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)

With Effect from Academic Year 2026-27

Prepared by: - Board of Studies in CSE (Artificial Intelligence)

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

Chairman – BOS **Chairman – Academic Board**

INDEX

Vision of the Institution

"To be 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 Citizen.

Vision of the Department

Excellence in Computer Science and Engineering, specialized in Artificial Intelligence, fostering skilled professionals and innovators for holistic societal Development

Mission of the Department

- To equip aspiring engineers for industry, academia and entrepreneurship by providing quality education in emerging Artificial Intelligence techniques.
- To impart value-added, research-oriented technical education by inculcating life skills, critical thinking and human values.
- To educate and empower learners with professional integrity through innovation, industry engagement and higher studies in Artificial Intelligence for sustainable solutions.
- To nurture ethical and socially responsible professionals by fostering lifelong learning skills for holistic societal development.

Program Educational Objectives (PEOs)

PEO	PEO Focus	PEO Statements- AI
PEO1	**Core competence**	Demonstrate core c**ompetence** in principles of computing and AI based technologies.
PEO2	**Breadth**	Apply AI principles, methodologies, algorithms, and tools, to effectively **design**, develop, and implement AI-driven solutions.
PEO3	**Professionalism**	Excel as professionalism with the necessary **soft skills** to work collaboratively in interdisciplinary teams.
PEO4	**Learning Environment**	Aim for **continuing education** and **entrepreneurship** in emerging areas of computing and Artificial Intelligence.

List of Program 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.
PO 12	**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.

Program Specific Objectives (PSO)

PSO1: Demonstrate proficiency in essential concepts of computer science and programming solutions.

PSO2: Formulate robust software design, execution, and testing strategies employing a software paradigms and Artificial Intelligence knowledge to solve real word problems.

PSO3: Adapt and exhibit expertise in evolving areas of computer science, engineering and technology.

Course Name Nomenclature as per NEP (For FY and SY)

BSC: Basic Science Course	MDOE: Multi-Disciplinary Open Elective
ESC: Engineering Science Course	CC: Co-curricular Course
PCC: Program Core Course	HSSM: Humanities Social Science and Management
PEC: Program Elective Course	IKS: Indian Knowledge System
ELC: Experiential Learning Course	FP: Field Project
MD: Multi Disciplinary	INT: Internship

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 and Engineering – Artificial Intelligence	CI
7	Assessment	Laboratory Continuous Assessment	CA
8	Assessment	Mid Semester Assessment	MSA
9	Assessment	End Semester Assessment	ESE
10	Assessment	Home Assignment	HA
11	Assessment	Course Project	CP
12	Assessment	Group Discussion	GD
13	Assessment	PowerPoint Presentation	PPT
14	Assessment	Class Test –1	CT1
15	Assessment	Class Test –2	CT2
16	Assessment	Mid Semester Examination	MSE
17	Assessment	End Semester Examination	ESE
18	Assessment	Written Examination	WRT
19	Assessment	Multiple Choice Questions	MCQ
20	Assessment	Laboratory	LAB

Title: Course Structure **FF No. 653**

Branch: Computer Science & Engineering (Artificial Intelligence) **Year:** T.Y. **A.Y.:** 2026-27 **Module: V**

Sub No.	Sub Code	Subject Name	Teaching Scheme (Hrs/Week)			Examination Scheme										Total	Credits
			Th	Lab	Tut	CVV	CP	LAB	GD/PPT	HA	MSE(R)	ESE (R)	ESE TH(W)	ESE TH(O)	PRACT + CVV		
S1	CI3001	Deep Learning	3	2	-								40		60	100	4
S2	CI3202	Operating System	2	2		20	30	10					40			100	3
S3	CI3003D	MLOPS	3	2		20	40		20	20						100	4
S3	CI3203B	Distributed and federated Learning	3	2		20	40		20	20						100	4
S3	CI3203A	Ethical and Responsible AI	3	2		20	40		20	20						100	4
S3	CI3203C	Information Security	3	2		20	40		20	20						100	4

S4	CI3004	Coursera (MOOC Course)	-	-		-		-	-						-	100	3
S5	CI3004	Open Elective I (Coursera)	-	-		-		-	-						-	100	4
S6	MDM3	Automata Theory	2	-	1		30		20	10			40			100	3
S6	MDM3	Design Thinking and Product Lifecycle Management (PLM)	2	-	1		30		20	10			40			100	3
S7	VESC-DT	Design Thinking – 5	-	-	-	-		-	-						-	100	1
S8	ERI	Engineering Research and Innovation									30	70				100	2
S9	AC*	Audit Courses*	-	-	-	-		-	-						-	-	0
		Total															**24**

Coursera Courses*

Subject Code	Subject Name
MD3106	Microsoft Power BI Data Analyst
MD3112	Google Cybersecurity
MD3113	Google Data Analytics
MD3118	IBM Data Science
MD3121	IBM DevOps and Software Engineering
MD3141	AWS Cloud Technology Consultant

TY Open Elective Bucket

Subject Code	Subject Name
MD3201	Bucket 1: FinTech
MD3202	Bucket 2: Digital Marketing
MD3203	Bucket 3: Entrepreneurship
MD3203	Bucket 4: Management

Title: Course Structure **FF No. 653**

Branch: Computer Science & Engineering (AI) **Year: B.Tech** **A.Y.:** 2026-27 **Module: VII (Course Work-Course Work)**

Subject No.	Subject Code	Subject Name	Teaching Scheme (Hrs/Week)			Examination Scheme								Credits
			Th	Lab	Tut.	CA		MSA		ESA			Total	
						HA (%)	LAB (%)	MSE (%)	PPT (%)	ESE (%)	GD (%)	Viva (%)		
OE1	Linked in*	Linked in *	2							100			100	2
OE2	CI4002	Swayam	2	-	-	10	-	30	-	30	-	30	100	2
OE2	CI4001	Generative AI	2	-	-	10	-	30	-	30	-	30	100	2
Major Project	CI4008	Major Project	0	20	-	-	-	30	-	70	-	-	100	9
-	CI4007	Design Thinking-7	-	-	-	-	-	-	-	100	-	-	100	1
		Total	**4**	**20**	-	**20**	-	**90**	-	**330**	-	**60**	**500**	**16**

Linked in Courses*

Subject Code	Subject Name	Subject Code	Subject Name
MD4274	Large Language Models Skill Development	MD4282	Natural Language Processing Skill Development
MD4275	Mastering Microsoft Power BI	MD4283	Prompt Engineering Skills
MD4276	Generative AI Skills for Developers	MD4284	Essentials in Generative AI
MD4277	Career in Data Analysis	MD4285	Python in Finance
MD4278	Concepts of Data Visualization and Storytelling	MD4286	Understanding Quantum Computing
MD4279	AWS Certified Solutions Architect	MD4287	Foundational Maths for Machine Learning
MD4280	IT Security Specialist	MD4281	Technical Program Management

Title: Course Structure **FF No. 653**

Branch: Computer Science & Engineering (AI) **Year: B.Tech** **A.Y.:** 2026-27 **Module: VIII (Course Work-Course Work)**

Subject No.	Subject Code	Subject Name	Teaching Scheme (Hrs/Week)			Examination Scheme								Credits
			Th	Lab	Tut.	CA		MSA		ESA			Total	
						HA (%)	LAB (%)	MSE (%)	PPT (%)	ESE (%)	GD (%)	Viva (%)		
OE1		LinkedIn Learning*	1							100			100	2
OE2	CI4014	Swayam Course Reinforcement Learning	1	-	-	10	-	30	-	30	-	30	100	2
OE3	CI4005	Natural Language Processing	2	-	-	10	-	30	-	30	-	30	100	2
Major Project	CI4016	Major Project	0	20	-	-	-	30	-	70	-	-	100	10
		Total	**4**	**20**	-	**20**	**-**	**90**	**-**	**230**	**-**	**60**	**400**	**16**

LinkedIn Learning Courses*

Subject Code	Subject Name	Subject Code	Subject Name
MD4274	Large Language Models Skill Development	MD4282	Natural Language Processing Skill Development
MD4275	Mastering Microsoft Power BI	MD4283	Prompt Engineering Skills
MD4276	Generative AI Skills for Developers	MD4284	Essentials in Generative AI
MD4277	Career in Data Analysis	MD4285	Python in Finance
MD4278	Concepts of Data Visualization and Storytelling	MD4286	Understanding Quantum Computing
MD4279	AWS Certified Solutions Architect	MD4287	Foundational Maths for Machine Learning
MD4280	IT Security Specialist	MD4281	Technical Program Management

Title: Course Structure **FF No. 653**

Branch: Computer Science & Engineering (AI) **Year: B.Tech** **A.Y.:** 2026-27 **Module: VII (Internship - Internship)**

Subject No.	Subject Code	Subject Name	Teaching Scheme (Hrs/Week)			Examination Scheme								Credits
			Th	Lab	Tut.	CA		MSA		ESA			Total	
						HA (%)	LAB (%)	MSE (%)	PPT (%)	ESE (%)	GD (%)	Viva (%)		
S1	CI4010	Industry Internship	-	32	-	-	-	30	-	70	-	-	100	16
S1	CI4011	Project Internship	-	32	-	-	-	30	-	70	-	-	100	16
S1	CI4012	Research Internship	-	32	-	-	-	30	-	70	-	-	100	16
S1	CI4013	Global Internship	-	32	-	-	-	30	-	70	-	-	100	16
S2	CI4007	Design Thinking-7	-	-	-	-	-	-	-	100	-	-	100	1
		Total		**32**	-	-	-	**30**	-	**170**	-		**200**	**16**

Title: Course Structure **FF No. 653**

Branch: Computer Science & Engineering (AI) **Year: B. Tech** **A.Y.:** 2026-27 **Module: VIII (Internship - Internship)**

Subject No.	Subject Code	Subject Name	Teaching Scheme (Hrs/Week)			Examination Scheme								Credits
			Th	Lab	Tut.	CA		MSA		ESA			Total	
						HA (%)	LAB (%)	MSE (%)	PPT (%)	ESE (%)	GD (%)	Viva (%)		
S1	CI4017	Industry Internship	-	32	-	-	-	30	-	70	-	-	100	16
S1	CI4019	Project Internship	-	32	-	-	-	30	-	70	-	-	100	16
S1	CI4018	Research Internship	-	32	-	-	-	30	-	70	-	-	100	16
S1	CI4020	Global Internship	-	32	-	-	-	30	-	70	-	-	100	16
		Total		**32**	-	-	-	**30**	-	**170**	-		**200**	**16**

Title: Course Structure **FF No. 653**

Branch: Computer Science & Engineering (AI) **Year: B.Tech** **A.Y.:** 2026-27 **Module: VII (Course Work-Internship)**

Subject No.	Subject Code	Subject Name	Teaching Scheme (Hrs/Week)			Examination Scheme								Credits
			Th	Lab	Tut.	CA		MSA		ESA			Total	
						HA (%)	LAB (%)	MSE (%)	PPT (%)	ESE (%)	GD (%)	Viva (%)		
OE1	Linked in*	Linked in *								100			100	4
OE2	CI4002	Swayam	2	-	-	10	-	30	-	30	-	30	100	2
OE2	CI4001	Generative AI	2	-	-	10	-	30	-	30	-	30	100	2
Major Project	CI4008	Major Project	0	20	-	-	-	30	-	70	-	-	100	9
-	CI4007	Design Thinking-7	-	-	-	-	-	-	-	100	-	-	100	1
		Total	**4**	**20**	-	**20**	-	**90**	-	**330**	-	**60**	**500**	**16**

Title: Course Structure **FF No. 653**

Branch: Computer Science & Engineering (AI) **Year: B.Tech** **A.Y.:** 2026-27 **Module: VIII (Course Work- Internship)**

Subject No.	Subject Code	Subject Name	Teaching Scheme (Hrs/Week)			Examination Scheme								Credits
			Th	Lab	Tut.	CA		MSA		ESA			Total	
						HA (%)	LAB (%)	MSE (%)	PPT (%)	ESE (%)	GD (%)	Viva (%)		
S1	CI4017	Industry Internship	-	32	-	-	-	30	-	70	-	-	100	16
S1	CI4019	Project Internship	-	32	-	-	-	30	-	70	-	-	100	16
S1	CI4018	Research Internship	-	32	-	-	-	30	-	70	-	-	100	16
S1	CI4020	Global Internship	-	32	-	-	-	30	-	70	-	-	100	16
		Total		**32**	-	-	-	**30**	-	**170**	-		**200**	**16**

Title: Course Structure **FF No. 653**

Branch: Computer Science & Engineering (AI) **Year: B.Tech** **A.Y.:** 2026-27 **Module: VII (Internship-Course Work)**

Subject No.	Subject Code	Subject Name	Teaching Scheme (Hrs/Week)			Examination Scheme								Credits
			Th	Lab	Tut.	CA		MSA		ESA			Total	
						HA (%)	LAB (%)	MSE (%)	PPT (%)	ESE (%)	GD (%)	Viva (%)		
S1	CI4010	Industry Internship	-	32	-	-	-	30	-	70	-	-	100	15
S1	CI4011	Project Internship	-	32	-	-	-	30	-	70	-	-	100	15
S1	CI4012	Research Internship	-	32	-	-	-	30	-	70	-	-	100	15
S1	CI4013	Global Internship	-	32	-	-	-	30	-	70	-	-	100	15
S2	CI4007	Design Thinking-7	-	-	-	-	-	-	-	100	-	-	100	1
		Total		**32**	-	-	-	**30**	-	**170**	-		**200**	**16**

Title: Course Structure **FF No. 653**

Branch: Computer Science & Engineering (AI) **Year:** B.Tech **A.Y.:** 2026-27 **Module: VIII (Internship-Course Work)**

Subject No.	Subject Code	Subject Name	Teaching Scheme (Hrs/Week)			Examination Scheme								Credits
			Th	Lab	Tut.	CA		MSA		ESA			Total	
						HA (%)	LAB (%)	MSE (%)	PPT (%)	ESE (%)	GD (%)	Viva (%)		
OE1		LinkedIn Learning*	1							100			100	2
OE2	CI4014	Swayam Course Reinforcement Learning	1	-	-	10	-	30	-	30	-	30	100	2
OE3	CI4005	Natural Language Processing	2	-	-	10	-	30	-	30	-	30	100	2
Major Project	CI4016	Major Project	0	20	-	-	-	30	-	70	-	-	100	10
		Total	**4**	**20**	-	**20**	**-**	**90**	**-**	**230**	**-**	**60**	**400**	**16**

LinkedIn Learning Courses*

Subject Code	Subject Name	Subject Code	Subject Name
MD4274	Large Language Models Skill Development	MD4282	Natural Language Processing Skill Development
MD4275	Mastering Microsoft Power BI	MD4283	Prompt Engineering Skills
MD4276	Generative AI Skills for Developers	MD4284	Essentials in Generative AI
MD4277	Career in Data Analysis	MD4285	Python in Finance
MD4278	Concepts of Data Visualization and Storytelling	MD4286	Understanding Quantum Computing
MD4279	AWS Certified Solutions Architect	MD4287	Foundational Maths for Machine Learning
MD4280	IT Security Specialist	MD4281	Technical Program Management

T.Y. B. Tech.

Computer Science & Engineering (Artificial Intelligence)

AY 2026-27

Syllabus Template **FF No.: 654**

CI3201: Deep Learning

Credits: 4 **Teaching Scheme**

Theory:3 Hours/Week

Lab: 2 Hours/Week

Course Prerequisites: Machine Learning, Python Programming, Linear Algebra, Probability and Statistics, Full Stack Development

Course Objectives:

1. To introduce the fundamental concepts, architectures, and tools used in deep learning.
2. To develop an understanding of neural networks, activation functions, and learning algorithms.
3. To study deep learning architectures such as CNN, RNN, and GRU for processing various types of data.
4. To explore convolutional neural network architectures and techniques for image representation and visual recognition tasks.
5. To learn modern deep learning techniques including transfer learning, autoencoders, transformers, and optimization methods.
6. To apply deep learning models for solving real-world problems in image, text, audio, video, and time-series domains.

Course Relevance:

This course focuses on the design, training, and optimization of deep neural network architectures for learning complex patterns from large-scale data. It equips students with techniques such as CNNs, RNNs, LSTMs, Transformers, and transfer learning for solving advanced AI problems. The course forms the foundation for developing state-of-the-art solutions in computer vision, natural language processing, speech analytics, and generative AI applications.

Section 1: Topics/Contents

Unit-I Fundamental of Deep Learning **06 Hours**

Introduction to Artificial Intelligence, Machine Learning, and Deep Learning, Limitations of machine learning, Advantage and challenges of deep learning, Evolution and applications of **Deep** Learning, Introduction to TensorFlow, Keras, and PyTorch, GPU and cloud-based platforms (Google Colab), Overfitting and Underfitting, Regularization Techniques, Batch Processing and Epochs, Data Preparation for Deep Learning: Dataset collection and preprocessing, Data normalization and scaling, Training, validation, and testing datasets.

Unit-II Perceptron and Neural Network Architecture **06 Hours**

Biological neuron vs artificial neuron, Perceptron model and multilayer perceptron, Weight and Bias, Activation functions: Sigmoid, Tanh, ReLU, Softmax, Loss functions and optimization basics, Notation, Calculation of Trainable Parameters, Backpropagation and Forward propagation, XOR Problem and Limitations of Perceptron.

Unit-III RNN, LSTM and GRU Architectures 09 Hours

Introduction to Sequential Data, Need for Recurrent Neural Networks, Applications of RNN in Real-world Problems, Architecture of Recurrent Neural Network, Difference Between Feed Forward Network and RNN, Layers in RNN, Mathematical Representation of RNN, Forward Propagation, Backpropagation Through Time (BPTT), Training Process, Vanishing Gradient Problem, Exploding Gradient Problem, Limitations of Traditional RNN. Introduction to Long Short-Term Memory (LSTM), Architecture, Gates in LSTM, Mathematical Equations of LSTM Gates, Calculation of Trainable Parameters in RNN and LSTM, Applications of LSTM in NLP and Time Series Forecasting, Gated Recurrent Unit (GRU), Need for GRU, Architecture, Limitations of Traditional RNN, Architecture of GRU, Components of GRU, Comparison Between RNN, LSTM, and GRU

Section2: Topics/Contents

Unit-IV Convolutional Neural Networks (CNN) 06 Hours

Introduction to CNN: Need for CNN in image processing, Convolution operation and feature extraction, CNN Architecture: Convolution layer, Filters, Kernels, Stride, and Padding.
Activation Functions in CNN, Pooling Layers (Max Pooling, Average Pooling), Fully Connected Layers, Batch Normalization and Dropout, CNN Hyperparameters. Popular CNN Models: AlexNet, VGG, ResNet, EfficientNet, Applications of CNN

Unit-V Advanced Deep Learning Architectures 07 Hours

Limitations of Traditional Neural Networks, Transfer Learning: Pre-trained models and fine-tuning, Autoencoders and Generative Models: Autoencoders and dimensionality reduction, Bidirectional Encoder Representations (BERT), Generative Adversarial Networks (GANs), Graph Neural Networks (GNN), Vision Transformers (ViT)

Unit-VI Applications of Deep Learning 08 Hours

Deep Learning for Image Processing: Image segmentation and enhancement, Disease prediction using medical images Deep Learning for Text Processing: Natural Language Processing (NLP), Chatbots, Text classification and summarization, Deep Learning for Audio and Video Processing: Speech-to-text systems, Audio classification, Video analysis Deep Learning for Numerical and Sensor Data: Time-series prediction, financial forecasting, IoT and smart sensor applications

List of Practical's:

1. Install and configure TensorFlow/Keras in Google Colab. Perform data preprocessing, normalization, train-test splitting, and data visualization on a sample dataset.
2. Design and implement a Multilayer Perceptron (MLP) for classification of the Iris or Wine dataset, and evaluate its performance using accuracy and a confusion matrix.
3. Implement forward propagation and backpropagation using TensorFlow/Keras. Analyze the effect of different learning rates and the number of epochs on model performance.
4. Develop an LSTM-based model for time-series forecasting using stock price, weather, or sales datasets.
5. Implement and compare RNN, LSTM, and GRU models for sequence classification, and analyze their performance using appropriate evaluation metrics.
6. Design and implement a Convolutional Neural Network (CNN) for image classification using the Tomato or Soybean disease dataset.
7. Implement transfer learning using pre-trained AlexNet, VGG16, ResNet50, and EfficientNetB0 models for image classification, and compare their performance.
8. Implement a pre-trained BERT model for sentiment analysis or text classification on a sample dataset.
9. Perform image classification using a pre-trained Vision Transformer (ViT) model and compare its performance with a CNN model.
10. Design and implement a deep learning application for audio classification that incorporates Speech-to-Text (STT) and Text-to-Speech (TTS) using an appropriate deep learning architecture, and evaluate its performance using suitable metrics.

List of Course Project areas:

1. Plant Disease Detection and Severity Assessment System: Develop a deep learning model to identify plant diseases and estimate disease severity from leaf images.
2. AI-Based Brain Tumor Classification Using MRI Images: Design a transfer learning-based system to classify brain MRI scans into different tumor categories.
3. Fake News Detection Using Deep Learning and NLP: Develop an intelligent system that classifies news articles as genuine or fake using text analysis.
4. Deepfake Image and Video Detection Framework: Build a deep learning model to detect manipulated images and videos using visual artifacts.
5. Speech Emotion Recognition System:Design a model to recognize human emotions from speech recordings using deep learning techniques.
6. Image Caption Generation Using CNN and LSTM: Develop a system that automatically generates meaningful captions for input images.
7. Intelligent Traffic Density Estimation and Signal Optimization: Create a deep learning-based traffic analysis system to estimate vehicle density and optimize signal timing.
8. Human Activity Recognition Using Wearable Sensor Data: Develop an LSTM-based system to identify human activities from smartphone or IoT sensor data.

9. Sign Language Recognition Using Deep Learning: Design a vision-based system that converts sign language gestures into text or speech.
10. Multilingual Chatbot Using Transformer Models: Develop an AI chatbot capable of understanding and responding in multiple languages.
11. Audio-Based Bird Species Classification System: Build a deep learning model to identify bird species from audio recordings of bird calls.
12. Network Intrusion Detection Using Deep Neural Networks: Develop a cybersecurity system that detects malicious network activities using deep learning.
13. Automatic Medical Report Classification and Summarization: Design an NLP-based system to categorize and summarize medical reports automatically.
14. Financial Fraud Detection Using Deep Learning: Develop a predictive model to identify fraudulent financial transactions from historical data.
15. AI-Based Resume Screening and Candidate Ranking System: Create a deep learning system that analyzes resumes and ranks candidates based on job requirements.
16. Image Forgery Detection Using CNN and Attention Mechanisms: Develop a model to detect tampered or manipulated digital images.
17. Automatic Waste Classification for Smart Cities: Design a deep learning-based waste segregation system for recyclable and non-recyclable materials.
18. Cervical Cancer Cell Classification Using Transfer Learning: Design a deep learning framework for automated classification of cervical cancer cell images.

Assessment Scheme:
Course Project: End Semester Examination: 100 Marks converted to 30 Marks
Laboratory and Comprehensive Viva Voce: End Semester Examination: 100 Marks converted to 30 Marks
Theory: End Semester Examination: 60 Marks converted to 40 Marks
Assessment scheme covers following aspects of Modified Blooms Taxonomy:
L2 Understanding, L3 Apply, L3 Design, L3 Apply, L4 Analyze and L5 Evaluate

Text Books: ***(As per IEEE format)***

1. N. Buduma, N. Buduma, and J. Papa, Fundamentals of Deep Learning, 2nd ed. Sebastopol, CA, USA: O'Reilly Media, 2022.
2. C. C. Aggarwal, Neural Networks and Deep Learning. Cham, Switzerland: Springer, 2018.
3. F. Chollet, Deep Learning with Python. Shelter Island, NY, USA: Manning Publications, 2018.

Reference Books: ***(As per IEEE format)***

1. O. Campesato, Artificial Intelligence, Machine Learning, and Deep Learning. Dulles, VA, USA: Mercury Learning and Information LLC, 2020.

2. S. Weidman, Deep Learning from Scratch: Building with Python from First Principles, 1st ed. Sebastopol, CA, USA: O'Reilly Media, 2019.
3. Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, 2nd ed. Sebastopol, CA, USA: O'Reilly Media, 2019.
4. Zhang, Z. C. Lipton, M. Li, and A. J. Smola, Dive into Deep Learning. Cambridge, U.K.: Cambridge University Press, 2023.

MOOCs Links and additional reading material:

1. https://nptel.ac.in/courses/106106184 [Deep Learning - IIT Ropar]
2. https://www.coursera.org/specializations/deep-learning [Deep Learning Specialization]
3. https://www.udemy.com/course/deeplearning/?couponCode=MT260629G2 [Deep Learning A-Z [2026]: DL, AI in Python & AWS + LLM Prize]
4. https://www.edx.org/learn/deep-learning [Learn deep learning]
5. https://course.fast.ai/ [Practical Deep Learning]

Course Outcomes:

After completion of the course, student will be able to

1. Explain the fundamentals of deep learning, data preparation techniques, and deep learning frameworks for building intelligent systems.
2. Apply perceptron and neural network architectures, activation functions, and backpropagation algorithms for solving classification problems.
3. Examine sequential data using RNN, LSTM, and GRU architectures for prediction and classification tasks.
4. Analyze the architecture, components, and performance of convolutional neural network models for image processing and computer vision applications.
5. Apply advanced deep learning architectures and optimization techniques to solve complex learning problems across diverse application domains.
6. Implement deep learning solutions for real-world applications involving image, text, audio, video, and numerical data.

CO-PO Map:

	Program Outcomes (PO)												PSO		
CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	2	1	-	-	-	-	-	-	-	-	-	2	2	-
CO2	3	2	2	-	2	-	-	-	-	-	-	2	3	3	2
CO3	3	3	2	2	2	-	-	-	-	-	-	2	3	3	2
CO4	3	3	2	2	2	-	-	-	-	-	-	2	3	3	2
CO5	3	3	2	2	2	2	2	-	-	-	-	2	3	3	2
CO6	3	3	2	2	2	2	2	2	2	2	2	2	3	3	2
Average	2.83	2.67	1.83	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.83	2.83	2.00

Attainment Levels: 2,3,4,4,3,3

CO Attainment levels:
Weights for attainment levels: L1 - Easy-0.75 L2 - Comfortatble-0.7 L3 – Medium – 0.65 L4 – Somewhat difficult – 0.6 L5 – Difficult – 0.55

CO1 – L2, CO2 – L3, CO3 – L4, CO4 – L4, CO5 – L3 and CO6 – L3

Future Course Mapping: Advanced Deep Learning, Natural Language Processing, Computer Vision, Generative AI, Reinforcement Learning

Job Mapping: Deep Learning Engineer, AI Engineer, Computer Vision Engineer, Generative AI Engineer, Robotics AI Engineering.

Syllabus Template **FF No.: 654**

CI3202: Operating Systems

Credits: 3

Teaching Scheme

Theory:2 Hours/Week

Lab: 2 Hours/Week

Course Prerequisites: C/C++ Programming, Data Structures, Computer Organization and Architecture

Course Objectives:

1. To learn the operating system's core structure, functions, and design principles.
2. To examine the internals of processes and analyze different Scheduling strategies.
3. To examine process synchronization and inter communications.
4. To study and analyze various memory management schemes.
5. To have exposure about IO and Disk management approaches.
6. To Understand Android architecture and its relationship with the Linux kernel.

Course Relevance:

The Operating System (Operating System) is a core subject that provides fundamental knowledge about how computer systems manage hardware and software resources efficiently. It explains key concepts such as process scheduling, memory management, file systems, and inter-process communication. This subject is essential for understanding system-level programming and the

internal working of modern computing devices. It also forms the base for advanced areas like cloud computing, distributed systems, and cybersecurity.

Section 1: Topics/Contents

Unit-I Operating Systems Overview **04 Hours**

Introduction to Operating Systems: Operating System Objectives and Functions, The Evolution of Operating Systems, Developments Leading to Modern Operating Systems. Operating System Kernel Fundamentals: Operating system structures, Kernel Functions and Responsibilities, Monolithic, Microkernel, Hybrid, and Modular Kernels. Linux OS: Shell commands and Shell Programming

Unit-2: Process Concepts and CPU Scheduling **04 Hours**

Process Concepts: process states, process control (process creation, program execution, process Termination), Process Control Block in Linux, Threads: processes and threads, threads creation and execution in Linux, types of threads. CPU Scheduling: Scheduling Criteria, Types of scheduling: Preemptive, Non-preemptive, Scheduling algorithms: FCFS, SJF, RR, Priority, Multilevel Feedback Queue Scheduling, Traditional UNIX & Modern UNIX Scheduling, Unix Process Control.

Unit 3: Process Synchronization and Deadlock **06 Hours**

Process Synchronization: Principles of Concurrency, Critical section problem, Peterson's solution, Semaphores, classical synchronization problems. Deadlocks: System model, deadlock characterization, Methods for Handling Deadlocks, deadlock prevention, Deadlock Avoidance, Deadlock Detection and Recovery, Process Synchronization in Unix/Linux.

Section 2: Topics/Contents

Unit 4: Memory Management **06 Hours**

Main Memory: Memory Management requirements, Memory partitioning: Fixed and Variable Partitioning, Memory Allocation: Allocation Strategies (First Fit, Best Fit, and Worst Fit), Fragmentation, Swapping, Paging, and Segmentation. Virtual Memory: Concepts, Caching and Virtual Memory, Demand Paging, Page Replacement Policies (FIFO, LRU, Optimal, Other Strategies), Thrashing, memory allocation strategies using Buddy System and Slab Allocation.

Unit 5: File and I/O Management **04 Hours**

I/O Management: I/O Devices, Organization of I/O function, I/O Buffering, Disk Scheduling-Disk Scheduling policies like FIFO, LIFO, STTF, SCAN, C-SCAN. File Management: Concept, Access methods, Directory Structure, Protection, File System implementation, Directory Implementation, Allocation methods, Free Space management, Linux and Windows File Systems.

Unit 6: Operating System Kernel Internals and Android OS **04 Hours**

Linux Kernel: Linux kernel Architecture, User Space and Kernel Space, Linux Boot Process, System call interface. Android Operating Systems: Overview of Mobile Operating Systems, Evolution of Android, Android Open-Source Project (AOSP). Android Architecture: Linux Kernel Layer, Hardware Abstraction Layer (HAL), Native Libraries, Android Runtime (ART), Application Framework, Applications Layer

List of Practical's:

1. Shell Programming: Implement student database using shell and awk programming to perform following operations:
Create database
View database
Insert a record
Delete a record
Modify a record
Display result of a particular student
Exit

2.Process Control: Process Creation Using fork ()
Zombie and Orphan Processes
Generate process tree with multiple levels
Program Execution Using exec ()

3.CPU Scheduling: Write a menu driven program to simulate the behavior of following CPU scheduling algorithms:
FCFS, SJF (Preemptive), Priority (Non-Preemptive) and Round Robin (Preemptive).

4.Multithreaded programming using pthread
Parallel Sum of Array Elements
Matrix Addition Using Threads
Matrix Multiplication Using Pthreads
Parallel File Processing:
Count words, lines, and characters using multiple threads. Merge results.

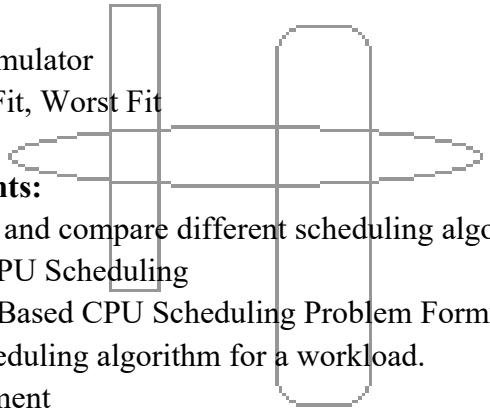
5.Inter Process Communication
Anonymous Pipe Communication
Bidirectional Communication Using Two Pipes
File Transfer Using Pipes (Transfer file contents through pipes.)
Named Pipes (FIFO)
Client-Server Communication Using FIFO
FIFO-Based Chat Application

Message Queues
Producer-Consumer Using Message Queues
Exchange messages using System V Message Queues.
Shared Memory
Share data between processes using shared memory.
Perform computations using shared memory.
(Client writes operands. Server performs arithmetic operations)
Shared Memory File Copy (Process 1 reads file. Process 2 writes data to another file.)

6.Deadlock handling: Write a program to simulate Banker's algorithm: Safety algorithm Resource Request algorithm

7.Paging and Virtual Memory
Write a program to simulate following Page replacement algorithms:
FIFO
LRU
Optimal

8.Memory Management Simulator
Implement: First Fit, Best Fit, Worst Fit



AI enabled OS Assignments:
1.Apply Linear Regression and compare different scheduling algorithms such as: FCFS and SJF and predicted SJF AI for CPU Scheduling
2.Reinforcement Learning Based CPU Scheduling Problem Formulation
3.Recommend the best scheduling algorithm for a workload.
4.Intelligent Page Replacement
5.Memory Usage Prediction
6.AI for Deadlock Management (Predict Deadlock occurrence)

List of Project areas:
Project 1: Multithreaded programming using pthread
Parallel Search Engine
Parallel Log Analyzer
Multithreaded Chat Server

Project 2: Linux Chat Messenger
IPC Used: FIFO, Message Queue
Project 3: Shared Memory Database
Concurrent record access

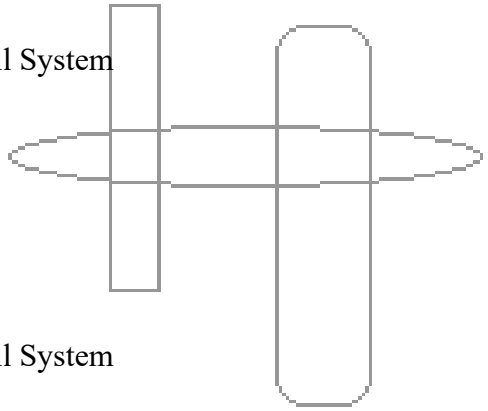
Data synchronization

Project 4: Multi-Process File Processing System
IPC Used: Shared Memory, Pipes
Features: Parallel file processing, Result aggregation

Project 5: Multi-Process File Processing System
IPC Used: Shared Memory, Pipes
Features: Parallel file processing, Result aggregation

Project 6: Library Management System using Bash
Features
Add books
Search books
Issue books
Return books
View inventory

Project 7: Employee Payroll System
Features
Store employee details
Calculate salary
Generate payslips
Maintain payroll records



Project 8: Employee Payroll System
Features
Store employee details
Calculate salary
Generate payslips
Maintain payroll records

Project 9: Attendance Management System
Features
Add student records
Mark attendance
Calculate attendance percentage
Generate attendance reports

Project 10: File Organizer Utility

Features
Automatically organize files into folders:
Images
Documents
Videos
Archives

Project 11: Multi-Threaded File Downloader
Download files using multiple threads.

Project 12: Use file related system calls/Functions and develop student record management system.

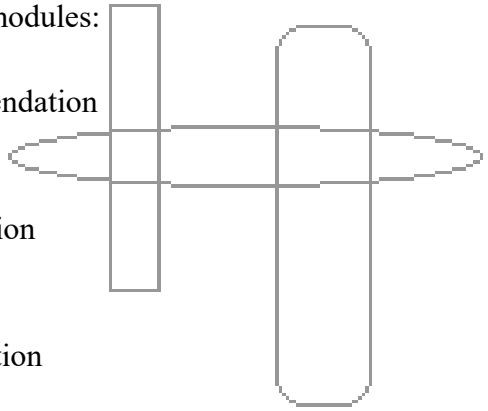
Project 13: Linux Log Analyzer Using Python
Problem Statement: Linux systems continuously generate logs that contain information about:
System activities
User logins
Security events
Application errors
Kernel messages
Manually analyzing thousands of log entries is difficult. The proposed system automates log analysis and generates useful reports.

Project 14: Linux Task Automation System
Features
Menu Driven Interface
Execute Commands Using execl()
Create Child Processes Using fork()
Synchronize Using wait()
Maintain Execution Log
Display Process Information
Commands Supported
ls
pwd
date
who
ps
free
uptime

Project 15: Design and Implementation of a Reader-Writer Synchronization Based File Handling Utility Using Python
Objective: Develop a file utility where:Multiple readers can read a file simultaneously.
Only one writer can modify the file at a time. No reader can access the file while a writer is writing.

Project 16: Develop a file handling utility using posix threads which can perform following operations:
File Compression Utility
Duplicate File Detector
Log File Analyzer
Multi-Threaded Text Search Engine
Student Record Management System
Multi-Threaded Backup Utility
Linux Log Monitoring System
Concurrent File Server

Project 17: Intelligent Operating System Analytics and Resource Management Platform in python which includes following modules:
Process Monitoring
CPU Scheduling Recommendation
Memory Usage Prediction
Deadlock Prediction
Page Replacement Simulation
Log Analysis
Performance Dashboard
AI-Based Resource Allocation



Assessment Scheme:
Laboratory Continuous Assessment: 100 Marks converted to 10 Marks
Course Project: End Semester Examination: 100 Marks converted to 30 Marks
Theory: End Semester Examination Theory (Written): 60 Marks converted to 40 Marks
Comprehensive Viva Voce: End Semester Examination: 100 Marks converted to 20 Marks
Assessment scheme covers following aspects of Modified Blooms Taxonomy:
L2 Understanding, L3 Apply, L3 Design, L3 Apply, L4 Analyze and L5 Evaluate

Text Books: *(As per IEEE format)*

1. Silberschatz, A., Galvin, P.B, and Gagne, G., “Operating System Principles”, Eight Edition, John Wiley & Sons.
2. Stallings William., "Operating Systems", Fourth Edition, Prentice Hall of India,2001

3. Dhamdhere D., "Systems Programming and Operating Systems", McGraw Hill, ISBN 0 -07 - 463579 – 4.
4. Modern Operating Systems -By Andrew S. Tanenbaum (PHI).

Reference Books : *(As per IEEE format)*

1. Unix Concepts And Applications, Sumitabha Das, McGraw Hill India
2. Operating System, 3/e Harvey M. Deitel, David R. Choffnes and Paul J. Deitel, Pearson
3. Stevens, W. R., & Rago, S. A. (2013). Advanced Programming in the UNIX Environment: Advanced Programming in the UNIX Environment. Addison-Wesley.
4. Linux System Programming: Talking Directly to the Kernel and C Library,

MOOCs Links and additional reading material:

https://www.udemy.com/course/system-programming/?couponCode=LETSLEARNNOWPP
https://archive.nptel.ac.in/courses/106/105/106105214/

Course Outcomes: After completion of the course, student will be able to

1. Describe OS basic structures and functions
2. Explain about processes and do the detail analysis of different Scheduling strategies.
3. Demonstrate process synchronization strategies.
4. Compare and contrast various memory management schemes
5. Exhibit the file handling and IO activities at OS level.
6. Analyze Linux kernel components and Android system architecture.

CO-PO Map:

	Program Outcomes (PO)												PSO		
CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	-	-	1	-	-	-	-	-	-	1	3	2	1
CO2	3	3	2	1	1	-	-	-	-	-	-	1	3	2	2
CO3	3	3	3	2	1	-	-	1	1	-	-	1	3	2	3
CO4	3	3	2	2	1	-	-	-	-	-	-	1	3	2	3
CO5	3	2	2	1	2	-	-	-	-	-	-	1	3	3	2
CO6	3	3	2	2	2	-	-	-	-	-	-	2	2	3	3
Average	3.00	2.67	2.25	1.60	1.33	-	-	1.00	1.00	-	-	1.17	2.83	2.33	2.33

Attainment Levels: 2,3,3,3,3, 3

CO Attainment levels: Weights for attainment levels: L1 - Easy-0.75 L2 - omfortatble-0.7
L3 – Medium – 0.65 L4 – Somewhat difficult – 0.6 L5 – Difficult – 0.55

CO1 – L1, CO2 – L3, CO3 – L4, CO4 – L3, CO5 – L4 and CO6 – L4

Future Course Mapping: Cloud Computing, Computer Networks, Cyber Security, System Programming / Linux Administration

Job Mapping: System Software Engineer, DevOps Engineer, Cloud Engineer / Cloud Architect Cyber Security Analyst, Linux System Administrator

Syllabus Template **FF No.: 654**

CI3203D: Machine Learning and Operations

Credits: 4

Teaching Scheme

Theory: 3 Hours/Week

Lab: 2 Hours/Week

Course Prerequisites: Programming Fundamentals and Python, Basics of Machine Learning, Introduction to Cloud and DevOps Concepts

Course Objectives:

1.Understand the basics of MLOps and ML lifecycle.
2.Learn data versioning, experiment tracking, and automation concepts.
3. Build simple ML pipelines using industry tools.
4. Deploy machine learning models using APIs and containers.
5. Understand monitoring, Explainability, and responsible AI concepts.
6. Develop practical skills for production-ready ML systems.

Course Relevance:

Machine Learning Operations (MLOps) is an essential discipline that bridges the gap between machine learning model development and real-world deployment. It combines Machine Learning, DevOps, and Cloud Computing practices to automate, deploy, monitor, and maintain AI systems efficiently. As industries increasingly adopt AI-driven solutions, professionals with MLOps skills are in high demand. This course equips students with the knowledge and practical skills required to build scalable, reliable, and production-ready machine learning systems for modern industry applications.

Section 1: Topics/Contents

Unit-I Introduction to ML Ops and ML Lifecycle 07 Hours

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-II Data, Feature, and Experiment Management 07 Hours

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-III ML Pipeline Automation 07 Hours

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-IV Model Deployment and Serving 07 Hours

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-V Responsible and Explainable AI Systems 07 Hours

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-VI Cloud-based ML Ops and Industry Applications 07 Hours

Introduction to Cloud-based ML Ops , Overview of cloud ML services,
Introduction to AWS SageMaker: Studio, Training Jobs, Model Deployment, and Monitoring

ML Ops 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

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. Train and deploy a Machine Learning model using AWS SageMaker
11. Mini project on complete MLOps workflow

List of Course 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

Assessment Scheme:

Assessment scheme covers following aspects of Modified Blooms Taxonomy:
L2 Understanding, L3 Apply, L3 Design, L3 Apply, L4 Analyze and L5 Evaluate

Comprehensive Viva Voce: End Semester Examination: 100 Marks converted to 20 Marks
Course Project: End Semester Examination: 100 Marks converted to 40 Marks
GD/PPT: End Semester Examination: 100 Marks converted to 20 Marks
HA: 100 Marks converted to 20 Marks

Text Books (Networking): ***(As per IEEE format)***

1.M. Treveil and A. Shukla, Introducing MLOps: How to Scale Machine Learning Projects with Continuous Delivery, Monitoring, and Governance. Sebastopol, CA, USA: O'Reilly Media, 2020.
2.D. Brinkmann, E. Raj, and M. Daoust, MLOps Engineering at Scale. Sebastopol, CA, USA: O'Reilly Media, 2023.

Reference Books (Networking) : ***(As per IEEE format)***

1.C. Osipov, MLOps: Continuous Delivery and Automation Pipelines in Machine Learning. Shelter Island, NY, USA: Manning Publications, 2022.
2.N. Gift and A. Deza, Practical MLOps. Sebastopol, CA, USA: O'Reilly Media, 2021.
3.E. Ameisen, Building Machine Learning Powered Applications. Sebastopol, CA, USA: O'Reilly Media, 2020.

MOOCs Links and additional reading material:

1. Practical MLOps (Coursera - DeepLearning.AI)

Link: https://www.coursera.org/specializations/mlops-machine-learning-duke

2. MLOps Specialization (DeepLearning.AI - Coursera)

Link: https://www.coursera.org/courses

3. NPTEL - DevOps and MLOps (IIT Ropar)
4. End-to-End Machine Learning with MLOps (Google Cloud)

Link:https://docs.cloud.google.com/architecture/mlops-continuous-delivery-and-automation-pipelines-in-machine-learning

5. MLflow: End-to-End Machine Learning Lifecycle (Databricks)

Link: https://docs.databricks.com/gcp/en/mlflow/end-to-end-example

6. Kubernetes for Machine Learning (edX)

Link: https://www.edx.org/learn/kubernetes

Course Outcomes:

Understanding Machine Learning Concepts: Students will be able to -

1. To explain the key differences between software development and ML lifecycles.
2. To apply MLOps tools like MLflow, DVC in workflows.
3. To manage data, features, and experiments effectively.
4. To automate ML pipelines using CI/CD integration.
5. To deploy and monitor ML models with appropriate tools.

To implement responsible AI practices, ensuring fairness, privacy, and compliance.

CO-PO Map:

	Program Outcomes (PO)												PSO		
CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	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	3
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	1.5

Attainment Levels: 2,3,4,3,4, 4

CO Attainment levels:

Weights for attainment levels: L1 - Easy-0.75 L2 - Comfortatble-0.7 L3 – Medium – 0.65
L4 – Somewhat difficult – 0.6 L5 – Difficult – 0.55

CO1 – L1, CO2 – L3, CO3 – L4, CO4 – L3, CO5 – L4 and CO6 – L4

Future Course Mapping:
Advanced Machine Learning, Deep Learning, Cloud Computing, Generative AI and LLMs

Job Mapping:
Machine Learning Engineer, MLOps Engineer, AI Engineer, Data Scientist, Cloud AI Engineer

Syllabus Template **FF No.: 654**

CI3203 B: Distributed and Federated Learning

Credits: 4 **Teaching Scheme Theory: 2 Hours/Week**
Lab: 2 Hours/Week

Course Prerequisites: Machine Learning

Course Objectives:

1. To provide students with an understanding of Federated Learning and its key concepts.
2. To equip students with the skill set of Federated and Distributed Machine Learning Models to implement it in real-world applications.
3. To introduce students to the Optimization Techniques in Federated Learning
4. To provide students with an understanding of Privacy, Security and applications in recent advancements in Federated Learning
5. To Understand the fundamentals of federated learning architectures
6. To develop understanding of algorithms and communication strategies in distributed learning.

Course Relevance:
This course provides an in-depth understanding of distributed machine learning algorithms, parallel model training, and federated optimization techniques for large-scale AI systems. Students learn data-parallel and model-parallel training strategies, distributed gradient descent,

parameter server architectures, and communication-efficient learning mechanisms. The course covers federated learning frameworks, privacy-preserving techniques, secure aggregation, differential privacy, and decentralized model synchronization. It enables students to design scalable, efficient, and secure AI solutions for cloud, edge, and Internet of Things (IoT) environments.

Section 1: Topics/Contents

Unit-I Fundamentals of Distributed and Federated Learning 6 Hours

Introduction to Distributed Computing and Machine Learning, Evolution from Centralized AI to Distributed and Federated Learning, Concepts and Terminology in Distributed Federated Learning,

Overview of Federated Learning: Definition, History, and Applications, Concepts and Terminology, Federated Learning Architecture, Machine Learning perspective, Security & Privacy, Federated Learning vs Centralized Learning: Comparison and Contrast

Case Study: FL for Google Assistant 'Hey Google' or Alexa

Unit-II Federated and Distributed Machine Learning Paradigms 6 Hours

Distributed Machine Learning Paradigms, Client–Server and Peer-to-Peer Architectures

Parameter Server Model, Decentralized Federated Learning, Federated Averaging (FedAvg) Algorithm. Tree based models in FL Semantic vectorization: Text and Graph based models, Personalization, Personalized and Robust FL with Fed+,Map Reduce algorithm Horizontal FL: Definition, Architecture, Federated Averaging algorithm, Improvement of FedAvg algorithm

Vertical FL: Definition, Architecture, VFL algorithms: Secure Federated Linear Regression, Secure Federated Tree Boosting, Federated Transfer Learning: Framework, Homomorphic encryption, FTL training process, FTL prediction process, Security analysis, Secret sharing based FTL.

Case study: Decentralized Federated Forest.

Unit-III Optimization Techniques 6 Hours

Federated learning with Non-HD Data: Heterogeneity, Stratification and Local

Updated rules. Advanced optimization techniques: Adaptive Learning rate, Momentum and Weight decay, Meta learning for FL: Model Selection and hyper parameter tuning.

Case Study: FL for Autonomous Vehicles

Section2: Topics/Contents

Unit-IV Privacy, Security and Applications in Federated learning 6 Hours

Protecting against Data Leakage in FL, Private parameter aggregation for FL,Data leakage in FL, Dealing with Byzantine threats to Neural Networks. Case Study: Federated Learning for Collaborative Financial Crimes Detection Federated Learning for Healthcare: Privacy preserving Diagnosis and Treatment planning Finance: Fraud detection and Risk Assessment IoT: Edge Computing and Distributed Sensing.

Case study: Case study for COVID-19

Unit-V Systems, Frameworks and Communication Strategies 6 Hours

System architectures-Centralized Architecture, Distributed Architecture, Federated Architecture Communication Strategies-Client–Server Communication, Distributed Communication Models Federated Learning Frameworks and Tools-TensorFlow Federated (TFF), Flower Framework, PySyft. NVIDIA FLARE, OpenFL

Unit-VI Emerging Trends in Distributed Federated Learning and Industrial application 4 Hours

Emerging Trends - Federated AI in 6G Networks, Edge AI and TinyML for Resource-Constrained Devices, Generative AI with Federated Learning, Blockchain-Enabled Federated Systems, Federated Learning for Cybersecurity Applications.

Industrial Application- Smart Transportation, Industry 4.0 and Smart Manufacturing, Healthcare and Smart Hospitals.

List of Practical's:

1.Design and implement a basic simulation of a Federated Learning system in Python where multiple clients train local models on their own data without sharing raw datasets. A central server aggregates the locally trained model parameters using averaging to form a global model, demonstrating the core federated learning workflow.

2.To perform data pre-processing and partitioning for a Federated Learning system using a real dataset such as a student performance dataset (e.g., study hours, attendance, internal marks, final score). The dataset is cleaned by handling missing values and normalization, then split into multiple local datasets and distributed among participating devices or nodes so that each client trains on its own private data without sharing raw information.

3.Implement a Federated Learning framework for a linear regression task where multiple clients collaboratively learn a global prediction model without sharing their raw data. For a house price dataset can be split across clients, each client trains a local linear regression model, and a central server applies Federated Averaging (FedAvg) to combine the local model parameters into a global model while preserving data privacy.

4.Local Model Training and Model Update Transmission to Central Server in Federated Learning

5.Server-Side Model Aggregation Using Weighted Federated Averaging and Global Model Distribution in Federated Learning

6.Implementation of Vertical Federated Learning for Joint Feature-Based Prediction

7.To implement a Federated Learning framework where multiple healthcare institutions collaboratively train a machine learning model for disease diagnosis or treatment recommendation without sharing sensitive patient data. (Use Diabetic patient data)

8.set up a client-server FL simulation where each client trains a local model, sends its parameters to the server, and receives the updated global model to continue training. This demonstrates privacy-preserving collaborative learning.
Example Dataset: House prices or student performance.
9.Implement a Federated Learning system where clients train local models on private data and share only noisy model updates with the server using differential privacy, so the server can build a global model without accessing raw data.
10.Design any basic Industrial application- Smart Transportation, Industry 4.0 and Smart Manufacturing, Healthcare and Smart Hospitals

List of Course Project topics:

1. Healthcare and Medical System
2. Smart Transportation and Autonomous Vehicle
3. Industry 4.0 and Smart Manufacturing
4. Cybersecurity and Privacy
5. Internet of Things (IoT) and Edge Computing
6. Finance and banking

Assessment Scheme:

Comprehensive Viva Voce: End Semester Examination: 100 Marks converted to 20 Marks
Course Project: End Semester Examination: 100 Marks converted to 40 Marks
GD/PPT: End Semester Examination: 100 Marks converted to 20 Marks
HA: 100 Marks converted to 20 Marks

Text Books: ***(As per IEEE format)***

1.Heiko Ludwig and Nathalie Baracaldo, Federated Learning: Comprehensive Overview of Methods and Applications, 1st Edition, Springer Nature Switzerland AG, 2022.
2.Qiang Yang, Yang Liu, Yong Cheng, Yan Kang, Tianjian Chen and Han Yu, Federated Learning, Morgan & Claypool Publishers, 2019.

Reference Books: ***(As per IEEE format)***

1.Kiyoshi Nakayama and George Jeno, Federated Learning with Python, O'Reilly Media, Inc., 2023.
2.Emily Glanz and Nova Fallen, what is Federated Learning? O'Reilly Media, Inc., 2022.
3.Qiang Yang, Yang Liu, Tianjian Chen and Yongxin Tong, Federated Learning, Springer, 2023.
4.Rajkumar Buyya, Anwesha Mukherjee and Sajal K. Das, Federated Learning: Foundations and Applications, Elsevier, 2026.
5.Kai Hwang, Geoffrey Fox and Jack Dongarra, Distributed and Cloud Computing: From Parallel Processing to the Internet of Things, 1st Edition, Morgan Kaufmann, 2011.

6.Yaochu Jin, Hangyu Zhu, Jinjin Xu and Yang Chen, Federated Learning: Fundamentals and Advances, Springer, 2023.

MOOCs Links and additional reading material:

https://research.aimultip1e.com/federated-learning/
https://towardsdatascience.com
https://www.youtube.com/watch?v=I7j5x18igp4
https://www.youtube.com/watch?v=YsFVdDAgkcU
https://www.nature.com/articles/s4l746-021-00431-6
https://www.persistent.com/blogs/privacy-preserving-ai-private-ai-the-rise-of-federated-learning/

Course Outcomes:

The student will be able to –

1. Describe the key concepts and architecture of Federated Learning.
2. Apply different methods and models to develop federated learning systems.
3. Apply optimization techniques in Federated Learning.
4. Construct and scale a simple federated system considering the privacy and security measures.
5. Understand system architectures used in intelligent systems
6. Apply communication strategies for distributed systems

CO-PO Map:

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

CO Attainment levels:

Weights for attainment levels: L1 - Easy-0.75 L2 - Comfortatble-0.7 L3 – Medium – 0.65
L4 – Somewhat difficult – 0.6 L5 – Difficult – 0.55
CO1 – L1, CO2 – L3, CO3 – L2, CO4 – L3, CO5 – L4 and CO6 – L5

Future Course Mapping:

Privacy-Preserving Artificial Intelligence, Distributed Systems, Advanced Data Engineering
AI Security and Trustworthy AI

Job Mapping: Distributed Systems Engineer, Federated Learning Engineer, Cloud AI Engineer
AI Infrastructure Engineer

Syllabus Template **FF No.: 654**

CI3203 A: Ethical and Responsible AI

Credits: 4 **Teaching Scheme Theory: 2 Hours/Week**
Lab: 2 Hours/Week

Course Prerequisites: Programming Fundamentals and Python, Basics of Machine Learning/Deep Learning/Natural Language Processing

Course Objectives:

1. Understand the fundamental concepts, ethical theories, and principles governing the development and deployment of Artificial Intelligence systems
2. Analyze issues related to fairness, bias, transparency, explainability, accountability, and trustworthiness in AI-driven decision-making.
3. Evaluate privacy, security, legal, regulatory, and governance challenges associated with AI technologies in diverse application domains.
4. Apply responsible AI frameworks, ethical guidelines, and risk assessment methodologies to design human-centered and socially beneficial AI solutions
5. Develop the ability to identify, mitigate, and manage ethical risks in AI systems through appropriate technical and organizational practices.
6. Examine emerging trends, challenges, and future directions in Responsible AI, including Generative AI, AI governance, sustainability, and societal impacts.

Course Relevance:

This course focuses on the technical foundations of trustworthy AI, including algorithmic fairness, bias quantification and mitigation, explainable AI (XAI), model interpretability, robustness, and privacy-preserving machine learning. Students learn methods for fairness evaluation, adversarial robustness testing, differential privacy, federated learning, secure AI pipelines, and AI governance mechanisms. The course covers model auditing, accountability frameworks, risk assessment, and compliance with emerging AI regulations and standards. It equips students to design, validate, and deploy AI systems that are reliable, transparent, secure, and aligned with responsible AI principles.

Section 1: Topics/Contents

Unit-I Introduction to Artificial Intelligence and Intelligent Agents **6 Hours**

Introduction, history, and applications of AI; how AI works; AI Lifecycle Governance; Codes of Conduct; Indian AI Policy; AI Ethics in Healthcare and Finance; IP Rights and AI; intelligent agents and environments. Rationality, rational agents, task environments; agent types — Reflex,

Model-Based, Goal-Based, Utility-Based, Learning Agents. Case Study: Comparing AI agents in Healthcare, Finance, Smart Cities, and Education by agent type and rationality.

Unit-II Ethical Decision-Making in Artificial Intelligence 6 Hours

Introduction to AI Ethics; Ethical Theories — Utilitarianism, Deontology, Virtue Ethics; Human-Centered Design. Ethical Decision-Making Models; Responsible Research and Innovation (RRI); the ART of AI — Accountability, Responsibility, Transparency.

Case Study: Autonomous-vehicle dilemmas and real AI-failure incidents analyzed through Utilitarian, Deontological, and Virtue Ethics lenses.

Unit-III Ethical AI and Responsible Innovation 6 Hours

Machine Ethics; Designing Artificial Moral Agents; Levels of Ethical Behavior in Machines; Ethical Status of AI Systems.AI and Human Values; Inclusion and Diversity in AI Development; Superintelligence and Existential Risks. Responsible AI Principles overview — Fairness, Transparency, Accountability, Privacy, Inclusiveness; AI for Social Good. Case Study: Autonomous-vehicle moral-agent design; AI for Social Good initiatives.

Section2: Topics/Contents

Unit-IV Fairness, Explainability, Privacy and Security in AI 6 Hours

Fairness and Bias in AI; Sources of Bias; Algorithmic Discrimination; Fairness Metrics and Bias Mitigation Techniques. Explainable AI (XAI); Black Box Problem; Model Interpretability; LIME and SHAP Techniques; Human-in-the-Loop Systems. Privacy-Preserving AI; Data Ethics and Governance; Differential Privacy; Federated Learning; Adversarial and Inference Attacks.

Case Study: Bias audit of real-world datasets (e.g., COMPAS); explainability case study using LIME/SHAP on loan-approval or medical-diagnosis models.

Unit-V Governance, Accountability and Human-Centered AI 6 Hours

Accountability, Transparency, and Auditability; AI Risk and Impact Assessment; Human-Centered and Accessible AI Design; Sustainable AI; AI and Employment. Regulatory Frameworks — GDPR, Case Study: AI risk/impact assessment for a healthcare or finance deployment; GDPR vs. NITI Aayog compliance comparison.

Unit-VI Emerging Trends and Future of Responsible AI 6 Hours

Generative AI and LLM Ethics; Deepfakes and Synthetic Media; Autonomous Weapons and Military AI. AGI and the AI Alignment Problem; AI and Democracy/Misinformation; Global AI Governance and Standards; Responsible AI for SDGs. Case Study: Deepfake/synthetic-media detection; analysis of a recent LLM or Generative AI controversy.

List of Practical's:

1. Implement a Simple Reflex Agent for a 4×4 vacuum-cleaner grid world; print the percept and action for each cell; extend to a Goal-Based Agent and compare outputs.
2. Given five hardcoded AI decisions (loan denial, content removal, etc.), score each under Utilitarianism, Deontology, and Virtue Ethics using a fixed weight table; display a heatmap using Seaborn.
3. Feed ChatGPT/Gemini a paragraph describing the COMPAS recidivism tool; ask it to identify Accountability, Responsibility, and Transparency issues; document and critique the response in a one-page worksheet.
4. Design five targeted prompts to test ChatGPT on Fairness, Transparency, Accountability, Privacy, and Inclusiveness (one prompt each); score responses 1–5 and fill a provided scorecard template.
5. Ask a GenAI tool to list core principles of IEEE, ACM, Google AI Principles, and EU AI Act; paste results into a pre-formatted comparison table and mark any factual errors found via a quick web check.
6. Load the UCI Adult Income CSV; compute Demographic Parity Difference by gender using pandas; plot a grouped bar chart of positive-prediction rates; state whether bias exists and by how much.
7. Load a pre-trained scikit-learn Random Forest (Heart Disease dataset, provided pickle); run SHAP on five test samples; plot the summary bar chart and one force plot; write two-line interpretation of each.
8. Simulate Federated Learning with two clients on the Iris dataset: each client trains a Logistic Regression on its half; average the coefficients on a central server; compare server-model accuracy with a single centralized model.
9. Paste the model details from Practical 7 or 8 into ChatGPT; prompt it to generate a Model Card (purpose, performance, limitations, bias, intended use); review output against NITI Aayog Responsible AI checklist and note gaps.
10. Upload three images (real and AI-generated) to a free deepfake-detection site (e.g., Hive Moderation); record confidence scores; then prompt ChatGPT to analyse one recent deepfake or LLM misinformation incident using Utilitarianism and Accountability

List of Course Project topics:

1. Bias Detection and Fairness Analysis in AI Systems
2. Explainable AI (XAI) for Machine Learning Models
3. Responsible AI-Based Healthcare Diagnosis System
4. Deepfake Detection and Verification
5. AI Ethics Audit and Compliance Framework
6. Human-Centered Recommendation System
7. Privacy-Preserving Machine Learning using Federated Learning
8. Responsible Chatbot for Education and Public Services

Assessment scheme: covers following aspects of Modified Blooms Taxonomy:
L2 Understanding, L3 Apply, L3 Design, L3 Apply, L4 Analyze and L5 Evaluate

Comprehensive Viva Voce: End Semester Examination: 100 Marks converted to 20 Marks
Course Project: End Semester Examination: 100 Marks converted to 40 Marks
GD/PPT: End Semester Examination: 100 Marks converted to 20 Marks
HA: 100 Marks converted to 20 Marks

Text Books: ***(As per IEEE format)***

1.M. Coeckelbergh, AI Ethics. Cambridge, MA, USA: The MIT Press, 2020.
2. Q. Lu, L. Zhu, J. Whittle, and X. Xu, Responsible AI: Best Practices for Creating Trustworthy AI Systems, 1st ed. Boston, MA, USA: Addison-Wesley Professional, 2024.

Reference Books: ***(As per IEEE format)***

1.S. J. Russell and P. Norvig, Artificial Intelligence: A Modern Approach, 4th ed. Hoboken, NJ, USA: Pearson, 2021.
2 L. Floridi, The Ethics of Artificial Intelligence: Principles, Challenges, and Opportunities. Oxford, U.K.: Oxford University Press, 2023.
3 V. Dignum, Responsible Artificial Intelligence: How to Develop and Use AI in a Responsible Way. Cham, Switzerland: Springer, 2019.
4 Q. Lu, L. Zhu, J. Whittle, and X. Xu, Responsible AI: Best Practices for Creating Trustworthy AI Systems, 1st ed. Boston, MA, USA: Addison-Wesley Professional, 2024.

MOOCs Links and additional reading material:

1. https://www.coursera.org/learn/ethics-of-artificial-intelligence
2. https://www.coursera.org/learn/ai-ethics
3. https://www.coursera.org/learn/ai-ethics-responsible-use-and-creativity
4. https://onlinecourses.nptel.ac.in/noc25_cs118/preview
5. https://ethics-of-ai.mooc.fi
6. https://www.coursera.org/specializations/explainable-artificial-intelligence-xai
7. https://www.coursera.org/learn/responsible-ai-principles

Course Outcomes:

The student will be able to –

1. Explain the concepts, ethical principles, and societal implications of Artificial Intelligence and Responsible AI.
2. Analyze AI systems for fairness, bias, transparency, explainability, and accountability using appropriate ethical frameworks.
3. Evaluate privacy, security, trustworthiness, and governance issues associated with AI applications and data-driven systems.

4. Apply responsible AI principles, risk assessment methods, and regulatory guidelines in the design and deployment of AI solutions.
5. Develop human-centered, inclusive, and sustainable AI solutions that address societal and organizational challenges responsibly.
6. Assess emerging ethical challenges related to Generative AI, Large Language Models, autonomous systems, and global AI governance.

CO-PO Map:

	Program Outcomes (PO)												PSO		
CO/PO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**	**PSO3**
CO1	3	2	-	-	1	2	1	3	-	-	-	2	2	1	-
CO2	2	3	-	2	1	2	-	3	-	-	-	2	2	1	-
CO3	2	3	1	2	2	3	1	3	-	-	-	2	2	2	1
CO4	2	3	2	3	2	3	1	3	-	-	-	2	2	3	1
CO5	2	2	3	2	2	3	3	3	1	1	-	3	2	3	2
CO6	2	3	2	2	2	3	2	3	1	1	-	3	2	3	2
Average	**2.17**	**2.67**	**1.33**	**1.83**	**1.67**	**2.67**	**1.33**	**3.00**	**0.33**	**0.33**	**0.00**	**2.33**	**2.0**	**2.17**	**1.0**

Attainment Levels: 2, 3, 3, 3, 3, 3

CO Attainment Levels:

Weights for attainment levels: L1 – Easy – 0.75 | L2 – Comfortable – 0.70 | L3 – Medium – 0.65 | L4 – Somewhat Difficult – 0.60 | L5 – Difficult – 0.55

CO1 – L2, CO2 – L3, CO3 – L4, CO4 – L3, CO5 – L5, CO6 – L4

Future Course Mapping:

Explainable Artificial Intelligence (XAI), Privacy-Preserving Machine Learning, AI Security and AI Auditing and Compliance

Job Mapping:

Responsible AI Engineer, AI Ethics Specialist, AI Auditor, AI Compliance Officer, Research Scientist (AI Ethics)

Syllabus Template **FF No.: 654**

CI3203C: **Information Security**

Credits: 4 **Teaching Scheme Theory: 3 Hours/Week**

Lab: 2 Hours/Week

Course Prerequisites: Basic knowledge of Computer Networks, Operating Systems, and Programming concepts.

Course Objectives:

1. To understand the fundamentals of information security and cyber threats.
2. To study cryptographic techniques and secure communication mechanisms.
3. To analyze system and network security mechanisms.
4. To understand ethical hacking and vulnerability assessment concepts.
5. To explore cyber laws and emerging security technologies.
6. Design and recommend secure computing solutions by integrating modern security technologies, cyber laws, and best security practices.

Course Relevance:

This course provides the fundamentals of information security, including cryptography, network and system security, secure authentication, and vulnerability assessment. It equips students with practical knowledge of ethical hacking, cyber defense, and emerging security technologies such as cloud, IoT, blockchain, and AI security. The course prepares learners to design, implement, and manage secure computing systems in modern digital environments

Section 1: Topics/Contents

Unit-I Fundamentals of Information Security **7 Hours**

Introduction to Information Security: Need for security, asset, threat, vulnerability, attack, countermeasure, CIA Triad: - Confidentiality, Integrity, Availability, authentication, non-repudiation, Threats and Vulnerabilities: - Passive/active attacks, insider and outsider threats, Malware and Cyber Attacks: - Virus, worm, Trojan, ransomware, spyware, rootkits. Social Engineering: - Phishing, vishing, baiting, tailgating and prevention.

Unit-II Cryptography **7 Hours**

Introduction to Cryptography:- Plaintext, ciphertext, encryption, decryption, classical ciphers, Public key infrastructure, Key pair generation, Symmetric Cryptography:- DES, AES and secret key encryption Asymmetric Cryptography:- RSA, Diffie-Hellman and public key encryption , Elliptic Curve Cryptography basics, Hash Functions and Digital Signatures:- SHA-256, MD5, integrity and authentication, Digital signature verification, SSL/TLS and Certificates:- HTTPS, certificate authority and certificate chain

Unit-III System and Network Security **7 Hours**

Firewalls: - Packet filtering, stateful inspection and application firewalls, IDS and IPS: Intrusion detection and intrusion prevention techniques, VPN and Wireless Security: VPN concepts, WPA2 and WPA3, Authentication and Authorization: -Passwords, OTP, biometrics and RBAC, Multi-factor authentication, Access Control and System Hardening: - DAC, MAC, RBAC and least privilege principle

Section2: Topics/Contents

Unit-IV Ethical Hacking and Vulnerability Assessment **7 Hours**

Ethical Hacking Concepts: - Types of hackers and penetration testing basic, Reconnaissance and Scanning: - WHOIS, DNS lookup, Google dorking and Nmap Vulnerability Assessment: - CVE, CVSS and Nessus overview, SQL Injection: - SQL injection attacks and prevention techniques, Penetration Testing: - Pentest phases, reporting and responsible disclosure

Unit-V Security Management and Cyber Laws **7 Hours**

Security Policies: - Password policy, acceptable use policy and ISO 27001 overview, Risk Management: - Risk identification, mitigation, BCP and DRP basics, Cyber Ethics: - Responsible use, privacy and professional ethics, Cyber Laws and Data Protection: - IT Act 2000, GDPR and DPDP Act 2023

Unit-VI Emerging Trends in Information Security **7 Hours**

Cloud Security:- IaaS, PaaS, SaaS, IAM and encryption, IoT Security: - IoT vulnerabilities and security measures, Blockchain Security: -Distributed ledger security, hash chaining, Merkle tree, wallet security, private key protection, AI in Cyber Security: - Threat intelligence, anomaly detection and deepfakes

List of Practical's:

1. Study of Basic Linux Security Commands

Learn and practice important Linux security commands such as creating users, changing file permissions, setting passwords, using sudo access, and monitoring running processes. Understand how Linux helps in protecting system resources and user data.

2.Implementation of Classical Cryptographic Algorithms

Write programs for Caesar Cipher and Vigenère Cipher using Java/Python. Encrypt and decrypt simple messages to understand the basics of classical encryption techniques.

3.AES Encryption and Decryption

Develop a program to perform AES encryption and decryption for secure data transmission. Explore different AES modes and understand how modern encryption protects sensitive information.

4. RSA Encryption and Decryption

Create a program using RSA algorithm to generate keys, encrypt messages, and decrypt them. Learn how public-key cryptography is used for secure communication.

5.SHA-256 Hashing Demonstration

Generate SHA-256 hashes for different inputs and observe how small changes in data create completely different hash values. Understand the use of hashing in password security and data integrity.

6.Packet Capture and Analysis using Wireshark

Use Wireshark to capture and analyze network packets. Identify different protocols and compare HTTP and HTTPS communication to understand secure data transfer.

7. Network Scanning using Nmap

Perform basic network scanning using Nmap to identify open ports and running services on a system. Learn how network scanning helps in security testing and system analysis.

8.Firewall Configuration and Rule Management

Configure simple firewall rules to allow or block network traffic. Understand how firewalls help in protecting systems from unauthorized access.

9. SQL Injection Demonstration using DVWA

Use DVWA to understand how SQL Injection attacks work in web applications. Perform basic demonstrations in a safe environment and learn methods to prevent such attacks.

10. Password Security and Hash Cracking

Study password security concepts by testing weak passwords, generating password hashes, and understanding salting techniques. Learn why strong passwords are important for cybersecurity.

11. Vulnerability Assessment using Nessus/OpenVAS

Perform vulnerability scanning using Nessus/OpenVAS tools to identify security weaknesses in systems and networks. Study scan reports and understand basic security recommendations.

12.Mini Project / Case Study

Develop a mini project or prepare a case study related to cybersecurity topics such as secure applications, encryption tools, vulnerability analysis, or recent cyber-attacks and their prevention methods.

List of Course Project topics:

1. Secure Authentication and Access Control Systems
2. Cryptographic Applications and Secure Communication
3. Network Security and Intrusion Detection Systems (IDS)
4. Ethical Hacking and Penetration Testing
5. Vulnerability Assessment and Penetration Testing (VAPT)
6. Web Application Security and OWASP Top 10
7. Cloud Security and Identity and Access Management (IAM)
8. Internet of Things (IoT) Security
9. Blockchain Security and Secure Digital Transactions
10. Artificial Intelligence and Machine Learning for Cybersecurity
11. Digital Forensics and Incident Response
12. Malware Analysis and Ransomware Detection

13. Security Operations Center (SOC) and Threat Intelligence
14. Secure Database and Data Privacy Systems

Assessment Scheme: Suggest an Assessment scheme that is best suited for the course. Ensure 360-degree assessment and check if it covers all aspects of Bloom's Taxonomy.
Comprehensive Viva Voce: End Semester Examination: 100 Marks converted to 20 Marks
Course Project: End Semester Examination: 100 Marks converted to 40 Marks
GD/PPT: End Semester Examination: 100 Marks converted to 20 Marks
HA: 100 Marks converted to 20 Marks

Text Books

1. W. Stallings, *Cryptography and Network Security: Principles and Practice*, 8th ed. Harlow, U.K.: Pearson Education, 2023.
2. B. A. Forouzan and D. Mukhopadhyay, *Cryptography and Network Security*, 4th ed. New Delhi, India: McGraw Hill Education, 2015.
3. N. Godbole and S. Belapure, *Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives*, New Delhi, India: Wiley India, 2011.
4. C. P. Pfleeger, S. L. Pfleeger, and J. Margulies, *Security in Computing*, 5th ed. Upper Saddle River, NJ, USA: Prentice Hall, 2015.
5. M. Gregg, *CEH Certified Ethical Hacker All-in-One Exam Guide*, 6th ed. New York, NY, USA: McGraw-Hill Education, 2021.

Reference Books: ***(As per IEEE format)***

1.W. Stallings, *Network Security Essentials: Applications and Standards*, 7th ed. Harlow, U.K.: Pearson Education, 2023.
2.B. Schneier, *Applied Cryptography: Protocols, Algorithms, and Source Code in C*, 20th Anniversary ed. Indianapolis, IN, USA: Wiley, 2015.
3.D. R. Stinson and M. B. Paterson, *Cryptography: Theory and Practice*, 4th ed. Boca Raton, FL, USA: CRC Press, 2019.
4.W. Stallings and L. Brown, *Computer Security: Principles and Practice*, 5th ed. Harlow, U.K.: Pearson Education, 2023.
5.E. Cole, *Network Security Bible*, 2nd ed. Indianapolis, IN, USA: Wiley Publishing, 2011.
6.W. Easttom, *Computer Security Fundamentals*, 5th ed. Hoboken, NJ, USA: Pearson, 2022.
7.J. Erickson, *Hacking: The Art of Exploitation*, 2nd ed. San Francisco, CA, USA: No Starch Press, 2008.
8.A. Silberschatz, P. B. Galvin, and G. Gagne, *Operating System Concepts*, 10th ed. Hoboken, NJ, USA: Wiley, 2018.

MOOCs Links and Additional Reading Material

1. Coursera – IBM Cybersecurity Analyst Professional Certificate

2. Coursera – Google Cybersecurity Professional Certificate
3. LinkedIn Learning – Cybersecurity Foundations
4. LinkedIn Learning – Ethical Hacking Foundations
5. TryHackMe Cyber Security Labs

Additional Reading Material

1. OWASP Web Security Resources
2. NIST Cybersecurity Framework (CSF) 2.0
3. CISA Cybersecurity Resources
4. MITRE ATT&CK Framework
5. SANS Cyber Aces Online
6. EC-Council Learning Resources

Course Outcomes: The student will be able to –

1. Explain concepts of information security and cyber threats
2. Apply cryptographic techniques for secure communication
3. Analyse network and system security mechanisms
4. Perform vulnerability assessment using security tools
5. Evaluate cyber laws and emerging trends
6. Design and recommend secure computing solutions by integrating modern security technologies, cyber laws, and best security practices.

CO-PO Map:

	Program Outcomes (PO)												PSO		
CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	-	-	-	-	-	1	-	-	-	2	3	2	–
CO2	3	3	2	-	2	-	-	-	-	-	-	2	3	3	3
CO3	3	3	2	2	3	-	-	-	-	-	-	2	3	3	2
CO4	3	3	3	2	3	-	-	-	1	-	-	2	2	2	1
CO5	2	2	1	-	-	2	1	3	-	1	1	2	3	2	2
CO6	3	3	2	2	3	1	–	–	2	2	1	3	3	3	2
Average	2.8	2.5	2	2	2.3	1.6	2	3	2	1.6	1	2.17	2.8	2.5	2

CO Attainment levels:

Weights for attainment levels: L1 - Easy-0.75 L2 - Comfortatble-0.7 L3 – Medium – 0.65
L4 – Somewhat difficult – 0.6 L5 – Difficult – 0.55
CO1 – L1, CO2 – L2, CO3 – L3, CO4 – L3, CO5 – L4 and CO6 – L5

Future Course Mapping:

Network Security, Cryptography and Network Security, Cyber Forensics and Incident Response, Cloud Security, Application Security and Secure Software Engineering, IoT Security,

Blockchain Security, Artificial Intelligence and Machine Learning Security, Security Operations Center (SOC) and Threat Intelligence, Digital Forensics and Malware Analysis, DevSecOps and Secure Cloud Computing, Advanced Ethical Hacking and Penetration Testing.

Job Mapping:

Cyber Security Analyst, Information Security Engineer, Security Operations Center (SOC) Analyst, Penetration Tester (Ethical Hacker), Vulnerability Assessment and Penetration Testing (VAPT) Engineer, Network Security Engineer, Cloud Security Engineer, Security Consultant, Incident Response Analyst, Digital Forensics Analyst, Cyber Threat Intelligence Analyst, Governance, Risk and Compliance (GRC) Analyst.

MM0901A: Automata Theory

Credits: 3 **Teaching Scheme Theory: 2 Hours/Week**

Tutorial: 1 Hours/Week

Course Prerequisites: Discrete Mathematics

Learning Objectives:

1. To introduce the students to basics of Automata Theory
2. To study abstract computing models to provide a formal connection between algorithmic problem solving and the theory of languages
3. To learn Grammar, Pushdown Automata and Turing Machine for language processing and algorithm design
4. To learn about the Automata theory computability and complexity for algorithm design
5. To learn about Turing Machine for the different requirements outlined by theoretical computer science
6. To understand & analyze different classes of problems, and study concepts of NP completeness

Course Relevance:

This course provides a rigorous mathematical foundation of computation through formal models such as Deterministic and Non-Deterministic Finite Automata (DFA/NFA), Pushdown Automata (PDA), and Turing Machines. It covers language classification under the Chomsky hierarchy, regular languages, context-free languages, closure properties, and equivalence transformations between automata and grammars. Students analyze computational limits using decidability, reducibility, and complexity of problems. The course also supports implementation aspects of

lexical analysis and parsing in compiler design through regular expressions and context-free grammars. It strengthens formal reasoning required for algorithm design, verification, and theoretical aspects of machine learning model behavior in AI systems.

Section 1: Topics/Contents

Unit-I: Finite Automata and Formal Language Theory 4 Hours

Basic Concepts: Symbols, Strings, Language, Formal Language, Natural Language. Basic Machine and Finite State Machine. Finite Automata (FA): An informal picture of FA, Finite State Machine (FSM), Language accepted by FA, Definition of Regular Language. FA without output: Deterministic and Nondeterministic FA (DFA and NFA), epsilon- NFA and inter-conversion. FA with output: Moore and Mealy Machines-Definition, models, inter-conversion

Unit II: Regular Expressions of Regular Language 5 Hours

Introduction, Operators of RE, Precedence of operators, Algebraic laws for RE, Language to Regular Expressions, Equivalence of two REs. Conversions: RE to NFA, DFA, DFA to RE using Arden's theorem, Pumping Lemma for Regular languages, Closure and Decision properties of Regular languages

Unit III: Context Free Grammar (CFG) and Context Free Language (CFL) 5 Hours

Basic Elements of Grammar, Formal Definition of Context Free Grammar, Chomsky Hierarchy, Sentential form, Derivation and Derivation Tree/ Parse Tree, Context Free Language (CFL), Ambiguous Grammar, writing grammar for language. Grammar (RG) Conversions: LRG to RLG, RLG to LRG, RG to FA, FA to RG. Simplification of CFG: Eliminating Є-productions, unit productions, useless production, and useless symbols. Normal Forms: Chomsky Normal Form, Greibach Normal Form, Pumping Lemma for CFG, Closure properties of CFL

Section2: Topics/Contents

Unit IV: Pushdown Automata 5 Hours

Introduction, Formal definition of PDA, Equivalence of Acceptance by Final State and Empty stack, Non-deterministic PDA (NPDA) and Deterministic PDA (DPDA), PDA and Context Free Language, Equivalence of PDA and CFG, PDA vs CFLs.

UNIT V: Turing Machines 5 Hours

Introduction, Formal definition of Turing Machines, Language Acceptability by Turing Machines, Universal Turing Machines, Multi-Tape Turing Machines, Multi-Stack Turing Machines, Multi-Track Turing Machines, Halting Problem of TM, Recursion Theorem

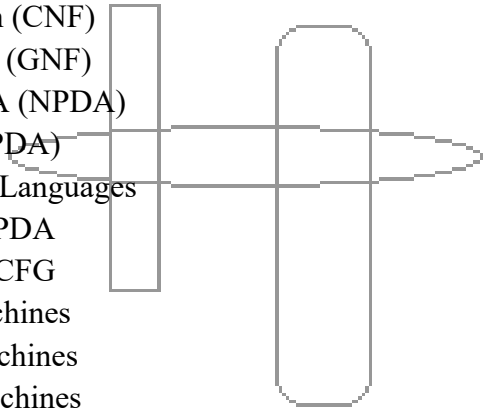
Unit VI: Computability and Complexity Theory 4 Hours

Computability Theory: Decidable Problems and Un-decidable Problems, Church-Turing Thesis.

Reducibility: Undecidable Problems that is recursively enumerable, A Simple Un-decidable. Complexity Classes: Time and Space Measures, The Class P, Examples of problems in P, The Class NP, Examples of problems in NP, P Problem Versus NP Problem, NP-completeness and hard Problems.

List of Tutorials:

1. Basic Machine Concepts and Finite State Machines
2. Deterministic Finite Automata (DFA)
3. Non-Deterministic Finite Automata (NFA)
4. Epsilon-NFA (ε-NFA) and its Representation
5. Conversion of NFA to DFA
6. Conversion of ε-NFA to NFA/DFA
7. Construction of RE from Languages
8. Equivalence of Regular Expressions
9. Conversion of RE to NFA
10. Conversion of RE to DFA
11. Ambiguous Grammar and Ambiguity Removal
12. Chomsky Normal Form (CNF)
13. Greibach Normal Form (GNF)
14. Non-Deterministic PDA (NPDA)
15. Deterministic PDA (DPDA)
16. PDA and Context Free Languages
17. Conversion of CFG to PDA
18. Conversion of PDA to CFG
19. Multi-Tape Turing Machines
20. Multi-Stack Turing Machines
21. Multi-Track Turing Machines
22. Undecidable Problems
23. Time Complexity and Space Complexity
24. Complexity Class P & NP



Assessment Scheme:

L2 Understanding, L3 Apply, L3 Design, L3 Apply, L4 Analyze and L5 Evaluate
Course Project: End Semester Examination: 100 Marks converted to 30 Marks
GD/Presentation: End Semester Examination: 100 Marks converted to 20 Marks
Home Assignment: End Semester Examination: 100 Marks converted to 10 Marks
Theory: End Semester Examination (Written): 60 Marks converted to 40 Marks

Text Books (Networking) : ***(As per IEEE format)***

1. Vivek Kulkarni, “Theory of Computation”, Oxford University Press, ISBN0 - 19 – 808458

2. John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman, "Introduction to Automata Theory Languages and Computation", Addison-Wesley, ISBN 0-201-44124-1
3. Daniel Cohen, "Introduction to Computer Theory", Wiley & Sons, ISBN97881265133454

Reference Books (Networking) : ***(As per IEEE format)***

1. Sanjeev Aroraand Boaz Barak, "Computational Complexity: A Modern Approach", Cambridge University Press, ISBN: 0521424267 97805214242643
2. John Martin, "Introduction to Languages and The Theory of Computation", 2nd Edition, McGraw Hill Education, ISBN-13:978-1-25-900558-9, ISBN-10: 1-25-900558-5
3. J. Carroll & D Long, "Theory of Finite Automata", Prentice Hall, ISBN 0-13-913708-45.

MOOCs Links and additional reading material:

1. **https://nptel.ac.in/courses/106/104/106104148/**
2. **https://nptel.ac.in/courses/106/104/106104028/**

Course Outcomes:

The student will be able to –

1. Understand formal language, translation logic, essentials of translation, alphabets, language representation and apply it to design Finite Automata and its variants
2. Construct regular expression to present regular language and understand pumping lemma for RE
3. Design Context Free Grammars and learn to simplify the grammar
4. Construct Pushdown Automaton model for the Context Free Language
5. Design Turing Machine for the different requirements outlined by theoretical computer science
6. Understand different classes of problems, classify and analyze them and study concepts of NP completeness

CO-PO Map:

	Program Outcomes (PO)												PSO		
CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	1	1	1	-	-	-	-	-	-	1	2	2	1
CO2	2	3	1	2	1	-	-	-	-	-	-	1	2	2	1
CO3	2	2	2	2	1	-	-	-	-	-	-	1	2	2	2
CO4	2	2	2	2	2	-	-	-	-	-	-	1	2	2	2
CO5	2	2	2	2	2	-	-	-	-	-	-	2	2	2	2
CO6	2	2	2	2	1	-	-	1	-	-	-	2	2	2	2
Average	2.17	2.17	1.67	1.83	1.33	-	-	0.17	-	-	-	1.33	2.00	2.00	1.67

Attainment Levels: 2,3,4,4,4,5

CO Attainment levels:

Weights for attainment levels: L1 - Easy-0.75 L2 - Comfortatble-0.7 L3 – Medium – 0.65 L4 – Somewhat difficult – 0.6 L5 – Difficult – 0.55

CO1 – L2, CO2 – L3, CO3 – L4, CO4 – L4, CO5 – L4 and CO6 – L5

Future Course Mapping: Compiler Design, Natural Language Processing (Syntax Analysis), Quantum Computing Fundamentals

Job Mapping: Compiler Engineer, Algorithm Engineer, Systems Programmer

Syllabus Template **FF No.: 654**

MM0901B: Design Thinking and Product Lifecycle Management (PLM)

Credits: 3 **Teaching Scheme Theory: 2 Hours/Week**

Tutorial: 1 Hours/Week

Course 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.

Learning Objectives:

1. To apply Design Thinking principles to identify and solve real-world problems.
2. To understand the stages and significance of Product Development Life Cycle.
3. To analyze and evaluate different PDLC and software development models.
4. To use AI tools and techniques in product planning, development, testing, and maintenance.
5. To develop practical skills through prototyping, testing, and case-study- based learning.
6. To study industrial practices followed in modern AI-driven product companies.

Course Relevance:

The Design Thinking Product Lifecycle Management (PDLC) course is highly relevant for Artificial Intelligence and Engineering students as it provides practical knowledge of transforming innovative ideas into real-world products. The course integrates Design Thinking, AI-assisted development, Agile methodologies, testing, deployment, and industrial case studies, helping students develop industry-oriented skills required in modern software and product engineering domains.

Section 1: 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, Introduction to Reverse Engineering, Reverse Engineering Process and Techniques, Product Analysis and Feature Extraction, Competitive Product Analysis, Product Redesign and Improvement Strategies.
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.

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: AI-based Smart Attendance and Analytics System

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

Assessment Scheme: Suggest an Assessment scheme that is best suited for the course. Ensure 360-degree assessment and check if it covers all aspects of Bloom's Taxonomy.
Assessment scheme covers following aspects of Modified Blooms Taxonomy:
L2 Understanding, L3 Apply, L3 Design, L3 Apply, L4 Analyze and L5 Evaluate
Course Project: End Semester Examination: 100 Marks converted to 30 Marks
GD/PPT: End Semester Examination: 100 Marks converted to 20 Marks
Home Assignment: 100 Marks converted to 10 Marks
Theory: End Semester Examination (Written): 60 Marks converted to 40 Marks

Text Books (Networking) : ***(As per IEEE format)***

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. 2T. Brown, Change by Design: How Design Thinking Creates New Alternatives for Business and Society. New York, USA: Harper Business, 2009, ISBN: 978- 0061766084.
3. 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 (Networking) : *(As per IEEE format)*

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.

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)

Course Outcomes:

The 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.

CO-PO Map:

	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

CO Attainment levels:

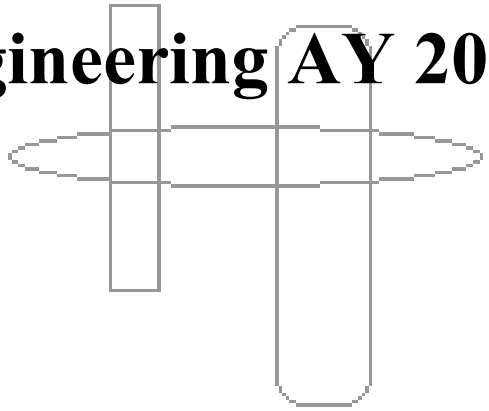
Weights for attainment levels: L1 - Easy-0.75 L2 - Comfortatble-0.7 L3 – Medium – 0.65
L4 – Somewhat difficult – 0.6 L5 – Difficult – 0.55

CO1 – L1, CO2 – L2, CO3 – L3, CO4 – L4, CO5 – L4 and CO6 – L5

Future Course Mapping: Smart Manufacturing Systems, Business Analytics for Product Strategy, System Design and Architecture

Job Mapping: Product Manager, UX/UI Designer, Product Designer, Business Analyst, Innovation Manager

B. Tech. Final Year Computer Engineering AY 2026-27



Module VII and VIII
Course Content

Syllabus Template **FF No.: 654**

Coursera Specialization Courses **Credits: 2**

Sr. No.	Course code	Specialization Name	Link
1	MD4228	IBM Full Stack Software Developer	https://www.coursera.org/professional-certificates/ibm-full-stack-cloud-developer
2	MD4230	IBM Back-End Developer	https://www.coursera.org/professional-certificates/ibm-backend-development
3	MD4248	IBM DevOps and Software Engineering	https://www.coursera.org/professional-certificates/devops-and-software-engineering
4	MD4262	Salesforce Sales Development Representative	https://www.coursera.org/professional-certificates/salesforce-sales-operations
5	MD4238	Microsoft Cybersecurity Analyst	https://www.coursera.org/programs/faculty-development-program-d5iiv/professional-certificates/microsoft-cybersecurity-analyst?source=search
6	MD4243	IBM Data Engineering	https://www.coursera.org/professional-certificates/ibm-data-engineer
7	MD4245	IBM Data Science	https://www.coursera.org/ professional-certificates/ibm-data-science
8	MD4247	IBM Data Warehouse Engineer	https://www.coursera.org/professional-certificates/data-warehouse-engineering
9	MD4257	IBM Mainframe Developer	https://www.coursera.org/professional-certificates/ibm-mainframe-developer
10	MD4269	Google UX Design	https://www.coursera.org/professional-certificates/google-ux-design

Syllabus Template **FF No.: 654**

CI 4001: Generative AI

Credits: 2 **Teaching Scheme Theory: 2 Hours/Week**

Course Prerequisites:
Python, Basic ML, Linear Algebra, Neural Networks

Course Objectives

1. To introduce the foundational concepts and evolution of Generative AI models such as GANs, VAEs, and Diffusion models.
2. To explain the basics of text processing and natural language understanding using real-world examples.
3. To provide a conceptual understanding of sequence modeling using RNNs, LSTMs, and GRUs.
4. To describe advanced NLP techniques including Attention Mechanism, Transformers, and BERT.
5. To familiarize students with the architecture and functioning of Large Language Models (LLMs) such as GPT and BERT.
6. To analyze practical applications and ethical concerns associated with Generative AI.
7. To promote awareness of responsible AI practices through case-based learning and discussions.

Unit-I: Processing Text with Python Essential Training **4 Hours**

Introduction to Text and NLP, Text Cleaning and Tokenization, Stemming and Lemmatization, Stop Words and POS Tagging, Named Entity Recognition (NER), Text Vectorization (BoW, TF-IDF, Embeddings), Sentiment Analysis, Text Classification Basics, Introduction to Text Generation, NLP Libraries: NLTK, spaCy, Hugging Face

Unit-II: Hands-On Natural Language Processing **5 Hours**

Text Data Loading and Preprocessing, Tokenization and POS Tagging with spaCy, Named Entity Recognition (NER), Text Classification using Scikit-learn, Word Embeddings (Word2Vec, GloVe), Introduction to Transformers and BERT

Unit-III: Advanced NLP with Python for Machine Learning **5 Hours**

Text Feature Engineering, N-grams and Language Models, Topic Modeling with LDA, Sequence Models with RNN and LSTM, Attention Mechanism Basics, Introduction to Transfer Learning in NLP

Unit-IV Deep Learning Foundations: Natural Language Processing with Tensor Flow **4 Hours**

Introduction to TensorFlow for NLP, Text Vectorization with TensorFlow, Building Neural Networks for Text Classification, Word Embeddings with Embedding Layer, Implementing RNNs and LSTMs in TensorFlow, Introduction to Transformer Models using TensorFlow

Unit-V Recurrent Neural Networks **5 Hours**

Introduction to Sequential Data and RNNs, RNN Architecture and Working, Vanishing Gradient Problem, Long Short-Term Memory (LSTM) Networks, Gated Recurrent Unit (GRU), Text Generation with RNNs

Unit VI Generative AI: Working with Large Language Models **5 Hours**

Overview of Large Language Models (LLMs), Pretraining and Fine-Tuning of LLMs, Prompt Engineering Techniques, In-Context and Few-Shot Learning, Introduction to Retrieval-Augmented Generation (RAG), Applications of LLMs in Text Generation and Chatbots

1. Reference Print Books

1. D. Foster; Generative Deep Learning: Teaching Machines to Paint, Write, Compose, and Play; 1st Edition; O'Reilly Media; 2019
2. J. Shane; You Look Like a Thing and I Love You: How AI Works and Why It's Making the World a Weirder Place; 1st Edition; Voracious; 2019
3. T. Mitchell; Machine Learning; 1st Edition; McGraw Hill; 1997
4. S. Russell, P. Norvig; Artificial Intelligence: A Modern Approach; 4th Edition; Pearson; 2020

Reference Electronic Books

1. H. Schmidt-Walter, R. Kories; Electrical Engineering: A Pocket Reference; Artech House, 2007; Accessed: Oct. 16, 2016; [Online]. Available: https://ebookcentral.proquest.com
2. F. Chollet; Deep Learning with Python; 2nd Edition; Manning Publications; 2021; Accessed: May30,2024; Available: https://www.manning.com/books/deep-learning-with-python-second-edition

3. I. Goodfellow, Y. Bengio, A. Courville; Deep Learning; MIT Press; 2016; Accessed: Apr. 22, 2024; Available: https://www.deeplearningbook.org

MOOCs and Other Learning Resources

1.I.Goodfellow,Y.Bengio;Introduction to Generative Adversarial Networks; DeepLearning.AI(Coursera); https://www.coursera.org/learn/build-basic-generative-adversarial-networks-gans; Accessed: May 28, 2024
2. L. Fei-Fei, J. Ngiam; CS231n: Convolutional Neural Networks for Visual Recognition; Stanford University; http://cs231n.stanford.edu; Accessed: May 25, 2024
3.C.Olah;TheIllustrated Transformer; Distill.pub; https://distill.pub/2016/deconv-checkerboard/; Accessed: May 26, 2024
4.OpenAI; ChatGPT Prompt Engineering for Developers; OpenAI; https://platform.openai.com/education; Accessed: May 29, 2024

Syllabus Template **FF No.: 654**

CI4005: Natural Language Processing

Credits: 2 **Teaching Scheme Theory: 2 Hours/Week**

Course Prerequisites: Theory of Computer Science, Compiler Design

Course Objectives:

1. To understand morphology for given natural language
2. To learn how to design lexical analyzer for given natural language
3. To learn how to design Syntactic Analyzer for given natural language
4. To learn how to design type dependency parser using pragmatic approach for given natural language
5. To understand the scientific process for machine transliteration, machine translation and information retrieval for given natural language using statistical approach

Course Relevance: Although Natural Language Processing (NLP) has been with us for quite some time, it has only recently gained industry-wide attention, thanks to Deep Learning. Today, NLP is a core competence area in Data Science and IT, with applications spanning across sectors that rely on harnessing language data's potential. Essentially, NLP applications are designed to extract relevant and meaningful information from natural human language data and impart machines with the ability to interact with humans.

SECTION 1: TOPICS/CONTENTS

Unit-I Introduction **5 Hours**

What is natural language processing? Applications of NLP, Origins of NLP, Challenges of NLP, Language and Knowledge, Language and Grammar, Processing Indian Languages. Grammar-based language models, lexical functional Grammar(LFG), Government and Binding (GB), Lexical functional Grammar Model, Generative grammars, Statistical Language Model.

Unit-II Regular Expressions and Automata **5 Hours**

Formal Language Theory: Basic Notions, Basic Regular Expression Patterns, Disjunction, Grouping and Precedence, Advanced Operators, Substitution, Finite State Automata, NFSA. Words and Transducers, Morphology, Inflectional Morphology, Derivational Morphology, Finite State Morphological Parsing, Construction of Finite State Lexicon, Finite State Transducers, FST for Morphological Parsing.

Unit-III Theory of parsing / Syntactic Analysis **4 Hours**

Context Free Grammar, parsing, Top-down Parsing, Bottom-up parsing, Probabilistic parsing, Indian Languages parsing Semantic Analysis: Meaning Representation, Lexical Semantic, Ambiguity, Word Sense Disambiguation, Discourse processing, Natural Language Generation.

SECTION2: TOPICS/CONTENTS

Unit-IV Computer Linguistics **5 Hours**

Machine Transliteration using Statistical Language modeling: N-gram model, Machine Transliteration: Rule-based, Phonology and Stress Analysis based and Statistical based, Support vector machine, Memory Entropy Model, Hidden Markov Model, Conditional Random Fields, Evaluation Metrics

Unit-V Machine Translation **5 Hours**

Introduction, Problems in MT, Characteristics of Indian Languages, Machine Translation Approaches, Direct Machine Translation, Rule-Based MT, Corpus Based Machine Translation, Semantic/Knowledge based MT Systems, Translation involving Indian Languages, Statistical-Based using MT Tools - GIZA++, SRTLM and Moses, Evaluation Metrics

.

Unit-VI Information Retrieval **4 Hours**

Designing features for IR Systems, IR Models, Classical IR Models, Non Classical IR Models. Evaluation of IR Systems, NLP in IR, Relation Mapping, and Knowledge based Approaches, Conceptual Graphs in IR, Cross Language Information Retrieval, Evaluation Metrics.

List of Course Project Areas:

1. Biomedical Text Mining.

2. Computer Vision and also NLP.
3. Deep Linguistic Processing.
4. Controlled Natural Language.
5. Language Resources and also Architectures for NLP.
6. Sentiment Analysis and also Opinion Mining.
7. Recognizing Similar Texts
8. Inappropriate Comments Scanner
9. Language Identifier
10. Image-Caption Generator

List of Design based Home Assignments:

Design:

1. Use a simple method to classify positive or negative sentiment in tweets
2. Use a more advanced model for sentiment analysis
3. Use vector space models to discover relationships between words and use principal component analysis (PCA) to reduce the dimensionality of the vector space and visualize those relationships
4. Write a simple English-to-French translation algorithm using pre-computed word embeddings and locality sensitive hashing to relate words via approximate k-nearest neighbors search
5. Create a simple auto-correct algorithm using minimum edit distance and dynamic programming
6. Write a better auto-complete algorithm using an N-gram model (similar models are used for translation, determining the author of a text, and speech recognition)
7. Write your own Word2Vec model that uses a neural network to compute word embeddings using a continuous bag-of-words model
6. Train a neural network with GLoVe word embeddings to perform sentiment analysis of tweets
8. Train a recurrent neural network to perform NER using LSTMs with linear layers
Translate complete English sentences into French using an encoder/decoder attention model
9. Build a transformer model to summarize text
1. Build a chatbot using a reformer model.

Case Study:

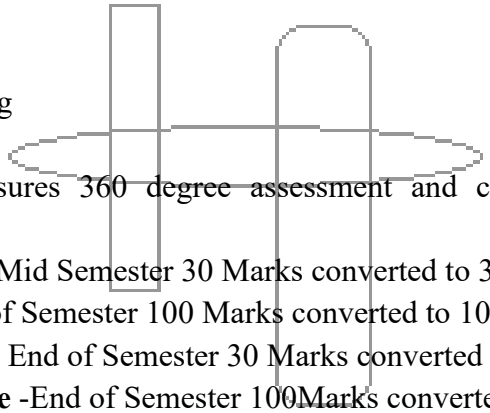
1. Clinical Documentation
2. Speech Recognition
3. Computer-Assisted Coding (CAC)
4. Data Mining Research
5. Automated Registry Reporting
6. Clinical Decision Support
7. Clinical Trial Matching
8. Prior Authorization

Blog:

1. Machine Translation: Rule-Based
2. Machine Translation: Statistical-Based
3. MT Tools - GIZA++, SRTLM and Moses
4. GIZA++, SRTLM and Moses
5. Natural Language Resources for Beginners
6. Natural Language Resources for Practitioners
7. Biomedical Text Mining
8. Computer Vision and also NLP
9. Deep Linguistic Processing
10. Controlled Natural Language.
11. Language Resources and Architectures for NLP
12. Sentiment Analysis and also Opinion Mining
13. NLP includes Artificial Intelligence

Survey:

1. Language Models
2. Top-down Parsing
3. Bottom-up parsing
4. Probabilistic parsing
5. Indian Languages parsing



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

MCQ Exam – Section I - Mid Semester 30 Marks converted to 30 equivalent Marks

Home Assignment - End of Semester 100 Marks converted to 10 equivalent Marks

MCQ Exam – Section II - End of Semester 30 Marks converted to 30 equivalent Marks

Comprehensive Viva Voce -End of Semester 100Marks converted to 30 equivalents Marks

Text Books:

1.Tanveer Siddiqui and U S Tiwary, "Natural Language Processing and Information Retrieval" Fourth Impression, Oxford, ISBN-13:978-019-569232-7.

2. Daniel Jurafsky and James H Martin., "Speech and Language Processing", 2nd edition, Pearson, Second Impression-2014,ISBN: 978-93-325-1841-43.

3. Christopher D.Manning and Hinrich Schutze,, "Foundations of Statistical Natural Language Processing" , MIT Press, 1999.

Reference Books:

1. *Alexander Clark (Editor), Chris Fox (Editor), Shalom Lappin (Editor), "The Handbook of Computational Linguistics and Natural Language Processing".*
2. *Natural Language Processing with Python – Analyzing Text with the Natural Language Toolkit, Steven Bird, Ewan Klein, and Edward Loper, O'Reilly Publication*
3. *Natural Language Processing with Python CookBook, Krishna Bhavsar, Naresh Kumar, Pratap Dangeti, Packt Publication.*

4. *Ralph Grishman, "Computational Linguistics: An Introduction (Studies in Natural Language Processing)", Cambridge University Press, ASIN : B01MQYCTOB.*

MOOCs Links and additional reading material:

- www.nptelvideos.in
- www.nfnlp.com
- https://www.mooc-list.com/tags/nlp
- https://www.my-mooc.com/en/mooc/natural-language-processing-nlp/
- https://huggingface.co/learn/nlp-course/
- https://www.coursera.org/learn/attention-models-in-nlp

Course Outcomes:

The student will be able to:

1. Interpret morphology for given natural language (2)
2. Construct shallow depth lexical analyzer and syntactic analyzer for given natural language (3)
3. Develop shallow depth type dependency parser for given natural language (3)
4. Develop shallow depth machine transliteration, machine translation, information retrieval for given natural language using either linguistic or statistical approach (3)
5. Correlate shallow depth machine transliteration, machine translation, information for given natural language (4)
6. Evaluate machine transliteration, machine translation and information retrieval results using standard evaluation metrics (5)

CO-PO Map:

	Program Outcomes (PO)												PSO		
CO/PO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**	**PSO3**
CO1	3	2		1	-	-	-	-	-	-	-	2	2	2	3
CO2	3	2	2	1	3	-	-	2	-	-	-	2	2	2	3
CO3	2	2	3	2	3	-	-	-	1	-	-	2	2	2	3
CO4	3	3	3	2	3	2	-	-	-	-	-	2	2	2	3
CO5	3	2		2	3	2	-	-	1	1	2	2	2	2	3
CO6	3	2		2	3	2	2	2		1	2	2	2	2	3
Average	**2.8**	**2.16**	**2.7**	**1.88**	**3**	**2.0**	**2.0**	**2.0**	**1.0**	**1.0**	**2.0**	**2.0**	**2.0**	**2.0**	3.0

CO Attainment levels:

Weights for attainment levels: L1 - Easy-0.75 L2 - Comfortatble-0.7 L3 – Medium – 0.65
L4 – Somewhat difficult – 0.6 L5 – Difficult – 0.55
CO1 – L2, CO2 – L3, CO3 – L3, CO4 – L3, CO5 – L4 and CO6 – L5

Future Course Mapping:

Computational Linguistics, ANN, RNN, Deep Learning

Job Mapping:

Application Developers, System programmer.

Syllabus Template **FF No.: 654**

Image Processing

Credits: 2 **Teaching Scheme Theory: 2 Hours/Week**

Course Prerequisites: Digital logic Design, Microprocessor, Computer Organization.

Course Objectives:

1.To describe different color models and image processing techniques.
2.To analyze image condition and deduce enhancement algorithms.
3.To apply image segmentation to identify the region of interest
4.To develop an algorithm to recognize the specified objects in the given image.
5.To study different image morphological operation.
6.To learn different image compression techniques.

Course Relevance:

Vision sense is the most powerful human sense organ. In the world where intelligent automation is taking place, image processing is a vital domain for research and development. In Industry 4.0, image processing systems built around industrial cameras are an essential component in automated production. Throughout all steps of production, from the inspection of raw materials and production monitoring (i.e., flaw detection) to final inspections and quality assurance, they are an indispensable part of achieving high efficiency and quality standards. In the Entertainment Industry, latest trends such as 4K video streaming requires high quality compression that can provide limited/no loss image quality with high fps. In social networking, sharing images has been a vital part. Creating innovative effects and overall manipulating the images will be explored

Section 1: Topics/Contents

Unit-I Introduction **4 Hours**

Elements of image processing system, Scenes and Images, Vector Algebra, Human Visual System, color vision color model: RGB, HVS, YUV, CMYK, YCbCr and some basic relationships between pixels, linear and nonlinear operations, Image sampling and quantization.

Unit-II Image Enhancements **5 Hours**

Memory-less operations, Spatial domain image enhancements: Denoising filters, Smoothing Operation, Sharpening Operation, and Contrast stretching /enhancement, histogram and histogram equalization.

Unit-III Image segmentation **5 Hours**

Classification of image segmentation techniques: Edge-based Segmentation, Region based techniques. Binarization: Global Thresholding, Adaptive thresholding. Types of Edge detector: derivative filters, Sobel, Canny. Edge linking. Feature Extraction.

Section2:Topics/Contents

Unit-IV Morphological Operation **4 Hours**

Binary Morphology, Erosion Dilation, Opening and Closing.

Unit-V Feature Extraction and Object Recognition **5 Hours**

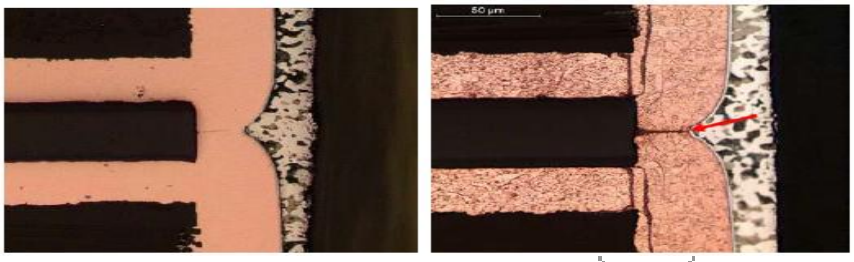
Feature points and feature detection (Line, circle and corner). Line detection: RANSAC, Hough Transform. Corner detection: Harris Corner Detector. Feature descriptors, Descriptor matching. SIFT, Boundary representation (Chain code), Boundary detection-based techniques.

Unit-VI Image Compression **5 Hours**

Introduction and need, Coding redundancy, classification of compression techniques (Lossy and lossless- JPEG, Run Length Coding, Huffman Coding, Shannon Fano coding).

List of Design based Home Assignments:

1.Design an algorithm to identify fault in a “PCB inspection system” as shown below

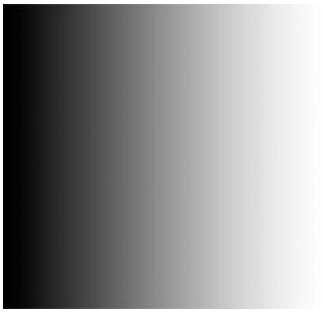


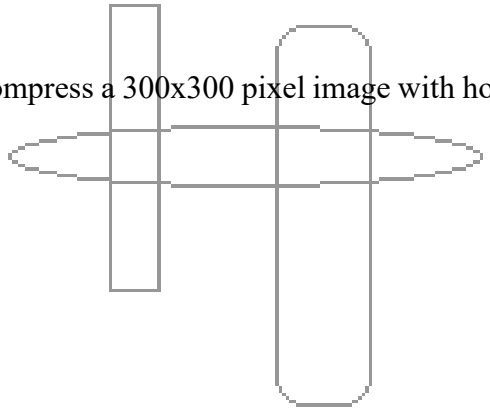
2.Design an algorithm to perform segmentation of the image below to extract the mango from its background.

3.Design an algorithm to get from image 1 to image 2.

4.Design an algorithm to recognize character “0” in the image below.

5.Design an algorithm to compress a 300x300 pixel image with horizontal black to white gradient as shown below.





List of Case Study based Home Assignments:

1. Cam-scanner: Document scanning app
2. Tesseract OCR library
3. Instagram filters
4. OpenCV
5. Google Street View

List of Blog based Home Assignments:

1. Image processing on Embedded platforms
2. Face recognition system security analysis for authentication
3. Image processing in MSME for effective automation

4. H.264 codec for image streaming
5. Role of mathematics in image processing

List of Survey based Home Assignments:

1. Image quality metrics
2. Vision based self-driving car safety
3. Compression techniques & codecs
4. State of the art applications such as AR/ XR
5. Human recognition in social networking apps like Facebook

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

MCQ Exam – Section I - Mid Semester 30 Marks converted to 30 equivalent Marks
Home Assignment - End of Semester 100 Marks converted to 10 equivalent Marks
MCQ Exam – Section II - End of Semester 30 Marks converted to 30 equivalent Marks
Comprehensive Viva Voce -End of Semester 100Marks converted to 30 equivalents Marks

Text Books: ***(As per IEEE format)***

1. *Rafael Gonzalez & Richard Woods, "Digital Image Processing," 3rd Edition, Pearson publications, ISBN 0132345633.*

2. *Anil K. Jain, "Fundamental of Digital Image Processing," 5th Edition, PHI publication, ISBN 13: 9780133361650.*

Reference Books: ***(As per IEEE format)***

1.*Pratt, "Digital Image Processing," Wiley Publication, 3rdEdition, ISBN 0-471- 37407-5.*

2. *K.R. Castleman, "Digital Image Processing,"3rd Edition, Prentice Hall: Upper Saddle River, NJ, 3, ISBN 0-13-211467 -4.*

MOOCs Links and additional reading material:

1. https://nptel.ac.in/courses/117/105/117105135/
2. https://nptel.ac.in/courses/106/105/106105032/
3. https://nptel.ac.in/noc/courses/noc19/SEM2/noc19-cs58/
4. https://www.coursera.org/learn/computer-vision-basics

Course Outcomes:

The student will be able to –

1.Recognize different color models and image processing techniques. (1)
2.Select image enhancement algorithm to improve the quality of image. (2)
3.Build image segmentation techniques to identify region of interest. (4)
4.Predict image morphological techniques to resize the image. (3)
5.Construct an algorithm to recognize the specified objects in the given image. (5)
6. Identify different image compression techniques to reduce the size of image. (3)

CO-PO Map:

	Program Outcomes (PO)												PSO			
CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3	PSO 4
CO1	3	2		1	-	-	-	-	-	-	-	2	2	2	3	3
CO2	3	2	2	1	3	-	-	2	-	-	-	2	2	2	3	3
CO3	2	2	3	2	3	-	-	-	1	-	-	2	2	2	3	3
CO4	3	3	3	2	3	2	-	-	-	-	-	2	2	2	3	3
CO5	3	2		2	3	2	-	-	1	1	2	2	2	2	3	3
CO6	3	2		2	3	2	2	2		1	2	2	2	2	3	3
Average	**2.8**	**2.16**	**2.7**	**1.88**	**3**	**2.0**	**2.0**	**2.0**	**1.0**	**1.0**	**2.0**	**2.0**	**2.0**	**2.0**	3.0	3.0

CO Attainment levels:

Weights for attainment levels: L1 - Easy-0.75 L2 - Comfortatble-0.7 L3 – Medium – 0.65
L4 – Somewhat difficult – 0.6 L5 – Difficult – 0.55

CO1 – L2, CO2 – L2, CO3 – L4, CO4 – L3, CO5 – L5 and CO6 – L3

Future Course Mapping:

1. Augmented Reality
2. Multimedia Processing

Job Mapping:

1. Augmented Reality Experience Designer
2. Automation Engineer
3. Embedded Software Developer
4. Image Processing Expert

Syllabus Template **FF No. : 654**

CI4008: Major Project

Credits:.10 **Teaching Scheme Theory: …26... Hours/Week**

Course Prerequisites: Project Based Learning

Aim

This course addresses the issues associated with the successful management of a project. The course emphasizes project life cycle phases requirement engineering, system analysis and system design. A further aim is for students to heighten personal awareness of the importance of developing strategies for themselves and working with peers to create desired outcomes. The Project Work can lead to:

- Transform existing Ideas into conceptual models.
- Transform conceptual models into determinable models.
- Use determinable models to obtain system specifications.
- Select optimum specifications and create physical models.
- Apply the results from physical models to create real target systems.

Project Group and Topic Selection and Synopsis:

The project work needs to be undertaken by a group of maximum FOUR and minimum of THREE students. The Project work will be jointly performed by the project team members. The student needs to identify a technological problem in the area of Computer Engineering or Information Technology of their choice and address the problem by formulating a solution for the identified problem. The Project Group will prepare a synopsis of the project work which will be approved by the concerned faculty member. The project should not be a reengineering or reverse engineering project. In some cases, reverse engineering projects will be permissible based on the research component involved in it. The project work aims at solving a real world technical problem. Hence ample literature survey is required to be done by the students. Application-oriented projects will not be acceptable. Low-level custom User Interface development and its allied mapping with a particular technology will not be accepted.

Overview of the Course:

1. The Student Project Group is expected to make a survey of situation for identifying the requirements of selected Technological Problem. The Student Project Group will be monitored by Internal Guides and External Guides (if any).
2. The project requires the students to conceive, design, implement and operate a mechanism (the design problem). The mechanism may be entirely of the student's own design, or it may incorporate off-the-shelf parts. If the mechanism incorporates off-the-shelf parts, the students must perform appropriate analysis to show that the parts are suitable for their intended purpose in the mechanism.
3. The project must be based on a Fresh Idea or Implementation of a Theoretical Problem – meaning that there is not a known Solution to the design problem Or Create a Better Solution.
4. The project must have an experimental component. Students must conceive, design, implement and operate an appropriate experiment as part of the project. The experiment might be to collect data about some aspect of the design (i.e., to verify that the design will work as expected). Alternatively, the experiment could be to verify that the final mechanism performs as expected.
5. Upon receiving the approval, the Student Project Group will prepare a preliminary project report consisting , Feasibility Study Document, System Requirement Specification, System Analysis Document, Preliminary System Design Document. All the documents indicated will have a prescribed format.

6. Upon project completion, the Student Project Group will prepare a detailed Project Report consisting Semester I Preliminary Project document along with Detailed System Design Document, Implementation and Testing Document with conclusion and future scope of the Project Work. All the documents indicated will have a prescribed format. The Project Report ideally should consist of following documents: (Exceptions may be there based on the nature of the project, especially if some of the following documents are not applicable to a particular project as determined by the project guide, coordinator and head of department).

Sr.	Project Item
1	Project Cover Front Page
2	Project Completion Certificate [Institute]
3	Project Completion Letter [In case of Sponsored Projects]
4	Acknowledgments
5	Table of Contents
6	List of Figures
7	List of Tables
8	Project Synopsis [Problem Background, Existing System Details, Proposed Solution]
9	Feasibility Study Report
10	Project Plan
11	System Requirement Specification
12	System Analysis Document: UML Use Case Diagrams
13	System Analysis Document: UML Sequence Diagrams
14	System Analysis Document: UML State Diagrams
15	System Design Document with Module Specifications
16	System Implementation
17	System Testing and Experimental Findings
18	Conclusion
19	References

7. The Project Work will be assessed jointly by a panel of examiners consisting faculty and industry experts. The Project Groups will deliver the presentation and demonstration of the Project Work which will be assessed by the panel.
8. The Student Project Group needs to actively participate in the presentation. The panel of examiners will evaluate the candidate's performance based on presentation skills,

questions based on the Project Work and overall development effort taken by the candidates.

Note:

The student needs to design and develop solution for the identified technological problem in the area of Computer Engineering or Information Technology of their choice. The Project Implementation needs to be completed using best possible use of available technologies as applicable to deal with the complexity of the project. The Project Group will prepare a detailed report of the project work which will be approved by the concerned faculty member. The Project Report need to be submitted both in Hard form and Soft form in CD. The Soft Copy of the Project Report must accompany other project deliverables as well.

Assessment: MSE and ESE

1. Mid Semester Assessment – 50 Marks to be converted to 30 Marks.
2. End Semester Assessment – 100 Marks to be converted to 70 Marks.

Mid Semester Assessment

Sr. No.	Parameter	Marks
1	Problem Statement	10
2	Literature Review	10
3	Group formation and identification of individual responsibility	10
4	Objective of Project activity	10
5	Knowledge of domain, latest technology and modern tools used /to be used	10
	TOTAL	**50**

End Semester Assessment

Sr. No.	Parameter	Marks
1	Realization of project as per problem statement	10
2	Design, Testing / Experimentation, Analysis / Validation	30
3	Documentation and Report Writing	20
4	Quality of Work	15
5	Performance in Question & Answers Session	15
6	Regular interaction with guide	10
	TOTAL	**100**

Course Outcomes:
Upon completion of the course, graduates will be able to -

CO1: Identify the real life problem from societal need point of view

CO2: Prepare the requirement engineering, feasibility analysis documents

CO3: Form the teams and share responsibilities according to individual skill strengths

CO4: Create design documents to build software solutions

CO5: Develop software solutions based on standard engineering specifications

CO6: Perform the verification and validation up to the mark

CO PO Map

	Program Outcomes (PO)												PSO		
CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO3
CO1	2	3		2					3			3	-		3
CO2	2	3	3	2	2				3	3	2	3	3	3	3
CO3	2	-	-	-	-				3		2	3	2	-	3
CO4	2	3	3	2	2	3	3.0	2.0	3	3	2	3	3	3	3
CO5	2	3	3	2	2				3	3	2	3	2	3	3
CO6	2	2	2	3	2				3	2	2	3	3	3	3
Average	**2.0**	**2.8**	**2.75**	**2.83**	**2.0**	**3.0**	**3.0**	**2.0**	**3.0**	**2.75**	**2.0**	**3.0**	**3.0**	**1.75**	**3.0**

CO attainment levels: CO1 -2 CO2 –2 CO3-3 CO4-5 CO5 -2 CO6-3

2:0.07, 3.0065, 5:.055

Syllabus Template **FF No.: 654**

CI4010, CI4011, CI4012, CI4013: Industry Internship, Project Internship, Research Internship, Global Internship, International Internship

Credits:.16 **Teaching Scheme Theory:32. Hours/Week**

Guidelines:

Internships are educational and career development opportunities, providing practical experience in a field or discipline. Internships are far more important as the employers are looking for employees who are properly skilled and having awareness about industry environment, practices

and culture. Internship is structured, short-term, supervised training often focused around particular tasks or projects with defined time scales. Core objective is to expose technical students to the industrial environment, which cannot be simulated/experienced in the classroom and hence creating competent professionals in the industry and to understand the social, economic and administrative considerations that influence the working environment of industrial organizations. Engineering internships are intended to provide students with an opportunity to apply theoretical knowledge from academics to the realities of the field work/training.

The following guidelines are proposed to give academic credits for the internship undergone as a part of the B.Tech. Engineering curriculum.

Duration: Industry Internship will be started at the beginning of the semester 7 or semester 8 or yearlong for the duration 6 months or 12 months.

Identification of Internship work:

Student may choose to undergo Internship at Industry/Govt./NGO/MSME/ Innovation/IPR/Entrepreneurship. Contacting various companies for Internship and Internship work identification process should be initiated at the end of 6th semester in coordination with training and placement cell/ industry institute cell/ internship cell. This will help students to start their internship work on time. Students can take internship work in the form of online/onsite work from any of the following but not limited to:

- Working for consultancy/ research project
- Contribution in Incubation/ Innovation/ Entrepreneurship Cell/ Institutional Innovation Council/ start-ups cells of institute
- Industry / Government Organization Internship,
- Internship through Internshala
- In-house product development, intercollegiate, inter department research internship under research lab/group, micro/small/medium enterprise/online internship
- Research internship under professors, IISC, IIT's, Research organizations

Internship Documents Submission:

Students must submit internship offer letter, internship completion letter, FF 1029 (Students feedback form), FF 1030 (Industry feedback about interns).

Students must present their internship progress time to time to faculty mentors. Faculty mentors and industry mentors both can evaluate the progress of the intern combiningly.

Internship Work Evaluation:

In-semester and end semester internship evaluation and assessment will be done by internal (Faculty mentor) and external examiners - a supervisor from industry.

After completion of Internship, the student should prepare a comprehensive report to indicate what he has observed and learnt in the internship/training period. The student may contact Industrial Supervisor/ Faculty Mentor/TPO for assigning special topics and problems and should prepare the final report.

If the student remains absent without prior intimation to the department/institute/concern authority/T & P Cell, his entire training can be cancelled and he will fail.

Course Outcomes: Industry Internship

On the completion of course, students will able to-

1. Understand real-world applications, workplace environment and operating procedures
2. Adapt skill for learning and applying modern tools and technologies
3. Apply professional values and ethical standards
4. Perform as an individual and as a team member effectively to changing conditions
5. Encompass improved writing, verbal communication and documentation skills
6. Learn about career positions and occupations along with the qualities and training required to obtain those positions

CO-PO Map: Industry Internship

	Program Outcomes (PO)												PSO			
CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3	PSO 4
CO1	2	3		2					3			3				2
CO2	2	3	3	2	3	2	3	2	3		2	3	3	3	3	3
CO3	2	3	3	2				2	3		2	3	3	3	3	3
CO4	2								3			3				
CO5	2		2		3			2	3	3	2	3			3	
CO6	2					2			3			3				
Average	2.0	3.0	2.66	2.0	3.0	2.0	3.0	2.0	3.0	3.0	2.0	3.0	3.0	3.0	3.0	2.66

CO attainment levels:

CO1 – 3, CO2 –5, CO3 –2, CO4 –2, CO5 – 3, CO6 –2

Course Outcomes: Research Internship

On the completion of course, students will able to-

1. Develop an ecosystem to promote entrepreneurship and research culture among the students.

2. Learn first-hand to apply techniques, resources, modern engineering tools for prediction modelling to complex engineering activities.
3. Apply reasoning informed by the contextual knowledge to access societal, health, safety, legal and cultural issues and consequent responsibilities.
4. Perform as an individual and as a team member.
5. Understand Engineering and Management Principles.
6. Exercise R & D aptitude focusing on the knowledge creation and dissemination through engineering artifacts creation, construction and presentation.

CO-PO Map: Research Internship

	Program Outcomes (PO)												PSO			
CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3	PSO 4
CO1	2											3				3
CO2	3	3	3	2	3	2	2	2				3	3	3	3	3
CO3	3	3	3	2		3						3	3		3	3
CO4									3			3				3
CO5	3	2	3	2								3	3		3	3
CO6	3	3	2	3			3	2.0		3.0	2.0	3			3	3
Average	**2.8**	**2.75**	**2.75**	**233**	**3.0**	**2.5**	**2.5**	**2.0**	**3.0**	**2.75**	**2.0**	**3.0**	**3.0**	**3.0**	**3.0**	**3.0**

CO1 – 3, CO2 –4, CO3 –5, CO4 –2, CO5 – 3, CO6 –4