



Bansilal Ramnath Agarwal Charitable Trust's

Vishwakarma Institute of Technology

(An Autonomous Institute affiliated to Savitribai Phule Pune University)

Structure & Syllabus

B.Tech. Computer Science & Engineering (Artificial Intelligence and Machine Learning)

With Effect from Academic Year 2026-27

Prepared by: - Board of Studies in CSE(AIML)

Approved by: - Academic Board, Vishwakarma Institute of Technology, Pune

Chairman – BOS



Chairman – Academic Board

INDEX

SN	Particular	Page No
1	Vision and Mission of Institute	III
2	Vision and Mission of Department	IV
3	Board of Studies	V
4	Program Educational Objectives	VI
5	List of Program Outcomes	VII
6	List of Program Specific Outcomes	IX
7	NEP Nomenclatures	X
8	Third Year Module -V Content	14
9	Third Year Module -VI Content	61

Vision of the Institution

"To be a globally acclaimed Institute in Technical Education and Research for holistic Socio-economic development".

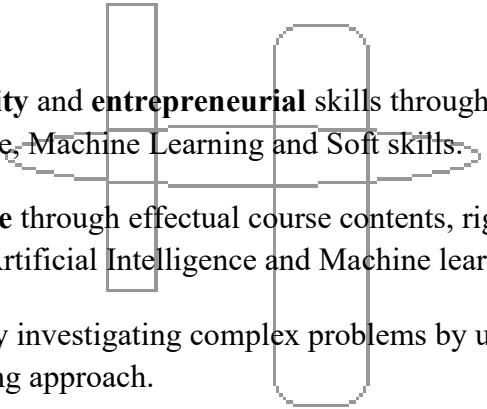
Mission of the Institution

- To ensure that 100% students are employable and employed in Industry, Higher Studies, become Entrepreneurs, Civil / Defense Services / Govt. Jobs and other areas like Sports and Theatre.
- To strengthen Academic Practices in terms of Curriculum, Pedagogy, Assessment and Faculty Competence.
- Promote Research Culture among Students and Faculty through Projects and Consultancy.
- To make students Socially Responsible Citizens.

Vision of the Department

“To be an academic excellence center for developing globally competent engineers and researchers for holistic Socio-economic development”.

Mission of the Department

- 
- To ensure students' **employability** and **entrepreneurial** skills through knowledge of principles of computing, Artificial Intelligence, Machine Learning and Soft skills.
 - To enhance **academic excellence** through effectual course contents, rigorous hands-on and active participation of industry in the Artificial Intelligence and Machine learning domain.
 - To cultivate **research culture** by investigating complex problems by using computing, Artificial Intelligence and Machine learning approach.
 - To develop a sense of responsibility and ethics among students to make them **responsible citizens**.

Board of Studies Computer Department AY 2026-27

Sr. No	Name	Affiliation in Committee	Designation
1	Prof. (Dr.) Premanand Ghadekar	Head of Department (Chairman)	Head & Professor, CSE(AI&ML)
2	Prof. (Dr.) Gitanjali Shinde	Entire Faculty of Each Specialization Representative	Associate Professor
3	Dr. Sanjeevkumar Angadi	Entire Faculty of Each Specialization Representative	Associate Professor
4	Dr. Jyoti Kanjalkar	Entire Faculty of Each Specialization Representative	Assistant Professor
5	Dr. Varsha Dange	Entire Faculty of Each Specialization Representative	Assistant Professor
6	Ms. Kalyani Ghuge	Entire Faculty of Each Specialization Representative	Assistant Professor
7	Prof. (Dr.) Ramakrishna Upadrasta	Subject Expert from Outside University nominated by Academic Council	Associate Professor, Department of CSE,IIT Hyderabad
8	Prof. (Dr.) Y. V. Haribhakta	Subject Expert from Outside University nominated by Academic Council	Associate Professor, Department of CSE,COEP Technological University
9	Prof.(Dr.) Nilesh Uke	VC Nominee	Professor, Principal, Indira College of Engineering and Management, Pune.
10	Mr.Prashant Kamble	Industry Representative	Executive Director - AI and Data Team MSCI Inc. ,Ex-Microsoft MBA, AI/ML/GenAI Mentor
11	Mr.Girish Bharambe	Meritorious Alumnus Nominated by Director	NvidiaSenior Manager GPU Compilers at NVIDIA

Program Educational Objectives (PEOs)

PEO	PEO Focus	PEO Statements
PEO1	Core competence	Demonstrate core competence in principles of computing and AIML based Technologies.
PEO2	Breadth	Apply engineering knowledge to solve industry problems with creative and innovative design , development tools and AIML techniques.
PEO3	Professionalism	Develop ethical and professional practices effectively to gain desired soft skills in social and global context.
PEO4	Learning Environment	Aim for continuing education and entrepreneurship in emerging areas of computing and AIML.

List of Programme Outcomes [PO]

PO	PO Statements
PO1	Engineering Knowledge: Apply the knowledge of mathematics, science, engineering fundamentals and an engineering specialization to the solution of complex engineering problems..
PO2	Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences
PO3	Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations
PO4	Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
PO5	Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
PO6	The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice

PO7	Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
PO8	Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
PO9	Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
PO10	Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
PO11	Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
PO12	Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Programme Specific Outcomes [PSO]

Programme Specific Outcome Statements

- PSO1:** Demonstrate competence in programming with sound knowledge of principles of computing.
- PSO2:** Formulate coherent design, execute proficient implementation, and conduct testing to solve real world problems using software paradigms and AIML methodologies.
- PSO3:** Adapt and exhibit skills in emerging domains of computer science, engineering and technology.

Nomenclature for Teaching and Examination Assessment Scheme AY 2026-27

Sr No.	Category	Head of Teaching/ Assessment	Abbreviation used
1	Teaching	Theory	Th
2	Teaching	Laboratory	Lab
3	Teaching	Tutorial	Tut
4	Teaching	Open Elective	OE
5	Teaching	Multi-Disciplinary	MD
6	Teaching	Computer Science	CS
7	Teaching	Machine Learning	ML
8	Assessment	Laboratory Continuous Assessment	CA
9	Assessment	Mid Semester Assessment	MSA
10	Assessment	End Semester Assessment	ESE
11	Assessment	Home Assignment	HA
12	Assessment	Course Project	CP
13	Assessment	Group Discussion	GD
14	Assessment	PowerPoint Presentation	PPT
15	Assessment	Class Test –1	CT1
16	Assessment	Class Test –2	CT2
17	Assessment	Mid Semester Examination	MSE
18	Assessment	End Semester Examination	ESE
19	Assessment	Written Examination	WRT
20	Assessment	Multiple Choice Questions	MCQ
21	Assessment	Laboratory	LAB

Vishwakarma Institute of Technology Issue 1 : Rev No. 01 : Dt. 05/02/26

Title: Course Structure

FF No. 653

Branch: Computer Science & Engineering (AIML)

Year: T.Y.

Academic Year.: 2026-27

Semester Module: V Pattern:26-27

Sub No.	Subject Code	Subject Name	Teaching Scheme (Hrs/Week)			Examination Scheme										Total	Credits
			Th	Lab	Tut	CVV	CP	LAB CA	GD/ PPT/ HA	MSE (W/O)	T1(W/O)	T2 (W/O)	ESE(R)	ESE (W/O)	PRACT /CVV		
S1	ML3201	PCC: Design and Analysis of Algorithms	3	2	-	-	-	-	-	-	-	-	-	40(W)	60	100	4
S2	ML3202	PCC: Machine Learning	2	2	-	20	30	10	-	-	-	-	-	40(W)	-	100	3
S3	ML3203A	PEC1: Computer Network Technology	3	2	-	20	40	-	20(PPT) 20(HA)	-	-	-	-	-	-	100	4
	ML3203B	PEC1: Machine Learning And Operations			-			-		-	-	-	-	-	-		
	ML3203C	PEC1: Computer Network Security			-			-		-	-	-	-	-	-		
	ID3001	PEC1: Basics Of Game Development			-			-		-	-	-	-	-	-		
S4	MD3101	PEC2: Coursera	-	-	-	-	-	-	20	-	-	-	-	80	-	100	3
S5	MD3201	OE1: Coursera	-	-	-	-	-	-	20	-	-	-	-	80	-	100	4
S6	MM0301A	MDM : Cloud Computing	2	-	1	-	30	-	20(PPT) 10(HA)	-	-	-	-	40(W)	-	100	3
	MM1501D	MDM: Product Lifecycle Management (PLM)							-	-	-	-	-	-	-		
S7	ML3207	VSEC: Design Thinking-3	-	-	1	-	-	-	-	-	-	-	-	-	-	100	1
S8	ML3204	ERI - Engineering research and innovation-1	-	-	-	-	-	-	-	30	-	-	70	-	-	100	2
		Total	10	6	2	40	100	10	110	30	0	0	70	280	60	800	24

PEC2: Coursera Courses*

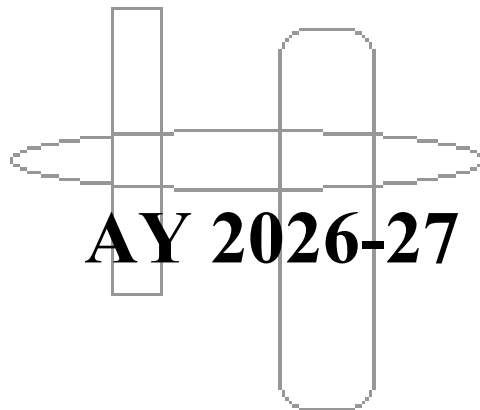
Subject Code	Subject Name
MD3101	IBM Full Stack Software Developer
MD3102	Meta Back-End Developer
MD3106	Microsoft Power BI Data Analyst
MD3112	Google Cybersecurity
MD3113	Google Data Analytics
MD3118	IBM Data Science
MD3119	Fractal Data Science
MD3121	IBM DevOps and Software Engineering
MD3141	AWS Cloud Technology Consultant
MD3147	Deep Learning.AI Deep Learning

OE1: Coursera Courses*

Subject Code	Subject Name
MD3201	Fintech
MD3202	Digital Marketing
MD3203	Entrepreneurship
MD3204	Management

Syllabus Of
Computer Science and Engineering (Artificial Intelligence
& Machine Learning)
(AY 2026-27)

T.Y. Computer Science and Engineering (Artificial Intelligence & Machine Learning)



Module V Course Content

Syllabus Template**FF No.: 654****ML3201: Design and Analysis of Algorithms**

Teaching Scheme	Examination Scheme
Credits : 4	PR/ CVV: 60 Marks
Lectures : 3 Hrs/week	ESE (W/O): 40 Marks
Practical : 2 Hrs/week	
Tutorial : -	

Prerequisites:	
●	Knowledge of Programming Fundamentals, Data Structures, Discrete Structures (Discrete Mathematics), and Theory of Computing.
Course Objectives:	
●	Students will gain understanding of asymptotic notations, recurrence relations, and mathematical techniques used for analyzing the time and space complexities of algorithms.
●	Students will develop the ability to formulate computational problems in an abstract and mathematically precise manner and establish the correctness of algorithms
●	Student will gain understanding of algorithm design paradigms such as Divide and Conquer, Dynamic Programming, Greedy, and Backtracking, and apply them to solve computational problems efficiently.
●	Students will develop the ability to analyze and compare alternative algorithmic solutions based on computational complexity and resource requirements
●	Student will gain understanding of complexity classes such as P, NP, NP-Complete, and NP-Hard, and their significance in computational problem solving
●	Students will develop the ability to design efficient solutions for complex computational problems using appropriate data structures, randomized algorithms, and algorithmic paradigms
Course Outcomes:	
	After completion of the course, student will be able to
1.	Formulate computational problems in an abstract and mathematically precise manner, and Analyze their time and space requirements using asymptotic notations and recurrence relations.
2.	Design and analyze efficient algorithms using the Divide and Conquer paradigm for solving computational problems.
3.	Apply Dynamic Programming techniques to develop optimal solutions for complex optimization problems
4.	Apply Greedy strategies to solve optimization problems and analyze their efficiency.

5.	Design solutions for combinatorial and constraint satisfaction problems using Backtracking techniques.
6.	Differentiate complexity classes (P, NP, NP-Complete, NP-Hard) and analyse the significance of NP-completeness and randomized algorithms.

Section1:	Topics/Contents
UNIT 1- Introduction to time and space complexity [07 Hours] Algorithm definition, insertion sort running time calculation, Notations Theta, big O, big Omega, little o notations with examples. Space complexity definition and examples like matrix multiplication, bubble sort, factorial computation recursive and iterative version. Recurrence relations and finding time complexity using substitution method, recursion tree and master theorem. Proof of correctness of algorithms. UNIT 2- Divide and conquer [07 Hours] General strategy of divide and conquer examples binary search, quick sort-best case, worst case and average case time complexities, merge sort and analysis, finding majority element and analysis, Order statistics- finding simultaneous maximum and minimum, selection problem. UNIT 3:- Dynamic programming [08 Hours] General strategy, Fibonacci sequence example dynamic and recursive with comparison. Travelling salesman problem by dynamic programming. And 0-1 knapsack problem using dynamic programming	
Section2:	Topics/Contents
UNIT 4:- Greedy strategy [08 Hours] General approach, fractional knapsack problem using greedy method, Job scheduling/sequencing using greedy approach examples, Huffman coding, minimum spanning tree-Kruskal's and Prim's algorithm UNIT 5:- Backtracking strategy [06 Hours] General approach, N-queens problem using backtracking, graph colouring problem with examples UNIT 6:- Complexity classes and Randomized algorithms [06 Hours] Introduction to P, NP, NPC and NP-Hard problems and their interrelations, Randomized algorithms, Las Vegas and Monte Carlo simple examples	
Text Books: (As per IEEE format)	
1	Cormen, Leiserson, Rivest and Stein "Introduction to Algorithms", 3rd edition, 2009. ISBN 81-203-2141-3, PHI

2	Horowitz and Sahani, Fundamentals of computer Algorithms, Galgotia, ISBN 81-7371-612-9
3	Jon Kleinberg, Eva Tardos “Algorithm Design”, 1st edition, 2005. ISBN 978-81-317-0310-6, Pearson
4	Dasgupta, Papadimitriou, Vazirani “Algorithms”, 1st edition (September 13, 2006), ISBN-10:9780073523408, ISBN-13: 978-0073523408, McGraw-Hill Education
Reference Books: (As per IEEE format)	
1	Anany Levitin, “Introduction to the Design & Analysis of Algorithm”, Pearson, ISBN 81-7758- 835-4
2	Gilles Brassard, Paul Bratle, Fundamentals of Algorithms, Pearson, ISBN 978-81-317-1244-3.
3	Motwani, Raghavan “Randomized Algorithms”, 1st edition (August 25, 1995), ISBN- 10:0521474655, ISBN-13: 978-0521474658, Cambridge University Press
4	Vazirani, “Approximation Algorithms”, ISBN-10: 3642084699, ISBN-13: 978-3642084690, Springer (December 8, 2010)

List of Practical's:

1. Implementation and timing analysis of matrix multiplication for square matrices
2. Implementation and analysis of quick sort
3. Finding out majority element from an array
4. Compare dynamic programming and divide and conquer using Fibonacci sequence
5. Huffman coding using Greedy strategy
6. Knapsack using dynamic programming

List of Project areas:

1. **Comparative Analysis of Sorting Algorithms**
 - Bubble Sort, Insertion Sort, Merge Sort, Quick Sort
 - Compare time and space complexities.
2. **Performance Evaluation of Divide and Conquer Algorithms**
 - Binary Search, Merge Sort, Quick Sort
 - Analyze best, worst, and average cases.
3. **Majority Element Finding Using Divide and Conquer**
 - Implement and compare with brute-force approach.
4. **Order Statistics Problem**
 - Finding Maximum, Minimum, and Kth Smallest Element efficiently.
5. **Analysis of Recursive and Iterative Algorithms**
 - Factorial, Fibonacci, Tower of Hanoi
 - Compare time and space complexities.
6. **Fibonacci Number Generation: Recursive vs Dynamic Programming**
 - Performance comparison using execution time graphs.
7. **Travelling Salesman Problem (TSP) Using Dynamic Programming**
 - Implement and analyze optimal route generation.
8. **0/1 Knapsack Problem Solver**
 - Dynamic Programming approach with performance analysis.

9. **Fractional Knapsack Problem Using Greedy Strategy**
 - Compare with 0/1 Knapsack.
10. **Job Scheduling for Maximum Profit**
 - Greedy algorithm implementation and analysis.
11. **Huffman Coding for Data Compression**
 - Encode and decode text files.
 - Measure compression efficiency.
12. **Minimum Spanning Tree Construction**
 - Implement Kruskal's and Prim's algorithms.
 - Compare execution times on different graph sizes.
13. **N-Queens Problem Solver**
 - Backtracking approach with visualization.
14. **Graph Coloring Using Backtracking**
 - Determine minimum colors required for different graphs.
15. **Randomized Algorithms: Monte Carlo vs Las Vegas**
 - Implement simple randomized search/sorting techniques.
 - Compare accuracy and running time.

Course Project Titles (Industry-Oriented)

1. Route Optimization System using TSP
2. Data Compression Tool using Huffman Coding
3. Task Scheduler for Cloud Computing
4. Network Design using Minimum Spanning Tree
5. Exam Timetable Generator using Graph Coloring
6. Chessboard N-Queens Visualizer
7. Stock Analysis using Divide and Conquer
8. Fast Search Engine using Binary Search Techniques
9. Algorithm Complexity Analyzer Tool

MOOCs Links and additional reading material:

<https://nptel.ac.in/courses/106106131>

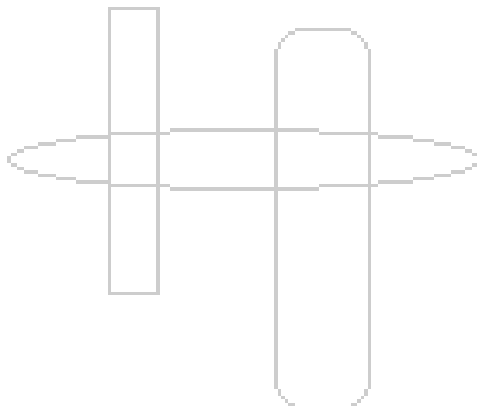
<https://nptel.ac.in/courses/106105157>

https://www.coursera.org/specializations/algorithms?utm_source=chatgpt.com

https://www.coursera.org/specializations/data-structures-algorithms?utm_source=chatgpt.com

CO-PO Mapping

CO	Program Outcomes (PO)												PSO		
CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO 11	PO12	PSO1	PSO2	PSO3
CO1	3	3	-	2	1	-	-	-	-	-	-	1	3	2	1
CO2	3	3	3	2	2	-	-	-	-	-	-	1	3	3	1
CO3	3	3	3	2	2	-	-	-	-	-	-	1	3	3	1
CO4	3	3	3	2	2	-	-	-	-	-	-	1	3	3	1
CO5	3	3	3	2	3	-	-	-	2	3	-	2	3	3	2
CO6	3	3	-	2	1	-	-	-	-	-	-	2	2	1	2
Average	3.00	3.00	2.00	2.00	1.83				0.33	0.50		1.33	2.83	2.50	1.33



Syllabus Template**FF No.: 654****ML3202: Machine Learning**

Teaching Scheme	Examination Scheme
Credits : 3	CP :30 Marks
Lectures : 2 Hrs./week	PPT: 20 Marks
Practical : 2 Hrs./week	LAB: 10 Marks
Tutorial : Hrs./week	ESE: 40 Marks

Prerequisites:	
•	Linear Algebra, Statistics, Calculus and programming language.
Course Objectives:	
•	To learn the concepts, techniques and building blocks of machine learning.
•	To learn mathematics for implementing machine learning algorithms.
•	To learn the supervised, unsupervised and reinforcement learning techniques.
•	To learn use of computational learning theory
•	To learn feature reduction on real life problems.
•	To learn Machine Learning models and implement them in real life scenarios for different applications.
Course Outcomes:	
	After completion of the course, student will be able to
1.	Understand the features of machine learning to apply on real world problems
2.	To develop various supervised machine learning algorithms in a wide range of real-world applications.
3.	Understand a wide variety of Statistical learning algorithms and apply on different applications
4.	Analyze the concept of Support Vector Machines, Naive Bayes Classifiers for learning linear and non-linear Classifiers
5.	To analyze Decision Tree algorithms to solve problems of real world.
6.	Demonstrate understanding of unsupervised learning, Reinforcement learning and their applications

Section1:	Topics/Contents
Unit-I: Introduction of Machine Learning [4 Hours] What is Machine Learning, Types of Learning: Supervised, Unsupervised, Reinforcement. Learning System, Well posed learning problem, Issues in machine learning. Concept Learning: Concept Learning, General-to-Specific Ordering: Task, search, Find S algorithm, Version space and the candidate elimination algorithm, List-then-eliminate algorithm, Bias, Variance, Underfitting, Overfitting, Inductive bias, Evaluation, Training, Testing, Cross-validation: Error Analysis, Error Metrics, Precision and recall.	

Unit-II : Supervised Learning: [4 Hours] Feature Engineering: Preprocessing of data: Normalization and Scaling, Standardization, Rationale and Basics: Learning from Observations, Bias. Metrics for assessing regression, Metrics for assessing classification, classification vs regression task. Unit-III : Statistical Learning [4 Hours] Machine Learning and Inferential Statistical Analysis, Descriptive Statistics in learning techniques, K-Nearest Neighbor Classifier. Linear Regression with Least Square Error Criterion, Logistic Regression for Classification Tasks, Neural network for supervised and unsupervised learning, Case study on KNN and logistic regression.	
Section2:	Topics/Contents
Unit-IV : Naive Bayes and Support Vector Machine [4 Hours] Bayes Theorem: Naive Bayes' Classifiers, Multinomial Naïve Bayes. Support Vector Machine (SVM)- Linear Support Vector Machines, Linear Classification, Kernel based classification, Non- linear Examples. Support Vector Regression. Case study on Naive Bayes and Support Vector Machine. Unit-V: Decision Trees and Ensemble Learning [6 Hours] Decision Tree Learning: Representation, Basic decision tree learning algorithm, Issues in decision tree learning. Introduction to Meta Classifier: Concepts of Weak and eager learner, Ensemble methods, Bagging, Boosting, Random Forests. Case study on decision tree. Dimensionality Reduction Techniques-Various Feature Selection Techniques (Wrapper, Filter and Embedded method) Sequential Forward Selection, Sequential Backward Selection. Introduction to Dimensionality Reduction, Principal Component Analysis (PCA). Unit-VI: Unsupervised and Reinforcement learning [6 Hours] Segmentation, Clustering Fundamentals- Basics, K-means: Finding optimal number of clusters, DBSCAN, Spectral Clustering. Evaluation methods based on Ground Truth-Homogeneity, Completeness. Hierarchical Clustering, Agglomerative Clustering, Case study on clustering. Rule Based Models: Rule learning for subgroup discovery, Association rules mining – Apriori Algorithm, Confidence and Support parameters. Reinforcement Learning: Exploration, Exploitation, Rewards, Penalties, Markov Decision Process, Q-Learning and Bellman Equation. Transfer Learning, Tiny ML, Oneshot, Few Shot Learning, Multimodal Dataset, Hyper parameter tuning etc.	
Text Books: <i>(As per IEEE format)</i>	
1	T. Mitchell, “Machine Learning”, McGraw-Hill, 1997
2	Anup Kumar Srivastava, Soft Computing, Alpha Science International limited. 2009.
Reference Books: <i>(As per IEEE format)</i>	
1	Ethem Alpaydin, "Introduction to Machine Learning", MIT press, 2004.
2	2. Jacek M. Zurada, “Introduction to Artificial neural System”, JAICO publishing house,2002

MOOCs Links and additional reading material:1 www.nptelvideos.in2 www.coursera.com**Course Outcomes:**

On the completion of course, student will able to

1. Demonstrate knowledge learning algorithms and concept learning
2. Develop various supervised machine learning algorithms in a wide range of real-world applications.
3. Understand a wide variety of Statistical learning algorithms and apply on different applications
4. Analyze the concept of Support Vector Machines, Naive Bayes Classifiers for learning linear and non-linear Classifiers
5. Aanalyze Decision Tree algorithms to solve problems of real world.
6. Demonstrate understanding of unsupervised learning, Reinforcement learning and their applications.

List of Practical's (Minimum Six):**Write a Python code to implement following algorithm using standard dataset..**

1. Linear Regression
2. Logistic Regression
3. KNN
4. Naive Bayes
5. Random forest Algorithm
6. Support vector machine
7. Decision Tree
8. Random Forest
9. PCA - Principal Component Analysis-Finding Principal Components, Variance and Standard Deviation calculations of principal components
10. K-means clustering
11. Hierarchical Clustering
12. MBA Rule based learning
13. Hidden markov model

List of Course Projects: USE the UCI ML or KDD Repository or any other dataset

- 1 Stock Market Price Prediction
- 2 Housing Prices Prediction
- 3 Sign Text Identification
- 4 Iris Flowers Classification
- 5 Fake News Detection
- 6 Product Delivery Drones
- 7 Smart City Water / Light Management System
- 8 Human Tracking System

- 9 Automatic Interview Conduction System
- 10 Student Information Chatbot Project.
- 11 Product Review Analysis for Genuine Rating.
- 12 Customer Targeted E-Commerce
- 13 MNIST Digit Classification
- 14 Bit coin Price Predictor
- 15 Credit Card Fraud Detection
- 16 Customer Segmentation
- 17 Design and apply ML techniques to solve real life problems

List of Course Seminar Topics:

1. Streaming Machine Learning Techniques
2. Error Analysis In Machine Learning
3. Parallel Infrastructures Such As Map-Reduce For ML
4. Ensemble Learning
5. Backpropagation Algorithm.
6. Practical Techniques For Reducing The Memory Requirements: Feature Hashing
7. Support Vector Machine
8. Practical Techniques For Reducing The Memory Requirements: Bloom Filters
9. Genetic Algorithm
10. Regression Analysis
11. K Means Algorithm
12. Decision Tree Learning Algorithm
13. Bayesian Learning, Bayes Theorem and Naïve Bayes Theorem.
14. Hidden Markov Model,
15. Principal Component Analysis (PCA)
16. Recommender Systems
17. Occam's Razor Principle and Overfitting Avoidance Heuristic Search in inductive Learning

List of Course Group Discussion Topics:

- 1 Machine Learning And Artificial Intelligence
- 2 Machine Learning And Data Science
- 3 Machine Learning Applications
- 4 Machine Learning Future
- 5 Machine Learning After 10 Years / 2030
- 6 Supervised Learning Techniques And Unsupervised Learning Techniques
- 7 Reinforcement Learning
- 8 Recommender Systems
- 9 Will Automation and ML Reduce Or Increase Jobs.
- 10 Cashless Economy Using ML
- 11 ML In Covid-19 Situations

List of Design based Home Assignments:

Design:

1. Heart Disease Prediction Using Machine Learning Algorithms
2. Detection Based Project For Social Cause
3. Classification Based Project For Social Cause
4. Clustering Based Project For Social Cause

5. Optimization Based Project For Social Cause
6. Recommender Systems Based Project For Social Cause
7. Identification Based Project For Social Cause
8. Machine Learning-Based Student's Native Place Identification For Real-Time

Case Study:

1. How Auto industry is preparing For The 4th Industrial Revolution using ML
2. How Indian Retail Giant Is Using ML to Prepare for the 4th Industrial Revolution
3. Rolls-Royce and Google Partner to Create Smarter, Autonomous Ships Based On ML
4. The Amazing Ways Tesla Is Using Machine Learning and Big Data
5. The Incredible Ways John Deere Is Using Machine Learning To Transform Farming

Blog

1. Machine Learning for Sentiment Analysis
2. Machine Learning for Character Recognition
3. Machine Learning for Heart Disease Detection
4. Machine Learning for Chatbot Development
5. Machine Learning for Agriculture
6. Machine Learning for Medical Field

Surveys

1. Adaption of Machine Learning ML AI in 2020
2. Machine Learning in Industry
3. Machine Learning in Digital Marketing
4. Machine Learning in Military
5. Machine Learning after Covid-19

Assessment Scheme: Ensures 360 degree assessment and covers all aspects of Bloom's Taxonomy.

Laboratory Continuous Assessment: 100 Marks converted to 10 Marks

Course Project: End Semester Examination: 100 Marks converted to 20 Marks

Presentation: End Semester Examination: 100 Marks converted to 20 Marks

Theory: End Semester Examination (Written): 60 Marks converted to 30 Marks

Comprehensive Viva Voce: End Semester Examination: 100 Marks converted to 20 Marks

CO-PO Mapping

	Program Outcomes (PO)												PSO			
CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3	
CO1	3	3	3	2	3	1		1	1	1		2	2	2	3	
CO2	3	3	3	2	3	2		1	1	1		2	2	3	3	
CO3	2	3	3	3	3	2		1	1	1		2	2	3	3	
CO4	3	3	3	3	3	2		1	1	1		2	2	3	3	
CO5	3	3	3	3	3	2		1	1	1		2	2	3	3	
CO6	3	3	3	3	3	2		1	1	1		2	2	3	3	
Average	3	3	3	2.66	3	1.83		1	1	1		2	3.0	2.83	3	

1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High)

CO Attainment levels:

- Co1 - Level 1
- Co2 - Level 2
- Co3 - Level 3
- Co4 - Level 5

Co5 - Level 4

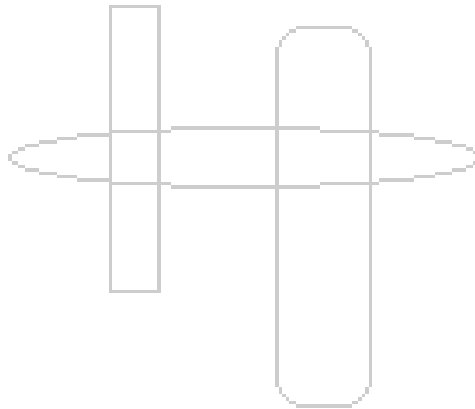
Co6 - Level 3

Future Course Mapping:

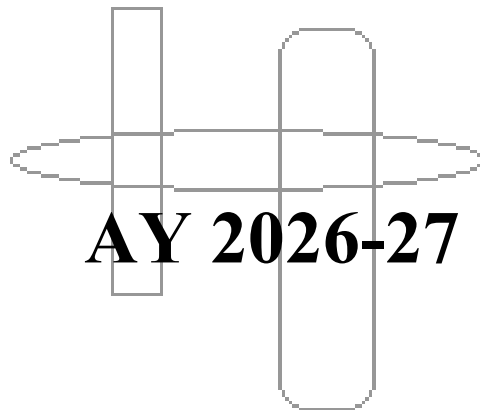
Soft Computing, Deep Learning

Job Mapping:

ML Scientist, ML Designer, ML Architectural Design, ML Developer, ML Data Analyst



**T.Y. Computer Science and Engineering
(Artificial Intelligence & Machine Learning)**



PEC1

Syllabus Template**FF No.: 654****ML3001: Computer Network Technology**

Teaching Scheme	Examination Scheme
Credits: 4	CP :40 Marks
Lectures : 3 Hrs./week	CVV: 20 Marks
Practical: 2 Hrs./week	PPT: 20 Marks
Tutorial: -	HA: 20 Marks

Prerequisites:	
•	Operating System, Database Management Systems
Course Objectives:	
•	To learn transmission mediums, networking devices, and topologies used in the Internet
•	To learn networking standards, IP packet switching, and routing used in the Internet
•	To learn transport layer and application layer protocols used on the Internet
•	To learn front-end technologies for website development
•	To learn single-page application development using REACT
•	To learn REST API based enterprise website development using REACT, Node JS, Spring Boot with different database technologies
Course Outcomes:	
	After completion of the course, the student will be able to
1.	Select topology, essential components of the physical layer, and networking devices to design computer networks.
2.	Build a wired and wireless intranet with correct communication and service frameworks.
3.	Develop Client-Servers by means of correct standards, protocols, and technologies
4.	Build single-page applications using REACT as a reusable UI component technology
5.	Write a Web API/RESTful API application programming interface to communicate with Spring Boot as the server-side technology.
6.	[Group Assignment] Design and develop a three-tier enterprise application using client-side, server-side, and back-end technologies

Section1:	Topics/Contents
Unit-I :Networking Fundamentals and Physical Layer [07 Hours] Network Organizations and Architectures: What are computer Networks, Network Topologies: Mesh, Star, and Hierarchical, Types of Computer Networks: LAN, MAN,	

WAN, PAN, Internet, and Intranet. Client-Server; Peer -to- Peer. Network Architecture Modes: Infrastructure and Ad-hoc mode.

Reference Models: OSI and TCP/IP. Design Issues for Layers.

Physical Layer: Transmission Media: Air, Vacuum, Cat5, Cat5e, Cat6, Cat6a, Cat7, Cat8, OFC - Single and Multicore.

Networking Devices Wired and Wireless: NIC, Repeater, Bridge, Switch, Modem, Router, Gateways, and Access Point.

Unit-II: Medium Access Control and Network Layer [07 Hours]

Medium Access Control: Legacy Standard: 10 Mbps IEEE 802.3 Standard(Ethernet), High-Speed Ethernet Standards: Fast, Gigabit, and 10Gigabit.

Wireless Standards: IEEE 802.11a/b/g/n/ac, IEEE 802.15, IEEE 802.15.4 and IEEE 802.16 Standards, CSMA/CA

Switching Techniques and IP Addressing: Circuit, Message, and Packet Switching. Logical Addressing: IPv4 and IPv6

Network Layer Protocols: Internet Protocol (IP), Internet Control Message Protocol(ICMP)

Unit-III : Transport Layer and Application Layer [07 Hours]

Transport Layer Protocols: Transmission Control Protocol (TCP), User Datagram Protocol (UDP). **Services:** Berkeley Sockets, Connection Establishment, Connection Release. **Application Layer:** Domain Name System (DNS) and File Transfer Protocol (FTP). **WWW:** Hypertext Transfer Protocol (HTTP1.1/1.2/2.0) and HTTPS with SSL. **Email:**SMTP, MIME, POP3, and Webmail.

Section2:	Topics/Contents
------------------	------------------------

Unit-IV: Client-Side Technologies [09 Hours]

HTML5: structure of HTML document, HTML elements: headings, paragraphs, line break, links, frames, lists, tables, images, and forms CSS3.0: Styles, colors, fonts, and Text Alignments

JavaScript: Basics of Document Object Model (DOM), Variable Declarations: Using var,let, and const, Reserved Keywords, Objects and Classes, Understanding Functions: Declarations, Expressions, Arrow Functions, Event Handling- Browser Events and Event Listeners, Form Validation, AJAX.

React

Introduction to React, React component, JSX, Render function, Component API, Component lifecycle, State, Props, Mixins, Component composition, Pass data from parent to child, Pass data from child to parent, Component styling, Forms, Events,

Refs, Keys, Router, Flux, Redux

Unit-V: Spring Boot**[05 Hours]**

Spring Framework, Spring Boot Framework, Installing Spring Boot, Build Tool Maven/Gradle/Ant, Core Features, Spring Security, Web Applications, JPA for database connectivity, working with SQL and NoSQL, Messaging, Testing, Deploying Spring Boot Applications, Monitoring and Testing. POSTMAN is a tool for API testing.

Unit-VI: NodeJS**[07 Hours]**

Introduction to Node JS, Installation of Node JS, Node JS Modules, Node Package Manager (NPM), Creating a Web server, File System, Express JS, Serving Static Resources, Database connectivity.

Text Books: *(As per IEEE format)*

1	Andrew S. Tanenbaum, "Computer Networks", 5th Edition, PHI, ISBN 81-203-2175-8.
2	Kurose, Ross "Computer Networking: A Top-Down Approach Featuring the Internet", Pearson; 6th edition (March 5, 2012), ISBN-10: 0132856204
3	Frouzan B., "Data Communications and Networking", 5th edition, Tata McGraw-Hill, Publications, 2006
4	Kumar, A., Web technologies, CRC Press, 2019 Gupta, R., Internet & Web Technologies, Engineering Handbook, 2019
5	Jeffrey C. Jackson, "Web Technologies: A Computer Science Perspective", Second Edition, Pearson Education, 2007, ISBN 978-0131856035.

Reference Books: *(As per IEEE format)*

1	Martin, M.G., Programming for Beginners: 6 Books in 1 – Swift+PHP+Java+Javascript+Html+CSS: Basic Fundamental Guide for Beginners, independently published, 2018
2	Kumar, A., Web technologies, CRC press, 2019 Gupta, R., Internet & Web Technologies, Engineering Handbook, 2019
3	Adam Bretz & Colin J Ihrig, "Full Stack Javascript Development with MEAN", SPD, First Edition 2015, Indian Reprint September 2015
4	Giulio Zamboni, "Beginning JSP, JSF and Tomcat", Apress Publication, Second Edition, 2013
5	Jeremy McPeak & Paul Wilton, "Beginning JavaScript", Wrox Publication, Fifth Edition, 2015
6	Robert W. Sebesta: Programming the World Wide Web, 4th Edition, Pearson education, 2008

List of Practical's (Minimum Six):**Unit-I and Unit II: Use two turn of lab**

1a) Setting up small wired computer network:

Set up a small wired network of 2 to 4 computers using Hub/Switch/. It includes Preparation of Cables and setting up wired network.

1b) Setting up small wireless computer network and hands-on networking command:

Set up a small wired network of 2 to 4 computers using access point and ask students to access it on their wireless gadgets.

Hands on for network commands - ping, pathping, ipconfig/ifconfig, arp, netstat, nbtstat, nslookup, route, traceroute/tracert, nmap.

Unit-II MAC and Network Layer

2) Write a program to find the shortest path using Dijkstra Equation for Link State Routing Protocol which is used by Open Shortest Path First Protocol (OSPF) in the Internet for the network flow provided by instructor.

Unit-III Transport Layer and Application Layers

3a) Write the client server programs using TCP Berkeley socket primitives for wired /wireless network for following

- a. to say Hello to Each other
- b. File transfer

3b) Write the client server programs using UDP Berkeley socket primitives for wired /wireless network for following

- a. to say Hello to Each other
- b. Calculator (Trigonometry)

3c) Understanding protocol stack of Intranet

Analyze packet formats of Ethernet, IP, TCP and UDP captured through Wireshark for wired networks.

Unit-IV Client-Side Technologies

4) Design and develop a website using toggleable or dynamic tabs or pills with bootstrap and JQuery to show the relevance of SDP, EDI, DT and Course projects in VIT.

Unit-V Springboot

5) Design and develop a responsive website to prepare one semester result of VIT

students using REACT, Springboot and MySQL/ MongoDB/Oracle. Take any four subjects with MSE Marks (30%) ESE Marks (70%).

Unit-VI NODE JS

6) Design and develop a responsive website for an online book store using REACT, Node JS/ PHP and MySQL/ MongoDB/Oracle having 1) Home Page 2) Login Page 3) Catalogue Page: 4) Registration Page: (database)

List of Project areas: Networking

1. Design and deploy website for TCP based Multithreaded HTTP client server for accessing student activity data in the institute.
2. Design and deploy website for TCP based Multithreaded FTP client server to share institute level notices.
3. Design and deploy website for UDP based Multithreaded TFTP client server for your class
4. Design and deploy website for TCP based Multithreaded SMTP and POP3 mail client server for your campus.
5. Design and deploy website for TCP based Multithreaded Chat client server for your class.
6. Design and deploy website for UDP based Multithreaded Chat client server for your class.
7. Design and deploy website for UDP based Multithreaded Audio-Conferencing client server for computer engineering department.
8. Design and deploy website for UDP based Multithreaded Video Conferencing client server for computer department
9. Design and deploy website to demonstrate implementation of RIP/OSPF/BGP using Packet Tracer
10. Design and deploy website to simulation of AODV routing protocol using Packet Tracer/ NS3/OMNet

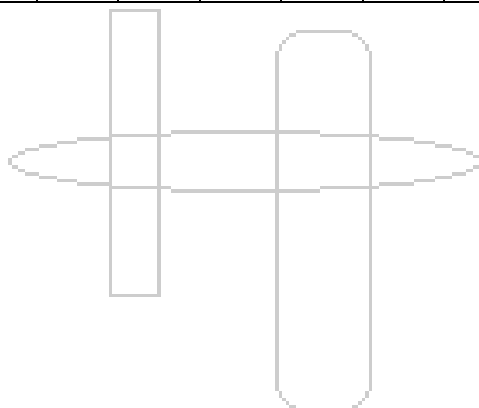
List of Course Project areas: Web Technology

1. Develop a responsive web application for Student Grievance System
2. Develop a responsive web application for Workflow Management System for MNC
3. Develop a responsive Gaming Website
4. Develop a responsive web application to help farmers to solve their farming problems
5. Develop a responsive web application for GST Billing Software for Small Business
6. Develop a responsive web application for online Crime Reporting System using PHP
7. Develop a responsive web application for online College Voting System
8. Develop a responsive web application for online Loan Processing System for Farmers.

9. Develop a responsive web application for restaurant food order management
10. Develop a responsive web application for e-book shop
11. Develop a responsive web application for on-line music store
12. Develop a responsive web application for guest visiting management to your society
13. Develop a responsive web application for web search engine

CO-PO Mapping

CO	Program Outcomes (PO)												PSO		
CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	1	1	0	2	0	0	1	0	0	0	0	0	0
CO2	3	2	3	1	3	2	2	2	1	0	0	0	3	3	0
CO3	3	3	3	1	3	2	0	2	1	0	0	0	3	3	0
CO4	3	3	3	1	3	2	0	2	1	0	0	0	3	3	0
CO5	3	2	3	1	3	2	0	2	1	0	0	0	3	3	0
CO6	3	3	3	2	3	3	0	3	3	2	2	0	3	3	2
Average	3	2.6	2.6	1.1	2.5	2.1	0.3	1.8	1.3	0.3	0.3	0	2.5	2.5	0.3



Syllabus Template

FF No.: 654

ML3203B: Machine Learning and Operations

Teaching Scheme	Examination Scheme
Credits : 04	CP: 40 Marks
Lectures : 3 Hrs/week	PPT: 20 Marks
Practical : 2 Hrs/week	CVV: 20 Marks
Tutorial : Hrs/week	HA : 20 Marks

Prerequisites:	
•	Programming Fundamentals and Python
•	Basics of Machine Learning
•	Introduction to Cloud and DevOps Concepts
Course Objectives:	
•	Understand the basics of MLOps and ML lifecycle.
•	Learn data versioning, experiment tracking, and automation concepts.
•	Build simple ML pipelines using industry tools.
•	Deploy machine learning models using APIs and containers.
•	Understand monitoring, Explainability, and responsible AI concepts.
•	Develop practical skills for production-ready ML systems.
Course Outcomes:	
	After completion of the course, student will be able to
1.	Explain MLOps concepts and ML workflow.
2.	Apply data and model versioning techniques.
3.	Create automated ML pipelines.
4.	Deploy ML models using Docker and APIs.
5.	Implement monitoring and CI/CD basics.
6.	Understand Explainability and ethical AI practices.

Section1:	Topics/Contents
Unit 1: Introduction to MLOps and ML Lifecycle	07 Hrs
Difference between Software Dev Lifecycle vs. ML Lifecycle Introduction to MLOps: Definition, Need, Challenges. Key Concepts: Versioning, Reproducibility, Automation, Monitoring Stages of ML Lifecycle: Problem Formulation, Data Collection and Preprocessing, Model Development, Training, Evaluation, Model Deployment and Monitoring, Overview of MLOps Architecture and Workflow, Overview of popular MLOps platforms and cloud services.	

Case Studies: Student Performance Prediction System using complete ML workflow				
Unit 2: Data, Feature, and Experiment Management				07 Hrs
Introduction to Data Versioning Concepts, DVC Basics Introduction to Dataset Management Experiment Tracking using MLflow, Feature Engineering Concepts: Encoding, Scaling, Normalization Model Registry Concepts (e.g., MLflow Model Registry) Need for Reproducibility in ML Basics of Data Validation Case Studies: Track ML experiments using version control and experiment tracking tools.				
Unit 3: ML Pipeline Automation				07 Hrs
Introduction to ML Pipelines, Workflow Automation Concepts Concept of Pipelines & DAGs, Tools Overview: Apache Airflow, Automating: Data Pre-processing, Feature Engineering, Model Training & Evaluation Basics of CI/CD for Machine Learning, Basic testing and validation concepts in ML pipelines. Case Studies: Build simple workflow automation using Airflow				
Section2:	Topics/Contents			
Unit 4: Model Deployment and Serving				07 Hrs
Introduction to Model Deployment REST API using FastAPI, Docker basics for ML deployment Batch vs Real-time inference Introduction to monitoring, Logging and alerting basics Basic concepts of drift detection Case Studies: Containerize ML model using Docker				
Unit 5: Responsible and Explainable AI Systems				07 Hrs
Introduction to Responsible AI , Bias and Fairness in ML Explainability using SHAP and LIME , Basic data privacy concepts Ethical issues in AI systems , Governance basics Case Studies: Monitoring ML models and understanding model drift concepts.				

Unit 6: Cloud-based MLOps and Industry Applications		07 Hrs
Introduction to Cloud-based MLOps , Overview of cloud ML services , MLOps use cases in healthcare, agriculture, finance Best practices in ML deployment, End-to-end MLOps workflow case study Case Studies: MLOps in Agriculture / Healthcare / Finance, Best Practices in Industrial MLOps Workflows		
Text Books: <i>(As per IEEE format)</i>		
1	M. Treveil and A. Shukla, <i>Introducing MLOps: How to Scale Machine Learning Projects with Continuous Delivery, Monitoring, and Governance</i> . Sebastopol, CA, USA: O'Reilly Media, 2020.	
2	D. Brinkmann, E. Raj, and M. Daoust, <i>MLOps Engineering at Scale</i> . Sebastopol, CA, USA: O'Reilly Media, 2023.	
Reference Books: <i>(As per IEEE format)</i>		
1	.C. Osipov, <i>MLOps: Continuous Delivery and Automation Pipelines in Machine Learning</i> . Shelter Island, NY, USA: Manning Publications, 2022.	
2	N. Gift and A. Deza, <i>Practical MLOps</i> . Sebastopol, CA, USA: O'Reilly Media, 2021.	
3	E. Ameisen, <i>Building Machine Learning Powered Applications</i> . Sebastopol, CA, USA: O'Reilly Media, 2020.	

List of Practical's:

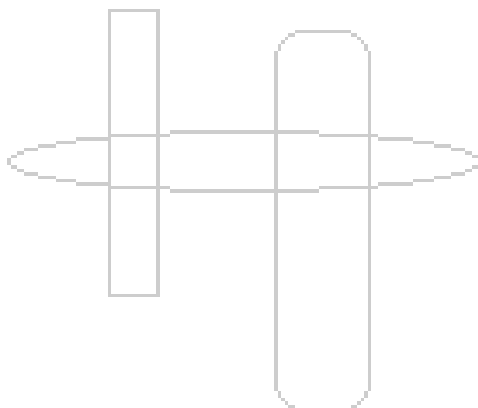
1. Implement dataset versioning using DVC
2. Track ML experiments using MLflow
3. Create feature engineering pipeline
4. Build workflow automation using Airflow
5. Create REST API using FastAPI
6. Containerize ML model using Docker
7. Implement CI/CD pipeline using GitHub Actions
8. Monitor model using Prometheus and Grafana
9. Apply SHAP/LIME for Explainability
10. Mini project on complete MLOps workflow

List of Project areas:

1. Setting up a complete MLOps pipeline
2. Deploying a machine learning model to production
3. Monitoring and maintaining a deployed model
4. Implementing CI/CD for a machine learning project

CO-PO Mapping

CO	Program Outcomes (PO)												PSO		
CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO12	PS O1	PS O2	PSO3
CO1	3	2	1	--	2	--	--	--	--	1	--	2	3	2	--
CO2	2	3	2	1	3	--	--	--	--	1	--	2	3	2	1
CO3	2	2	3	2	3	--	--	--	1	1	1	2	3	3	2
CO4	2	2	3	2	3	--	--	--	1	2	1	2	3	3	2
CO5	2	3	2	2	3	--	--	1	1	2	1	2	2	3	2
CO6	2	3	1	2	2	2	1	3	1	1	--	3	2	2	1
Average	2.2	2.5	2.0	1.5	2.7	0.3	0.2	0.7	0.5	1.3	0.5	2.2	2.7	2.5	1.3



Syllabus Template

FF No.: 654

ML3203C: Computer Network Security

Teaching Scheme	Examination Scheme
Credits : 4	CP: 40 Marks
Lectures : 3 Hrs/week	CVV: 20Marks
Practical : 2 Hrs/week	HA/GD/PPT : 20 Marks
Tutorial : Hrs/week	PPT: 20 Marks

Prerequisites:	
•	Computer Networks, Basics of Operating Systems, Data Structures and Algorithms
Course Objectives:	
•	To understand the fundamental concepts of network security, threats, and attacks
•	To study the principles of symmetric and asymmetric key cryptographic algorithms
•	To understand message authentication, hashing, digital signatures, and key management
•	To study network security protocols such as IPSec, SSL/TLS, PGP, and S/MIME
•	To understand system-level security mechanisms including firewalls and intrusion detection/prevention systems
•	To create awareness of malware, cyber-attacks, cyber laws, and ethical issues in security
Course Outcomes:	
	After completion of the course, student will be able to
1.	Explain the fundamental concepts of network security and classify various types of security attacks
2.	Apply symmetric and asymmetric cryptographic techniques to secure data and communications
3.	Analyze and design authentication, hashing, and key management mechanisms
4.	Implement and evaluate network security protocols for secure communication
5.	Configure and analyze firewalls and intrusion detection/prevention systems to protect networks
6.	Assess ethical, legal, and policy issues associated with cyber security and evolving security trends

Section1:	Topics/Contents
Unit I – Introduction to Network Security: Security goals (confidentiality, integrity, availability), security attacks (passive and active), security services and mechanisms, OSI security architecture, classical encryption techniques (substitution and transposition ciphers).Case Study: Airbnb Design Thinking Transformation	
Unit II – Symmetric Key Cryptography:	

Block cipher principles, Data Encryption Standard (DES), Triple DES, Advanced Encryption Standard (AES), block cipher modes of operation (ECB, CBC, CFB, OFB, CTR), stream ciphers (RC4), key distribution.	
Unit III – Asymmetric Key Cryptography: Principles of public key cryptosystems, RSA algorithm, Diffie-Hellman key exchange, Elliptic Curve Cryptography (ECC), key management and distribution.	
Section2:	Topics/Contents
Unit IV – Authentication and Message Integrity: Message authentication codes (MAC), cryptographic hash functions (MD5, SHA family), digital signatures, digital certificates, Public Key Infrastructure (PKI), Kerberos.	
Unit V – Network Security Protocols: IP Security (IPSec) – AH and ESP, Secure Socket Layer (SSL) and Transport Layer Security (TLS), Pretty Good Privacy (PGP), S/MIME, Secure Electronic Transaction (SET), Wireless network security basics.	
Unit VI – System Security and Cyber Law: Malicious software (viruses, worms, trojans, ransomware), Intrusion Detection and Prevention Systems (IDS/IPS), Firewalls (types and configurations), Denial of Service (DoS/DDoS) attacks, honeypots, Cyber Laws, IT Act, ethical and legal issues in cyber security, emerging trends (Cloud, IoT security).	
Text Books: (As per IEEE format)	
1	<i>William Stallings; Cryptography and Network Security: Principles and Practice; 7th Edition, Pearson Education</i>
2	<i>Behrouz A. Forouzan; Cryptography and Network Security; 3rd Edition, McGraw Hill Education</i>
3	<i>Atul Kahate; Cryptography and Network Security; 4th Edition, McGraw Hill Education</i>
Reference Books: (As per IEEE format)	
1	<i>Charlie Kaufman, Radia Perlman, Mike Speciner; Network Security: Private Communication in a Public World; 2nd Edition, Prentice Hall</i>
2	<i>William Stallings; Network Security Essentials: Applications and Standards; 6th Edition, Pearson Education</i>
3	<i>Bruce Schneier; Applied Cryptography: Protocols, Algorithms and Source Code in C; 2nd Edition, Wiley</i>

List of Practical's:

1. Implementation of Caesar Cipher and Mono-alphabetic Substitution Cipher
2. Implementation of the DES/AES symmetric encryption algorithm
3. Implementation of the RSA public key algorithm
4. Implementation of the Diffie-Hellman key exchange algorithm
5. Implementation of MD5/SHA hashing and digital signature generation and verification

6. Configuration of firewall rules using iptables
7. Study and configuration of an Intrusion Detection System (e.g., Snort)
8. Network scanning and vulnerability assessment using Nmap
9. Configuration of SSL/TLS for secure web communication
10. Mini project on network security (e.g., secure chat application or packet sniffer with traffic analysis)

List of Project areas:

1. Design of a secure network communication protocol
2. Development of an Intrusion Detection/Prevention System
3. Design and simulation of a Public Key Infrastructure (PKI)
4. Development of a network vulnerability scanner
5. Design of a secure email/messaging system using cryptographic techniques

CO-PO Mapping

CO	Program Outcomes (PO)												PSO		
CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
C01	3	2	1										2		
C02	3	3	2	1									2	1	
C03	2	3	3	2	1								2	2	1
C04	2	2	3	3	2	1							1	2	2
C05	1	2	2	3	3	2	1						1	2	3
C06		1	2	2	3	3	2	1					1	1	2
Average	2.2	2.2	2.2	2.2	2.3	2.0	1.5	1.0					1.5	1.6	2.0

Syllabus Template**FF No.: 654****ID3001: Basics of Game Development**

Teaching Scheme	Examination Scheme
Credits : 04	CP: 40 Marks
Lectures : 3 Hrs/week	PPT: 20 Marks
Practical : 2 Hrs/week	CVV: 20 Marks
Tutorial : Hrs/week	HA : 20 Marks

Prerequisites:	
•	Knowledge of OOP, Basic Java Script.
Course Objectives:	
•	learn Game Mechanics and web based game development concepts.
•	learn C# OOP fundamentals in order to do Unity 3D scripting.
•	learn programming patterns that are generally used in game development.
•	learn the basics of game services.
•	learn the basics of collision physics in gaming.
•	learn Game Mechanics and web based game development concepts.
Course Outcomes:	
	After completion of the course, student will be able to
1.	Explain the fundamentals of the gaming industry, game development lifecycle, gaming platforms, and business models.
2.	build simple interactive games using appropriate game development frameworks and programming languages for web and standalone platforms.
3.	Apply game programming concepts, software design patterns, and state management techniques to develop maintainable game applications.
4.	Apply essential game development features such as physics, collision detection, game services, networking, and asynchronous programming.
5.	Analyze game architectures, programming approaches, and implementation strategies for solving real-world game development scenarios and enhancing player experience..
6.	Design, develop, and present game-based solutions by integrating appropriate game development tools, programming techniques, and industry practices.

Section1:	Topics/Contents
Unit 1: Game Domain Gaming as business & Global market, Introduction to Gaming, Gaming Ecosystem and opportunities, Player time spending, what kind of games succeed, monetizing game, how to target the gaming crowd, what is working for real money gaming, game	

testing, types of game testing, Realism and compliance testing, multiplayer and network testing	
Unit 2: Web based Game programming Basics of HTML, CSS, Java Script, introduction to phaser.io js library	
Unit 3: Standalone Game Platform C# fundamentals with all OOPs concepts, Introduction to Unity 3D Engine, C# scripting in Unity 3D Engine	
Section2:	Topics/Contents
Unit 4: Programming Design Patterns in Game Development Observer Design Pattern, State Design Pattern, Command Design Pattern, Decorator Design Pattern	
Unit 5: Game Services Choosing a Gamer Service, Basic Setup, Lobbies and Match making, Networking, Player Statistics, Player Achievements Leaderboards, Other Service,	
Unit 6: Game Physics: Collision Detection The Collision Detection Pipeline, Broad Phase Collision Detection, Requirements, Bounding Volume Hierarchies, Hierarchies Building the Hierarchy, Sub-object Hierarchy, Spatial Partitioning, Binary Space Partitioning, Oct-Trees and Quad-Trees, Grids, Multiresolution Maps	
Text Books: <i>(As per IEEE format)</i>	
1	
2	
Reference Books: <i>(As per IEEE format)</i>	
1	Colt McAnlis, et al., “HTML5 Game Development Insights”, 2014, Apress, ISBN: 978-1-4302-6698-3. (https://doi.org/10.1007/978-1-4302-6698-3)
2	Joshua Glazer, Sanjay Madhav, “ Multiplayer Game Programming, Architecting Networked Games”, 2016, Addison-Wesley, ISBN-13: 978-013-403430-0
3	Ian Millington, “Game Physics Engine Development”, 2007, Morgan Kaufmann, ISBN-13: 978-0-12-369471-3

For MOOCs and other learning Resources :

1. Udemy
2. Coursera

List of Practical's:

1.Snake Game

Develop a simple Snake game using JavaScript and H5 and a game framework such as Phaser.io. The game should include the following features:

1. A snake that can move in four directions (up, down, left, right) using the arrow keys.

2. Food items that randomly appear on the game field.
3. The snake grows longer each time it eats a food item.
4. The game ends when the snake collides with itself or the walls.
5. A scoring system that tracks the number of food items eaten.

2. Tetris Game

Develop a simple Tetris game using C# and a game framework such as Unity. The game should include the following features:

1. Randomly generated Tetriminos (tetrominoes).
2. The ability to move and rotate Tetriminos.
3. The game field (grid) should be able to detect filled rows and clear them.
4. The game should end when the Tetriminos reach the top of the grid.
5. A scoring system that tracks the number of cleared lines.

3. State Machines in Gaming

Implementation Related to State Machines:

Implement a state machine to manage the state of a traffic light system.

Write the code for a traffic light state machine with states like "Green," "Yellow", "Red" and transitions between these states.

How would you handle asynchronous actions within a state machine?

Provide examples of handling asynchronous actions (like API calls) within a state machine using JavaScript.

4. Scenario Based statement

Describe a scenario where you would use JavaScript for UI and C# for game logic in a game. Provide an example where this separation of concerns is beneficial and explain how the integration would work.

How would you debug issues in a game with integrated JavaScript and C# components?

Outline the tools and techniques you would use to debug problems that arise from the interaction between JavaScript and C#. How would you implement cross-platform support for a game using JavaScript and C#?

Describe strategies for ensuring that a game runs smoothly on multiple platforms (e.g., web, mobile, desktop).

5. Core Gaming (Understanding, Analysis and Presentation)

Flowchart/Presentation

Title: Enhancing Player Engagement in Candy Crush/Temple Run/Puzzle game Using

Design Patterns

Observer Pattern

Description: Real-time notifications for friend activities and game events.

Impact: Increases social interaction and competitive spirit.

Visual: Diagram showing notification flow from game server to players.

State Pattern

Description: Manage different game states for varied gameplay.

Impact: Keeps the game fresh and challenging.

Visual: Flowchart showing transitions between Normal Play, Special Event, and Timed Challenge states.

Command Pattern

Description: Allow players to undo moves and activate abilities.

Impact: Enhances control and reduces frustration.

Visual: Sequence diagram illustrating the Command pattern for undoing a move.

Decorator Pattern

Description: Add dynamic features like power-ups and skins.

Impact: Provides customization and boosts player excitement.

Visual: Diagram showing how decorators add new features to game components.

Conclusion

Summary: Recap of the suggested patterns and their benefits.

Next Steps: Implementation plan and expected outcomes.

6. Async and sync Implementation in Gaming

How do you define an asynchronous function in JavaScript?

Provide examples of defining asynchronous functions using `async` and `await` keywords.

What are Promises and how are they used in JavaScript?

Explain the concept of Promises and their role in managing asynchronous operations.

Implement a basic game loop using synchronous functions in JavaScript.

Write code for a simple game loop that updates game state and renders frames synchronously.

Convert the above synchronous game loop into an asynchronous one.

Rewrite the game loop using `requestAnimationFrame` and asynchronous functions to prevent blocking the main thread.

How would you handle asynchronous loading of game assets (e.g., images, sounds) in JavaScript?

Provide code examples for loading game assets asynchronously using Promises or `async/await`.

Implement a function to fetch high scores from a server asynchronously in a JavaScript game

Write a function that uses `fetch` or `XMLHttpRequest` to retrieve high scores from an external server without blocking the game.

Describe how you would manage asynchronous player inputs in a multiplayer game

Explain techniques for handling player inputs asynchronously, ensuring that the game remains responsive and updates in real-time.

CO-PO Mapping

CO/PO	Program Outcomes (PO)												PSO			
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3	PSO4
CO1	3	2	3	-	3	1	-	-	-	-	-	-	3	1	-	-
CO2	3	2	3	-	3	1	-	-	2	-	-	-	3	2	-	-
CO3	3	2	3	-	3	1	-	-	2	-	-	-	3	3	2	-
CO4	3	2	3	-	3	2	1	1	1	-	1	-	3	2	-	-
CO5	3	2	3	-	3	2	1	2	2	-	2	2	3	1	1	1
CO6	3	2	3	-	3	3	1	3	3	3	3	3	3	1	-	2
Average	3	2	3	0	3	1.67	2	2	2	3	2	2.5	3	1.67	1.5	1.5

Syllabus Template**FF No.: 654****MM0301A: Cloud Computing**

Teaching Scheme	Examination Scheme
Credits : 3	CP: 30 Marks
Lectures : 2 Hrs/week	HA: 10Marks
Practical : -	HA/GD/PPT : 20 Marks
Tutorial : 1 Hrs/week	ESE (W/O): 40 Marks

Prerequisites:	
●	Operating Systems, Computer Networks, Database Management System
Course Objectives:	
●	To become familiar with cloud computing and its ecosystem
●	To acquire basics of virtualization and its importance
●	To evaluate in-depth analysis of Cloud Computing capabilities and its services.
●	To configure and implement storage services.
●	To analyze different cloud-based services to meet a set of given requirements.
●	To design security aspects for cloud computing
Course Outcomes:	
	After completion of the course, student will be able to
1.	Describe the main concepts, key technologies, strengths, and limitations of cloud computing and the possible applications for state-of-the-art cloud computing
2.	Explain the architecture and infrastructure of cloud computing, including SaaS, PaaS, IaaS, public cloud, private cloud, hybrid cloud, etc.
3.	Identify problems, and explain, analyze, and evaluate various cloud computing solutions.
4.	Choose the appropriate technologies, algorithms, and approaches for the related issues.
5.	Display new ideas and innovations in cloud computing.
6.	Collaboratively research and write a paper on the state of the art (and open problems) in cloud computing

Section1:	Topics/Contents
	Unit-I Introduction to Cloud Computing [4 Hrs] Recent trends in computing, Cluster computing, Distributed computing ,Evolution of cloud computing, Cloud versus traditional architecture, Cloud Computing Architecture, Google Cloud architecture, Infrastructure as a Service (IaaS), Platform as a Service (PaaS), Software as a Service (SaaS), Public cloud, Private cloud, Hybrid cloud, Community cloud.

Unit-II Virtualization [6 Hrs] Introduction to virtualization, Different approaches to virtualization, Hypervisors, Machine Image, Virtual Machine (VM), Compute options in the cloud, Exploring IaaS with Compute Engine, Configuring elastic apps with auto scaling, Basics of virtualization and implementation challenges. System virtualization technologies-architectures and internals. KVM, Xen, VMware. Amazon Elastic Compute Cloud EC2 as computing service. Memory virtualization-virtualization techniques, ballooning, deduplication and sharing. Network and storage virtualization, Virtual machine migration and replication techniques pre-copy and post-copy techniques, applicability to system availability.	
Unit-III Cloud Services [4 Hrs] Service Oriented Architecture (SOA), Web services, Web 2.0, Web OS. Introduction to IaaS, PaaS, SaaS. Cloud Platform and Management, Exploring PaaS with App Engine, Event driven programs with Cloud Functions, Containerizing and orchestrating apps with Google Kubernetes Engine Software as a Service (SaaS) Docker flow, orchestration with Docker, dynamic linking and legacy linking of containers. The GCP Console, understanding projects, Billing in GCP, Install and configure Cloud SDK, Use Cloud Shell, GCP APIs.	
Section2:	Topics/Contents
Unit-IV Cloud Storage [4 Hrs] Storage options in the cloud, Structured and unstructured storage in the cloud, unstructured storage using Cloud Storage, SQL managed services, Exploring Cloud SQL, Cloud Spanner as a managed service, NoSQL managed service options, Cloud Datastore, a NoSQL document store, Cloud Bigtable as a NoSQL option. OpenStack: NOVA, Neutron, Keystone Cinder, Swift and Glances, VMware Suit, Apache Cloud Stack, Data Lakes, Snowflake.	
Unit-V Service Management [4 Hrs] Service Level Agreements (SLAs), Billing and accounting, Billing in GCP Cloud Security: Introduction to security in the cloud, the shared security model, Encryption options, Authentication and authorization with Cloud IAM, Identify Best Practices for Authorization using Cloud IAM., Introduction to configuration and management tools Ansible, Architecture of DevOps.	
Unit-VI Cloud Network and Security [6 Hrs] Introduction to networking in the cloud, defining a Virtual Private Cloud, Public and private IP address basics, Google's network architecture, Routes and firewall rules in the cloud, Multiple VPC networks, building hybrid clouds using VPNs, interconnecting, and direct peering, Different options for load balancing. Introduction to security in the cloud, the shared security model, Encryption options	
Text Books: <i>(As per IEEE format)</i>	
1	Judith Hurwitz, R.Bloor, M.Kanfman, F.Halper, "Cloud Computing for Dummies", Wiley,India.
2	Ronald Krutz and Russell Dean Vines, "Cloud Security", Wiley-India
3	Gautam Shroff. "Enterprise Cloud Computing", Cambridge

Reference Books: (As per IEEE format)	
1	Barrie Sosinsky, "Cloud Computing Bible", Wiley India
2	Anthony T Velte, et.al, "Cloud Computing : A Practical Approach", McGraw Hill.
3	Michael Miller, "Cloud Computing", Que Publishing.
4	Tim Malhar, S.Kumaraswamy, S.Latif, "Cloud Security & Privacy", SPD,O'REILLY
5	Scott Granneman, "Google Apps", Pearson

List of Tutorials :

- 1) Install VirtualBox/VMware Workstation with different Linux or Windows Operating Systems.
- 2) Study Google Cloud Architecture.
- 3) Find a procedure to launch virtual machine
- 4) Find a procedure to transfer the files from one virtual machine to another virtual machine.
- 5) Simulate a cloud scenario using CloudSim and run a scheduling algorithm that is not present in CloudSim.
- 6) Install Google App Engine. Create Hello World App and other simple web applications using python/java.
- 7) Launch the Web Applications using GAE launcher.
- 8) Install Hadoop single node cluster and run simple applications like wordcount.
- 9) Use AWS Pricing Calculator: Create estimate for EC2 Compute cost for VM instance. Use region closest to you. Find On demand cost and compare the pricing for other regions.
- 10) Launch EC2 instance and explore Public/Private/Elastic IP

List of Practicals (Any Six):

- 1) To setup AWS accounts and launch instances.
- 2) To install an OS using VirtualBox/ VMWare Workstation. Add Storage to create new virtual disk.
- 3) To Deploy Virtual Machine on hypervisor such as KVM, ESXi. Take Backup and Migrate them.
- 4) To use Infrastructure as a Service to facilitates for creating and deleting compute resources. Create network and attach volumes to run instances.
- 5) To install docker on window/linux and build docker image from docker hub.
- 6) Deploy a stateless/stateful application on Kubernetes cluster.
- 7) To work on different Cloud Storage Service
- 8) To create login into AWS and use S3 Bucket Service for storage.
- 9) Develop elastic services for dynamic load scenario using AWS APIs. Build load balancer and explore on scalability, fault detection and performance.

Course Projects:

List of Course Project Topics

1. Creating Google Account to store files and programs.
2. Creating Account to Store Images.
3. Creating a Warehouse Application in Salesforce.com
4. Creating an Application in Salesforce.com using Apex programming Language.
5. To study and implement Web services in SOAP for JAVA Applications.
6. Implementation of Para-Virtualization using VMWare 's Workstation/ Oracle's Virtual Box and Guest Operator System.
7. Installation and Configuration of Hadoop.
8. AWS Case Study: Amazon.com.
9. Case Study of Google App Engine.
10. Case Study of Face book.

Seminars:

List of Course Seminar Topics:

1. Storage Cost Optimization on Cloud.
2. Cloud Security and Cryptography
3. Infrastructure as a Code (IAC)
4. Cloud Computing in Healthcare
5. Serverless
6. Deployment of Microservices in Kubernetes Engine
7. RPA Using AWS Cloud
8. Cloud Trends in Supporting Ubiquitous Computing
9. Mobile Cloud Computing
10. Modern Data Center Architecture

List of Home Assignments:

List of Design Based Home Assignments:

1. Serverless Web App to order taxi rides using AWS lambda.
2. Deploying App on Kubernetes.
3. Serverless web Application (GCP Cloud Functions).
4. Demonstration of EBS, Snapshot, Volumes.
5. Single Node Cluster Implementation (Hadoop).

List of Case Study Based Home Assignments

1. PayU Migration to AWS.
2. Cloud object storage.

3. Deployment and Configuration options in AWS.
4. Deployment and Configuration options in Microsoft Azure.
5. Deployment and Configuration options in GCP.

List of Blog Based Home Assignment

1. Comparing design of various cloud computing platforms.
2. AWS EKS and Google Cloud Functions.
3. App Engine.
4. Cloud Endpoints.
5. Cloud Pub/Sub.

List of Survey Based Home Assignments

1. Disaster Recovery in Cloud Computing.
2. Cloud Economics.
3. Data archiving solutions.
4. Salesforce.
5. Dropbox.

MOOCs Links and additional reading material:

<https://nptel.ac.in/courses/106/105/106105167/>
https://swayam.gov.in/nd1_noc20_cs55/preview
<https://www.coursera.org/specializations/cloud-computing>
<https://azure.microsoft.com/en-in/overview/what-is-cloud-computing/>
<https://aws.amazon.com/what-is-cloud-computing/>
<https://www.ibm.com/in-en/cloud/learn/cloud-computing>

CO-PO Mapping

	Program Outcomes (PO)												PSO		
CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	1			1								2	2	3
CO2	2	2	1	1	1								2	2	3
CO3	3	2	2	2	2		3	3					2	2	3
CO4	3	2	2	2	3	3			3				2	2	3
CO5	3	3	1	3	3				1		2		2	2	3
CO6	2	2	1	3	1					3		3	2	2	2
Average	2.50	2.00	1.40	2.20	1.83	3.00	3.00	3.00	2.00	3.00	2.00	3.0	2.0	2.0	2.83

Syllabus Template**FF No.: 654****MM1501D: Product Lifecycle Management (PLM)**

Teaching Scheme	Examination Scheme
Credits : 3	CP: 30 Marks
Lectures : 2 Hrs/week	HA: 10Marks
Practical : Hrs/week	HA/GD/PPT : 20 Marks
Tutorial : 1 Hrs/week	ESE (W/O): 40 Marks

Prerequisites:	
Students should have basic knowledge of programming, software engineering, and fundamentals of AI. Familiarity with problem-solving techniques, and basic application development concepts will help in understanding PDLC effectively.	
Course Objectives:	
- To apply Design Thinking principles to identify and solve real-world problems. - To understand the stages and significance of Product Development Life Cycle. - To analyze and evaluate different PDLC and software development models. - To use AI tools and techniques in product planning, development, testing, and maintenance. - To develop practical skills through prototyping, testing, and case-study-based learning. - To study industrial practices followed in modern AI-driven product companies.	
Course Outcomes:	
	After completion of the course, student will be able to
1.	Apply Design Thinking principles to identify user-centric problems and develop innovative product ideas.
2.	Design AI-assisted product plans, user journeys, prototypes, and Minimum Viable Products (MVPs).
3.	Explain and analyze various stages of the Product Development Life Cycle (PDLC) and product planning methodologies.
4.	Compare, evaluate, and select suitable product development models such as Agile, Waterfall, Spiral, Prototype, and DevOps.
5.	Apply testing, quality assurance, deployment, and maintenance techniques for reliable product development.
6.	Analyze the role of Artificial Intelligence in PDLC through real-world industrial case studies and practical applications.

Section1:	Topics/Contents
	Unit-I: Design Thinking for Product Innovation 4 Hours Introduction to Innovation and Product Development, Need for Product Development in AI-driven Industries, Introduction to Design Thinking, Human-Centered Product Design, Five Stages of Design Thinking, Customer Need Analysis, Problem Identification and Opportunity Mapping, Brainstorming and Ideation Techniques, Introduction to Real-world Product Innovation Case Study: Airbnb Design Thinking Transformation Unit II AI-Assisted Design Thinking and Product Planning 5 Hours Integration of Design Thinking with PDLC, AI in Product Ideation and Requirement Gathering, Functional and Non-functional Requirements, User Journey Mapping, Wireframing and Rapid Prototyping, Minimum Viable Product (MVP), Ethical and Sustainable AI Product Design, AI Tools for Product Planning and Documentation, Product Roadmap Preparation, Introduction to Product Analytics, Customer Feedback Collection Techniques. Case Study: OpenAI and ChatGPT MVP Evolution and Feature Planning. Unit III: Fundamentals of Product Development Life Cycle 5 Hours Introduction to PDLC, Product vs Project, Stages of Product Development Life Cycle, Product Data Management (Product Information Management, Document Management, Version Control, Revision Management, BOM (Bill of Materials, Workflow and Change Management)), Product Team Roles and Responsibilities, Introduction to Agile Product Culture. Case Study: Swiggy / Zomato Product Lifecycle Evolution.
Section2:	Topics/Contents
	Unit IV: Product Development Models and Methodologies 5 Hours Waterfall Model, Prototype Model, Incremental Model, Spiral Model, Agile Model, Scrum Framework, DevOps Methodology, Continuous Integration and Continuous Deployment (CI/CD), Comparative Study of PDLC Models, Cost Estimation using COCOMO II and Agile Estimation, Model Selection Criteria. Industry Tool Exposure: Introduction to Jira, User Story Creation, Product Backlog Management, Sprint Planning, Sprint Board and Burndown Charts. Case Study: Spotify Agile and DevOps Model. UNIT V: Product Testing, Quality Assurance and Maintenance 5 Hours Product Testing, Verification and Validation, Functional and Non-functional Testing, Unit Testing, Integration Testing, System Testing, User Acceptance Testing (UAT), Performance and Security Testing, Bug Tracking and Defect Management, Product Deployment and Maintenance, Product Monitoring and Feedback Systems. Case Study: Tesla Autonomous Product Testing and Deployment.

Unit VI: AI in PDLC and Industrial Product Case Studies		4 Hours
Introduction to Product Management, Role of Product Manager, Project Manager vs Product Owner, Team Coordination and Cross-functional Collaboration, Product Analytics Basics(Comparative Analysis of Diverse Product), Key Performance Indicators (KPIs), User Engagement Metrics, A/B Testing Concepts, Customer Feedback Analytics, Product Performance Monitoring, Cost and Budget Considerations, Industry Reporting and Documentation, Industry Tools: Jira, GitHub, Slack, Leadership and Communication Skills. Case Study: Google Product Ecosystem and Product Management Practices.		
Text Books: <i>(As per IEEE format)</i>		
1	K. T. Ulrich and S. D. Eppinger, Product Design and Development, 6th ed. New Delhi, India: McGraw-Hill Education, 2016, ISBN: 978-9352601851.	
2	T. Brown, Change by Design: How Design Thinking Creates New Alternatives for Business and Society. New York, USA: Harper Business, 2009, ISBN: 978-0061766084.	
3	I. Sommerville, Software Engineering, 10th ed. Boston, USA: Pearson Education, 2015, ISBN: 978-9332582699.	
4	E. Ries, The Lean Startup: How Today’s Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses. New York, USA: Crown Business, 2011, ISBN: 978-0307887894.	
5	M. Grieves, Product Lifecycle Management: Driving the Next Generation of Lean Thinking, New York, USA: McGraw-Hill Education, 2006, ISBN: 978-0071452304.	
Reference Books: <i>(As per IEEE format)</i>		
1	K. Schwaber, Agile Project Management with Scrum. Redmond, USA: Microsoft Press, 2004, ISBN: 978-0735619937.	
2	J. Humble and D. Farley, Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation. Boston, USA: Addison-Wesley Professional, 2010, ISBN: 978-0321601919.	
3	J. Stark, Product Lifecycle Management: 21st Century Paradigm for Product Realization, 3rd ed., London, UK: Springer, 2015, ISBN: 978-3319174433.	
4	R. Pichler, Agile Product Management with Scrum: Creating Products that Customers Love. Boston, USA: Addison-Wesley Professional, 2010, ISBN: 978-0321605788.	
5	A. Saaksvuori and A. Immonen, Product Lifecycle Management, 3rd ed., Berlin, Germany: Springer, 2008, ISBN: 978-3540781738.	
6	M. Perri, Escaping the Build Trap: How Effective Product Management Creates Real Value, Sebastopol, USA: O'Reilly Media, 2018, ISBN: 978-1491973790.	

List of Tutorials

- Problem Identification and Empathy Mapping using Design Thinking

- User Persona Creation and Customer Journey Mapping
- Product Vision, Value Proposition and MVP Canvas Preparation
- Requirement Analysis and SRS Preparation for Selected Product
- Product Roadmap Design and Feature Prioritization
- Select a real-world product and justify an appropriate PDLC model selection
- Jira-based User Story Creation, Product Backlog and Sprint Planning
- Design Test Cases and Product Quality Evaluation
- Product Analytics: KPI identification, customer feedback and A/B testing analysis
- Industry Mini Product Development Project (Problem identification → Persona → MVP → Sprint → Prototype → Testing → Final presentation) Case Study Analysis and Presentation on AI-enabled Products such as ChatGPT, Tesla, Google Gemini, Swiggy/Zomato, Healthcare AI, or Agriculture AI Systems.

List of Course Projects:

1. AI-based Smart Attendance and Analytics System
2. Personalized Learning Recommendation Platform
3. Smart Agriculture Advisory System
4. AI Resume Screening System
5. Healthcare Appointment and Disease Prediction System
6. AI-powered Campus Assistance Chatbot
7. Smart Traffic Monitoring System
8. Product Recommendation Engine
9. Intelligent Retail Analytics System
10. AI-enabled Waste Management System

List of GD/PPT Topics:

1. Agile vs Waterfall in modern industries
2. Role of AI in Product Development
3. Product Analytics and Business Decisions
4. MVP: Build Fast or Build Perfect?
5. Can AI replace Product Managers?
6. ChatGPT and Generative AI Product Evolution
7. Startup Ecosystem and Product Innovation
8. Ethical Issues in AI Products
9. DevOps Culture in Industry
10. Product Failures and Lessons Learned

List of Home Assignments:

1. Analyze a successful product using Design Thinking principles
2. Create user personas for an AI product
3. Write user stories and acceptance criteria
4. Prepare a product roadmap for a startup idea
5. Compare two PDLC models for a selected application

6. Create sprint plans using Jira concepts
7. Study KPIs used in Netflix/Spotify/Zomato products
8. Analyze customer reviews and suggest improvements
9. Create a bug report template
10. Study one industrial product lifecycle and prepare a report

MOOCs Links and additional reading material:

1. Coursera – Product Management, Agile, and Design Thinking Courses (<https://www.coursera.org>)
2. edX – Software Engineering and Product Development and AI Courses (<https://www.edx.org>)
3. Scrum Alliance – Scrum and Agile Learning Resources (<https://www.scrumalliance.org>)
4. Atlassian Agile Coach – Agile, Scrum and JIRA Tutorial (<https://www.atlassian.com/agile>)
5. Atlassian University – Jira Learning and Product Management Resources (<https://university.atlassian.com>)
6. Google for Developers – AI and Product Development Resources (<https://developers.google.com>)
7. OpenAI – Generative AI and AI Product Resources (<https://openai.com>)
8. GitHub Skills – Version Control, Collaboration and DevOps Tutorials (<https://skills.github.com>)
9. IBM Skills Build – AI, Product Development and Professional Skills Courses (<https://skillsbuild.org>)
10. NPTEL – Software Engineering, Design Thinking and Product Development Courses (<https://nptel.ac.in>)
11. MIT OpenCourseWare – Product Design and Software Engineering (<https://ocw.mit.edu>)

CO-PO Mapping

CO	Program Outcomes (PO)												PSO		
CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	3	2	2	1	-	2	2	-	-	1	2	3	2
CO2	2	2	3	2	3	-	-	2	2	2	-	1	3	3	2
CO3	3	3	2	2	2	-	-	-	1	-	1	2	2	3	2
CO4	2	3	3	3	2	-	-	-	2	1	1	2	2	3	2
CO5	2	2	3	2	3	-	-	1	2	1	2	2	3	3	2
CO6	2	3	2	3	2	2	1	2	2	2	1	2	3	2	3
Average	2.3	2.6	2.6	2.3	2.3	1.5	1.0	1.7	1.8	1.5	1.2	1.6	2.5	2.8	2.1

Syllabus Template

FF No.: 654

ML3207: Design Thinking -3

Teaching Scheme	Examination Scheme
Credits : 1	CP: Marks
Lectures : Hrs/week	GD/PPT/HA: Marks
Practical : Hrs/week	MSE (W/O):30 Marks
Tutorial : 1 Hrs/week	ESE (W/O):70 Marks

Prerequisites:	
•	Course Prerequisites: Problem Based Learning, Project Centric Learning Course Relevance:
Course Objectives:	
•	To provide ecosystem for students and faculty for paper publication and patent filing
Course Outcomes:	[Publication of paper or patent]
	After completion of the course, student will be able to
1.	Understand the importance of doing Research
2.	Interpret and distinguish different fundamental terms related to Research
3.	Apply the methodology of doing research and mode of its publication
4.	Write a Research Paper based on project work
5.	Understand Intellectual property rights
6.	Use the concepts of Ethics in Research , Understand Entrepreneurship and Business Planning
7.	

Section1:	Topics/Contents
	What is research? Importance of Paper Publication and Patents Importance of Paper Publication and Patents Structure of Paper Journal Publication Publication in conference Literature Review Research Paper Writing Journal Ratings and Evaluation How to rate a Journal? Intellectual property (IP) Research Ethics Entrepreneurship
Section2:	Topics/Contents
	Structure of The paper Journal List (Top 50 Journals) Selection of the journal Use of various online journal selection tools Plagiarism checking Improving contents of the paper Patent drafting Patent search Filing of patent

Writing answers to reviewer questions Modification in manuscript
 Checking of publication draft

CO – PO / PSO Articulation Matrix

Correlation: 1 = Low (Slight), 2 = Medium (Moderate), 3 = High (Substantial), '-' = no correlation.

CO	Program Outcome (PO)												PSO		
CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	1	1	1	1	1							1	1	2	2
CO2	1	1	1	1	1							1	1	2	2
CO3	2	2	3	3	2	2	1	2	2	3		1	1	3	3
CO4	3	3	3	3	3	2	1	2	2	3	1	1	1	3	3
CO5	1	1	1	1	1							1	1	2	2
CO6	2	2	2	2	2	2	1	3	2	3		1	1	3	3
Avg	1.7	1.7	1.8	1.8	1.7	2.0	1.0	2.3	2.0	3.0	1.0	1.0	1.0	2.5	2.5

Syllabus Template**FF No.: 654****ML3204: Engineering Research & Innovation**

Teaching Scheme	Examination Scheme
Credits : 2	CP: Marks
Lectures : Hrs/week	GD/PPT/HA: Marks
Practical : Hrs/week	MSE (W/O):30 Marks
Tutorial : Hrs/week	ESE (W/O):70 Marks

Prerequisites:	
•	Course Prerequisites: Research Based Learning Course Relevance: Research Centric Learning (RCL) is a powerful tool for students to work in areas of their choice and strengths. Along with course based research projects, curriculum can be enriched with semester long Engineering Research and Innovation courses, in which students can investigate socially relevant problems using various technologies and research methods from relevant disciplines. The various socially relevant domains can be like Health care, Agriculture, Defense, Education, Smart City, Smart Energy and Swaccha Bharat Abhiyan. To gain the necessary skills to tackle such research problems, students can select relevant online courses and acquire skills from numerous sources under guidance of faculty and enrich their knowledge in the research domain, thereby achieving research centric learning. Modern world sustained and advanced through the successful completion of research and innovation. In short, if students are prepared for success in life, we need to prepare them for a research-based world. It is a style of active learning and inquiry-based learning. Research centric learning will also redefine the role of teacher as mentor in the learning process. The RCL model focuses the student on a big open-ended question, challenge, or problem to research and respond to and/or solve. It brings students not only to know, understand and remember rather it takes them to analyze, design and apply categories of Bloom's Taxonomy.
Course Objectives:	
•	To develop critical thinking and problem solving ability by exploring and proposing solutions to realistic/social problems.
•	To Evaluate alternative approaches, and justify the use of selected tools and methods,
•	To emphasize learning activities those are long-term, inter-disciplinary and student-centric.
•	To engage students in rich and authentic learning experiences.

●	To provide every student the opportunity to get involved either individually or as a group so as to develop team skills and learn professionalism.
●	To develop an ecosystem to promote entrepreneurship and research culture among the students
Course Outcomes:	
	After completion of the course, student will be able to
1.	Identify a real-life research problem or gap from societal and technological need point of view
2.	Choose and compare alternative research methodologies to select the most feasible one
3.	Analyze and synthesize the identified research problem from a technological and scientific perspective.
4.	Design a reliable and scalable research methodology/experimental approach to meet the challenges.
5.	Evaluate the research outcomes based on the criteria specified
6.	Inculcate a long life learning and research attitude towards the societal problems

Section1:	Topics/Contents 
Preamble - The content and process mentioned below is the guideline document for the faculties and students to start with. It is not to limit the flexibility of faculty and students; rather they are free to explore their creativity beyond the guideline mentioned herewith. For all courses of ERI, laboratory course contents of “Engineering Research” are designed as a ladder to extend connectivity of research methodologies and emerging technologies to solve real world problems using an interdisciplinary approach. The ladder in the form of gradual steps can be seen as below: Industry Communication Standards, Research Databases and Literature Repositories (IEEE Xplore, Scopus, Google Scholar), Patent Search and IPR Tools, Statistical and Data Analysis Tools, Computational Biology (Biomedical and Bioinformatics), Robotics and Automation, Industry 4.0 (Artificial Intelligence, Machine Learning, 5G and IoT, Cloud Computing, Big Data and Cyber Security etc). Group Structure: There should be a team/group of 4-5 students. A supervisor/mentor teacher assigned to individual groups. It is useful to group students of different abilities and nationalities together. Selection of Research Problem/Topic: ● Students must focus on initiating the task/idea. The idea of inception and consideration shall be from following areas as a real world problem: -Health Care, Agriculture, Defense, Education, Smart City, Smart Energy, Swaccha Bharat Abhiyan, Environment, Women Safety. ● This is the sample list to start with. Faculty and students are free to include other areas which meet society’s requirements at large.	

- The model begins with the identifying of a problem, often growing out of a question or “wondering”. This formulated problem then stands as the starting point for learning. Students design and analyze the problem/project within an articulated disciplinary subject frame/domain.
- A problem can be theoretical, practical, social, technical, symbolic, cultural, and/or scientific and grows out of students’ wondering within different disciplines and professional environments. A chosen problem has to be exemplary. The problem may involve an interdisciplinary approach in both the analysis and solving phases.
- By exemplarity, a problem needs to refer to a particular practical, scientific, social and/or technical domain. The problem should stand as one specific example or manifestation of more general learning outcomes related to knowledge and/or modes of inquiry.

Teacher’s Role in RCL:

- Teacher is not the source of solutions rather he will act as the facilitator and mentor.
- To utilize the principles of problem solving, critical thinking and metacognitive skills of the students.
- To make the group aware about time management.
- Commitment to devote the time to solving student’s technical problems and interested in helping students to empower them better.

Student’s Role in RCL:

- Students must have ability to initiate the task/idea, they should not be mere imitators.
- They must learn to think.
- Students working in RCL must be responsible for their own learning.
- Students must quickly learn how to manage their own learning, instead of passively receiving instruction.
- Students in RCL are actively constructing their knowledge and understanding of the situation in groups.
- Students in RCL are expected to work in groups.
- They have to develop interpersonal and group process skills, such as effective listening or coping creatively with conflicts.

Developing Inquiry Skills:

- Students in PCL are expected to develop critical thinking abilities by constantly relating: What they read to do? What do they want to do with that information?
- They need to analyze information presented within the context of finding answers.
- Modeling is required so that the students can observe and build a conceptual model of the required processes.
- Use the following mechanism to maintain the track of moving towards the solution.

How effective is?

How strong is the evidence for?

How clear is?

- What are the justifications for thinking? Why is the method chosen?
- What is the evidence given to justify the solution?

Literature Survey – To avoid reinvention of wheel:

- It is an integral part of self- directed learning
- Identify the information needed to solve a given problem or issue
- Be able to locate the needed information
- Use the information to solve the given problem effectively.
- Skills required by students in information literacy include:
- · How to prepare for the search? How to carry out research
- Sorting and assessing of information in general

Use of Research Methodology: - investigation, collaboration, comprehension, application, analysis, synthesize and evaluation

Focus on the following skills while working in a team to reach to solution:

- Collaborative learning
- Interpersonal Skills
- Resources Evaluation
- Metacognitive Skills
- Reflection Skills

Section2:	Topics/Contents
------------------	------------------------

ERI Sample Research Case Studies: -

With the adaptation of industry communication standards and modern research tools and platforms, the following research problems can be taken up:

- 1) Research on soil moisture prediction and detection using sensor data and machine learning
- 2) Research on temperature monitoring and prediction using IoT sensors and analytics
- 3) Research on pressure sensing and anomaly detection techniques for industrial applications
- 4) Research on early smoke and fire detection using sensor fusion and AI
- 5) Research on human motion detection and activity recognition using sensors
- 6) Research on collision detection and avoidance systems using sensor fusion
- 7) Research on acoustic and sound-based anomaly detection using machine learning

...not limited to.....Faculty and students are free to include other areas which meet society's requirements at large.

Text Books: <i>(As per IEEE format)</i>	
1	Research Methodology: Methods and Techniques. By C.R. Kothari. New Age International Publishers. ISBN:978-8122415223; 2004
2	Research Methodology: A Step-by-Step Guide for Beginners. By Ranjit Kumar, SAGE Publications. 2019.
3	Practical Research: Planning and Design. By Paul D. Leedy, Jeanne Ellis Ormrod, Pearson. 2019
4	The Craft of Research. By Wayne C. Booth, Gregory G. Colomb, Joseph M. Williams. University of Chicago Press. 2016
Reference Books: <i>(As per IEEE format)</i>	
1	Creswell J.W.: Research Design: Qualitative, Quantitative and Mixed Methods Approaches. Sage Publications. 2014.
2	Research project management core textbook, second edition, Indian Edition, by Gopalan.
3	The Craft of Research. By Wayne C. Booth, Gregory G. Colomb & Joseph M. Williams

CO – PO / PSO Articulation Matrix

Correlation: 1 = Low (Slight), 2 = Medium (Moderate), 3 = High (Substantial), '-' = no correlation.

CO	Program Outcome (PO)												PSO		
CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	2	2	2					3		2	2	3	3	2
CO2	2	2	3	2	2		2		3		2	2	3	3	2
CO3	2	2	3	2	3		2		3	1	2	2	3	3	2
CO4	2	2	3	2	3	3		2	3		2	2	3	3	2
CO5	2	2	3	2	3	2			3		2	2	3	3	2
CO6	2	2	3	3	2				3		3	2	3	3	2
Average	2.0	2.0	2.83	2.83	2.6	2.5	2.0	2.0	3.0	1.0	2.16	2.0	3.0	3.0	2.0