distance vector routing protocol to find suitable path for transmission.

List of Assignments (Part B) - 8,9,10 mandatory and Any SIX from 11 to 15	
8	Implement Subnetting and Test IP Addressing Schemes using a packet tracer tool. (Compulsory)
9	Write a program for DNS lookup. Given an IP address as input, it should return URL and vice Versa. (Compulsory)
10	Capture and analyze HTTP, DNS, and SMTP traffic using Wireshark Packet Analyzer Tool. (Compulsory)
11	Configuration of 3 router networks using one of the following protocols RIP/OSPF/BGP Using packet Tracer tool.
12	Write a program using TCP socket for wired network for File transfer
13	Write a program using UDP Sockets to enable file transfer (Script, Text, Audio and Video one file each) between two machines.
14	Configure DHCP server-client setup using a packet tracer tool.
15	Write a program to analyze following packet formats captured through Wireshark for wired Network. 1. Ethernet 2. IP 3.TCP 4. UDP

List of Assignments (Part C) - Any ONE from 16 to 19	
16	Study and Analyze the performance of HTTP, HTTPS and FTP protocol using Packet tracer Tool.
17	To study the SSL protocol by capturing the packets using Wireshark tool while visiting any SSL secured website (banking, e-commerce etc.).
18	Illustrate the steps for implementation of S/MIME email security through Microsoft Office Outlook.
19	To study the IPSec (ESP and AH) protocol by capturing the packets using Wireshark tool

Savitribai Phule Pune University Third Year of Computer Science and Engineering(Data Science) (2024 Course) **PEC321CDSA: 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 and Statistics

Companion Course :

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 (NLP): 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): 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.

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

Case studies:

1. 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 Processing, Corpora and 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

Cast Studies - 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 Studies - Develop and analyze a Part-of-Speech (POS) tagging system using a probabilistic sequence model.

Unit IV - NLP Tasks (09 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 : Case Study: BERT-based text classification for customer support queries

Unit V - Modern Neural NLP and Transformers (09 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 Studies - 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. Geetha, T. V., Understanding Natural Language Processing – Machine Learning and Deep Learning Perspectives, 1st ed., Pearson, 2024.
2. Bird, S., Klein, E., & Loper, E., Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit, 1st ed., O'Reilly Media, 2009.
3. Indurkhya, N., & Damerau, F. J., Handbook of Natural Language Processing, 2nd ed., CRC Press, 2010.

E-Book

1. Introduction to Natural Language Processing, University of London, Study Guide, 2020.
2. Eisenstein, J., Natural Language Processing, 1st ed., MIT Press / Online, 2019.

MOOC/NPTEL/SWAYAM Course Links:

1. https://onlinecourses.nptel.ac.in/noc20_cs87/preview
2. https://nptel.ac.in/courses/106101007

Savitribai Phule Pune University Third Year of Computer Science and Engineering(Data Science) (2024 Course) **PEC321CDSC: Cloud Computing**		
Teaching /scheme	**Credits**	**Examination Scheme**
Theory : 03 Hours/Week	03	**CCE** : 30 Marks **End-Semester:** 70 Marks

Prerequisite Courses : Computer Networks and Security

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

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: Configure, Test and Deploy applications on Cloud
- CO4: Understand and Apply security in cloud applications
- CO5: Analyze emerging technologies in modern cloud computing

Course Contents

Unit I - Introduction to Cloud Computing (07 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 studies: 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

Cast Studies - 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 Studies - 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 -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. 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/NPTEL/SWAYAM Course Links: **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 of Computer Science and Engineering(Data Science) (2024 Course) **PCC-362C-CDS: Soft Computing**		
Teaching /scheme	**Credits**	**Examination Scheme**
Theory : 03 Hours/Week	03	**CCE** : 30 Marks **End-Semester:** 70 Marks

Prerequisite Courses :

Companion Course :

Course Objectives: The course aims to:

1. To introduce the concepts of soft computing and its applications.
2. To understand the fundamental concepts of Artificial Neural Networks.
3. To provide hands-on knowledge of Fuzzy Logic and Fuzzy Inference Systems.
4. To explore Genetic Algorithms and evolutionary computation.
5. To study hybrid systems like Neuro-Fuzzy models.

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

- CO1: Analyze the difference between hard and soft computing.
- CO2. Design and train artificial neural networks for classification problems.
- CO3. Apply fuzzy logic principles to model uncertainty in engineering applications.
- CO4. Optimize complex problems using evolutionary strategies and Genetic Algorithms.
- CO5. Evaluate the performance of hybrid Neuro-Fuzzy systems.

Course Contents

Unit I Introduction & Foundations of Fuzzy Logic

Soft Computing Overview: Introduction to Soft Computing vs. Hard Computing. Fuzzy Set Theory: Transition from classical "crisp" sets to fuzzy sets to represent uncertainty and vagueness. Membership Functions: Definitions and types of fuzzy membership functions. Operations: Fuzzy set operations including Union, Intersection, and Complement. Relations: Fuzzy relations and the concept of 2D binary fuzzy relations. Propositions: Fuzzy propositions and their canonical representations.

Case studies: Uncertainty Management in Medical Diagnosis or Student Grading

Unit II - Fuzzy Inference and Control Systems (8 Hours)

Decision Making: Using fuzzy logic to make decisions and control physical systems. Inference Mechanism: Fuzzy implications and fuzzy inferences.

Defuzzification: Converting fuzzy results back into crisp values using methods like Centroid and Mean of Maximum. Fuzzy Logic Controllers (FLC): Architecture, design, and practical engineering applications.

Cast Studies - Sendai Subway Train Control System (The "Gold Standard" of FLC)

Unit III - Genetic Algorithms (GA) (8 Hours)

Evolutionary Computing: Nature-inspired optimization techniques modeling natural selection to find optimal solutions. Foundations: Basic concepts and strategies of Genetic Algorithms. Encoding Schemes: Representing data using Binary, Octal, and Hexadecimal schemes. GA Operators: * Selection: Roulette wheel and Tournament selection. Crossover: Single-point and Multi-point techniques. Mutation: Techniques for maintaining genetic diversity.

Case Studies - VLSI Floor planning and Optimization

Unit IV - User Experience (UX) Design (8 Hours)

Complex Optimization: Exploring scenarios where multiple conflicting objectives must be optimized simultaneously. Optimality Concepts: The concept of Domination and Pareto-optimality. Non-Pareto Approaches: Vector Evaluated GA and related strategies. Pareto-based Approaches: Implementing NSGA (Non-dominated Sorting Genetic Algorithm) and NPGA for finding trade-offs in complex problems

Case study : Sizing of Hybrid Renewable Energy Systems (HRES)

Unit V - Artificial Neural Networks (ANN) (8 Hours)

Biological Inspiration: Computational models inspired by the brain that "learn" from data. Architectures: Introduction to Single-layer vs. Multi-layer ANN architectures.

Learning Paradigms: Training ANNs via Supervised and Unsupervised learning. Backpropagation: Algorithms and weight adjustment techniques. Hybrid Systems: Overview of integrated Soft Computing tools.

Case Studies - Fault Diagnosis in Chemical Continuous Stirred-Tank Reactors (CSTR)

Learning Resources

Text Books:

1. N. Sivanandam & S.N. Deepa: Principles of Soft Computing, Wiley India.
2. J.S.R. Jang, C.T. Sun, E. Mizutani: Neuro-Fuzzy and Soft Computing, PHI / Pearson Education.
3. S. Rajasekaran & G.A. Vijayalakshmi Pai: Neural Networks, Fuzzy Logic and Genetic Algorithm: Synthesis and Applications, PHI.
4. David E. Goldberg: Genetic Algorithms in Search, Optimization and Machine Learning, Addison Wesley.

Reference Books:

1. Ross, Timothy J., Fuzzy Logic with Engineering Applications, Wiley.
2. Haykin, Simon, Neural Networks: A Comprehensive Foundation, Pearson.
3. Mitchell, Melanie, An Introduction to Genetic Algorithms, MIT Press.
4. Karray, Fakhreddine O. and De Silva, Clarence, Soft Computing and Intelligent Systems Design, Pearson.
5. Bishop, Christopher M., Pattern Recognition and Machine Learning, Springer.

E_Books

1. Wiley India – Principles of Soft Computing
2. MIT OpenCourseWare – Neural Networks Course Material
3. Springer eBooks – Soft Computing Resources
4. NPTEL Soft Computing Archive

MOOC Courses (Web Links):

1. https://nptel.ac.in/courses/106105173
2. https://nptel.ac.in/courses/106106184
3. https://www.coursera.org/learn/neural-networks-deep-learning
4. https://www.coursera.org/learn/genetic-algorithms
5. https://www.edx.org
6. https://www.udemy.com/topic/fuzzy-logic/

YouTube/Video Links:

1. MIT OpenCourseWare YouTube Channel – https://www.youtube.com/@mitocw
2. Stanford Online YouTube Channel – https://www.youtube.com/@stanfordonline
3. Neural Networks Explained Playlist – https://www.youtube.com/results?search_query=neural+networks+explained+playl
4. Genetic Algorithm Tutorials Playlist – https://www.youtube.com/results?search_query=genetic+algorithm+tutorials+playl

Laboratory

Savitribai Phule Pune University Third Year of Computer Science and Engineering(Data Science) (2024 Course) **PEC321CCDS: Design and Analysis of Algorithm**		
Teaching /scheme	**Credits**	**Examination Scheme**
Theory : 03 Hours/Week	03	**CCE** : 30 Marks **End-Semester:** 70 Marks

Prerequisite Courses : Programming Fundamentals, Data Structures

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

Course Objectives: The course aims to:

1. Introduce fundamental concepts of algorithm design and computational complexity.
2. Develop analytical skills to evaluate algorithm efficiency using time and space complexity.
3. Apply major algorithm design paradigms to solve computational problems.
4. Strengthen problem-solving abilities for data-intensive and optimization challenges.

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

- CO1: Analyze the efficiency of algorithms using asymptotic notations.
- CO2:Apply appropriate algorithmic strategies for problem-solving.
- CO3:Compare different algorithms based on performance and scalability.
- CO4:Solve optimization and graph problems using advanced techniques.
- CO5: Evaluate computational complexity and NP-hardness of problems.

Course Contents

Unit I - Foundations of Analysis and Computational Complexity (07 Hours) (07 Hours)

Algorithmic Principles, asymptotic analysis (O, ,), iterative and recursive performance, recurrence relations, Master Theorem, amortized analysis.

Case Study: Mathematical Modeling of Divide-and-Conquer in Data Processing.

Case studies: : Mathematical Modeling of Divide-and-Conquer in Data Processing – Analyze recursive data partitioning algorithms using Master Theorem for scalability in analytics pipelines.

Unit II Sorting, Searching and Greedy Optimization(09 Hours)

Merge Sort, Quick Sort, Linear and Binary Search, Greedy paradigm, Fractional Knapsack, Job Sequencing, Activity Selection, Huffman Coding.

Case Study: Huffman Coding for Data Compression in Analytics Systems.

Cast Studies - : Huffman Coding for Data Compression in Analytics Systems – Implement optimal prefix encoding for efficient storage and transmission of large-scale datasets.

Unit III - Dynamic Programming and Intelligent State-Space Search (08 Hours)

Dynamic Programming: Principle of Optimality, Memoization, Applications: Binomial Coefficients, 0/1 Knapsack, Matrix Chain Multiplication, Edit Distance

Backtracking: 8-Queen, Graph Coloring, Sum of Subsets, Branch and Bound: FIFO, LIFO, Least Cost, TSP, Knapsack

Case Studies - State-Space Optimization for Logistics and Route Intelligence – Compare Branch-and-Bound strategies for smart routing systems

Unit IV - Graph Algorithms, Parallelism and Network Optimization (08 Hours)

Graph Exploration: BFS, DFS, connectivity applications, Shortest Path and MST: Dijkstra, Bellman-Ford, Prim, Kruskal, • Optimization: Network Flow, Linear Programming basics, • Parallel and Concurrent Models: RAM/PRAM, Amdahl's Law, Brent's Theorem, Dining Philosophers

Case study : Parallel Graph Analytics in Social Networks and Recommendation Systems – Analyze scalable graph data using PRAM concepts.

Unit V - Modern Algorithmics, Distributed Systems and Advanced Applications (07 Hours)

Complexity: Turing Machines, Deterministic vs Non-Deterministic, P, NP, NP-Complete, Advanced Topics: Vertex Cover, 3-SAT, Randomized and Approximation

Algorithms, Distributed & IoT: Bully Algorithm, Routing, Clustering, Trust Models, Software Systems: Boyer-Moore, KMP, Buddy Memory Allocation

Case Studies -Distributed Coordinator Selection in Cloud and IoT Systems – Implement Bully Algorithm for fault tolerance

Learning Resources

Text Books:

1. Introduction to Algorithms by Cormen, Leiserson, Rivest, and Stein (CLRS)
2. Fundamentals of Computer Algorithms by Horowitz, Sahni, and Rajasekaran
3. Algorithm Design by Jon Kleinberg and Éva Tardos

Reference Books:

1. Introduction to the Design and Analysis of Algorithms — Introduction to the Design and Analysis of Algorithms by Anany Levitin
2. The Design and Analysis of Computer Algorithms — by Alfred V. Aho, John Hopcroft, and Jeffrey Ullman
3. Algorithms — by Sanjoy Dasgupta, Christos Papadimitriou, and Umesh Vazirani
4. Data Structures and Algorithm Analysis in Python — by Mark Allen Weiss
5. Competitive Programming — by Steven Halim

E_ Books:

1. Open Data Structures — by Pat Morin
2. Algorithms (Free PDF) — by Jeff Erickson
3. Problem Solving with Algorithms and Data Structures Using Python — by Brad Miller and David Ranum
4. Grokking Algorithms — by Aditya Bhargava

MOOC/NPTEL/SWAYAM Course Links:

1. Design and Analysis of Algorithms – Prof. Madhavan Mukund (IIT Madras)
2. Introduction to Algorithms – Prof. S. Arunkumar (IIT Delhi)
3. Data Structures and Algorithms – Prof. Naveen Garg (IIT Delhi)

Savitribai Phule Pune University Third Year - Computer Science and Engineering(Data Science)(2024 Pattern)		
PEC321CDSA- Natural Language Processing Laboratory		
Teaching Scheme	**Credits**	**Examination Scheme**
Practical : 02 Hours/Week	01	Term Work: 25 Marks

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

Companion Course: Natural Language Processing

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. Description 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 the 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 student's 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 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

Guidelines for Practical Examination

1. Both internal and external examiners should collaboratively set problem statements for the practical examination.
2. During the assessment, maximum weightage should be given to the satisfactory implementation of the problem statement.
3. The evaluators may ask supplementary and relevant questions to assess the student's understanding of fundamental concepts and their ability to implement solutions effectively and efficiently.
4. The evaluation process should be carried out jointly by both internal and external examiners.

Suggested List of Laboratory Experiments/Assignments

Students must complete at least 8 assignments, selecting a minimum 4 from Group A and 4 from Group B. An additional assignment any 1 from (13–14)

Group A (Any 4 Compulsory from Assignments 1 to 5)	
1	NLP Pipeline and Linguistic AnalysisDevelop 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. Email ids Phone numbers dates extract patterns URLs Hashtags Design and analyze a Finite Automaton (DFA/NFA) for Binary strings ending with “01” 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 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 Implement: One-hot encoding Bag of Words (BoW) Document-Term Matrix TF-IDF Compare and analyze: Sparse vs dense representations Feature dimensionality and sparsity Interpret and visualize feature vectors.

Group B (Any 4 Compulsory from Assignments 6 to 12)	
6	Text Classification using Naïve Bayes and Logistic Regression Develop a text classification system to classify documents into predefined categories. The system should: Preprocess the text data (tokenization, stopword removal). Convert text into numerical features using TF-IDF. Implement: Naïve Bayes classifier Logistic Regression model Evaluate model performance using: Accuracy Precision, Recall, F1-score Confusion Matrix Compare the performance of both models and justify the results. Dataset: IMDB Movie Reviews Dataset SMS Spam Dataset
7	N-gram Language Model and Perplexity Analysis Develop a language model to predict word sequences using N-gram techniques. The system should: Build: Unigram Bigram Trigram models Estimate probabilities using Maximum Likelihood Estimation (MLE) Apply smoothing techniques (Laplace or Good-Turing). Compute perplexity for each model. Compare model performance and analyze the effect of smoothing. Dataset: Brown Corpus Gutenberg Corpus
8	POS Tagging using Hidden Markov Model (HMM) Develop a Part-of-Speech (POS) tagging system using Hidden Markov Models. The system should: Model POS tagging using HMM (states, observations). Estimate transition and emission probabilities. Implement the Viterbi algorithm to find the most probable tag sequence. Evaluate tagging accuracy on test sentences Analyze errors and discuss limitations of HMM. Dataset: Penn Treebank Dataset NLTK tagged corpus
9	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: 1. Preprocess input text data. 2. Apply NER using: Rule-based approach (pattern matching) Pre-trained statistical model (spaCy) 3. Extract and classify named entities. 4. Evaluate model performance using: Precision Recall F1-score 5. Compare rule-based and statistical approaches and justify the results. Dataset:CoNLL-2003 Dataset Custom news articles

List of Assignments (Part C) - Any ONE from 16 to 19	
16	Study and Analyze the performance of HTTP, HTTPS and FTP protocol using Packet tracer Tool.
17	To study the SSL protocol by capturing the packets using Wireshark tool while visiting any SSL secured website (banking, e-commerce etc.).
18	Illustrate the steps for implementation of S/MIME email security through Microsoft Office Outlook.
19	To study the IPSec (ESP and AH) protocol by capturing the packets using Wireshark tool

Savitribai Phule Pune University Third Year - Computer Science and Engineering(Data Science)(2024 Pattern)		
PEC321CDCB- Cloud Computing Laboratory		
Teaching Scheme	**Credits**	**Examination Scheme**
Practical : 02 Hours/Week	01	Term Work: 25 Marks

Companion Course: Cloud Computing

Course Objectives:

1. To understand the fundamental concepts, architecture, and service models of Cloud Computing.
2. To provide hands-on experience with cloud simulation tools, virtualization platforms, and distributed computing environments.
3. To develop practical skills in cloud application development using platforms such as Salesforce and Hadoop.
4. To explore cloud deployment, scheduling algorithms, big data processing, and cloud security issues.
5. To analyze real-world cloud service providers and evaluate cloud solutions for enterprise applications.

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

- CO1:Explain cloud computing concepts, architectures, deployment models, and service models.
- CO2: Simulate cloud environments using CloudSim and implement customized scheduling algorithms.
- CO3: Develop and deploy cloud-based applications using Salesforce Apex programming language.
- CO4: Install, configure, and execute applications on Hadoop distributed computing framework.
- CO5: Identify cloud security challenges and analyze industrial cloud platforms such as Amazon AWS and Microsoft Azure.

Guidelines for Laboratory Conduction- Cloud Computing

1. List of Assignment- Group A (Any THREE from 1 to 5, 6th and 7th is mandatory)
2. Study fundamental concepts of cloud, understand components of Cloud Architecture and Deployment models for Cloud.
3. Study the concept of Storage as a Service (SaaS) and implement cloud storage using a cloud platform.
4. Install Virtualbox/VMware Workstation with different flavors of linux or windows OS on top of windows 7 or 8.
5. Install a C compiler in the virtual machine created using virtual box and execute Simple Programs.
6. Implement procedure to transfer the files from one virtual machine to another virtual machine
7. Google App Engine (GAE) Setup: Install Google App Engine. Create hello world app or any other simple web applications using python/java.
8. 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 inCloudSim.
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 - Computer Science and Engineering(Data Science)(2024 Pattern)		
PEC321CDSC- Soft Computing Laboratory		
Teaching Scheme	**Credits**	**Examination Scheme**
Practical : 02 Hours/Week	01	Term Work: 25 Marks

Prerequisite Courses : Probability and Statistics, Artificial Intelligence

Companion Course: Cloud Computing

Course Objectives:

1. To introduce the fundamental concepts of Soft Computing and its significance in solving real-world engineering problems.
2. To understand fuzzy logic principles, fuzzy inference systems, and their applications in decision-making and control systems.
3. To study Artificial Neural Networks (ANN) including learning paradigms, architectures, and backpropagation techniques.
4. To explore Genetic Algorithms and evolutionary optimization techniques for solving complex optimization problems.
5. To develop knowledge of hybrid soft computing approaches such as Neuro-Fuzzy systems for intelligent system design and prediction applications.

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

- CO1: Differentiate between hard computing and soft computing techniques and analyze their applications in engineering domains.
- CO2: Design and implement fuzzy logic systems using membership functions, inference mechanisms, and defuzzification techniques.
- CO3: Apply Artificial Neural Networks for classification, prediction, and pattern recognition problems.
- CO4: Solve optimization problems using Genetic Algorithms and multi-objective evolutionary approaches.
- CO5: Develop and evaluate intelligent hybrid systems such as Neuro-Fuzzy models for real-world applications.

Guidelines for Laboratory Conduction- Cloud Computing

List of Assignment- Group A (Mandatory)

1. Introduction to Soft Computing – Comparative study of Hard Computing vs Soft Computing techniques (MS Word / PowerPoint, Computer/Laptop Compulsory Individual assignment, report submission)
2. Fuzzy Set Operations – Implement Union, Intersection, and Complement operations (Using Python Python, Google Colab/Jupyter Notebook, Laptop Compulsory Batch of 2 students, code execution with output screenshots)
3. Membership Functions Design – Implement and compare Triangular, Trapezoidal, and Gaussian membership functions (Python/MATLAB, Matplotlib, Laptop Compulsory Practical demonstration with graph plotting compulsory)
4. Defuzzification Techniques – Compare Centroid, Mean of Maximum, and Bisector methods (Python/MATLAB, scikit-fuzzy, Laptop Optional Graphical analysis and mathematical explanation required)
5. Single Layer Perceptron – Implement ANN for AND/OR gate classification (Python, NumPy, Jupyter Notebook, Laptop Compulsory Manual calculations and implementation required)

List of Assignment- Group B (1,2 and 3 Mandatory, 3,4 optional) Perform atleast Three

1. Fuzzy Inference System – Design student grading/performance evaluation system using fuzzy rules (MATLAB Fuzzy Toolbox / scikit-fuzzy, Team of 2–3 students, minimum 5 fuzzy rules, viva)
2. Fuzzy Logic Controller – Develop controller for temperature, traffic, or fan speed control system (Python/MATLAB, Simulation Tool, Simulation-based implementation with presentation)
3. Genetic Algorithm Basics – Implement GA for optimization of simple mathematical function (Python, NumPy, Google Colab, Individual coding assignment with dry run demonstration)
4. GA Operators Implementation – Perform selection, crossover, and mutation on sample chromosomes (Python, Jupyter Notebook, Laptop Compulsory Demonstration of genetic operations with outputs)
5. Function Optimization using GA – Optimize mathematical equation and analyze fitness over generations (Python, Matplotlib, Google Colab, graph plotting compulsory)

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

1. Multi-Objective Optimization – Generate Pareto-optimal solutions using NSGA/VEGA techniques(Python, Optimization Libraries with comparative Pareto analysis)
2. Multi-Layer Perceptron using Backpropagation – Train ANN for digit or pattern classification , Accuracy/loss graph compulsory using TensorFlow/Keras
3. Unsupervised Learning – Implement K-Means clustering for dataset grouping Python, scikit-learn, Matplotlib, Laptop Optional Dataset visualization and analysis mandatory.
4. Neuro-Fuzzy System – Develop hybrid Neuro-Fuzzy model for prediction/classification (MATLAB/Python, ANFIS Tool)
5. Apply Soft Computing techniques for medical diagnosis, smart traffic control, renewable energy optimization, crop disease detection, or fault diagnosis(Python/MATLAB, Google Colab, Sensors/IoT kits (optional))

Savitribai Phule Pune University Third Year - Computer Science and Engineering(Data Science)(2024 Pattern)		
PEC321CDSD - Design and Analysis of Algorithm Laboratory		
Teaching Scheme	**Credits**	**Examination Scheme**
Practical : 02 Hours/Week	01	Term Work: 25 Marks

Prerequisite Courses : fundamentals of Programming, Probability and Staticstics

Companion Course: Design and Analysis of Algorithm

Course Objectives:

1. To understand the fundamentals of algorithm design and analysis.
2. To learn how to evaluate algorithm efficiency using time and space complexity.
3. To study various algorithm design techniques such as divide and conquer, greedy method, dynamic programming, and backtracking.
4. To develop skills for solving computational problems using efficient algorithms.
5. To understand graph algorithms, searching, sorting, and string matching techniques.
6. To apply algorithmic concepts to real-world problem-solving.

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

- CO1: Understand the basics of algorithm design and analysis techniques.
- CO2: Analyze algorithms using asymptotic notations and recurrence relations.
- CO3: Apply divide and conquer, greedy, and dynamic programming approaches to solve problems.
- CO4: Design efficient algorithms for sorting, searching, graph, and string problems.
- CO5: Implement backtracking and branch-and-bound techniques for optimization problems.
- CO6: Apply algorithmic techniques to real-world computing and engineering problems.

Guidelines for Laboratory Conduction

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 the 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 student's 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 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 Computer Science and Engineering(Data Science) branch beyond

the scope of the syllabus.

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

programming language: C, C++, Python, Java

Programming tools recommended: - Visual Studio Code/ Code::Blocks, Jupyter Notebook,PyCharm, NetworkX. OpenMP. MPI

List of Assignment- Group A (Mandatory)

1. Analyze the efficiency of algorithms using asymptotic notations and recurrence relations. (Analyze (Software Requirement: C/C++/Python Compiler, Laptop/Desktop)
2. Implement recursive algorithms for mathematical and searching problems . (Python/C Compiler, VS Code/CodeBlocks)
3. Design and implement Merge Sort and Quick Sort algorithms. (C/C++/Python, Jupyter Notebook/VS Code)
4. Apply Binary Search for efficient searching in sorted datasets. (Python/C++)
5. Implement Greedy algorithms for Activity Selection and Huffman Coding problems. (Python, GCC Compiler, Laptop)
6. Develop Dynamic Programming solutions for Knapsack and Matrix Chain Multiplication problems. (Python/C++, Jupyter Notebook)
7. Design graph traversal algorithms using BFS and DFS techniques (Python/C++, Jupyter Notebook)
8. Apply shortest path and minimum spanning tree algorithms for network optimization. (Python, NetworkX Library, Laptop)
9. Implement String Matching algorithms such as KMP and Boyer-Moore (Python/C++, VS Code/CodeBlocks)

Group B (Any ONE)

1. Solve N-Queen and Graph Coloring problems using Backtracking . (Python/C++, VS Code)
2. Implement Branch-and-Bound strategies for Traveling Salesperson Problem (TSP) . (Python, Graph Visualization Tools)
3. Analyze parallel computing concepts using PRAM models and Brent's theorem . (OpenMP/MPI Simulator, Linux/Windows System)
4. Analyze NP-Complete problems and compare deterministic and non-deterministic approaches
5. Implement Distributed System algorithms such as Bully Algorithm for coordinator selection (Distributed System Simulator)

Group C (Miniproject Any ONE)

1. Develop a Smart Traffic Route Optimization System using Dijkstra's Algorithm for shortest path identification and traffic management.
2. Develop a Data Compression System using Huffman Coding Algorithm for efficient storage and transmission of large datasets.
3. Develop a Student Performance Prediction and Analysis System using Sorting, Searching, and Dynamic Programming techniques.
4. Develop a Social Network Graph Analysis System using BFS and DFS algorithms for connection and community analysis .
5. Develop a Smart Text Search Engine using KMP/Boyer-Moore String Matching Algorithms for efficient document searching .

Savitribai Phule Pune University Third Year_Computer Science and Engineering(Data Science)		
MDM331CDS- Next Generation of IoT		
Teaching Scheme	**Credits**	**Examination Scheme**
Tutorial : 02 Hours/Week **Practical: 02** Hours/Week	03	CCE : 30 Marks End-Semester: 70 Marks

Prerequisite Courses : Digital Electronics and Logic Design, Fundamentals of IoT.

Course Objectives:

- 1. Understand the fundamental concepts of IoT architecture, components, and end-to-end data flow in IoT systems.
- 2. Implement IoT solutions using edge devices and cloud platforms for real-time data acquisition, processing, and communication.
- 3. Analyze IoT-generated data using visualization tools and apply machine learning techniques for prediction and decision-making.
- 4. Develop secure and scalable IoT applications incorporating communication protocols, authentication, and data protection mechanisms.
- 5. Design and build complete IoT ecosystems by integrating sensors, edge devices, cloud services, machine learning models, and dashboards.

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

- 1. Understand IoT architecture and data flow
- 2. Implement edge and cloud-based IoT systems
- 3. Analyze IoT data using visualization and ML techniques
- 4. Develop secure and scalable IoT applications
- 5. Build complete IoT ecosystems integrating hardware and software

Course Contents

Unit I - IoT Fundamentals & System Architecture (3 Hours)

Introduction to IoT: Concepts, Applications, and Challenges, IoT Architecture: Device–Gateway–Cloud Model,Sensors and Actuators in IoT , Arduino IDE and Python setup for IoT , Basic IoT pipeline and data flow

Unit–II: Edge Computing & Communication Protocols (3 Hours)

Edge Computing Concepts and Benefits, ESP32 for Edge Processing, Local Data Processing and Decision Making, IoT Communication Protocols: HTTP, MQTT , Performance Metrics: Latency and Efficiency , Real-Time Data Streaming using MQTT and WebSockets

Unit–III: IoT Data Management & Cloud Integration (3 Hours)

IoT Data Storage: SQL vs NoSQL Databases ,MongoDB and Firebase for IoT,Data Cleaning and Preprocessing Techniques ,Cloud Computing in IoT: AWS IoT, Google Cloud IoT ,Cloud Data Ingestion and Management Pipelines

Unit-IV : Data Analytics, Visualization & Machine Learning (3 Hours)

Data Visualization using Python (Plotly, Dash) ,Dashboard Development and Cloud Visualization ,Time-Series Analysis: Moving Averages, Trend Detection ,Introduction to Forecasting Techniques ,Machine Learning on IoT Data: Prediction & Classification ,Anomaly Detection using Statistical Methods and ML

Unit V : Advanced IoT, Security & System Integration (3 Hours)

Edge AI Concepts and Lightweight ML Models ,Deployment on ESP32 / Edge Devices ,IoT Security and Privacy: TLS, Authentication, Vulnerabilities ,API Development using Flask/FastAPI ,Backend Integration for IoT Systems ,Smart System Integration: Sensors + Cloud + ML + Dashboard ,Building Complete IoT Ecosystem

Learning Resources

Text Books:

1. David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Rob Barton, Jerome Henry, “IoT Fundamentals – Networking Technologies, Protocols, and Use Cases for the Internet of Things”, 1st Edition, Published by Pearson Education, Inc, publishing as Cisco Press, 2017.
2. Hakima Chaouchi, “The Internet of Things - Connecting Objects to the Web”, 1st Edition, Wiley, 2010.
3. Arshdeep Bahga, Vijay Madisetti, “Internet of Things – Hands-On Approach”, 2nd Edition, Universities Press, 2016.
4. Perry Lea, “Internet of things For Architects”, 1st Edition, Packt Publication, 2018
5. Simone Cirani, Gianluigi Ferrari, Marco Picone, Luca Veltri, “Internet of Things:Architectures, Protocols and Standards”, Wiley.

Reference Books:

1. Raj Kamal, Internet of Things: Architecture and Design Principles, McGraw Hill Education, 1st Edition, 2017
2. Adrian McEwen & Hakim Cassimally, “Designing the Internet of Things”, 1st Edition, Wiley, 2014.
3. David Hanes, Gonzalo Salgueiro, IoT Fundamentals Networking Technologies, Protocols and Use Cases for Internet of Things, Cisco Press, 1st Edition, 2017.
4. Donald Norris, “Raspberry Pi – Projects for the Evil Genius”, 2nd Edition, McGraw Hill, 2014.

MOOC / NPTEL/YouTube Links: -

1. `https://nptel.ac.in/courses/106105195`
2. `https://nptel.ac.in/courses/108108179`
3. `https://nptel.ac.in/courses/108108098`
4. `https://nptel.ac.in/courses/106105166`

Guidelines for Laboratory/Term Work Assessment:

Continuous assessment of laboratory work should be done based on overall performance and Laboratory assignments performance of student. Each Laboratory assignment assessment should be assigned grade/marks based on parameters with appropriate weightage. Suggested parameters for overall assessment as well as each Laboratory assignment assessment include timely completion performance, innovation, efficient codes, punctuality and neatness.

Guidelines for Oral Examination:

Oral examination gauge students’ knowledge and skills based on the spoken word, typically guided by questions or small tasks. A pair of examiners must design appropriate questions for each learning outcome. They should focus on depth rather than breadth. They should include potential followup questions and prompts based on different types of answers. Examiners should standardize the number of questions, difficulty of questions, and the time allotted. Questions should be based on the practical assignments performed in the term work and not on the entire syllabus.

Suggested List of Assignment: (Any 8)

1. Study and Setup of IoT Architecture (Device–Gateway–Cloud) using Arduino IDE, Python, and Pipeline Demonstration

2. Implementation of IoT-Based Sensor Data Processing with Calibration and Store Structured Data (CSV/JSON)
3. Design and Implementation of Edge Computing using ESP32 with Local Processing, Decision Making, and Reduced Cloud Dependency
4. Design and Evaluation of IoT Communication Protocols using HTTP and MQTT for Latency and Efficiency
5. Study of Real-Time Data Streaming using MQTT Broker and WebSockets with Live Feed Development
6. Study & Implementation of IoT Data Storage in SQL/NoSQL Databases (MongoDB/Firebase) with Data Cleaning and Pre-processing
7. Study of Cloud-Based IoT Pipeline for Sending and Managing Data using AWS IoT and Google Cloud IoT
8. Design and Implementation of Data Visualization Dashboard using Python (Plotly, Dash) and Cloud Platforms for Trend and Anomaly Detection
9. Study of Time-Series Analysis of IoT Sensor Data using Moving Averages, Trend Detection, and Introduction to Forecasting
10. Implementation of Machine Learning on IoT Data for Temperature/Energy Prediction and Normal vs Anomaly Classification
11. Study of Anomaly Detection in IoT Systems using Statistical Methods and ML Models with Applications in Fault and Intrusion Detection
12. Study of Edge AI Implementation using Lightweight ML Model Deployment on ESP32/Edge Device for Real-Time Prediction
13. Study of IoT Security and Privacy with Secure Communication (TLS), Device Authentication, and Vulnerability Assessment in IoT Systems.
14. Design and Implementation of API Development for IoT Systems using Python (Flask/FastAPI) for Device-to-Backend Integration.
15. Design and Implementation of Smart System Integration using Sensors, Cloud, Machine Learning, and Dashboard for a Complete IoT Ecosystem
16. automation using sensors and actuators

Savitribai Phule Pune University Third Year_Computer Science and Engineering(Data Science)(2025 Pattern)		
ELC342CDS: Technical Seminar		
Teaching Scheme	**Credits**	**Examination Scheme**
Theory : 02 Hours/Week	01	Oral : 25 Marks

Course Objectives: The course aims to:

1. To develop research orientation and technical communication skills in emerging Computer Engineering, Artificial Intelligence and Data Science 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 Computer Engineering and Artificial Intelligence domains research problems through a systematic literature survey.
- CO2: Critically analyze research papers using technical, methodological, and ethical perspectives.
- CO3: Prepare structured structures technical reports and presentations.
- CO4: Deliver effective oral presentations demonstrating clarity, confidence, and domain understanding.
- CO5: Demonstrate professional ethics and academic integrity

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, Artificial Intelligence domains (last 3–5 years).
- Must involve a minimum of 5 recent research papers (IEEE, ACM, Elsevier, Springer, etc.). They should summarize the papers by reading abstracts and identifying ideas, conclusions, advantages of their approaches, and drawbacks. Generalize results from a research paper to related research problems. Compare approaches and identify weaknesses and strengths in recent research articles on the subject. Practical sessions on how to read, analyze, and summarize research papers should be included.
- Should not be a basic textbook topic.
- Must include: Problem statement, State-of-the-art analysis, Comparative study, Ethical & societal impact, Future research scope, 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 engineer- ing/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 - Computer Science and Engineering (Data Science)

2024 Pattern

Semester VI

With effect from Academic Year 2026-27

Savitribai Phule Pune University Third Year of Artificial Intelligence and Data Science (2024 Course) **PCC-351-CDS: Machine Learning**		
Teaching /scheme	**Credits**	**Examination Scheme**
Theory : 03 Hours/Week	03	**CCE** : 30 Marks **End-Semester:** 70 Marks

Prerequisite Courses : • Python Programming , Mathematical Foundation for Data Science – I,II, Data Storytelling and Visualization ,Artificial Intelligence,

Companion Course :

Course Objectives: The course aims to:

1.To provide a strong foundation of machine learning concepts and paradigms.
2. Build mathematical foundations to develop regression and classification models.
3. Expose learners to classification methods for structured and unstructured data.
4. Introduce clustering, ensemble techniques, and model optimization strategies.
5. Develop insight into reinforcement learning for sequential decision making systems..

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

- CO 1.: Describe machine learning paradigms, different model types, and basic feature engineering methods.
- CO 2. Use regression techniques and performance measures to build prediction models for real world problems.
- CO 3. Apply classification algorithms and measure their performance on various datasets.
- CO 4. Examine clustering and ensemble methods to identify ways of improving model accuracy.
- CO 5. Develop reinforcement learning models for sequential and adaptive systems.

Course Contents

Unit I- Machine Learning Fundamentals (07 Hours)

Evolution of Machine Learning in Data Science ML vs AI vs Deep Learning vs Data Analytics, Learning Paradigms: Supervised, Unsupervised, Semi-supervised, Reinforcement, Model Types: Parametric vs Non-parametric, Probabilistic vs Deterministic, Feature Engineering: Feature Scaling, Encoding, Feature Selection Methods (Filter, Wrapper, Embedded), Dimensionality Reduction:PCA (Geometric intuition)LDA (Discriminative approach)Bias-Variance Tradeoff .

Case studies: Student Performance Prediction with Feature Optimization

Unit II - Regression & Predictive Modeling (08 Hours)

Regression Fundamentals and Assumptions, Types: Linear, Multiple, Polynomial, Non-linear Regression Models.
Advanced Regression: Decision Tree Regression, Random Forest Regression, Support Vector Regression (SVR)Regularization: Ridge (L2), Lasso (L1), ElasticNet,
Model Evaluation's, RMSE, MAE,R^2 and Adjusted R^2,Cross, Validation Techniques.

Cast Studies - Comparative Study of Regression Techniques for Real Estate Price Prediction System

Unit III - Classification & Model Evaluation (08 Hours)

Classification Fundamentals, Types: Binary, Multiclass, Multi-label, Imbalanced Classification Handling (SMOTE concept),
Algorithms: Logistic Regression, KNN, Support Vector Machine (Linear & Kernel SVM), Kernel Tricks: RBF, Polynomial, Sigmoid,
Evaluation Metrics: Confusion Matrix, Precision, Recall, F1-score, ROC-AUC Curve, Macro, Micro, Weighted Averaging,
Multiclass Strategies: One-vs-One, One-vs-All.

Case Studies - Email Spam Detection / Fraud Detection

Unit IV - Clustering Techniques & Ensemble Methods (08 Hours)

Clustering Techniques: K-Means, K-Medoids, Hierarchical Clustering (Agglomerative, Divisive) DBSCAN, Gaussian Mixture Models, Cluster Validation: Elbow Method, Silhouette Score, Ensemble Learning: Voting (Hard/Soft),Bagging, Random Forest,Boosting:AdaBoost,Gradient Boosting, Stacking, Applications in business analytics.

Case study : Customer Segmentation and Sales Prediction

Unit V - Reinforcement Learning (06 Hours)

Fundamentals of Reinforcement Learning, RL vs Supervised Vs Unsupervised Learning,Markov Decision Process (MDP): States, Actions, Rewards, Policy and Value Functions, Bellman Equation, Exploration vs Exploitation-Learning Algorithm, Applications:Robotics,Game AI,Smart Systems

Case Studies -Self Driving Cars

Learning Resources

Text Books:

1. Murphy, Kevin P., "Probabilistic Machine Learning: Advanced Topics," 1st Edition, MIT Press, 2023, ISBN: 978-0262048439.
2. Géron, Aurélien, "Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow," 3rd Edition, O'Reilly Media, 2022, ISBN: 978-1098125967.
3. 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.

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.

Ebooks:

1. Machine Learning Yearning": https://www.deeplearning.ai/machine-learning-yearning/
2. Probabilistic Machine Learning: An Introduction:https://probml.github.io/pml-book/book1.html
3. Reinforcement Learning: An Introduction:
4. https://www.andrew.cmu.edu/course/10-703/textbook/BartoSutton.pdf

MOOC Courses (Web Links):

1. NPTEL Course: Introduction to Machine Learning, by Prof.Balaraman Ravindran , IIT Madras https://nptel.ac.in/courses/10
2. NPTEL Course: Introduction to Machine Learning, by Prof.Sudeshna Sarkar.IIT kharagpur https://nptel.ac.in/courses/10610

YouTube/Video Links:

1. Fundamentals of Machine Learning: https://www.youtube.com/watch?v=65BlnPBLbXM
2. Machine Learning Basics: https://www.youtube.com/watch?v=bytGpKPWvIY

Savitribai Phule Pune University Third Year of Artificial Intelligence and Data Science (2024 Course) **PCC-351-CDS: Machine Learning Laboratory**		
Teaching /scheme	**Credits**	**Examination Scheme**
Theory : 03 Hours/Week	03	**CCE** : 30 Marks **End-Semester:** 70 Marks

Prerequisite Courses : • • Python Programming , Mathematical Foundation for Data Science – I,II, Data Storytelling and Visualization ,Artificial Intelligence

Companion Course : Machine Learning (PCC-351-CDS)

Course Objectives: The course aims to:

1. Develop the ability to preprocess and prepare real-world datasets by handling missing values, outliers, and feature inconsistencies for effective machine learning applications.
2. Build competence in selecting, constructing, and evaluating appropriate machine learning models including regression, classification, and ensemble techniques for data-driven problem solving.
3. Enhance analytical skills in assessing model performance, handling imbalanced data, and optimizing models using feature selection and regularization techniques.
4. Enable the design and implementation of clustering and reinforcement learning approaches for solving complex real-world decision-making and pattern discovery problems.

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

1. Apply data preprocessing and feature engineering techniques to prepare real-world datasets for machine learning model development.
2. Apply regression and classification algorithms to build predictive models and analyze their performance using appropriate evaluation metrics.
3. Analyze the impact of feature selection, regularization, and imbalanced data handling techniques on model accuracy and generalization.
4. Apply clustering, ensemble, and reinforcement learning techniques to solve real world problems and analyze their effectiveness in improving decision-making and prediction performance

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 Lab Journal

The laboratory assignments are to be submitted by the 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 student's 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, and punctuality.

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.

Guidelines for Practical Examination

1. Both internal and external examiners should collaboratively set problem statements for the practical examination.
2. During the assessment, maximum weightage should be given to the satisfactory implementation of the problem statement.
3. The evaluators may ask supplementary and relevant questions to assess the student's understanding of fundamental concepts and their ability to implement solutions effectively and efficiently.
4. The evaluation process should be carried out jointly by both internal and external examiners.

Suggested List of Laboratory Experiments/Assignments
(All Assignments are Compulsory)

1. Perform data preprocessing by identifying and treating outliers, missing values, and feature inconsistencies on a suitable open-access dataset to improve data quality for machine learning tasks.
2. Construct feature selection pipelines using filter, wrapper, and embedded methods on a suitable dataset and compare their impact on model accuracy.
3. Analyze the bias–variance trade-off by training models with varying complexity on a suitable dataset and interpreting underfitting and overfitting behavior.
4. Demonstrate the use of nonlinear regression model on a suitable real-world dataset and evaluate its performance using appropriate regression metrics.
5. Compare multiple regression models (Linear, Tree-based, SVR) on a suitable dataset to identify the most suitable model for prediction tasks.
6. Evaluate the effect of regularization techniques (Ridge, Lasso, ElasticNet) on a suitable dataset to control overfitting and improve generalization.
7. Develop a multiclass classification model using suitable algorithms on a real-world dataset and evaluate performance using macro and micro averaging metrics.
8. Analyze the impact of imbalanced data handling techniques such as SMOTE on classification performance using a suitable dataset.
9. Design a classification pipeline integrating feature selection and model tuning on a suitable dataset to improve prediction accuracy.
10. Construct hierarchical clustering models on a suitable dataset and interpret cluster structures using dendrogram analysis.
11. Compare ensemble techniques such as Bagging, Boosting, and Stacking on a suitable dataset to evaluate performance improvement over base learners.
12. Design a reinforcement learning model using Q-learning for a grid-based environment and evaluate policy optimization through reward maximization.

Learning Resources

Text Books:

1. Murphy, Kevin P., "Probabilistic Machine Learning: Advanced Topics," 1st Edition, MIT Press, 2023, ISBN: 978-0262048439.
2. Géron, Aurélien, "Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow," 3rd Edition, O'Reilly Media, 2022, ISBN: 978-1098125967.
3. 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.

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.

Ebooks:

1. Machine Learning Yearning”: https://www.deeplearning.ai/machine-learning-yearning/
2. Probabilistic Machine Learning: An Introduction:https://probml.github.io/pml-book/book1.html
3. Reinforcement Learning: An Introduction:
4. https://www.andrew.cmu.edu/course/10-703/textbook/BartoSutton.pdf

Virtual Laboratory (links):

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

MOOC Courses (Web Links):

1. NPTEL Course: Introduction to Machine Learning, by Prof.Balaraman Ravindran , IIT Madras https://nptel.ac.in/courses/10
2. NPTEL Course: Introduction to Machine Learning, by Prof.Sudeshna Sarkar.IIT kharagpur https://nptel.ac.in/courses/10610

YouTube/Video Links:

1. Fundamentals of Machine Learning: https://www.youtube.com/watch?v=65BlnPBLbXM
2. Machine Learning Basics: https://www.youtube.com/watch?v=bytGpKPWvIY

Savitribai Phule Pune University Third Year of Computer Science and Engineering(Data Science)(2024 Course) **PEC321CDSB: Cyber Security and Data Privacy**		
Teaching /scheme	**Credits**	**Examination Scheme**
Theory : 03 Hours/Week	03	**CCE** : 30 Marks **End-Semester:** 70 Marks

Prerequisite Courses : Operating Systems, and basic programming (C or Python)

Companion Course :

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 and Data Privacy (08 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 studies: Data Exposure in a Student Mobile Application

Unit II - Data Encryption Techniques (08 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.

Cast Studies - Analyzing Password Breach (like the 2012 LinkedIn)

Unit III - Secure Engineering and Vulnerability Management (08 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 Studies - Secure SDLC review for a student portal.

Unit IV - Security and Threats Handling (08 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 (08 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 Studies -Ransomware incident response plan with evidence collection and recovery steps.

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. Nina Godbole, Sunit Belapure, “Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives”, Wiley India Pvt. Ltd., ISBN: 978-81-265-2179-1.
2. William Stallings, “Cryptography and Network Security: Principles and Practice”, Pearson Education Limited, 7th Edition, ISBN: 978-9332585225.
3. Natraj Venkataramanan, Ashwin Shriram “Data Privacy Principles and Practice “, CRC Press ISBN: 9781498760122

E-Book

1. Charles P. Pfleeger, Shari Lawrence Pfleeger, “Security in Computing”, Pearson India, 5th Edition, ISBN: 978-9352866533.
2. Michael Howard, David LeBlanc, “Writing Secure Code”, Microsoft Press, 2nd Edition, ISBN: 978-0735617223.
3. ISO/IEC 27701:2025 – Privacy Information Management Systems (PIMS) Requirements and Guidance, International Organization for Standardization, 2nd Edition, ISBN: 978-92-67-22701-1.

MOOC/NPTEL/SWAYAM Course Links:

1. NPTEL Course: Cyber Security and Privacy - Course
2. NPTEL Course Internet Crimes and Cyber Security - Course

Savitribai Phule Pune University Third Year of Computer Science and Engineering(Data Science) (2024 Course) **PCC-361B-CDS: Blockchain Technology**		
Teaching Scheme	**Credits**	**Examination Scheme**
Theory : 03 Hours/Week	03	**CCE** : 30 Marks **End-Semester:** 70 Marks

Prerequisite Courses : Data Structures, Database Management Systems, Computer Network

Companion Course :

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: Upon successful completion of this course, students will be able to:
- CO2: Describe the fundamentals of blockchain technology
- CO3: Demonstrate the use of cryptocurrency wallets to perform blockchain transactions.
- CO4: Analyze appropriate consensus mechanisms for distributed systems
- CO5: Construct smart contracts using Ethereum and Solidity for data-driven applications
- CO6: Develop blockchain-based solutions for applications in AI and data science

Course Contents

Unit I Introduction to Blockchain and Cryptographic Foundations

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

Case studies: Hash-based integrity verification

Unit II - Cryptocurrency and Bitcoin (8 Hours)

Introduction: Bitcoin and the Cryptocurrency, Cryptocurrency, Bitcoin Transactions and Scripts, Wallets and Keys, Mining Process, Blockchain Security Mechanisms, Cryptocurrency Economics Types of Cryptocurrencies: Cryptocurrency Usage, Crypto wallets: MetaMask, Coinbase, Binance

Types of Blockchain Platforms: Bitcoin, Ethereum, Hyperledger, IoTA, Corda, and R3.

Cast Studies - Create your own wallet for cryptocurrency using any of the blockchain platforms..

Unit III - Consensus Mechanism (8 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 Time, Proof of Activity, Proof of Burn, Proof of Authority, Scalability and Performance

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

Unit IV - Ethereum and Smart Contracts (8 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 (8 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 Studies -Secure model sharing using blockchain

Learning Resources

Text Books:

1. Imran Bashir, "Mastering Blockchain," Packt Publishing, ISBN: 9781803230214, 4th Edition, 2023
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. 1. Andreas M. Antonopoulos, Mastering Bitcoin: Programming the Open Blockchain, 3rd Edition, O'Reilly Media, Sebastopol, California, USA, 2023
2. 2. Melanie Swan, Blockchain: Blueprint for a New Economy, O'Reilly Media, Sebastopol,
3. California, USA, 2015.
4. 3. Daniel Drescher, Blockchain Basics: A Non-Technical Introduction in 25 Steps, Apress, New
5. York, USA, 2017.
6. 4. William Mougayar, The Business Blockchain: Promise, Practice, and Application of the Next
7. Internet Technology, Wiley, Hoboken, New Jersey, USA, 2016.
8. 5. Arshdeep Bahga and Vijay K. Madisetti, Blockchain Applications: A Hands-On Approach, VPT Publications, Blacksburg, Virginia, USA, 2017.

Additional Reference

1. Research papers and technical documentation from AR/VR platforms
2. Online developer documentation for Unity, ARCore and ARKit
3. Industry case studies in immersive technology

MOOC Courses (Web Links):

1. https://onlinecourses.nptel.ac.in/noc26_cs03/preview
2. https://onlinecourses.nptel.ac.in/noc25_cs87/preview
3. https://www.classcentral.com/course/swayam-augmented-virtual-reality-foundations-and-applications-500287
4. https://elearn.nptel.ac.in/shop/completed-courses/short-term-programs-completed/foundation-course-on-virtual-reality-and-augmented-reality

Ebooks:

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

MOOC Courses (Web Links):

1. NPTEL Course on "Introduction to Blockchain Technology & Applications" Link: https://nptel.ac.in/courses/106/104/1061042
2. NPTEL Course on "Blockchain and its Applications" Link: https://nptel.ac.in/courses/106/105/106105184/
3. Coursera: "Blockchain Basics" by University at Buffalo
4. edX: "Blockchain for Business" by The Linux Foundation and "Quantum Computing" by Prof. Debabrata Goswami, IIT Kanpur. Link: https://nptel.ac.in/courses/104104082

YouTube/Video Links:

1.1. https://www.youtube.com/watch?v=ipwxYa-F1uY
2. https://www.youtube.com/watch?v=M576WGiDBdQ
3. https://www.youtube.com/watch?v=gyMwXuJrbJQ
4. https://www.youtube.com/watch?v=ipwxYa-F1uY
5. https://www.youtube.com/watch?v=EH6vE97qIP4
6. https://www.youtube.com/watch?v=3681ZYbDSSk

Savitribai Phule Pune University Third Year of Computer Science and Engineering(Data Science) (2024 Course) **PCC-361C-CDS: Augmented Reality and Virtual Reality**		
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 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.
- CO2. Apply basic 3D scene representation and transformation concepts in immersive environments.
- CO3. Describe interaction techniques and user interface design principles in virtual environments.
- CO4. Explain tracking, registration and spatial mapping techniques used in augmented reality systems.
- CO5. Analyze real-world applications, performance issues and ethical considerations in AR/VR systems.

Course Contents

Unit I Introduction to AR and VR (9 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 studies: 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 (9 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.

Cast Studies - 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 (9 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 Studies - 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 (9 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 (9 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 Studies -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-We

Reference Books:

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

Additional Reference

1. Research papers and technical documentation from AR/VR platforms
2. Online developer documentation for Unity, ARCore and ARKit
3. Industry case studies in immersive technology

MOOC Courses (Web Links):

1. https://onlinecourses.nptel.ac.in/noc26_cs03/preview
2. https://onlinecourses.nptel.ac.in/noc25_cs87/preview
3. https://www.classcentral.com/course/swayam-augmented-virtual-reality-foundations-and-applications-500287
4. https://elearn.nptel.ac.in/shop/completed-courses/short-term-programs-completed/foundation-course-on-virtual-reality-and-augmented-reality

YouTube/Video Links:

1. https://www.youtube.com/watch?v=zLMgdYI82IE
2. Foundation for Virtual and AugmentedReality Systems, IIT Guwahati : https://onlinecourses.nptel.ac.in/elearning/preview/
3. Foundations of Virtual Reality, IIT Madras, Prof. M. Manivannan : https://onlinecourses.nptel.ac.in/elearning/preview/noc2
4. MobileVirtual Reality and Artificial Intelligence : https://onlinecourses.nptel.ac.in/e-learning/preview/

Savitribai Phule Pune University Third Year of Computer Science and Engineering(Data Science) (2024 Course) **PCC-361D-CDS: 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: Analyz equantum 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 (9 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: Pauligates (X, Y, Z), Hadamard gate, Controlled-NOT gate (CNOT). Quantum Circuits and Measurement:Quantum circuit model and measurement in quantum systems.

Case studies: Classical Bit vs Quantum Bit: Superposition Experiment

Unit II - Mathematical Foundations of Quantum Mechanics (9 Hours)

Linear Algebra Framework for Quantum Mechanics: Vector spaces, complex Hilbert spaces, innerproducts, 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 measure ment theory, density operator formalism, composite quantum systems. Fundamentals of Quantum Principle: No-cloning theorem and implications for quantum information and quantum AI.

Cast Studies - Detection of Eavesdropping in a Quantum AI Communication System using the No Cloning Principle.

Unit III - Quantum Algorithms (9 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 Studies - Classical vs Quantum Search: Understanding Grover's Algorithm.

Unit IV - Quantum Machine Learning(8 Hours)

Introduction to quantum-classical hybrid learning, data encoding and quantum feature maps, vi brational 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 (9 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 Studies -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 Courses (Web Links):

1. NPTELcourse: "QuantumComputing"byProf. DebabrataGoswami, IITKanpurhttps://www.youtube.com/playlist?list=PLqGn
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/noc
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. Centerfor Excellence in QuantumTechnology: https://research.ibm.com/blog/next-wave-quantum centric-supercomputing

Savitribai Phule Pune University Third Year of Computer Science and Engineering(Data Science) (2024 Course) **PCC-362A-CDS: Generative AI**		
Teaching /scheme	**Credits**	**Examination Scheme**
Theory : 03 Hours/Week	03	**CCE** : 30 Marks **End-Semester:** 70 Marks

Prerequisite Courses :

Companion Course :

Course Objectives: The course aims to:

1. Understand the foundations and evolution of Generative AI systems.
2. Explore transformer architectures and large language models.
3. Apply prompt engineering and fine-tuning techniques for AI applications.
4. Develop generative AI applications using modern frameworks and APIs.
5. Analyze ethical, legal, and societal implications of Generative AI.

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

CO1: Understand the fundamentals and architectures of Generative AI systems.

CO2 :Apply transformer and LLM concepts for intelligent content generation.

CO3 :Develop AI applications using prompt engineering and fine-tuning methods.

CO4 :Implement multimodal and generative AI solutions for real-world problems.

CO5 :Analyze ethical and responsible AI practices in generative systems.

Course Contents

Unit I Foundations of Generative AI (9 Hours)

Introduction to Artificial Intelligence and Generative AI ,Evolution of Generative Models ,Generative vs Discriminative AI ,Neural Networks Review , Deep Learning fundamentals , Introduction to transformers ,Applications of Generative AI in industry

Case studies: AI-Based Content Generation for Digital Marketing

Study of AI tools used for automated content, advertisement, and blog generation.

Unit II -Large Language Models and Transformers (9 Hours)

Transformer architecture ,Self-attention mechanism ,Encoder–decoder models ,GPT architecture , BERT and foundation models ,Tokenization and embeddings ,Pre-training and fine-tuning ,Prompt-based learning

Cast Studies - Chatbot Development using Large Language Models

Design and analysis of conversational AI systems for customer support automation.

Unit III - Prompt Engineering and AI Application Development (9 Hours)

Prompt engineering fundamentals,Zero-shot, one-shot, and few-shot prompting ,Chain-of-thought prompting , AI agents and autonomous systems ,API integration with Generative AI tools ,Retrieval-Augmented Generation (RAG) ,AI copilots and coding assistants ,Building Generative AI applications

Case Studies - AI Coding Assistant for Software Development

Implementation of AI-assisted code generation and debugging systems.

Unit IV - Generative Models and Multimodal AI (9 Hours)

Generative Adversarial Networks (GANs) ,Variational Autoencoders (VAEs) ,Diffusion Models ,Text-to-image generation ,Multimodal AI systems ,Audio and video generation, Image captioning and synthetic media generation , AI for creativity and design

Case study : AI Image Generation for Media and Entertainment

Analysis of diffusion-based image generation systems in creative industries.

Unit V -Responsible AI and Future Trends (9 Hours)

Ethics in Generative AI ,Bias and fairness in AI systems ,Hallucination in LLMs , AI safety and governance ,Copyright and legal considerations , Privacy and security issues
,Explainable and Responsible AI ,Future trends in Generative AI

Case Studies - **Responsible AI in Healthcare Systems**
Evaluation of ethical challenges and AI governance in healthcare-based generative AI applications

Learning Resources

Text Books:

1. Generative Deep Learning – David Foster
2. Deep Learning – Ian Goodfellow, Yoshua Bengio, Aaron Courville
3. Natural Language Processing with Transformers – Lewis Tunstall, Leandro von Werra, Thomas Wolf
4. Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow – Aurélien Géron

Reference Books:

1. Transformers for Natural Language Processing – Denis Rothman
2. Pattern Recognition and Machine Learning – Christopher M. Bishop
3. Artificial Intelligence: A Modern Approach – Stuart Russell and Peter Norvig
4. Speech and Language Processing – Dan Jurafsky and James H. Martin

E_Books

MOOC Courses (Web Links):

1. Programming with Generative AI – IISc Bangalore (NPTEL)
2. NPTEL Official Portal
3. Foundations to Machine Learning – Generative AI Module (IIT Madras)
4. DeepLearning.AI Generative AI Courses
5. Coursera – Generative AI Specialization
6. Udemy – Generative AI and LLM Courses
7. Google AI Learning Platform

Savitribai Phule Pune University Third Year of Computer Science and Engineering(Data Science) (2024 Course) **PEC-361C-CDS: Ethical Hacking**		
Teaching /scheme	**Credits**	**Examination Scheme**
Theory : 03 Hours/Week	03	**CCE** : 30 Marks **End-Semester:** 70 Marks

Prerequisite Courses :

Companion Course :

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:

1. CO1. Apply knowledge of ethical hacking frameworks, lifecycle stages, and professional guidelines to assess modern threat landscapes.
2. CO2. Utilize reconnaissance methodologies and tools to identify exposed assets and map attack surfaces effectively
3. CO3. Identify common vulnerabilities and demonstrate basic exploitation concepts in controlled environments.
4. CO4. Analyze identity-based attacks, credential misuse, and post-compromise activities to understand attacker behavior.
5. 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 (7 Hours)

IEthical 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 studies: 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 & Attack Surface Analysis World (8 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 .

Cast Studies - 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 (8 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 Studies - IWeb 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 (8 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 → AiTM credential capture → MFA bypass → lateral movement via Pass-the-Hash → data exfiltration from
SharePoint. Analysis includes detection opportunities at each stage.

Unit V - Modern Attack Domains: Web, API, Cloud & AI Threats (8 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 Studies -: 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. Peter 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 (continuously updated).

MOOC Courses (Web Links):

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/Video 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

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 of Computer Science and Engineering(Data Science) (2024 Course) **PCC-362C-CDS: 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)

IWhat 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 studies: 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.

Cast Studies - 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 Studies - Analyze product search, checkout flow, recommendation systems, and visual hierarchy (Amazon, Flipkart, Myntra)

Unit IV - Wireframing Fundamentals (9 Hours)

Introduction to wireframes, purpose and importance in UX design, low-fidelity vs high-fidelity wire frames, 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 Studies -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. 4.Jeff Gothelf & Josh Seiden – Lean UX, O'Reilly.

E_Books

MOOC Courses (Web 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_Computer Science and Engineering(Data Science)		
VSE372CDS- Solar Technology and Maintenance		
Teaching Scheme	**Credits**	**Examination Scheme**
Practical: 02 Hours/Week	01	Term Work : 50 Marks

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

Course Objectives:

- Apply Safely install, wire, and commission basic solar PV systems while measuring key performance parameters.
- Analyze Break down the impact of environmental and operational factors on solar system efficiency and diagnose common faults.
- Evaluate Judge the effectiveness of maintenance and troubleshooting procedures for solar PV components and systems.
- 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: Createsimplepractical 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 and
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 Charge controller Battery 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 Engineers Author: 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/noc24_ph26/preview?u
2. SkillCat or Other Free Solar Training (Installation Focus). https://www.skillcatapp.com/solar installation-training

Savitribai Phule Pune University		
Third Year - Computer Science and Engineering(Data Science)(2024 Pattern)		
ELC381CDS - 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. In ternship 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. To develop professional, technical, and communication skills.
4. To encourage self-learning and problem-solving abilities.
5. To provide hands-on experience in emerging technologies

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

- **CO1:** Apply theoretical knowledge to solve real-world problems.
- **CO2:** Demonstrate technical competency in tools/technologies used in industry.
- **CO3:** Exhibit professional ethics and teamwork.
- **CO4:** Communicate effectively through reports and presentations.
- **CO5:** Analyze and document industrial workflows and processes.

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 (mandatory 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 voc

Deliverables

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

Nature of Internship

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
- WebandMobile 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

Guidelines for Internship Report Writing

E-Books

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

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

Programme Coordinator

Dr. Vaishali P. Vikhe - Member, Board of Studies - Computer Engineering

Team Members for Course Design

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, Vilad,Ahilyanagar
Mr. Anshuman Vilas Jadhav	Independent Consultant

Cyber Security and Data Privacy	
Dr. Chhaya Gosavi	Cummins COEW,Pune
Dr. Bhagyashree Dhakulkar	Ajeenkya DY Patil School of Engineering, Lohegaon Pune
Dr. Bajirao Subhash Shirole	Loknete Gopinathji Munde Institute of Engineering Education and Research, Nashik
Dr. Sonali Bharat Shirke	Keystone School of Engineering,Pune
Prof.Pradip Nanasaheb Shendage	Vidya Pratishthan's Kamalnayan Bajaj Institute of Engineering,Baramati
Prof.Sagar Rajebhosale	Keystone School of Engineering, Pune

Ethical Hacking	
Dr. Dikshendra Sarpate	ISBM College Of Engineering, Pune
Dr Prakash Ramesh 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 Meenal Kamlakar	Cummins College of Engineering for Women,Pune
Mr.Sachin Kulkarni	flydubai

Theory of Computation	
Dr.Shilpa Prafull Khedkar	MES Wadia College of Engineering Pune.
Dr.P.M.Yawalkar(MET)	MET's Institute of Engineering, Nashik
Dr.Shwetal K. Patil	Marathwada Mitramandal's Institute of Technology, Lohgaon

Ms.Varsha Mhaske	College of Engineering Malegaon
Mrs.Pradnya Kothawade	Genba Sopanrao Moze College of Engineering

Computer Networks and Laboratory	
Dr.S.R.Lahane	RH Sapat College of Engg, Management Studies and Research
Dr.Ranjana Dahake	MET's Institute of Engineering, Nasik
Dr.Salunke Mangesh	Marathwada Mitra Mandal's Institute of Technology
Dr.Megha Kadam	Marathwada Mitra Mandal's Institute of Technology
Mrs.Pallavi Rahinj	Rajiv Gandhi College of Engineering,Karjule Harya
Mr. Prasad Bhosle	Trinity Academy of Engineering,Pune

UI/UX + Lab	
Prof. Pallavi M. Shimpi	Department of Computer Engg, ADYPSoE, Pune
Dr. Dipali Ujlamkar	AISSMS College of Engineering
Dr.Deepali Baviskar	MITWPU, Pune
Mr. Chaitnya Bhosale	MAE Alandi,Pune
Mr. Yogesh Mali	ADYPU,PUNE
Mr.Abhishek Bangar	Jaihind College of Engineering, Tal Junnar Dist Pune

AR/VR + Lab	
Dr N F Shaikh	MESCOE, Pune
Dr. Shabnam Sayyad	AISSMSCOE, Pune
Mr. Sagar Shinde	MESCOE, Pune
Mr. Yogesh Survase	FirebirdVR Pvt. Ltd
Mr. Akshay Rathod	FirebirdVR Pvt. Ltd

Bockchain Technology + Lab	
Dr. Vandana Navale	Ajeenkya DY Patil School of Engineering,Lohegaon,Pune
Dr. Divya Sharma	Ajeenkya DY Patil School of Engineering,Lohegaon,Pune
Dr Priyanka Kedar	Entrata India Pvt Ltd, Koregaon Park, Pune
Dr Varsha Dange	Vishwakarma Institute of Technology , Bibwewadi Pune
Mr M. R. Kharde	Pravara Rural Engineering College, Loni Ahliyanagr
Mr. Gajanan Vinayak Malvade	Millennium Management Corporation, Dublin

Artificial Intelligence+Lab	
Dr. Neeta Deshpande	RHSapat College of Engg, Management Studies and Research
Dr.Vaishali Tidke	MVPS's KBTCOE Nashik
Dr.ChandrakantD.Kokane	Vishwakarma Institute of Technology, Pune
Dr.Vaibhav Dabhade	MET's Institute of Engineering
Dr. Suvarna Patil	DY Patil International University,Akurdi,Pune
Mrs.MeghaAmeyPatil	RHSapat College of Engg, Management Studies and Research

Next Generation IoT	
Dr. Khairnar Prerna	SVIT, Chincholi, Nashik
Dr. Lavahte Seema S.	Pravara Rural Engineering College, Loni

Soft Computing	
Miss Trupti B Katte	Pravara Rural Engineering College Loni
Mrs Sakshi Mandke	MKSSS Cummins College of Engineering for Women, Pune

Cloud Computing	
Dr.Kamini Ashutosh Shirsath	Sandip Institute of Engineering and Management Nasik
Dr.Renuka Gound	Nutan Maharashtra Institute of Engineering and Technology, Pune
Dr.Madhuri Shinde	MET's Institute of Engineering, Nashik
Mrs.Nanda Kulkarni	Siddhant College of Engineering Pune
Mr.Bhushan Pawar	Software Technology Architect, Congnizant

Design and Analysis of Algorithm	
Dr. Shinde P. Jayshri	MMIT, Pune
Miss Supriya Shirsath	Pravara Rural Engineering College Loni, Ahilyanagar

Machine Learning + Lab	
Prof. Pradip Ghorpade	Vidya Pratishthan'sKamal nayan Bajaj Institute of Engineering& Technology, Baramati
Dr.Sulochana Sonkamble	JSPM Narhe Technical Campus, Pune
Dr.Girisha Bombale	SND COE & RC, Yeola
Prof.Varsha P.Chavan	Anantrao Pawar college of Engineering and Research
Prof.S.D.Pawar	Vidya Pratishthan's Kamalnayan Bajaj Institute of Engineering, Baramati

Data Science and Big Data Analytics + Lab	
Mrs. Shital Sandip Patil	SVIT, Chincholi, Nashik
Mrs. Pranjal Deshmukh	MMIT, Pune

Quantum Artificial Intelligence	
Dr. Depika Adalkar	GH Raisoni college of engineering and Management
Dr. Shinde P. Jayshri	MMIT, Pune
Prof. Pradip Ghorpade	Vidya Pratishthan'sKamal nayan Bajaj Institute of Engineering& Technology, Baramati

Generative AI	
Dr Vaishali P Vikhe	Pravara Rural Engineering College Loni
Mr Sushil Shinde	Credentia Pvt Ltd Pune
Miss Tarranum Sayyed	Team Lead (AI&DS) Credentia Pvt Ltd Pune

Computer Vision	
Dr.VinayS.Nalawade	SB Patil College of Engineering, Indapur
Mrs. Padma Nibhore	MIT AOE, Alandi
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 DY Patil University, Pune
Prof.S.D.Pawar	Vidya Pratishthan's Kamalnayan Bajaj Institute of Engineering, Baramati
Mr.Pradip D.Kulkarni	Visdin Solutions

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

***