Savitribai Phule Pune University

(Formerly University of Pune)

Four Year Degree Program in
Bachelor of Science (B.Sc)

With

Major: ARTIFICIAL INTELLIGENCE (AI) and MACHINE LEARNING (ML)

(Faculty of Science and Technology)

Syllabi for

(For Colleges Affiliated to Savitribai Phule Pune University)

Choice Based Credit System (CBCS) Syllabus
Under National Education Policy (NEP)

To be implemented from Academic Year 2025-2026

Syllabus Structure as per NEP Guidelines
B.Sc. (Artificial Intelligence & Machine Learning)
SY (Level 5.0) SEMESTER III (A.Y. 2025-26)

Course Type	Course code	Course Name	Credits		Teaching Scheme Hrs/Week		Examination Scheme and Marks		
			TH	PR	TH	PR	CE	EE	Total
Major Mandatory	AIML-201-MJ-T	Data Structures	2	--	2	--	15	35	50
	AIML-202-MJ-T	Software Design and Project Management	2	--	2	--	15	35	50
	AIML-203-MJ-P	Lab course on AIML201MJ + AIML202MJ	--	2	--	4	15	35	50
VSC	AIML-221-VSC-P	Advanced Python Programming	--	2	--	4	15	35	50
IKS	AIML-201-IKS	Indian Knowledge System in Computing	2	--	2	--	15	35	50
FP/OJT/ CEP	AIML-231-FP	Mini Project	--	2	--	4	15	35	50
Minor	AIML-241-MN-T	Linear Algebra	2	--	2	--	15	35	50
	AIML-242-MN-P	Practical on AIML241MN	--	2	--	4	15	35	50
GE/OE*	OE-201-AIML-T OE-202-AIML-T OE-203-AIML-T OE-204-AIML-T	E-Commerce-I / Web Design-I/ Digital Marketing-I / AI for everyone - I	2	--	2	--	15	35	50
AEC	AEC-201-ENG	From University Basket	2	--	2	--	15	35	50
CC	CC-201-PE/NSS /NCC	From University Basket	2	--	2	--	15	35	50
		Total	**14**	**08**	**14**	**16**			**550**

* These subjects are offered to other faculty students under GE/OE vertical. The students of B.Sc. (AI & ML) will opt the subjects offered by other faculty given in University Basket.

Syllabus Structure as per NEP Guidelines
B.Sc. (Artificial Intelligence & Machine Learning) from 2025-26
SY (Level 5.0) SEMESTER IV

Course Type	Course code	Course Name	Credits		Teaching Scheme Hrs/Week		Examination Scheme and Marks		
			TH	PR	TH	PR	CE	EE	Total
Major Mandatory	AIML-251-MJ-T	Microservices using Python	2	--	2	--	15	35	50
	AIML-252-MJ-T	Artificial Intelligence - I	2	--	2	--	15	35	50
	AIML-253-MJ-P	Practical based on AIML251MJ + AIML252MJ	--	2	--	4	15	35	50
VSC	AIML-271-VSC-P	Databases – II	--	2	--	4	15	35	50
FP/OJT/ CEP	AIML-281-FP	Mini Project	--	2	--	4	15	35	50
Minor	AIML-291-MN-T	Mathematical Logic	2	--	2	--	15	35	50
	AIML-292-MN-P	Practical based on AIML291MN	--	2	--	4	15	35	50
GE/OE*	OE-251-AIML-T OE-252-AIML-T OE-253-AIML-T OE-254-AIML-T	E-Commerce-II / Web Design-II / Digital Marketing-II / AI for Everyone - II	2	--	2	--	15	35	50
SEC	SEC-251-AIML-T	DAA - I	2	--	--	2	15	35	50
AEC	AEC-251-T	From University Basket	2	--	2	--	15	35	50
CC	CC-251-T	**From University Basket**	2	--	2	--	15	35	50
		Total	**14**	**08**	**12**	**18**			**550**

*These subjects are offered to other faculty students under GE/OE vertical. The students of B.Sc. (AI & ML) will opt the subjects offered by other faculty given in University Basket.

The students of B.Sc (AI &ML) will opt the subjects offered by other faculty given in University Basket

Exit option: Award of UG Diploma in Major and Minor with 88 credits and an additional 4 credits core as per university guidelines

OR

Continue Option: Continue with Major and Minor

Detailed Syllabus

B.Sc (AI & ML)

Semester III

S.Y. B.Sc. (AI and ML)
Semester – III
AIML-201-MJ-T -Data Structures

No. of Credits: 2	Teaching scheme: Theory: 2 hrs/week	Evaluation Pattern: Continuous Evaluation: 15 End Sem Evaluation: 35	Total Lectures: 30

Prerequisites: Python Programming, OOP

Objectives:
The course is designed to teach:

- basic techniques of algorithm analysis using mathematical techniques
- estimate algorithm complexity using step and frequency counts
- iterative and recursive methods for several sub-quadratic sorting algorithms including quicksort, merge sort and heapsort
- basic linear data structures such as stacks and queues and operations on them
- implementation of linked data structures such as linked lists and binary trees
- advanced data structures such as balanced search trees, hash tables and priority queues
- graph algorithms such as shortest path and minimum spanning tree
- use of data structure library of major programming language

Contents:

Unit	Contents	Number of lectures	Text Book
1	Algorithms- Definition, Characteristics, Complexity- Analysis of Algorithms, Asymptotic Analysis, Order of Growth - Best, Average and Worst Cases, Asymptotic Notations, Step & frequency count, Recursion and its types, Introduction to Abstract data types (ADTs), Linear and Non-linear ADTs, Concept of Dynamic memory allocation, Heap memory Vs Stack memory	**6**	**1,2**
2	Array as an ADT, Operations, Searching - Linear Search and Binary search, Sorting- Insertion, Selection, Quick and Merge sort with complexities, Static and Dynamic arrays Array-Based Sequences - Sequence Types, Low-Level Arrays, Referential Arrays, Compact Arrays, Dynamic Arrays and Amortization, implementing a Dynamic Array, Amortized Analysis of Dynamic Arrays, Use of Python's Sequence Types	**8**	**1,2**
3	Stacks - Concept, Operations, Applications, Simple Array-Based Stack Implementation, Reversing Data Using a Stack, Matching Parentheses and HTML Tags Queues - Concept, types, Operations on queues, Applications, Array-Based Queue Implementation Linked list - Concept, types, Operations on lists, Applications, Singly Linked List, implementing a Stack using a Singly Linked List, implementing a Queue using a Singly Linked List, Circularly Linked Lists - Round-Robin Schedulers, implementing a Queue with a Circularly Linked List, Doubly Linked Lists - Basic Implementation	**8**	**1,2,3**

	of a Doubly Linked List, Implementing a Deque with a Doubly Linked List Priority Queues – Concept, Implementing a Priority Queue, Implementation with an Unsorted List, Implementation with a Sorted List		
4	Graphs- Concept, types, Representations, traversals, SP, MCST, Applications Trees – Concept, types, representation, Hash Tables, Tree Structures and Heaps Revision of Python's List and Tuple Classes, String Class, Strings, Files, Lists - List Class, Dictionaries, Tuples, Hash maps – Concept, operations with examples	**8**	**1,2,3**

Outcomes:

After competing this course, a student will be able to:

- CO1: Understand the concept of algorithms and their complexity using mathematical notation, dynamic memory management, abstract data types
- CO2: Evaluate algorithms and data structures in terms of time and memory complexity of basic operations
- CO3: Express the basic types for data structure, implementation and application.
- CO4: Judge the strength and weakness of different data structures.
- CO5: Deduce an appropriate data structure in context of solution of given problem like sorting, searching, insertion and deletion of data
- CO6: Develop programming skills which require to solve given problem involving graphs, trees and heaps
- CO7: Determine bugs in program and compare and contrast the operation of common data structures

Text books:

1. Ellis Horowitz, Sartaj Sahni, Susan Anderson Freed, Fundamentals of Data Structures In C, University Press, 2018, ISBN: 9788173716058
2. Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser, Data Structures and Algorithms in Python, John Wiley & Sons, Inc, 2021 (Indian adaption), ISBN: 9789354247866
3. Benjamin Baka, Python Data Structures and Algorithms, Packt Publishing, 2017, ISBN: 9781786467355

Reference Books:

1. Rance D. Necaise, Data structures and algorithms using Python, John Wiley & Sons, 2011, ISBN: 978-0470618295
2. Narasimha Karumanchi, Data Structure and Algorithmic Thinking with Python, CareerMonk Publications, 2015, ISBN: 9788192107592
3. Brad Miller, David Ranum, Problem Solving with Algorithms and Data Structures using Python, Franklin, Beedle & Associates Inc, 2006, ISBN: 978-1590280539
4. Jay Wengrow, A Common-Sense Guide to Data Structures and Algorithms in Python, Volume 1: Level Up Your Core Programming Skills, 1st Ed., Pragmatic Bookshelf, 2024, ISBN: 9798888650356

Format of theory paper for all S.Y. B.Sc (AI&ML) Computer science subjects

External Examination: 35 marks **Duration: 2.0 Hours**

Question No.	Total number of questions		Marks			Question Pattern
	Total	Compulsory	Each question	With Option Total Marks	Total Marks	
1	8	8	1	8	8	Theory
2	6	4	2	12	8	Theory+ Problems/ Programs
3	3	2	4	12	8	Theory+ Problems/ Programs
4	3	2	4	12	8	Theory+ Problems/ Programs
5	2	1	3	6	3	Theory+ Problems/ Programs
			Total Marks	50	35	

S.Y. B.Sc. (AI and ML)
Semester – III
AIML-202-MJ-T: Software Design and Project Management

No. of Credits: 2	Teaching scheme: Theory: 2 hrs/week	Evaluation Pattern: Continuous Evaluation: 15 End Sem Evaluation: 35	Total Lectures: 30

Prerequisites: Basic knowledge of a programming language and databases

Objectives:
The course is designed to teach to:

1. design & implementation of complex software solutions using state of the software engineering techniques.
2. identify a software system that needs to be developed
3. document methods in form of Software Requirements Specification (SRS) for the identified system.
4. the requirements and design representation methods using of UML (Unified Modeling Languages) for a given case study - the design use cases and develop the use case model.
5. the method of identifying the conceptual classes and develop a Domain Model to derive class diagram from that, using the identified scenarios, find the interaction between objects and represent them, construction of relevant State Chart and Activity Diagrams for the a system/case under study.
6. Implementation of the system as per the detailed design
7. inculcate and excel working capabilities as part of software team and develop significant projects under a tight deadline time / schedule and present the project in a professional manner
8. familiarize the students with the basic features of agile development

Contents:

Unit	Contents	Number of lectures	Text Book
1	**Software Engineering Process**- Introduction to Software Engineering, Software Process, SDLC, Waterfall model **Analysis:** Software Requirement: Functional and Non-functional Requirements, Known and Unknown Requirements. Characteristics of a Good Requirement, Software Requirements Specification Document: Nature of the SRS Document, Organization of the SRS Documents, Requirements Change Management, Overview of Analysis, Analysis Object Models and Dynamic Models, Entity, Boundary, and Control Objects, Structured Analysis versus Object-Oriented Analysis, Identification of Classes: Entity Classes, Interface Classes, Control Classes, Identification of Relationships: Association, Aggregation, Multiplicity, Composition, Dependency, Generalization, Modeling Relationships.	**12**	**1, 2**
2	**Introduction of UML** - Overview of UML, Conceptual Model of UML, Basic and Advanced Structural Modeling and Basic Behavioral Modeling - Class Diagram, Advanced classes, Advanced Relationship, Interface, Types and	**6**	**1,2**

	Roles, Packages, Object Diagram, Use case diagram, activity diagram, state diagram, sequence diagram.		
3	**Object Oriented Design:** Interaction Diagrams, Refinement of Use Case Description, Construction of Detailed Class diagram, Development of Detailed Design and Creation of Software Design Document	**3**	**1-4**
4	**The Agile Methodology-** Need of Agile software development, Introduction to Agile Models - Adaptive Software Development (ASD), Scrum, Dynamic Systems Development Method (DSDM), Crystal, Feature Drive Development (FDD), Lean Software Development (LSD), Agile Modeling (AM), Agile Unified Process (AUP) **Agile context**– Manifesto, Principles, Methods, Values, Roles, Artifacts, Stakeholders, and challenges. business benefits of software agility, Traditional Vs Agile Models **SCRUM:** Need of scrum, Scrum practices –Working of scrum, Project velocity, Burn down chart, Sprint backlog, Sprint planning and retrospective, Daily scrum, Scrum roles– Product Owner, Scrum Master, Scrum Team. Extreme Programming- Core principles, values and practices. Kanban, Feature-driven development, Lean software development **XP:** Introduction, Values, When to use XP, Life Cycle of XP, Values and rules of XP methodology	**9**	**5, 6**

Outcomes:

After competing this course, a student will be able to:

CO1: define software and relate to the importance of software models
CO2: elicit, analyze and specify software requirements through a productive working relationship with various stakeholders of the project
CO3: perform formal analysis on specifications, use SRS to document software requirements
CO4: apply agile methodology to manage a software project
CO5: use UML diagrams for analysis and design.
CO6: build class diagrams for a software project/case
CO7: explain fundamentals of Agile methodology and principles
CO8: identify software project characteristics that would not be suitable for an agile proces and apply Scrum principles
CO9: apply practices of XP and Incremental design

Text Book(s)

1. Grady Booch, Object-Oriented Analysis And Design With applications, Addison-Wesley, second ed., 2019, ISBN: 0-8053-5340-2
2. Ivar Jacobson et. al., Object Oriented Software Engineering: A Use Case Driven Approach, Pearson India, 2002, ISBN: 978-8131704080
3. Roger Y. Lee, Object-Oriented Software Engineering with UML: A Hands-On Approach, Nova Science Publishers Inc, 2019, ISBN: 978-1536147551
4. James Martin and James Odell, Principles of Object-Oriented Analysis and Design, Prentice Hall, 1992, ISBN: 978-0137208715
5. Andrew Stellman, Jill Alison Hart, Learning Agile, O'Reilly, 2015.
6. Robert C. Martin, Clean Code: A Handbook of Agile Software Craftsmanship, PHI, 2017, ISBN: 978-0132350884

Reference Books

1. Roger S. Pressman, Bruce Maxim, Software Engineering: A Practitioners Approach, 9th Edition, Mc Graw-Hill, 2023, ISBN: 978-9355325044
2. Subhash Mehta and Suresh Basandra, Object Oriented Software Engineering, Galgotia Publications Pvt Ltd, 1995, ISBN: 978-8175150584
3. Bernd Bruegge and Allen H. Dutoit, Object-Oriented Software Engineering: Using UML, Patterns and Java, Second Edition, independently published, 2008.
4. E.Gamma, R. Helm, R. Johnson, and J. Vlissides, Design Patterns: Elements of Reusable Object-Oriented Software, Addison-Wesley, 1994, ASIN: B000SEIBB8
5. J. Rumbaugh, I.Jacobson and G. Booch, The Unified Modeling Language Reference Manual, Addison Wesley, 2004, ISBN: 978-0321245625
6. Craig Larman, Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design, and the Unified Process, Prentice-Hall, 2nd Edition, 2001, ISBN: 978-0130925695
7. Andrew Stellman, Jennifer Green, Head first Agile, O'Reilly, 2017.
8. Rubin K, Essential SCRUM: A practical guide to the most popular Agile process, Addison-Wesley, 2013.
9. Agile Development with Scrum, Ken Schwaber & Mike Beedle, Prentice Hall, 2001
10. Integrating Agile Development in the Real World, Peter Schuh, Charles River Media, 2005

S.Y. B.Sc. (AI and ML)
Semester – III
AIML-203-MJ-P- Lab course on AIML201MJ + AIML202MJ
(Data Structures and Software Design and Project Management)

No. of Credits: 2	Practicals per week: 1 (4 hrs)	Evaluation Pattern: Continuous Evaluation: 15 End Sem Evaluation: 35	Total Practicals: 15

Pre-requisites: C++ / Python Programming

Objectives:
The course is designed to teach:

1. Use of step and frequency count to compute the complexity of any code
2. sorting and searching algorithms of polynomial time complexity
3. stack and queue concept and operations on both, developing simple applications
4. various types of linked lists and operations on them.
5. concept of hash tables, graph and trees and their applications
6. requirements gathering and system analysis
7. methods of applying OO Design Principles to build class diagrams
8. use of UML diagrams in design of a software project and designing test cases

Contents:

Unit	Description
1	Implementation of searching techniques: Linear, binary Implementation of sorting techniques: Selection, Insertion, Merge, Quick Sorts
2	Implementation of list- Operations on SLL and DLLs.
3	Implementation of stack and stack operations Implementation of Applications of stack - Implementation of polish notation and its conversion, Parenthesis balancing/checking in expressions Program for recursion removal using stack
4	Program for implementation of queue and operations on it
5	Program for storing data as tree structure and implementation of various traversal Techniques, Heap Sort
6	Program for storing data as graph structure and implementation of various traversal Techniques
7	Program to implement hash table and its operations
8, 9	**Case studies-based approach** to OO Design- Identify a software system that needs to be developed, document the Software Requirements Specification (SRS) for the identified system.
10, 11, 12	Identify use cases and develop the Use Case model Identify the conceptual classes and develop a Domain Model and also derive a Class Diagram from that.
13, 14, 15	Using the identified scenarios, find the interaction between objects and represent them using UML Sequence Diagrams Draw relevant State Chart and Activity Diagrams for the same system

Note: A Lab book shall be prepared related to the practicals that a student needs to perform. All programs should use OOP concepts

Guidelines related to Case Studies:

a) Case studies can be submitted as a group of maximum 3 students.
b) Relevant case study to be completed by every student and a report (hand written/typed) should be submitted that includes (but not limited to) the following:
   1. Description of the case under study
   2. SRS (in proper format)
   3. ERD (if required), Use cases and Class diagram
   4. Sequence Diagram
   5. State Chart/Activity diagram

Outcomes:

After competing this course, a student will be able to:

CO1: understand basic data structures, their implementation and some of their standard applications
CO2: develop the ability to design and analyze basic algorithms
CO3: implement sorting algorithms ad compare and contrast them
CO4: implement basic searching algorithms
CO5: design and analyze real-life programming problem statements and design a solution using appropriate data structures
CO6: use libraries to implement data structures.
CO7: identify a software system that needs to be developed
CO8: document the Software Requirements Specification (SRS) for the identified system
CO9: identify use cases and develop the Use Case model.
CO10: identify the conceptual classes and develop a Domain Model and derive a Class Diagram
CO11: use the identified scenarios, find the interaction between objects and represent them using UML
CO12: Draw Sequence diagrams, State Chart and Activity Diagrams for the case under study

Text Books:

1. Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser, Data Structures and Algorithms in Python, John Wiley & Sons, Inc, 2021 (Indian adaption), ISBN: 9789354247866
2. Ivar Jacobson et. al., Object Oriented Software Engineering: A Use Case Driven Approach, Pearson India, 2002, ISBN: 978-8131704080
3. Robert C. Martin, Clean Code: A Handbook of Agile Software Craftsmanship, PHI, 2017, ISBN: 978-0132350884

Format of Practical slips for AIML-203-MJ-P- Lab course on AIML201MJ + AIML202MJ (Data Structures and Software Design and Project Management)

External Examination: 35 marks **Duration: 3.0 Hours**

Question	Description	Marks
1	Program based on Data Structures	15
2	Case Study based Viva	15
3	Viva	05

S.Y. B.Sc. (AI and ML)
Semester – III
AIML-221-VSC-P - Advanced Python Programming

No. of Credits: 2	Teaching scheme: Theory: 2 hrs/week	Evaluation Pattern: Continuous Evaluation: 15 End Sem Evaluation: 35	Total Lectures: 30

Prerequisites: Python Programming

Objectives:
The course is designed:

1) To enhance programming skills using advanced Python concepts.
2) To introduce functional programming techniques using decorators.
3) To provide hands-on experience in data handling and manipulation using NumPy and pandas.
4) To enable database interaction using Python.
5) To introduce Django framework for web application development.

Contents:

Unit	Contents	Number of lectures	Text Book
1	**Advanced Function & Decorators: -** Functions as first-class objects, Closures and inner functions, Function and class-based decorators, Built-in decorators (@staticmethod, @classmethod, @property)	**4**	**T1(ch5 & ch7), T2(ch 9)**
2	**Introduction to NumPy:** Arrays and their Types – Indexing and Slicing - Array Operations - Reading and Writing Data using NumPy, shape, reshape, iterating, join, split, search, sort, filter **Data Handling with Pandas:** Series and DataFrame - Indexing and Filtering - Data Cleaning and Transformation - Aggregation and Grouping - Importing and Exporting Data	**10**	**T3(ch4, 5, 6,7 & 9)**
3	**Database Connectivity in Python:** Python Database API (DB-API) - Connecting to PostgreSQL – CURD operations, Transactions & Error handling, Basic ORM Overview	**7**	**T4(ch 17)**
4	**Introduction to Django Framework:** Django Installation and Setup - Project Structure and App Creation - Models, Views, Templates (MVT Pattern) - Simple Web Application Development - Admin Interface Basics	**9**	**T5(ch1, 2, 3,5 & 6)**

Outcomes:
After competing this course, a student will be able to:

CO1: Implement reusable and modular code using decorators.
CO2: Apply NumPy and Pandas for data analysis.
CO3: Perform database operations using Python connectors.
CO4: Develop basic web applications using Django
CO5: Integrate Python skills into complete data-to-web pipelines.

Text books:

1. Luciano Ramalho, Fluent Python, O'Reilly Media, 2nd Edition, ISBN: 978-1-492-05635-5
2. Eric Matthes, Python Crash Course, 2nd Edition, ISBN-13: 978-1-59327-928-8

3. Wes McKinney, Python for Data Analysis, O'Reilly Media, 2nd Edition, ISBN: 978-1-449-31979-3
4. Mark Lutz**,** Programming Python, 4th Edition, O'Reilly, ISBN:978-0-596-15810-1
5. William S. Vincent, Django for Beginners, 2nd Edition

Reference Books:

1. Samir Madhavan, Mastering Python for Data Science, Packt publishing, ISBN 978-1-78439-015-0
2. Mark Lutz, Learning Python, O'Reilly Media, 5th Edition, ISBN: 978-1-449-35573-9
3. Antonio Mele, Django by Example, Packt publishing, ISBN 978-1-78439-191-1

Web Resources:

1. Python Documentation: https://docs.python.org/3/
2. NumPy Documentation: https://numpy.org/doc/stable/
3. Pandas Documentation: https://pandas.pydata.org/docs/
4. Django Documentation: https://docs.djangoproject.com/en/stable/
5. Real Python: https://realpython.com
6. GeeksforGeeks Python: https://www.geeksforgeeks.org/python-programming-language/

S.Y. B.Sc. (AI and ML)
Semester – III
AIML-201-IKS: Indian Knowledge System in Computing

No. of Credits: 2	Teaching scheme: Theory: 2 hrs/week	Evaluation Pattern: Continuous Evaluation: 15 End Sem Evaluation: 35	Total Lectures: 30

Objectives:

1. To introduce Vedic mathematical techniques and their relevance to modern computational methods.
2. To understand Nyaya's logical framework and its application in reasoning and AI.
3. To explore the algorithmic structure of Panini's grammar and Chandasastra's binary system in computational linguistics and mathematics.
4. To explore real-world applications of IKS concepts in computational sciences

Contents:

Unit	Contents	Number of lectures
1	**Vedic Mathematics & Computational Thinking-** Introduction to Vedic Mathematics: Origins and importance in ancient India, Sutras and their logical foundation Basic Arithmetic using Vedic Methods: Addition, subtraction, multiplication, and division tricks Algebraic Applications of Vedic Mathematics: Squaring, square roots, cube roots, and factorization	8
2	**Introduction to Nyaya (Indian Logic) –** Introduction to Nyaya Philosophy: Introduction to Nyaya (Indian Logic), Overview of Indian philosophical schools, Importance of Nyaya in logical reasoning, Types of reasoning (Anumana, Pramana, etc.) Nyaya's Four Sources of Knowledge (Pramaṇa): Perception, inference, comparison, verbal Testimony Types of Argumentations in Nyaya Vada (truth-based), Jalpa (debate-focused), Vitanda (criticism) Applications in AI & Machine Learning**:** Logical reasoning models, expert systems, and rule-based AI	8
3	**Nyaya Shastra (Indian Logic) & AI Reasoning-** Introduction to Panini's Astadhyayi: Historical background and linguistic importance Rule-Based System of Sanskrit Grammar: Sutras, meta-rules, recursion, and transformations Chandasastra's Binary logic and combinatorial techniques	8
4	**Jyotish Shastra & AI in Predictive Analytics-** Mind and cognition in Samkhya and Yoga: AI insights Machine Learning and Indian philosophies: Understanding of human cognition in Indian philosophical schools (Advaita, Samkhya and Yoga) Cryptography and Security: Ancient cryptographic methods in Kautilya's Arthashastra, protecting information: analogies from Indian traditions	6

Outcomes:

After competing this course, a student will be able to:

CO1: Understand the computational foundations of Indian Knowledge Systems by applying Vedic mathematical techniques in problem-solving.

CO2: Use Nyaya's logical reasoning in AI and decision-making.

CO3: Explore the connection between Panini's grammar and NLP technologies.

CO4: Recognize the applications of IKS in modern computing fields

Text books:

1. Vedic Mathematics, Jagadguru Swami Bharati Krishna Tirtha, Motilal Banarsidass Publishing House, New Delhi.
2. The Power of Vedic Maths – Atul Gupta, JAICO publishing
3. Nyaya Theory of Knowledge – S.C. Vidyabhusana
4. A Primer of Indian Logic– Kuppuswami Sastri, Hassell Street Press.2021
5. Indian Logic: A Reader– Jonardon Ganeri
6. Aṣṭādhyāyī of Pāṇini (Volumes 1 & 2) – Rama Nath Sharma, Munshirm Manoharlal publication
7. Panini: His Work and Its Traditions – George Cardona, Motilal Banarsidass Publishing House
8. The Mathematics of Metre – Satyanarayana Das
9. Samkhya and Science – Debabrata Sen Sharma
10. Explores the cognitive science aspects of Samkhya and Yoga in AI research.
11. AI and Indian Philosophy– Sangeet Kedia
12. Kautilya's Arthashastra – R. Shamasastry (Translation)
13. History of Indian Cryptography– Subhash Kak
14. Discusses coded messages, steganography, and security concepts in ancient India.
15. Saubhagya Vardhan, AI in Land of Vedas, Notion Press, 2023

S.Y. B.Sc. (AI and ML)
Semester – III
AIML-241-MN-T – Linear Algebra

No. of Credits: 2	Teaching scheme: Theory: 2 hrs/week	Evaluation Pattern: Continuous Evaluation: 15 End Sem Evaluation: 35	Total Lectures: 30

Prerequisites: Basic Mathematics

Objectives:
The course is designed to teach:

1. fundamental concepts such as defining and manipulating vectors in different dimensions, vector spaces and their properties.
2. matrix operations and analyzing linear transformations and their representation using matrices.
3. Solving systems of linear equations, techniques of Gaussian elimination to solve systems of equations and finding the existence and uniqueness of solutions to linear systems.
4. Use of Eigenvalue and Eigenvector analysis, calculating eigenvalues and eigenvectors of matrices.
5. Understanding the geometric meaning of eigenvalues and eigenvectors in relation to linear transformations and its applications

Contents:

Unit	Contents	Number of lectures
1	MATRICES AND SYSTEM OF LINEAR EQUATIONS: Matrices - Definition, types, Matrix operations, elementary transformations, minor, rank, normal form, inverse, adjoint. Row echelon form - Rank - System of linear equations - Consistency - Gauss elimination method - Gauss Jordan method, Cramer's Rule – LU Decomposition VECTOR SPACES: Vector spaces over Real fields - Subspace – Linear space - Linear independence and dependence - Basis and dimension.	**10**
2	LINEAR TRANSFORMATION: Linear transformation, Rank space and null space - Rank and nullity, Dimension theorem (without proof), Matrix representation of linear transformation.	**6**
3	INNER PRODUCT SPACES: Inner product and norms, Properties – Orthogonality, Orthogonal sets, Orthogonal basis, Gram Schmidt orthogonalization process	**6**
4	EIGEN VALUE PROBLEMS AND MATRIX DECOMPOSITION: Determinants, properties of determinants, Eigen values and eigen vectors of linear transformation, Eigen value Problems: Diagonalization. Power method, Jacobi rotation method – Singular value decomposition – QR decomposition.	**8**

Outcomes:
After competing this course, a student will be able to:

CO1: visualize the space of vectors and the interrelation of vectors with matrices.
CO2: apply vectors, inner products, and linear transformations to real world situations
CO3: apply linear transformation and its corresponding matrix

CO4: solve linear systems of equations using a variety of techniques and to select the best technique for a given system
CO5: develop an algebraic understanding of eigenvalues and eigenvectors and eigenspaces.

Text books:

1) Gilbert Strang, Introduction to Linear Algebra, 5 th edition, Wellesley Cambridge Press,2016
2) Linear Algebra by Kenneth Hoffman and Ray Kunze, Pearson Education (low priced edition), New Delhi.
3) Linear Algebra by Stephen H. Friedberg et al Prentice Hall of India Pvt. Ltd. 4th Edition 2007
4) Gilbert Strang Linear Algebra and its Applications, 4th edition,2016
5) Andrilli, S., & Hecker, D. (2016). Elementary Linear Algebra (5th ed.). Elsevier India.
6) Linear Algebra by J.N. Sharma and A.R. Vasista, Krishna Prakashan Mandir, Meerut
7) Matrices by Shanti Narayana, published by S. Chand Publications.
8) Friedberg, Stephen H., Insel, Arnold J., & Spence, Lawrence E. (2003). Linear Algebra (4th ed.). Prentice-Hall of India Pvt. Ltd. New Delhi.

Reference books:

1) Jorge Brasil, Before Machine Learning Volume 1 - Linear Algebra for A.I, Packt Publishing, 2024, ISBN: 9781836208952
2) Jorge Brasil, Before Machine Learning Volume 2 - Calculus for A.I: The fundamental mathematics for Data Science and Artificial Intelligence, Packt Publishing,
3) Lay, David C., Lay, Steven R., & McDonald, Judi J. (2016). Linear Algebra and its Applications (5th ed.). Pearson Education.
4) Kolman, Bernard, & Hill, David R. (2001). Introductory Linear Algebra with Applications (7th ed.). Pearson Education, Delhi. First Indian Reprint 2003.

S.Y. B.Sc. (AI and ML)
Semester – III
AIML-242-MN-P – Practical on Linear Algebra (AIML-241-MN-T)

No. of Credits: 2	Practicals per week: 1 (4 hrs)	Evaluation Pattern: Continuous Evaluation: 15 End Sem Evaluation: 35	Total Practicals: 15

Prerequisites: Basic Mathematics

Objectives:
The course is designed to teach:
1. the space of vectors and the interrelation of vectors with matrices.
2. the concepts of bases, dimension and minimal spanning sets in vector spaces
3. concepts of eigen value eigen vector decomposition of square matrices on SVD.

Contents:

Unit	Description
1,2,3	Problems based on Matrix Operations, elementary transformations, minor, rank , normal form, inverse, adjoint. Row echelon form - Rank - System of linear equations - Consistency - Gauss elimination method - Gauss Jordan method. Cramer's Rule, LU decomposition
4,5	Problems based on Vector Space- Subspace – Linear space - Linear independence and dependence - Basis and dimension.
6,7	Linear transformation - Rank space and null space - Rank and nullity - Dimension theorem (without proof)– Matrix representation of linear transformation
8,9,10	Problems based on Inner product and norms - Properties – Orthogonality, Orthogonal sets, Orthogonal basis, Gram Schmidt orthogonalization process
11-15	Practical based on python Matrix construct, eye(n), zeros (n, m) matrices. Addition, Subtraction, Multiplication of matrices, powers and invers of a matrix. Accessing Rows and Columns, Deleting and Inserting Rows and Columns Determinant, reduced row echelon form, null space, column space, Rank Solving systems of linear equations (Gauss Elimination Method, Gauss Jordan Method, LU- decomposition Method) Eigenvalues, Eigenvectors, and Diagonalization

Outcomes:
After competing this course, a student will be able to:
CO1: employ techniques to classify and solve linear systems of equations
CO2: illustrate the use of matrices and determinants
CO3: utilize vector spaces and linear transformations
CO4: explore the concept of orthogonality in vector spaces
CO5: compute eigenvalues and eigenvectors of matrices

Text books:

1. Gilbert Strang, Introduction to Linear Algebra, 5 th edition, Wellesley Cambridge Press,2016
2. Linear Algebra by Kenneth Hoffman and Ray Kunze, Pearson Education (low priced edition), New Delhi.
3. Linear Algebra by Stephen H. Friedberg et al Prentice Hall of India Pvt. Ltd. 4th Edition 2007
4. Gilbert Strang Linear Algebra and its Applications, 4th edition,2016

5. Andrilli, S., & Hecker, D. (2016). Elementary Linear Algebra (5th ed.). Elsevier India.
6. Linear Algebra by J.N. Sharma and A.R. Vasista, Krishna Prakashan Mandir, Meerut
7. Matrices by Shanti Narayana, published by S. Chand Publications.
8. Friedberg, Stephen H., Insel, Arnold J., & Spence, Lawrence E. (2003). Linear Algebra, (4th ed.). Prentice-Hall of India Pvt. Ltd. New Delhi.

S.Y. B.Sc. (AI and ML)
Semester – III
OE-201-AIML-T – E-Commerce-I

No. of Credits: 2	Teaching scheme: Theory: 2 hrs/week	Evaluation Pattern: Continuous Evaluation: 15 End Sem Evaluation: 35	Total Lectures: 30

Prerequisites:

1. Basic Computer and Internet Knowledge
2. Fundamentals of Business, Commerce and Digital Marketing Basics
3. Financial and Payment Systems

Objectives:

1. To understand basic concepts about E-commerce.
2. To understand the applications of E-commerce.
3. To learn Business model knowledge.
4. To enable knowledge about E-payment system.
5. To get a general idea of M-commerce

Contents:

Unit	Contents	Number of lectures
1	**E- Commerce and Business Model Concepts-** Introduction to E-Commerce: Definition, Goals, Technical Components, Functions, Status, Prospects, Significance, Advantages, Disadvantages E-Commerce Business Models: Major Business to Consumer (B2C) Business Model Portal, E-tailor, Major Business to Business (B2B) Business Mode, E Distributor, E-Procurement, Exchanges Business models in Emerging E-Commerce Areas - C2C, P2P, and B2G, case studies.	7
2	**E-Marketing and E- Commerce Application-** Introduction, Identifying Goals, Definitions, Browsing Behavior Model Online Marketing - E-commerce and retailing, E-commerce and banking E-Advertising –Introduction, Purpose, Goals, advantages, disadvantages. Internet Marketing Trends, Target Markets, E-Branding Marketing Strategies Consumer Online: The Internet Audience and Consumer Behaviors E-cycle of Internet Marketing	7
3	**E-Commerce Payment Method-** The requirements of an electronic payment system, Traditional payment system, electronic payment technology. Electronic payment gateways. B2B electronic payments, Third-party payment processing, electronic or digital currency, characteristics, operation. Online credit card payments and smart cards.	6
4	**E-Commerce Laws & Regulations-**	5

	Introduction to E-commerce Laws, Information Technology (IT) Act, 2000 (India), GDPR (General Data Protection Regulation) – EU, Other Global E-commerce Laws.	
5	**Future Trends in E-commerce-** AI & Chatbots in E-commerce, AR/VR Shopping Experiences, Personalization & Data Analytics, The Role of IoT in E-Commerce.	**5**

Outcomes:

On completion of the course, student will be able to–

CO1: Learn and implement basic concepts and applications of E-Commerce.
CO2: Understand operations of electronic payment system.
CO3: Compare and analyze various business models.
CO4: Understand regulatory framework for E-Commerce.

Text books:

1. Kenneth C. Laudon, E-Commerce: Business, Technology, Society, 4th Edition, Pearson
2. S. J. Joseph, E-Commerce: An Indian perspective, PHI
3. Alan Davidson, E-Commerce Law: National and Transnational Topics
4. Richard Boire, Artificial Intelligence in E-Commerce

S.Y. B.Sc. (AI and ML)
Semester – III
OE-202-AIML-T – Web Design-I

No. of Credits: 2	Teaching scheme: Theory: 2 hrs/week	Evaluation Pattern: Continuous Evaluation: 15 End Sem Evaluation: 35	Total Lectures: 30

Prerequisites:

1. Basic computer knowledge and the ability to work with files.
2. Knowledge of other programming languages will enhance your abilities with HTML and web development.
3. Knowledge and understanding of Internet.

Objectives:

1. To learn HTML tags and programming concepts and techniques.
2. To develop the ability to logically plan and develop web pages.
3. To learn to write, test, and debug web pages using HTML.
4. To learn to design Style sheets.
5. To know to link and publish Web pages.

Contents:

Unit	Contents	Number of lectures
1	**Introduction to Web Design –** Introduction, Working of the Internet, Role of Web Servers, Clients (Communication), Web Browsers Working of the Internet, Intranet and WWW E-Mail Servers and Protocols E-mail Clients and Web Based Mail Access using Browser Messenger Services and Clients (Chat) Advantages and Disadvantages of Internet Concept of effective Web Design (Web site, classification of website, Advantages and Disadvantages. Of website) Fundamental Principles of Web page design and issues	**8**
2	**Getting Started with HTML –** Introduction to scripting Languages HTML Editing Tools WYSISYG Authoring Tools - HTML Script, Basic HTML Document Structure, Common HTML Tags and its attributes, Design HTML Tags, Text Formatting and Styles, Images and Graphics, Button, Formatting and Style, Lists, Hyperlinks Multimedia, Frames, HTML Forms, Linking Web pages, Publishing Web Pages	**6**
3	**Tables-** Table Structure, Table tags, Affecting table appearance, Table troubleshooting, Tips and tricks, Standard table templates, Multipart images in tables	**6**
4	**Frame / Forms –** Introduction to frames, Basic frameset structure, the frame function, appearance and Targeting frames, The Inline (Floating) frames and Frame design tips and tricks	**6**

	Forms: FORM elements, FORM attributes, Unconventional use of FORM elements Demystifying CGI Retrieving parameter value using getParameter () method	
5	**Case Studies-** Case study 1: Creation of forms, small case study to create HTML pages using all the above learnt techniques. Case study 2: Creation of Forms layout designing by using div element with CSS property Case study 3: Create Multiple Web pages link them to publish a small website.	**4**

Outcomes:

On completion of the course, student will be able to–

CO1: Learn and use the HTML Tags
CO2: Understand and resolves written HTML codes.
CO3: Design and develop the page using HTML codes
CO4: Implement and develop Web pages

Text books:

1. Computer Programming For Beginners:Learn The Basics Of HTML5-Joseph Connor
2. The Complete Reference HTML & CSS-Fifth Edition-Thomas A.Powell
3. Learning Web Design: A beginner's Guide to HTML, CSS, Javascript, and Web Graphics - Jennifer Robbins
4. HTML5: The Missing Manual - Matthew MacDonald.

S.Y. B.Sc. (AI and ML)
Semester – III
OE-203-AIML-T – Digital Marketing-I

No. of Credits: 2	Teaching scheme: Theory: 2 hrs/week	Evaluation Pattern: Continuous Evaluation: 15 End Sem Evaluation: 35	Total Lectures: 30

Prerequisites:

Creative & Logical thinking ability, Digital devices operational skills and Knowledge

Objectives:

1. To understand Digital Marketing as the most powerful marketing tool.
2. Learn to create digital marketing artworks.
3. Learn how to use email campaigns, blogging to produce worthwhile, pertinent material that draws in and engages a target audience.

Contents:

Unit	Contents	Number of lectures
1	**History of Digital Marketing –** Introduction Offline digital marketing - Electronic billboards, Radio marketing, T.V. marketing, Phone marketing Online marketing - Search Engine Optimization (SEO), Social media marketing, E-mail marketing Difference between conventional marketing and online marketing.	8
2	**Internet Marketing -** Structure of Website - Team Types of websites - Static Website, Dynamic website, Personal, Commercial, Governmental, Non- profit organization Web Portals - Type of Portals	8
3	**Classification of E-commerce –** Business to Business (B2B) Model Business to Consumer(B2C) Model Consumer to Consumer(C2C) Model Consumer to Business (B2B) Model	8
4	**Case Studies-** Case Study 1: Digital Platform like subscription-based streaming service that offers a vast library of TV shows, movies, documentaries, and original content. (eg. Prime Video, Netflix etc) Case Study 2: Any Private sector bank in India–Digital Transformation and Customer Engagement, Retail banking, corporate banking, credit cards, loans, digital banking, etc. Case Study 3: Multispecialty Hospital in 2- Tier city -Cardiology and Orthopaedic Departments Case Study 4: Content Marketing - like HubSpot's Content Marketing creates valuable content like blog posts, e-books and infographics that address their target audience and establishing themselves as though leaders and attracting potential customers.	6

	Case Study 5: Any Newspaper like The New York Times – Digital Subscription Growth-Transition from print to digital revenue through subscriptions	

Outcomes:

On completion of the course, student will be able to–

CO1: Learn and visualize power to explore new ideas.

CO2: Analyze and develop their Marketing skills.

CO3: Analyze and understand tacts with corporate objectives and produce quantifiable outcomes from e-payment mechanisms.

CO4: Identify and apply target demographics and company objectives, increasing brand awareness and boosting conversions.

Text books:

1. SEO 2025: Learn Search Engine Optimization with Smart Internet Marketing Strategies by Adam Clarke
2. Digital Marketing: Nitin Kamat, Chinmay Kamat (Himalaya Publishing House)
3. Made to Stick: Why Some Ideas Survive and Others Die by Chip Heath and Dan Heath
4. Digital Marketing: Strategy, Implementation, and Practice by Dave Chaffey and Fiona Ellis-Chadwick

S.Y. B.Sc. (AI and ML)
Semester – III
OE-204-AIML-T – AI for Everyone - I

No. of Credits: 2	Teaching scheme: Theory: 2 hrs/week	Evaluation Pattern: Continuous Evaluation: 15 End Sem Evaluation: 35	Total Lectures: 30

Objectives:

1. Understand the basics of Artificial Intelligence (AI) and its subfields.
2. Explore real-world applications of AI across different industries.
3. Gain insights into the ethical, social, and economic implications of AI.
4. Develop an appreciation for the potential of AI to drive innovation and transformation

Contents:

Unit	Contents	Number of lectures
1	**Introduction to Artificial Intelligence –** Definition and scope of AI, Historical overview and key milestones, Differentiating AI from human intelligence, Types of AI tools: Text, image, audio, video, coding, and automation, Where to find free AI tools? (Google AI, Open AI, Hugging Face, etc.)	**8**
2	**AI Subfields -** Machine learning: Supervised, unsupervised, and reinforcement learning, Deep learning and neural networks, Natural language processing (NLP) and computer vision	**6**
3	**Applications of AI** AI in healthcare: Diagnosis, treatment, and medical imaging AI in finance: Fraud detection, algorithmic trading, and risk assessment AI in transportation: Autonomous vehicles and traffic optimization AI in customer service and chatbots AI in education: Personalized learning and intelligent tutoring systems	**8**
4	**Ethical and Social Implications of AI -** Bias and fairness in AI systems, Privacy and data protection concerns, Impact of AI on employment and the workforce, AI and social inequality	**8**

Outcomes:

On completion of the course, student will be able to–

CO1: Learn and analyse the fundamental concepts and subfields of AI.
CO2: Understand the potential of AI to drive innovation and transformation in different domains.
CO3: Identify and apply AI tools across various industries.
CO4: Analyse and apply the ethical, social, and economic implications of AI.

Text books:

1. Artificial Intelligence: A Guide for Thinking Humans" – Melanie Mitchell
2. The AI Revolution in Medicine: GPT-4 and Beyond" – Peter Lee, Carey Goldberg, Isaac Kohane
3. AI 2041: Ten Visions for Our Future" – Kai-Fu Lee, Chen Qiufan
4. The Business of AI: AI Technologies and How to Leverage Them for Business Success" – Anirudh Koul
5. AI-Powered Marketing: Harness the Future of Marketing with AI" – Peter Gentsch
6. The AI Marketing Handbook" – Ryan McKenzie

Detailed Syllabus

B.Sc (AI & ML)

Semester IV

S.Y. B.Sc. (AI and ML)
Semester – IV
AIML-251-MJ-T- Microservices using Python

No. of Credits: 2	Teaching scheme: Theory: 2 hrs/week	Evaluation Pattern: Continuous Evaluation: 15 End Sem Evaluation: 35	Total Lectures: 30

Prerequisites: Python Programming

Objectives:
The course is designed to teach:

1. To introduce students to the concept of Microservices architecture, including its benefits, challenges, and differences compared to monolithic architectures.
2. To provide hands-on knowledge of Django REST Framework (DRF) for building RESTful APIs, focusing on both basic and advanced API development techniques.
3. To enable students to design and implement REST-based microservices using Django, including service decomposition, database design, and inter-service communication.
4. To familiarize students with practical tools and techniques for logging, error handling, and basic security in microservices-based applications.
5. To expose students to real-world case studies and best practices for designing reliable, maintainable, and secure microservices applications.

Contents:

Unit	Contents	Number of lectures	Text Book
1	**Introduction to Microservices:** What are Microservices? Monolithic vs Microservices Architecture, Characteristics and Advantages of Microservices, Real-world Examples and Case Studies, Challenges in Microservices Implementation	**4**	**T2 (ch1 & ch2)**
2	**Introduction to Django REST Framework (DRF):** Django REST Framework (DRF), Differences between Django (Web) and Django REST Framework (API), REST API Basics (HTTP methods – GET, POST, PUT, DELETE), Installing DRF in an existing Django project Configuring DRF in settings.py, Creating a simple Django project with DRF installed	**4**	**T1 (ch1 & 2)**
3	**Developing REST APIs with Django REST Framework:** Creating Models, Serializers, and Views, Class-based vs Function-based Views, CRUD Operations using DRF, URL Routing, Testing APIs using Postman	**8**	**T1 (ch4, ch5, ch6 & ch7)**
4	**Microservices Design and Inter-service Communication:** Decomposing Applications into Microservices, Service Separation - Database per Service, REST-based Communication Between Microservices, Basic Error Handling between Services, Service Discovery and Basic API Gateway Concept (Overview)	**8**	**T3 (ch3 & ch5)**
5	**Basic Logging and Error Handling:**	**6**	**T2(ch8 & ch9)**

	Basic Logging in Microservices, Structured Logging in Django, Basic Security for Microservices - Authentication and Authorization Overview, Error Handling and Circuit Breaker Pattern (Introduction)		

Outcomes:

After competing this course, a student will be able to:

- CO1: Explain the concepts, characteristics, advantages, and challenges of Microservices architecture, and differentiate it from Monolithic architecture.
- CO2: Set up and configure Django REST Framework (DRF) within Django projects and develop basic REST APIs using serializers, views, and URL routing
- CO3: Build fully functional REST APIs using both Function-based Views (FBVs) and Class-based Views (CBVs), handle CRUD operations, and test them using Postman.
- CO4: Design microservices-based systems, including service decomposition, database separation, and REST-based communication between services.
- CO5: Apply basic logging, error handling techniques, and security practices (such as authentication and authorization) in microservices applications using Django.
- CO6: Demonstrate the ability to use structured logging, error handling mechanisms, and API gateways to enhance the resilience and observability of microservices applications.

Text books:

1. William S. Vincent, Django for APIs: Build Web APIs with Python and Django, Lean Publishing; 3.1
2. Sam Newman, Building Microservices, O'Reilly, ISBN13: 978-1-491-95035-7
3. Tarek Ziade, Python Microservices Development, Packt Publishing; ISBN13: 978-1-78588-111-4

Reference Books:

1. Chris Richardson, Microservices Patterns, Manning Publications Co, ISBN: 9781617294549
2. Shayank Jain, Designing Microservices Using Django, BPB Publications, First Edition, 2020, ISBN: 978-93-89328-790
3. Akos Hochrein, Designing Microservices with Django, Apress, ISBN13: 978-1-4842-5358-8

Web Resources:

1. Django REST Framework Documentation: https://www.django-rest-framework.org
2. https://learndjango.com/courses/django-for-apis/
3. Microservices.io (Design Patterns and Concepts): https://microservices.io
4. Real Python – Django & REST Framework Tutorials: https://realpython.com
5. TutorialsPoint – Django Tutorial: https://www.tutorialspoint.com/django

S.Y. B.Sc. (AI and ML)
Semester – IV
AIML-252-MJ-T- Artificial Intelligence – I

No. of Credits: 2	Teaching scheme: Theory: 2 hrs/week	Evaluation Pattern: Continuous Evaluation: 15 End Sem Evaluation: 35	Total Lectures: 30

Prerequisites: Python Programming, Object Oriented Programming, Logic

Objectives:
The course is designed to teach:

1. fundamental concepts of Artificial Intelligence (AI)
2. architecture and functioning of intelligent agents
3. identification of problem-solving components
4. different informed and uninformed search algorithms.
5. search method using classical approach of a problem-space.
6. concept of constraint satisfaction problem and various search methods.
7. implementation of knowledge representation and explore reasoning under uncertainty

Contents:

Unit	Contents	Number of lectures	Text Book
1	**Introduction - W**hat Is AI?, The Foundations of Artificial Intelligence, Brief History of Artificial Intelligence, Applications-State of the Art, Intelligent Agents- Agents and Environments, The Concept of Rationality, The Nature of Environments, The Structure of Agents	**6**	**1**
2	**Problem-solving - Solving Problems by Searching -** Problem-Solving Agents, Example Problems, Searching for Solutions, Uninformed Search Strategies (BFS, DFS, uniform cost search), Informed (Heuristic) Search Strategies (Generate and Test, best-first search, A* search, Hill climbing and simulated annealing), Problem Reduction, Mean- End Analysis	**8**	**1,2**
3	**Beyond Classical Search-** Local Search Algorithms and Optimization Problems, Local Search in Continuous Spaces, Searching with Nondeterministic Actions and Partial Observations, Online Search Agents and Unknown Environments Constraint Satisfaction Problems (CSP)- Defining Constraint Satisfaction Problems, Constraint Propagation: Inference in CSPs, Backtracking Search for CSPs, Local Search for CSPs	**8**	**1,2**
4	**Knowledge-Based Agent in Artificial intelligence:** Knowledge representation, Approaches in Knowledge representation, Issues in knowledge representation, First-Order Logic (FOL), Inference in first-order logic. Uncertainty Knowledge and Reasoning- Probabilistic reasoning. Weak slot-and-filler structure- semantic net, Strong slot-and-filler structure- Conceptual Dependency (CD), scripts.	**8**	**1,2**

Outcomes:

After competing this course, a student will be able to:

CO1: identify and apply suitable intelligent agents for various ai applications
CO2: interpret AI methods and describe their foundational principles and applications in solving real-world problems.
CO3: analyze and illustrate the role of search techniques in problem solving.
CO4: demonstrate knowledge of reasoning and knowledge representation techniques to address and solve real-world problems in AI.
CO5: implement ideas underlying modern logical inference systems
CO6: represent complex problems using expressive but constrained languages like semantic networks, FOL, allowing for efficient reasoning and problem-solving

Text books:

1. Russell, S. and Norvig, P. 2015. Artificial Intelligence - A Modern Approach, Prentice Hall, 3rd edition, ISBN: 978-0-13-604259-4.
2. Rich, E., Knight, K and Shankar, B. 2009. Artificial Intelligence, Tata McGraw Hill, 3rd edition, ISBN:978-0-07-008770-5

Reference Books:

1. Introduction to Artificial Intelligence & Expert Systems, Dan W Patterson, PHI., 2010., ISBN: 0134771001.
2. Introduction to Machine Learning, Alpaydin, E. 2010., MIT, 2nd edition, ISBN: 978026201243-0.
3. Nilsson Nils J, Artificial Intelligence: A new Synthesis, Morgan Kaufmann Publishers Inc. San Francisco, CA, ISBN: 978-1-55-860467-4.

S.Y. B.Sc. (AI and ML)
Semester – IV
AIML-253-MJ-P -Practical based on Microservices using Python and Artificial Intelligence – I
(AIML-251-MJ-T + AIML-252-MJ-T)

No. of Credits: 2	Practicals per week: 1 (4 hrs)	Evaluation Pattern: Continuous Evaluation: 15 End Sem Evaluation: 35	Total Practicals: 15

Prerequisites: Python Programming

Objectives:
The course is designed to teach:

1. Installation of the Django REST framework
2. Use of Django REST framework in applications
3. creating models, serializers, and views, CRUD operations and API testing
4. implement microservice and handle errors
5. the study and design of the state space representation for an AI problem
6. develop code for searching in a state space representation
7. use of CSP in local search
8. few methods of knowledge representation and knowledge inference

Contents:

Unit	Description
1	Installation and real life case study
2, 3, 4	Django framework usage, Django REST Framework, Programs based on use of HTTP methods – GET, POST, PUT, DELETE, Installing DRF in an existing Django project, Configuring DRF in settings.py, Creating a simple Django project with DRF installed
5, 6,	Programs on creating models, serializers, and views, Class-based vs Function-based Views, CRUD Operations using DRF, URL Routing, Testing APIs using Postman
7, 8, 9	Implementing microservice, REST-based Communication Between Microservices, Basic Error Handling between Services, Service Discovery Error handling Programs
10	Identify a case study involving rational agents, study of any rational agent, their environments
11	State space representation of an AI agent
12	Implementing Search algorithms
13	Problem on CSP and use of local search algorithms in CSP
14, 15	Knowledge Representation exercise – FOL, Semantic Nets, CD Inferencing using FOL

Outcomes:
After competing this course, a student will be able to:

CO1: obtain an understanding of microservices: the benefits, the challenges, and how using microservices differs from a monolithic approac
CO2: design and build robust, user friendly, scalable applications in microservices using Python
CO3: deploy microservices using Docker
CO4: demonstrate use of error handling with exception handling

CO5: construct a state space representation for an AI case study
CO6: solve AI problems using searching and backtracking
CO7: implement simple reasoning systems using either backward or forward inference
CO8: derive the best-case scenario of knowledge representation for a given AI case study

Text books:

1. Russell, S. and Norvig, P. 2015. Artificial Intelligence - A Modern Approach, Prentice Hall, 3rd edition, ISBN: 978-0-13-604259-4.
2. Rich, E., Knight, K and Shankar, B. 2009. Artificial Intelligence, Tata McGraw Hill, 3rd edition, ISBN:978-0-07-008770-5

Format of Practical slips for AIML-253-MJ-P
Practical based on Microservices using Python and Artificial Intelligence – I
(AIML-251-MJ-T + AIML-252-MJ-T)

External Examination: 35 marks **Duration: 3.0 Hours**

Question	Description	Marks
1	Program based on Microservices using Python	15
2	Problem solving / Program based on Artificial Intelligence - I	15
3	Viva	05

S.Y. B.Sc. (AI and ML)
Semester – IV
AIML-271-VSC-P- Databases – II

No. of Credits: 2	Practicals per week: 1 (4 hrs)	Evaluation Pattern: Continuous Evaluation: 15 End Sem Evaluation: 35	Total Practicals: 15

Prerequisites: Basic DBMS Concepts

Objectives:
The course is designed to teach hands-on:

1. concept and creating views
2. use of stored procedures and dynamic queries
3. concept of functions and triggers and its use
4. concepts of transactions and implementation of locking concepts

Contents:

Unit	Practical Title
1	Revision of basic DDL and DML Commands (using PostgreSQL)
2, 3	**Views-** Concept, Basic Views (Create views), Updatable Views, Recursive Views, Managing Views – Alter and list views in current database
4,5,6,7,8	**Stored Procedures** - Anatomy of a Stored Procedure (CREATE, DECLARE, BEGIN, END, Nested Declare- Begin-end), Alias, RENAME, Simple and Composite variables, Function Arguments, Declarations, PERFORM, Execution Control Structures (RETURN, Exceptions and Messages, IF-THEN-ELSE, NULLIF, CASE, LOOP, WHILE, FOR), Dynamic Queries- EXECUTE.
9,10	**SQL functions -** Basic functions, functions returning a set of elements, functions returning a table, Polymorphic SQL functions Control structures: Conditional statements: IF statements, CASE statements, Loop statements, The record type, Exception handling statements Deleting Functions
11,12	**Triggers:** OLD, NEW variables, Rules on INSERT- ALSO, INSTEAD OF, Rules on UPDATE / DELETE - Creating the new_tags table, creating two tables, managing rules on INSERT, DELETE, and UPDATE events: INSERT rules, DELETE rules, UPDATE rules, Event Triggers
13- 15	**Transaction and Locking**- Overview of Grouping Data Changes into Logical Units, Concurrent Multiuser Access to Data, ACID Rules, Transaction Logs. Transactions with a Single User- Transactions Involving Multiple Tables, Transactions and Savepoints, Transaction Limitations

Outcomes:
After competing this course, a student will be able to:

CO1: create views for different class of users
CO2: write and implement functions on tables
CO3: build triggers on tables to validate data being inserted/deleted/modified in tables
CO4: implement stored procedures to handle complex computing tasks on databases
CO5: apply the concept of transactions and deadlock avoidance

Text books:

1. Neil Matthew And Richard Stones, Beginning Databases with PostgreSQL From Novice to Professional, APress, Second Edition, 2005, ISBN (pbk): 1-59059-478-9

Reference Books:

1. Luca Ferrari, Enrico Pirozzi, Learn PostgreSQL: Use, Manage, and Build Secure and Scalable Databases with PostgreSQL 16, Packt Publishing; 2nd ed. Edition, ISBN: 978-1837635641
2. Regina Obe and Leo Hsu**,** PostgreSQL: Up and Running: A Practical Guide to the Advanced Open Source Database, O'Reilly, 3rd ed., 2017, ISBN: 978-1491963418

S.Y. B.Sc. (AI and ML)
Semester – IV
AIML-281-FP – Mini Project

No. of Credits: 2	Practicals per week: 1 (4 hrs)	Evaluation Pattern: Continuous Evaluation: 15 End Sem Evaluation: 35	Total Practicals: 15

Prerequisite:
Student should have completed the AIML-231-FP course

Objectives:
The course is designed to teach-

1. equip students with practical skills and knowledge to successfully design software projects.
2. methods to inculcate and excel working capabilities as part of software term and develop significant
3. methods of knowledge acquisition, skill development, and a deeper understanding of real-world applications
4. managing projects using a suitable agile methodology.
5. presentation of projects in a professional manner
6. approaches to put to the test the knowledge students have acquired in classrooms, and to acquire new knowledge and skills directly related to the issues and realities they encounter in real life like software projects.

Contents:

Unit	Description
1-12	Implementation of the case studied in semester III
13, 14	Reports generated (if any), experiences while undergoing the course
15	Report writing and Presentation of work via demos of developed project

Outcomes:
After competing this course, a student will be able to:

CO1: Use software engineering techniques to analyse user requirements and documentation methods of a software project
CO2: Develop engineering solutions to complex problems by designing and developing appropriate UI, business logic, reports etc.
CO3: Use agile techniques to effectively manage process involved in software development
CO4: Implement effective methods to communicate with stakeholders and work collaboratively in teams
CO5: develop cross-disciplinary skills of implementation of the case

S.Y. B.Sc. (AI and ML)
Semester – IV
AIML-291-MN-T - Mathematical Logic

No. of Credits: 2	Teaching scheme: Theory: 2 hrs/week	Evaluation Pattern: Continuous Evaluation: 15 End Sem Evaluation: 35	Total Lectures: 30

Objectives:

The course is designed to teach:

1. to formalize reasoning in symbolic languages with precisely defined meanings and precisely defined rules of inference.
2. translate fragments of natural language into the symbolic languages
3. give mathematically precise meanings (semantics) to the terms and sentences of the symbolic languages
4. construct formally correct arguments in the logics, mirroring valid arguments in mathematical, philosophical, or ordinary reasoning
5. understand the idea—and some specific examples—of algorithms for deciding the validity or consistency of logical formulas, as well as the idea of undecidability
6. use the precise syntax and semantics of predicate logic to disambiguate sentences of natural language
7. a concise base of Mathematical Logic

Contents:

Unit	Contents	Number of lectures
1	**Basic Set Theory and Semi-formal Introduction to Propositional Logic**- What is propositional logic? From natural language to logical notation, Logical connectives, Conditional statements, Validity of Arguments **Formal Syntax and Semantics of Propositional Logic (PL)**- The Language of Propositional Logic –Formulas, Construction, Induction, Recursion, Semantics for PL- Truth, Valid Argument Forms, Validity, Equivalence Semantics for the Propositional Language- Satisfiability	**10**
2	**Basic Theory of Propositional Logic (**Natural Deduction) – Truth Functions, Truth-Functional Completeness, An Algorithm for CNF **Proofs in Propositional Logic-** Conditional Introduction, Conditional Elimination, Reiteration, Conjunction, Biconditional, Negation Introduction, Negation Elimination, Reductio Ad Absurdum, Disjunction Introduction, Disjunction Elimination	**10**
3	**Syntax and Semantics of Monadic Predicate -** Syllogistic Logic, Sets and Operations on Sets, relation between sets and propositions, Pure Monadic Predicate Logic, Constants, Function Symbols, The Identity Predicate, Substitution, Syntax and Semantics of Full Predicate Logic- Predicates of Higher Arity, Functions of Higher Arity **Proofs for Predicate Logic** – Identity, Universal Elimination, Universal Introduction, Existential Introduction, Existential Elimination	**10**

Outcomes:

After competing this course, a student will be able to:

CO1: Use truth tables and logical operators to solve problems.
CO2: Translate between narrative arguments and propositional logic
CO3: Prove logical equivalency, contingency, tautology, and contradictions
CO4: explain and apply basic notions of symbolic logic
CO5: analyze natural language arguments by means of symbolic propositional logic
CO6: analyse propositions and arguments in propositional logic by natural deduction method
CO7: Apply inference rules to solve problems
CO8: Prove or disprove assertions using predicate logic

Text Books:

1. Stephen Cole Kleene, Mathematical Logic, Dover Books on Mathematics), 2002, ISBN: 978-0486425337
2. Dirk van Dalen, Logic and Structure, 4th Edition, Springer, 2008, ISBN: 978-3540208792
3. P. T. Johnstone Notes on Logic and Set Theory, Cambridge University Press, 1st Edition, 1987, ISBN: 978-0735587298

Reference Books:

1. Raymond Smullyan, A Beginner's Guide to Mathematical Logic, Dover Publications Inc, 2014, ISBN: 978-0486492377
2. Laszlo Csirmaz, Zalán Gyenis, Mathematical Logic: Exercises and Solutions (Problem Books in Mathematics, 2023, ISBN: 978-3030790127
3. Yu. I. Manin, B. Zilber, Neal Koblitz, A Course in Mathematical Logic for Mathematicians, Springer; 2nd ed. 2010 edition, ISBN: 978-1441906144
4. Ian Chiswell, Wilfrid Hodges, Mathematical Logic, Oxford University Press, 1st Edition, 200 ISBN: 978-0198571001
5. Angelo Margaris, First Order Mathematical Logic, Dover Publications Inc, 2003, ISBN: 978-0486662695
6. http://www.logicinaction.org/

S.Y. B.Sc. (AI and ML)
Semester – IV
AIML-292-MN-P- Practical based on Mathematical Logic (AIML-291-MN-T)

No. of Credits: 2	Practicals per week: 1 (4 hrs)	Evaluation Pattern: Continuous Evaluation: 15 End Sem Evaluation: 35	Total Practicals: 15

Objectives

The course is designed to teach:

1. teach the application of the framework for defining mathematical statements, proofs, and the relationships between them
2. develop skills in formulating proofs systematically
3. provide hands-on experience with logic and verify the correctness of their reasoning
4. understand the limitations of formal reasoning and the nature of mathematical truth
5. apply logical concepts to real-world scenarios and reinforce theoretical understanding and improve analytical and problem-solving abilities

Contents:

Unit	Practical based on AIML-291-MN-T
1,2	Basic set theory problem solving, Problems based on representing natural language to logical notation
3, 4	Problem solving using prepositional logic
5, 6	CNF representation
7,8	Problems based on proof in PL
9,10	Problem solving based on Syllogistic Logic
11, 12	Problems based on Pure Monadic Predicate Logic
13 – 15	Problems based on Proofs for Predicate Logic

Outcomes:

After competing this course, a student will be able to:

CO1: translate natural language into logical formulas
CO2: explain validity of natural language arguments from the symbolic logical point of view
CO3: define proposition and argument, explain propositional connectives, explain and exemplify truth value status of a proposition, explain and and exemplify validity of an argument, explain and exemplify tautology, contradiction and contingency
CO4: analyze natural language arguments by means of symbolic propositional logic
CO5: represent complex statements and relationships using predicates, variables, and quantifiers
CO6: understand the syntax and semantics of predicate logic
CO7: perform logical inferences predicate logic expressions and apply predicate logic concepts in fields of artificial intelligence viz. knowledge representation

Text books:

1. Stephen Cole Kleene, Mathematical Logic, Dover Books on Mathematics), 2002, ISBN: 978-0486425337
2. Dirk van Dalen, Logic and Structure, 4th Edition, Springer, 2008, ISBN: 978-3540208792
3. P. T. Johnstone Notes on Logic and Set Theory, Cambridge University Press, 1st Edition, 1987, ISBN: 978-0735587298

S.Y. B.Sc. (AI and ML)
Semester – III
SEC-251-AIML-T- Design and Analysis of Algorithms - I

No. of Credits: 2	Teaching scheme: Theory: 2 hrs/week	Evaluation Pattern: Continuous Evaluation: 15 End Sem Evaluation: 35	Total Lectures: 30

Prerequisites: Data Structures, Basic Algebra

Objectives:
The course is designed to teach the:

1. notion of pseudo conventions used in algorithms and the use of the RAM model
2. concept of complexity of an algorithm using mathematical notations
3. guidelines for asymptotic analysis and properties of asymptotic notations
4. use of recurrence relations and methods to solve recurrence relations
5. use of Brute force / Exhaustive search strategy and its pros and cons
6. application of divide and conquer and its variations in various problems

Contents:

Unit	Contents	Number of lectures	Text Book
1	**Introduction -** Notion of an Algorithm: Pseudo conventions – variables, data types, data structures, recursive algorithms, tail recursion- usage & advantages, the Random Access Machine (RAM) model. **Algorithmic Complexity -** How to Compare Algorithms? What is Rate of Growth? Commonly Used Rates of Growth, Types of Analysis, Asymptotic Notation, Big-O Notation [Upper Bounding Function], Omega-Ω Notation [Lower Bounding Function], Theta-Θ Notation [Order Function], Why is it called Asymptotic Analysis? Guidelines for Asymptotic Analysis, Simplifying properties of asymptotic notations, Practical complexities	**8**	**1,2**
2	**Complexity analysis**: - sorting algorithms: Insertion, Merge, Quick, Counting, Radix, Shell Sort Searching algorithms: Linear, Binary, Jump, Fibonacci Search and the Ubiquitous Binary Search **Algorithm Classes-** Recursion or Iteration, Exact or Approximate, Serial or Parallel or Distributed Algorithms	**8**	**1,2**
3	**Algorithm Classification by Design method -** Brute Force / Exhaustive Search– General method, Selection sort, String Matching and substring search, Polynomial evaluation, Closest-Pair and Convex-Hull Problems, Travelling Salesman Problem, Knapsack Problem, Coin change problem, Brute Force - Strengths and Weaknesses,	**6**	**1,2**

4	**Divide and Conquer -** Binary search, Recurrence equation for divide and conquer, Merge sort, Quick sort, advantages and disadvantages of divide and conquer. **Decrease and Conquer Approach-** Introduction**,** top down and bottom-up approach **Decrease by a constant-** Introduction, Application to - Insertion Sort, Depth First Search, Breadth First Search, Topological Sorting, Algorithms for Generating Combinatorial Objects **Decrease by constant factor** - Introduction, Application to - binary search, fake-coin, Russian peasant multiplication **Variable size decrease-** Introduction, Application to - Computing median and selection problem, Interpolation Search, Euclid's algorithm **Transform and Conquer** Approach - Pre-sorting, Gaussian Elimination, Balanced Search Trees, Heaps and Heap sort, Problem Reduction	**8**	**1,2**

Outcomes:

After competing this course, a student will be able to:

CO1: compute the complexity of an algorithm using mathematical concepts

CO2: interpret recurrence relations and find the complexity for the same using different methods

CO3: understand the classes of algorithm and their strategies

CO4: derive a solution to a computational problem using the brute force approach and explore the pros and cons of the applied strategy

CO5: apply various forms of the divide and conquer strategy to problems and derive the complexity

Text books:

1) Ellis Horowitz, SatrajSahni and Rajasekharam, *Fundamentals of Computer Algorithms*, Galgotia publications Pvt.Ltd.,2007, ISBN: 978- 8173716126

2) T Cormen, C Leiserson, R Rivest, *Introduction to Algorithms,* MIT Press, Third Edition*,* 2010, ISBN: 978-0262033848

Reference Books:

1) Donald Knuth, *Art of Computer Programming, The: Volume 1: Fundamental Algorithms (Art of Computer Programming Volume 1)*, Third edition, Addison-Wesley, 1997, ISBN: 978- 0201896831

2) Donald Knuth, *Art of Computer Programming, Volume 3: Sorting and Searching*, Second Edition, Addison-Wesley,1998, ISBN: 978- 0201896855

3) Steven S Skiena, *The Algorithm Design Manual*, Second Edition, Springer, 2008, ISBN: 978-1849967204

4) Parag Himanshu Dave, Himanshu Bhalchandra Dave, *Design and Analysis Algorithms*, Publisher: Pearson, ISBN:978-8177585957

S.Y. B.Sc. (AI and ML)
Semester – IV
OE-251-AIML-T – E-Commerce-II

No. of Credits: 2	Teaching scheme: Theory: 2 hrs/week	Evaluation Pattern: Continuous Evaluation: 15 End Sem Evaluation: 35	Total Lectures: 30

Prerequisites:

1. Knowledge of basic E-Commerce concepts, business models and payment systems.
2. Understanding of digital marketing, financial systems, and internet technologies

Objectives:

1. To understand the technical and security aspects of E-Commerce.
2. To explore data-driven decision-making and analytics in E-Commerce.
3. To study supply chain and logistics management in E-Commerce.
4. To gain insights into global E-Commerce trends and challenges.
5. To learn about the integration of AI, Blockchain, and Cloud Computing in E-Commerce.

Contents:

Unit	Contents	Number of lectures
1	**E-commerce Logistics and Supply Chain Management** E-Commerce Supply Chain Overview, Inventory Management in E-Commerce, Role of Warehousing and Fulfilment Centers, Last-Mile Delivery Challenges, Reverse Logistics and Return Policies, Green and Sustainable Supply Chains	6
2	**Data Analytics and Consumer Behavior in E-commerce-** Importance of Data Analytics in E-Commerce, Key Performance Indicators (KPIs) and Metrics, Web Analytics (Google Analytics, Heatmaps, A/B Testing), Predictive Analytics & Customer Insights, Recommendation Engines & Personalization, Conversion Rate Optimization (CRO), Fraud Detection Using AI in E-Commerce	7
3	**E-Commerce Security and Privacy-** Importance of Security in E-commerce, Threats to E-Commerce (Phishing, Fraud, Cyber Attacks, Cryptography & Secure Transactions (SSL/TLS, Encryption), Digital Signatures & Certificates, Firewalls & Intrusion Detection Systems	6
4	**Advanced E-Commerce Technologies-** Machine Learning for Product Recommendations, Block chain for Secure Transactions and Smart Contracts, Cloud Computing and SaaS Platforms for E-Commerce, The Role of 5G in E-Commerce Growth	5
5	**Global Trends of E-Commerce –** Cross-Border E-Commerce and Global Expansion, Mobile Commerce (M-Commerce) Innovations, Subscription-Based E-Commerce Models, Social Commerce (Instagram, Facebook Shops, TikTok Commerce), Ethical and Sustainable E-commerce Practices, Future Challenges in E-commerce	6

Outcomes:

On completion of the course, student will be able to–

CO1: Implement secure E-Commerce transactions and protect user data.

CO2: Apply analytics tools to track and enhance E-Commerce performance.
CO3: Manage E-Commerce logistics and understand global trends.
CO4: Use emerging technologies such as AI, Blockchain, and Cloud for E-Commerce applications.

Text books:

1. E-Commerce Analytics: Analyze and Improve the Impact of Your Digital Strategy, Judah Phillips, Pearson
2. Cloud Computing: Principles and Paradigms, Rajkumar Buyya, James Broberg, Andrzej Goscinski, Wiley
3. Global E-Commerce: Theory and Case Studies, Jie Lin, Fei Gao, Springer
4. Logistics and Supply Chain Management, Martin Christopher, Pearson

S.Y. B.Sc. (AI and ML)
Semester – IV
OE-252-AIML-T – Web Design-II

No. of Credits: 2	Teaching scheme: Theory: 2 hrs/week	Evaluation Pattern: Continuous Evaluation: 15 End Sem Evaluation: 35	Total Lectures: 30

Prerequisites:

1. Knowledge and understanding of HTML is essential for structuring web pages.
2. Basic design principles can enhance your web design abilities.
3. Knowledge of programming concepts like variables, loops, and functions can be helpful when learning JavaScript.

Objectives:

1. To learn to define the structure and content of XML documents using XML.
2. To know and learning how to use the DOM to access and manipulate XML data within applications.
3. To prepare the learners with the fundamentals of CSS programming and scripting languages.
4. Learners should know how to create and interact with web pages effectively, develop static and dynamic websites, and understand how they work together.

Contents:

Unit	Contents	Number of lectures
1	**CSS(Cascading Style Sheets) –** Introduction of CSS and its Syntax, Ways to Insert CSS and Background image handling, Background colour management using CSS, Text and Font management using CSS, Managing Hyperlinks and List using CSS, Designing Borders and Outline, Setting Page Margin using CSS	**6**
2	**XML(Extensible Markup Language) -** XML Namespaces and Infoset and Document Type, Definitions (DTDs), XML Schemas and XML-Parser, Data Modeling, Document and Object Model (DOM), Displaying XML with XSLT	**6**
3	**Introduction to JavaScript –** Concept of Script, Types of Scripts, Scripting Languages, Introduction to JavaScript, Variables, identifier and Operator, Control structure, Examples on JavaScript Operators, Functions, Event Handling in JavaScript with examples.	**8**
4	**Case Studies-** Case study 1: Creation of forms, small case study to create HTML pages using all the above learnt techniques. Case study 2: Redesigning the Website of a Small Business. Case study 3: Create a Styled Web Page for a Coffee Shop.	**10**

Outcomes:

On completion of the course, student will be able to–

1. Apply CSS to design Webpages.
2. Link web pages to each other and publish the designed webpages as a website.
3. Implement XML in defining custom tags to represent structured information and define and store data in a shareable manner
4. create engaging user experiences by adding functionality like animations, real-time updates, and

event handling using JavaScript
5. Develop dynamic and interactive web pages.

Text books:

1. Learning Web Design: A beginner's Guide to HTML, CSS, JavaScript, and Web Graphics - Jennifer Robbins
2. HTML5: The Missing Manual - Matthew MacDonald
3. HTML and JavaScript – Ivan Bayross
4. Mastering HTML, CSS &JavaScript, Web Publishing

S.Y. B.Sc. (AI and ML)
Semester – IV
OE-253-AIML-T – Digital Marketing-II

No. of Credits: 2	Teaching scheme: Theory: 2 hrs/week	Evaluation Pattern: Continuous Evaluation: 15 End Sem Evaluation: 35	Total Lectures: 30

Prerequisites:

Creative & Logical thinking ability, Digital devices operational skills and Knowledge

Objectives:

1. To understand Digital Marketing as the most powerful marketing tool.
2. To Learn to create digital marketing artworks.
3. To use social media sites like Facebook, Instagram, Twitter, LinkedIn, and others to raise sales, engage customers, and establish a brand for a product.

Contents:

Unit	Contents	Number of lectures
1	**Online Consumer Behavior Analysis –** Consumer Behavior, Segmentation and Targeting online customers, Psychological Responses, Social Trends	**8**
2	**Social Media Marketing –** Social Media Sites, Influence of Social Media Marketing, Power of social media, Monetization through social media	**8**
3	**Future of Digital Marketing –** Use of Artificial Intelligence (AI) in Digital Marketing, Common use of household gadgets for online marketing, Digital Marketing strategies.	**8**
4	**Case Studies-** Case Study 1 Experiential Learning: Creating a website. Case Study 2 Online Consumer Behavior Analysis for an E-Commerce Fashion Brand or any related case	**6**

Outcomes:

On completion of the course, student will be able to–

CO1: Understand and learn marketing strategies and results effectively to stakeholders.
CO2: Assess and enhance digital marketing campaigns' return on investment.
CO3: Analyze and implement practical experience with industry-standard digital marketing tools.
CO4: Analyze and use variety of social media channels to create and interact with communities, raise awareness of a brand.

Text books:

1. Digital Marketing: Nitin Kamat, Chinmay Kamat (Himalaya Publishing House)
2. Digital Marketing for Dummies by Ryan Deiss and Russ Henneberry
3. Influence: The Psychology of Persuasion by Robert B. Cialdini
4. Social Media Marketing Workbook: How to Use Social Media for Business by Jason McDonald

S.Y. B.Sc. (AI and ML)
Semester – IV
OE-254-AIML-T – AI for Everyone - II

No. of Credits: 2	Teaching scheme: Theory: 2 hrs/week	Evaluation Pattern: Continuous Evaluation: 15 End Sem Evaluation: 35	Total Lectures: 30

Objectives:

1. Understand the basics of artificial intelligence and its subfields.
2. Explore real-world applications of AI across different industries.
3. Gain insights into the ethical, social, and economic implications of AI.
4. Develop an appreciation for the potential of AI to drive innovation and transformation.

Contents:

Unit	Contents	Number of lectures
1	**Advanced AI Fundamentals –** Deep Dive into AI, Machine Learning & Deep Learning, Types of AI Models: Generative AI, NLP, Computer Vision, Reinforcement Learning, Latest AI Trends: AGI, Large Language Models (LLMs), and multimodal AI, Exploring AI Frameworks & APIs: Open AI, Hugging Face, Google AI	**6**
2	**AI for Advanced Text & Content Creation –** AI for Long-form Writing & Reports, Automating Research & Citation Management, AI for SEO & Content Optimization, Using AI for Professional Emails & Business Writing	**8**
3	**Advanced AI for Image & Video Processing -** AI Image Generation Beyond Basics, Deepfake Technology & Ethical Concerns, AI Video Editing & Creation, AI Animation & Motion Capture	**8**
4	**AI Tools –** Chat GPT (Open AI), Google Gemini (Bard AI), Canva, Beautiful. AI, Gamma**,** Slides AI, Rytr, Grammarly	**8**

Outcomes:

On completion of the course, student will be able to–

CO1: Understand different types if AI Models
CO2: Learn and use content optimization using AI.
CO3: Compare and implement Animations and motions in AI
CO4: Understand and analyse AI tools.

Text books:

1. Artificial Intelligence: A Modern Approach – Stuart Russell & Peter Norvig.
2. Practical AI for Business Leaders – Anand S. Rao
3. AI-Powered Automation Handbook – Will Kelly
4. AI for Content Creators: How to Use AI Tools for Writing and Marketing – Rob Lennon
5. Human Compatible: Artificial Intelligence and the Problem of Control – Stuart Russell
6. https://www.grammarly.com/ai/ai-writing-tools/article-writer