COURSE TITLE: Computer Vision

Course Code: MMCADCV325						Examination Scheme	T	P
Total number of Lecture Hours: 60						External	72	-
						Internal	28	-
Lecture (L):	4	Practical (P):	-	Tutorial (T):	-	Total Credits		4

Course Objectives

- To Gain a comprehensive understanding of the principles, methodologies, and challenges in computer vision, including image formation, representation, and basic processing techniques.
- To develop the ability to extract meaningful features from images and apply advanced segmentation techniques, preparing for complex tasks like object recognition.
- To learn to apply computer vision algorithms for motion tracking, enabling the analysis and tracking of object movement in real-world video sequences.
- To acquire the skills to perform optical flow analysis, feature matching, and depth estimation, with a focus on 3D reconstruction using multi-camera systems.

Course Content	TEACHING HOURS
UNIT I: Introduction.	**15 Hrs**
Introduction to Computer Vision: Overview, History, and Applications of Computer Vision, Key Challenges, Image Formation and Representation, Basic Image Processing Techniques, Overview of Computer Vision Algorithms.	
UNIT II: Feature Extraction and Image Segmentation	**15 Hrs**
Feature Extraction: Edges (Canny, LOG, DOG), Line Detectors (Hough Transform), Corners (Harris and Hessian Affine), Orientation Histogram (SIFT, SURF, HOG, GLOH), Scale-Space Analysis (Image Pyramids and Gaussian Derivative Filters), Other Filters (Gabor Filters and DWT). Image Segmentation: Region Growing, Edge-Based Approaches to Segmentation, Graph-Cut, Mean-Shift, MRFs, Texture Segmentation.	
UNIT III: Object Motion & Tracking	**15 Hrs**
Object Motion & Tracking: Understand Motion Models (Define Object Movement Over Time), Analyze Videos as Sequences of Individual Image Frames, Programmatically Track a Single Point Over Time, implement a Method for Tracking a Set of Unique Features Over Time.	
UNIT IV: Optical Flow & Feature Matching	**15 Hrs**
Optical Flow & Feature Matching: Optical Flow (Track a Moving Car Using Optical Flow), Feature Matching (Match Features from One Image Frame to Another), Depth Estimation and Multi-camera Views (Perspective, Binocular Stereopsis, Camera and Epipolar Geometry, Homography, Rectification, DLT, RANSAC, 3-D Reconstruction Framework, Auto-calibration).	

Textbooks:

1. Computer Vision: Algorithms and Applications" by Richard Zaleski, Springer Nature, 2nd Edition, 2022
2. Multiple View Geometry in Computer Vision" by Richard Hartley and Andrew Zisserman, Cambridge University Press, 2nd Edition, 2004
3. Digital Image Processing" by Rafael C. Gonzalez and Richard E. Woods, Pearson Publishing, 4th Edition 2018.

Reference Books:

1. Computer Vision: A Modern Approach" by David A. Forsyth and Jean Ponce, Pearson Publishing, 2nd Edition, 2012.

COURSE LEARNING OUTCOMES (CLO):

CLO1: Students will describe the history, applications, and challenges in computer vision.

CLO2: Students will extract features using methods like SIFT, SURF, HOG, and Hough Transform.

CLO3: Students will model object motion and track features over time across video frames.

CLO4: Students will apply optical flow and feature-matching algorithms to image sequences.

CLO-PLO Matrix for the Course

Unit-Wise CLOs	PLO 1	PLO 2	PLO 3	PLO 4	PLO 5	PLO 6	PLO 7	PLO 8	PLO 9	PLO 10	Average (CLO)
MMCADCV325.1	3	1	1	1	2	0	1	1	1	2	**1.3**
MMCADCV325.2	2	3	2	1	3	0	1	2	2	3	**1.9**
MMCADCV325.3	2	3	3	1	3	0	1	2	2	3	**2.0**
MMCADCV325.4	2	3	2	1	3	0	1	2	2	3	**1.9**
Average (PLO)	**2.3**	**2.5**	**2.0**	**1.0**	**2.8**	**0.0**	**1.0**	**1.8**	**1.8**	**2.8**	**1.78**

COURSE TITLE: Enterprise Resource Planning

Course Code: MMCADER325						**Examination Scheme**	**T**	**P**
Total number of Lecture Hours: 60						**External**	**72**	-
						Internal	**28**	-
Lecture (L):	**4**	**Practical (P):**	-	**Tutorial (T):**	-	**Total Credits**		**4**

Course Objectives

- To understand the evolution, definition, and growth of ERP, including its advantages, various modules, and relevance to different business models, particularly in the context of India.
- To analyse the relationship between ERP and related technologies, such as Business Process Reengineering (BPR), Management Information Systems (MIS), and Supply Chain Management (SCM).
- To evaluate the ERP implementation lifecycle, including planning, system selection, training, data migration, and the roles of consultants, vendors, and employees in successful ERP implementation.
- To assess the post-implementation challenges of ERP systems, the factors influencing their success or failure, and explore emerging trends such as extended ERP systems, CRM, SCM, and web-enabled ERP solutions.

Course Content	**TEACHING HOURS**
UNIT I: Introduction to ERP	**15 Hrs**
Evolution of ERP; what is ERP? Reasons for the Growth of ERP; Scenario and Justification of ERP in India; Evaluation of ERP; Various Modules of ERP; Advantage of ERP. An Overview of Enterprise. Integrated Management Information; Business Modeling; ERP for Small Business; ERP for Make to Order Companies; Business Process Mapping for ERP Module Design; Hardware Environment and its Selection for ERP Implementation	
UNIT II: ERP and related Technologies	**15 Hrs**
ERP and Related Technologies; Business Process Reengineering (BPR); Management Information System (MIS); Executive Information System (EIS); Decision support System (DSS); Supply Chain Management (SCM	
UNIT III: ERP Implementation	**15 Hrs**
ERP Implementation: Planning Evaluation and selection of ERP systems - Implementation life cycle - ERP implementation, Methodology and Frame work-Training – Data Migration - People Organization in implementation-Consultants, Vendors and Employees	
UNIT IV: Post Implementation and Emerging Trends	**15 Hrs**
Post Implementation: Maintenance of ERP- Organizational and Industrial impact; Success and Failure factors of ERP Implementation Emerging Trends on ERP: Extended ERP systems and ERP add-ons -CRM, SCM, Business analytics - Future trends in ERP systems-web enabled.	

Textbooks

1. Manufacturing Resource Planning (MRP II) with Introduction to ERP; SCM; an CRM by Khalid Sheikh, Publisher: McGraw-Hill
2. ERP and Supply Chain Management by Christian N. Madu, Publisher: CHI.

Reference Books

1. Jagan Nathan Vaman, ERP in Practice, Tata McGraw-Hill, 2008
2. Alexis Leon, Enterprise Resource Planning, second edition, Tata McGraw-Hill, 2008.
3. Mahadeo Jaiswal and Ganesh Vanapalli, ERP Macmillan India, 2009
4. Vinod Kumar Grag and N.K. Venkitakrishnan, ERP- Concepts and Practice, PHI.

COURSE LEARNING OUTCOMES (CLO):

By the end of this course, students will be able to:

CLO1: Explain the evolution, significance, and advantages of ERP systems, and evaluate ERP modules and their applicability to different business environments, including small and make-to-order enterprises.
CLO2: Analyze the relationship between ERP systems and associated technologies such as BPR, MIS, EIS, DSS, and SCM, and assess their role in enhancing organizational efficiency.
CLO3: Demonstrate understanding of ERP implementation life cycles, methodologies, and frameworks, including vendor and consultant roles, training, and data migration processes.
CLO4: Evaluate the post-implementation impact of ERP systems on organizations, identify critical success and failure factors, and explore emerging trends like CRM, SCM integration, and web-enabled ERP systems.

CLO-PLO Matrix for the Course

Unit-Wise CLOs	PLO 1	PLO 2	PLO 3	PLO 4	PLO 5	PLO 6	PLO 7	PLO 8	PLO 9	PLO 10	Avg (CLO)
MMCADER325.1	3	2	3	2	2	1	1	2	1	2	**1.90**
MMCADER325.2	2	3	3	2	3	1	2	2	2	3	**2.30**
MMCADER325.3	3	3	3	2	3	2	2	2	2	3	**2.50**
MMCADER325.4	3	2	3	3	3	2	2	3	2	3	**2.60**
Average (PLO)	**2.75**	**2.50**	**3.00**	**2.25**	**2.75**	**1.50**	**1.75**	**2.25**	**1.75**	**2.75**	**2.32**

DCE-VI

COURSE TITLE: Natural Language Processing

Course Code: MMCADNL325						**Examination Scheme**	
Total number of Lecture Hours: 60						**External**	**72**
						Internal	**28**
Lecture (L):	**4**	**Practicals(P):**	**0**	**Tutorial (T):**	**0**	**Total Credits**	**4**

Course Learning Objectives

- To provide foundational knowledge of natural language processing principles, including text representation, linguistic structures, and computational models.
- To enable students to understand and analyze key NLP algorithms, such as tokenization, part-of-speech tagging, and sentiment analysis, and their role in text processing.
- To develop practical skills in NLP programming using tools like NLTK, spaCy, and Transformers, enabling the design and implementation of NLP pipelines.
- To identify and assess challenges in NLP, including data sparsity, bias in language models, and multilingual processing.

Course Content	TEACHING HOURS
UNIT 1: Introduction to Natural Language Processing	**15 Hrs**
Introduction to NLP: definitions, goals, applications, Components of NLP: tokens, lexicon, grammar, semantics. Text preprocessing techniques: tokenization, normalization, stop-word removal, stemming vs. lemmatization, Basic feature extraction: bag-of-words, TF–IDF.	
UNIT 2: Linguistic Analysis	**15 Hrs**
Part-of-Speech tagging and morphological analysis, Syntactic parsing: constituents vs. dependency trees, Chunking and shallow parsing, Syntax-based feature generation, Evaluating tagging and parsing performance	
UNIT 3: Semantics & Language Modeling Word semantics: embeddings (Word2Vec, GloVe), Contextual representations (e.g., BERT overview), Language modeling: n-grams and neural models, Word sense disambiguation applications, Semantic similarity and clustering	**15Hrs**
UNIT 4: Applications and Future Trends	**15 Hrs**
Sentiment analysis and opinion mining , Text classification and document clustering, Information extraction: NER, relation extraction, Introduction to question answering and summarization, Advanced interpretability (e.g. feature importance, opening the door to your SHAP work)	

Textbooks:

1. Speech and Language Processing, Daniel Jurafsky and James H. Martin, Prentice Hall, 2020.

2. Deep Learning for Natural Language Processing, Palash Goyal, Sumit Pandey, and Karan Jain, Apress, *2018.*

Reference Books

1. Natural Language Processing with Python, Steven Bird, Ewan Klein, and Edward Loper, O'Reilly Media, 2009.
2. Foundations of Statistical Natural Language Processing, Christopher D. Manning and Hinrich Schütze, MIT Press, 1999.
3. Transformers for Natural Language Processing, Denis Rothman, Packt Publishing, 2021.

COURSE LEARNING OUTCOMES (CLO):

CLO1: Explain the foundational concepts of natural language processing, including text representation, preprocessing, and word embeddings.
CLO2: Analyze and apply key NLP algorithms such as HMMs, CRFs, RNNs, LSTMs, and transformer-based models.
CLO3: Develop NLP pipelines using standard programming libraries and frameworks (e.g., NLTK, spaCy, Hugging Face).
CLO4: Evaluate real-world NLP applications, assess challenges like bias and multilinguality, and explore emerging trends like LLMs and conversational AI.

LEVEL OF CO-PO MAPPING TABLE

	PLOs										
Unit wise CLOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**Avg (CLO)**
MMCADNL325.1	3	2	2	2	2	1	1	2	1	2	**1.8**
MMCADNL325.2	3	3	3	2	3	1	2	2	2	2	**2.3**
MMCADNL325.3	3	3	3	2	3	2	2	2	2	2	**2.4**
MMCADNL325.4	3	3	2	2	2	3	3	3	2	3	**2.6**
Average (PLO)	**3.0**	**2.75**	**2.5**	**2.0**	**2.5**	**1.75**	**2.0**	**2.25**	**1.75**	**2.25**	**2.28**

COURSE TITLE: Software Quality Assurance

Course Code: MMCADSQ325						**Examination Scheme**	**T**	**P**	
Total number of Lecture Hours: 60						**External**	**72**	**-**	
						Internal	**28**	**-**	
Lecture (L):	**4**	**Practical (P):**	**-**	**Tutorial (T):**	**-**	**Total Credits**		**4**	

Course Objectives

- To understand the components of the Software Quality Assurance System, including Pre-Project Software Quality Components, Contract Review, and Development and Quality Plans.
- To apply various quality engineering tools and techniques, such as the Seven Basic Quality Tools, Statistical Process Control, and Failure Mode and Effect Analysis (FMEA).
- To analyze different software quality standards and models, including ISO 9000 series, CMMI, Six Sigma, and the integration of AI techniques in software testing and quality assurance.
- To implement Scrum in an organization, understanding the steps for transitioning to Scrum, Scrum artifacts, and the use of Agile project management tools like JIRA and Trello.

Course Content	TEACHING HOURS
UNIT I: Software Quality	**15 Hrs**
Definition of Software Quality, Quality Planning, Quality system. Quality Control vs Quality Assurance , Product life cycle , Project life cycle models. The Software Quality Challenge, Software Quality Factors, Components of the Software Quality Assurance System. Pre-Project Software Quality Components, Contract Review, Development and Quality Plans	
UNIT II: Software Quality Engineering Tools And Techniques	**15 Hrs**
Supporting quality activities- Metrics, Reviews, SCM , Software quality assurance and risk management, Seven basic Quality tools, Checklist, Pareto diagram, Cause and effect diagram, Run chart, Histogram, Control chart, Scatter diagram, Poka Yoke, Statistical process control ,Failure Mode and Effect Analysis, Quality Function deployment, Continuous improvement	
UNIT III: Quality Assurance Models and AI Assurance	**15 Hrs**
Software Quality Standards, ISO 9000 series, CMM, CMMI, P-CMM , Six Sigma, Malcolm Baldrige Quality, Introduction to AI in Software Quality Assurance-Definition and Importance of AI Assurance, Overview of AI in Software Development and Testing, Benefits and Challenges of Integrating AI in QA, Overview of AI Techniques in Software Testing, Automated Test Case Generation-Techniques for Automated Test Case Generation, Ensuring AI Model Quality-Verification and Validation of AI Models, Testing AI Models for Accuracy, Robustness, and Fairness	
Unit IV: Scrum and Agile Model	**15 Hrs**
Introduction to Agile-Definition and History of Agile, Principles of Agile Manifesto, Benefits of Agile Methodology, Overview of Scrum Framework-Definition of Scrum, Scrum Values and Principles, Scrum Artifacts-Product Backlog, Sprint Backlog, Increment, Agile Project Management Tools-Introduction to Tools (e.g., JIRA, Trello, Asana), Implementing Scrum in an Organization-Steps to Transition to Scrum	
Textbooks:	

1. Galin, D. (2018). Software Quality Assurance: From Theory to Implementation (2nd ed.). Pearson.
2. Godbole, N. S. (2017). Software Quality Assurance: Principles and Practice (1st ed.). Alpha Science International Ltd.

Reference Books:

1. Cohn, M. (2006). Agile Estimating and Planning. Prentice Hall.
2. Partridge, D. (1992). Artificial Intelligence in Software Engineering. Routledge.
3. Black, R. (2002). Managing the Testing Process: Practical Tools and Techniques for Managing Hardware and Software Testing (2nd ed.). Wiley.
4. Jones, C. (2017). Applied Software Measurement: Global Analysis of Productivity and Quality (3rd ed.). Addison-Wesley Professional.

COURSE LEARNING OUTCOMES (CO):

CLO1: Understand software quality principles, quality planning, assurance components, and their relevance across software and project life cycles.

CLO2: Apply quality engineering tools and risk management strategies to ensure continuous software quality improvement.

CLO3: Analyze software quality standards and evaluate AI-assisted quality assurance methods and model validation techniques.

CLO4: Demonstrate the ability to implement Agile principles and Scrum framework using modern project management tools.

CLO-PLO Matrix for the Course

Unit-Wise CLOs	PLOs										
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**Average (CLO)**
MMCADSQ325.1	3	2	3	1	3	3	2	2	2	3	**2.4**
MMCADSQ325.2	3	3	3	1	3	3	2	2	2	3	**2.5**
MMCADSQ325.3	3	2	3	1	3	2	2	2	3	3	**2.4**
MMCADSQ325.4	2	3	3	2	3	2	2	2	2	3	**2.4**
Average (PLO)	**2.75**	**2.5**	**3.0**	**1.25**	**3.0**	**2.5**	**2.0**	**2.0**	**2.25**	**3.0**	**2.43**

COURSE TITLE: Deep Learning

Course Code: MMCADDL325						Examination Scheme	T	P
Total number of Lecture Hours: 60						External	72	-
						Internal	28	-
Lecture (L):	4	Practical (P):	0	Tutorial (T):	0	Total Credits		4

Course Objectives

- To provide a comprehensive understanding of deep learning principles, including the distinction between shallow and deep architectures.
- To equip students with the skills to design, implement, and train artificial neural networks (ANNs) and convolutional neural networks (CNNs) for various applications.
- To introduce students to advanced deep learning architectures, techniques, and challenges, including regularization, transfer learning, and neural architecture search.
- To explore cutting-edge topics in deep learning, such as Graph Neural Networks, Meta Learning, Auto encoders, Generative Adversarial Networks (GANs), and Deep Reinforcement Learning.

Course Content	TEACHING HOURS
UNIT I: Fundamentals of Neural Networks and Deep Learning	**15 Hrs**
Introduction to Machine Learning vs Deep Learning, Biological Neuron vs Artificial Neuron, Basics of Neural Networks: Perceptron, Multilayer Perceptron (MLP), Activation Functions: Sigmoid, Tanh, ReLU, Leaky ReLU, Softmax, Loss Functions: MSE, Cross-Entropy, Gradient Descent and Backpropagation, Overfitting and Underfitting, Bias-Variance Tradeoff	
UNIT II: Deep Neural Networks and Training Techniques	**15 Hrs**
Deep Neural Network Architectures, Batch Normalization and Dropout, Weight Initialization Methods, Optimization Algorithms: SGD, Adam, RMSProp, Hyperparameter Tuning: Epochs, Batch Size, and Learning Rate, Model Evaluation: Confusion Matrix, Precision, Recall, F1 Score, ROC-AUC	
UNIT III : Convolution Neural Networks	**15 Hrs**
Introduction to CNNs: Need and Applications, CNN Architecture: Convolutional Layers, Pooling Layers, Fully Connected Layers, Filters, Stride, Padding, Transfer Learning and Pre-trained Models (VGG, ResNet, etc.), Data Augmentation, Case Studies: Image Classification, Object Detection (basic intro)	
UNIT IV: Advanced Topics and Applications	**15 Hrs**
Recurrent Neural Networks (RNNs): Basics and Applications, Long Short-Term Memory (LSTM) and GRU, Introduction to Generative Adversarial Networks (GANs), Autoencoders and Variational Autoencoders, Applications of Deep Learning in NLP, Computer Vision, and Healthcare	
Textbooks	

1. Deep Learning by Ian GoodFellow, MIT Press.2016
2. Advanced Deep Learning with Python, Ivan Vasilev, 2019
3. Advances in Deep Learning, M. Arif Wani, 2019

Reference Books

1. Deep Learning with Python, Francois Chollet, 2nd edition, 2021
2. Deep Reinforcement Learning Hands-On, Maxim Lapan, 2nd edition, 2020
3. Automated Machine Learning Methods, Systems, Challenges, 2019
4. Deep Learning: A Visual Approach, Andrew Glassner, 2021
5. Selected Journal and Conference Papers.

COURSE LEARNING OUTCOMES (CO):

CLO1: Understand core concepts of deep learning and artificial neural networks.
CLO2: Apply convolutional neural network techniques and analyze their architectures.
CLO3: Evaluate neural architecture search methods and their challenges.
CLO4: Implement and assess advanced deep learning models in real-world applications.

LEVEL OF CO-PO MAPPING TABLE

UNIT-WISE CLOs	PLO 1	PLO 2	PLO 3	PLO 4	PLO 5	PLO 6	PLO 7	PLO 8	PLO 9	PLO 10	Avg (CLO)
MMCADDL325.1	3	2	2	0	0	0	0	0	0	0	**2.33**
MMCADDL325.2	0	0	3	3	2	0	0	0	0	0	**2.67**
MMCADDL325.3	0	0	0	0	3	2	0	0	0	0	**2.5**
MMCADDL325.4	0	0	0	0	0	0	3	3	0	0	**3.0**
Avg (PLO)	**1.5**	**1.0**	**1.67**	**1.5**	**1.5**	**1.0**	**1.5**	**1.5**	**0**	**0**	**2.63**

COURSE TITLE: Internet of Things

Course Code: MMCADIT325						Examination Scheme	T	P
Total number of Lecture Hours: 60						External	72	-
						Internal	28	-
Lecture (L):	4	Practical (P):	-	Tutorial (T):	-	Total Credits		4

Course Objectives

- To understand the fundamental concepts of cloud computing, including different cloud deployment models (public, private, hybrid) and service models (IaaS, PaaS, SaaS).
- To explore the technologies and processes involved in deploying and managing cloud-based applications and web services.
- To analyze the management aspects of cloud services, including reliability, availability, scalability, and the economic factors influencing cloud platform choices.
- To gain practical knowledge in cloud-based application development and service creation environments, focusing on the benefits and challenges of cloud architecture.

Course Content	TEACHING HOURS
UNIT 1:	**15 Hrs**
Definition & Characteristics of Iot, Physical Design of Iot, Things in Iot, Iot Protocols; Logical Design Of Iot: Iot Functional Blocks, Iot Communication. Models, Iot Communication APIs; IoT Levels and Templates. Domain. Specific IoTs – Home, City, Environment, Energy, Retail, Logistics, Agriculture, Industry, health and Lifestyle.	
UNIT 2:	**15 Hrs**
Wireless Sensor Networks, Cloud Computing, Big Data Analytic, Communication Protocols, Machine to Machine, Difference between IoT and M2M, Software define Network, Embedded Systems. Design challenges, Development challenges, Security challenges, Other challenges.	
UNIT 3:	**15 hrs**
Introduction, Functional View, Information View, Deployment and Operational View, Other Relevant architectural views. Real-World Design Constraints- Introduction, Technical Design constraints-hardware is popular again. Data representation and visualization, Interaction and remote control. Industrial Automation- Service oriented architecture-based device integration, SOCRADES: realizing the enterprise integrated Web of Things, IMC-AESOP: from the Web of Things to the Cloud of Things.	
UNIT 4 :	**15 hrs**
Setting up the Arduino development environment: Options for Internet connectivity, Interacting with basic sensors, Interacting with basic actuators, Configuring Arduino for the IoT. Grabbing the content from a web page, Sending data to the cloud, Monitoring sensor data from a cloud dashboard, Monitoring several Arduino boards at, Storing data on Google Drive. Basic local M2M interactions, Cloud M2M with IFTTT; Case Study: IoT based Flood Monitoring and Alert System.	
Textbooks	

1. Vijay Madisetti and Arshdeep Bahga, "Internet of Things (A Hands-on-Approach)", 1stEdition, VPT, 2014. (ISBN-13: 978-8173719547)
2. Schwartz, Marco. "Internet of Things with Arduino Cookbook". Packt Publishing Ltd, 2016.

Reference Books

1. Jan Holler, Vlasios Tsiatsis, Catherine Mulligan, Stefan Avesand, Stamatis Karnouskos, David Boyle, "From Machine-to-Machine to the Internet of Things: Introduction to a New Age of
2. Intelligence", 1st Edition, Academic Press, 2014. (ISBN-13: 978-0124076846)
3. The Internet of Things: How Smart TVs, Smart Cars, Smart Homes, and Smart Cities Are Changing the World, Michael Miller

COURSE LEARNING OUTCOMES (CLO):

CLO1: Students will be able to describe and differentiate between key IoT enabling technologies, such as Wireless Sensor Networks, Cloud Computing, and M2M communication, while understanding the role of software-defined networks and embedded systems.
CLO2: Students will gain knowledge of the challenges in IoT design and development, including security and technical constraints, and propose solutions to overcome these issues.
CLO3: Students will acquire skills in designing and analyzing IoT architecture, considering various views like functional, information, and deployment, and apply these concepts to industrial automation and real-world applications.
CLO4: Students will develop practical skills in using Arduino for IoT projects, including setting up the environment, connecting to sensors and actuators, and implementing cloud-based data monitoring and M2M interactions.

LEVEL OF CLO-PLO MAPPING TABLE

CLOs	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8	PLO9	PLO10	Average (CLO)
MMCADIT325.1	3	1	2	0	2	1	1	1	1	2	1.4
MMCADIT325.2	2	1	2	1	3	3	2	2	2	3	2.1
MMCADIT325.3	2	2	3	1	3	1	2	2	2	3	2.1
MMCADIT325.4	2	3	3	1	3	1	2	3	2	3	2.3
Average(PLO)	**2.25**	**1.75**	**2.5**	**0.75**	**2.75**	**1.5**	**1.75**	**2**	**1.75**	**2.75**	**1.97**

COURSE TITLE: Software Project Management

Course Code: MMCACSP325						**Examination Scheme**	
Total number of Lecture Hours: 60						**External**	**72**
						Internal	**28**
Lecture (L):	**4**	**Practicals(P):**	**0**	**Tutorial (T):**	**0**	**Total Credits**	**4**

Course Learning Objectives

- To provide fundamental skills of software Project management emphasizing on issues & hurdles associated with delivering successful projects.
- To apply project management concepts through working in a group as team leader or active team member on an IT project.
- To utilize scheduling terminology, techniques, and tools to create accurate and feasible project timelines.
- To identify potential risks in software projects and develop mitigation strategies to minimize their impact on project outcomes.
- To define and measure key performance indicators (KPIs) to assess project success, including quality, scope adherence, and stakeholder satisfaction.

Course Content	TEACHING HOURS
UNIT 1: Introduction to SPM	**15 Hrs**
Fundamentals of Software Project Management (SPM), Need Identification, Vision and Scope Document, Project Management Cycle, SPM Objectives, SPM Framework, Software Project Planning, Planning Objectives, Project Plan, Types of Project Plan, Structure of a Software Project Management Plan. Software Project Estimation, Estimation Methods, Estimation Models, Decision Process	
UNIT 2: Project Organization and Scheduling Project Elements	**15 Hrs**
Work Breakdown Structure (WBS), Types of WBS, Functions, Activities and Tasks, Project Life Cycle and Product Life Cycle. Ways to Organize Personnel, Project Schedule, Scheduling Objectives, Building the Project Schedule, Scheduling Terminology and Techniques. Network Diagrams: PERT, CPM, Bar Charts: Milestone Charts, Gantt Charts	
UNIT 3: Project Monitoring and Control	**15 Hrs**
Dimensions of Project Monitoring & Control, Earned Value Analysis, Earned Value Indicators: Budgeted Cost for Work Scheduled (BCWS), Cost Variance (CV), Schedule Variance (SV), Cost Performance Index (CPI), Schedule Performance Index (SPI) Software Reviews, Types of Review: Inspections, Deskchecks, Walkthroughs, Code Re views	
UNIT 4: Software Quality Assurance	**15 Hrs**
Concept of Software Quality, Software Quality Attributes, Software Quality Metrics and Indicators, The SEI Capability Maturity Model (CMM). SQA Activities, Formal SQA Approaches: Proof of Correctness, Statistical Quality Assurance, Product versus process quality management, Introduction, types of contracts, stages in contract, placement, typical terms of a contract, contract management, acceptance.	

Textbooks:

1. Software Project Management, Bob Hughes and Mike Cotterell, McGraw Hill

Reference Books

1. Software Project Management A Unified Framework, Walker Royce, Addison-Wesley
2. A practitioner's Guide to Software Engineering, Roger Pressman, Tata McGraw Hill 2014 8th edition.

3. Basics of Software Project Management, NIIT, Prentice-Hall India, Latest Edition

COURSE LEARNING OUTCOMES (CLO):

CLO1: Understand the foundational concepts of Software Project Management, including project lifecycle, planning, and estimation techniques.
CLO2: Apply scheduling techniques (WBS, PERT, CPM, Gantt Charts) and organize personnel effectively in a software project environment.
CLO3: Analyze and apply project monitoring methods like Earned Value Analysis (EVA) and software reviews to ensure project control.
CLO4: Evaluate software quality using CMM, quality attributes, and formal SQA approaches; understand contracts and quality metrics in software projects.

LEVEL OF CLO-PLO MAPPING TABLE

	PLOs										
Unit wise CLOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**Average (CLO)**
MMCACSP325.1	3	3	3	2	3	1	2	2	2	3	**2.4**
MMCACSP325.2	3	3	3	2	3	1	2	3	2	3	**2.5**
MMCACSP325.3	3	3	3	2	3	1	2	2	2	3	**2.4**
MMCACSP325.4	3	3	3	2	3	2	2	2	2	3	**2.5**
Average (PLO)	**3.0**	**3.0**	**3.0**	**2.0**	**3.0**	**1.25**	**2.0**	**2.25**	**2.0**	**3.0**	**2.5**

COURSE TITLE: Data Science with Python Lab

Course Code: MMCALDS325						Examination Scheme	T	P
Total number of Practical Hours: 60						External	0	36
						Internal	0	14
Lecture (L):	0	Practical (P):	2	Tutorial (T):	0	Total Credits		2

Course Learning Objectives:

- To gain hands-on experience with Python programming syntax, data types, and control structures.
- To apply functional, modular, and object-oriented programming concepts to solve computational problems.
- To utilize Python libraries such as NumPy, Pandas, Matplotlib, and SciPy for data manipulation and analysis.
- To implement and understand the basics of machine learning algorithms using Python on real-world datasets.

Practical's

Week 1

- Install Python and set up IDEs like Jupyter Notebook or VS Code
- Write a "Hello, World!" program.
- Write a program to perform basic arithmetic operations: addition, subtraction, multiplication, and division.
- Write a program to print your name and age.

Week 2

- Write a program to create variables of different data types (int, float, complex, string) and print their values.
- Write a program to perform string operations: concatenation, slicing, and repetition.
- Write a program to demonstrate arithmetic, logical, and relational operations.

Week 3

- Write a program to create a list, perform slicing, and append elements to it.
- Write a program to demonstrate the use of tuple data type and its operations.
- Write a program to find the length, maximum and minimum value of a list.

Week 4

- Write a program to demonstrate the use of if, else, and elif statements.
- Write a program to print the first 10 natural numbers using a for loop.
- Write a program to print a pattern using nested loops (e.g., a pyramid).

Week 5

- Write a program to iterate over a string, list, and dictionary using loops.
- Write a program to demonstrate the use of while loops.
- Write a program to manipulate loops using pass, continue, break, and else.

Week 6

- Write a program to to define and call a function that adds two numbers.
- Write a program to demonstrate the use of lambda functions.
- Write a program with a function that takes a list as an argument and returns the sum of all its elements.

Week 7

- Write a program to create and import a custom module.
- Write a program to to use an external library (e.g., math or random).
- Write a program to organize code into a package.

Week 8

- Write a program to define a class and create objects.
- Write a program to demonstrate inheritance.
- Write a program to show polymorphism using method overriding.

Week 9

- Write a program to perform basic array operations with numpy arrays.
- Write a program to create and manipulate DataFrame objects using Pandas.
- Write a program to draw basic plots in Python program using Matplotlib.
- Write a program to perform a basic statistical analysis using SciPy.

Week 10

- Write a program to Count the frequency of occurrence of a word in a body of text is often needed during text processing..
- Write a program to compute weighted averages in Python either defining your own functions or using Numpy.

Write a python program to calculate the mean, median, mode, variance

Week 11

- Write a program to create a normal curve using python program.
- Write a python program for correlation with scatter plot
- Write a python program to compute correlation coefficient.

Week 12

- Write a program to demonstrate Regression analysis with residual plots on a given data set.
- Write a program to demonstrate the working of the decision tree-based ID3 algorithm.

Week 13

- Write a program to implement the Naïve Bayesian classifier for a sample training data set.
- Write a program to implement k-Nearest Neighbor algorithm to classify the iris data set.

COURSE LEARNING OUTCOMES (CO):

CLO1: Students will be able to write and execute basic Python programs using fundamental concepts such as variables, data types, operators, conditional statements, and loops.

CLO2: Students will apply functions, modules, packages, and object-oriented programming principles (classes, inheritance, and polymorphism) to develop modular and reusable Python programs.

CLO3: Students will use Python libraries such as NumPy, Pandas, Matplotlib, and SciPy to perform data manipulation, visualization, and statistical analysis.

CLO4: Students will implement basic machine learning and decision-making techniques such as regression analysis and decision tree classification using Python.

CLO-PLO Matrix for the Course

Unit-Wise CLOs	PLO 1	PLO 2	PLO 3	PLO 4	PLO 5	PLO 6	PLO 7	PLO 8	PLO 9	PLO 10	Avg (CLO)
MMCALDS325.1	2	3	2	1	2	0	1	1	0	2	**1.4**
MMCALDS325.2	2	3	2	1	2	0	1	2	1	2	**1.6**
MMCALDS325.3	2	3	3	1	3	0	2	2	2	3	**2.1**
MMCALDS325.4	2	2	3	1	3	0	1	2	2	3	**1.9**
Average (PLO)	**2.0**	**2.75**	**2.5**	**1.0**	**2.5**	**0.0**	**1.25**	**1.75**	**1.25**	**2.5**	**1.75**

COURSE TITLE: Web Programming Lab								
Course Code: MMCALWP325						**Examination Scheme**	**T**	**P**
Total number of Practical Hours: 60						**External**	-	**36**
						Internal	-	**14**
Lecture (L):	-	**Practical (P):**	**2**	**Tutorial (T):**	-	**Total Credits**		**2**

Course Objectives:

- **To introduce students about the foundational concepts of web development** through hands-on experience with **HTML, CSS, JavaScript, PHP, and MySQL**.
- **To enable students about the structure and design static web pages** using HTML and CSS, including text formatting, layout design, multimedia, forms, and semantic elements.
- **To teach students to apply styling techniques** through various types of CSS, explore the box model, and implement responsive design with media queries.
- **To provide a working understanding of JavaScript for client-side interactivity**, including functions, loops, conditionals, DOM manipulation, and event handling.
- **To familiarize students with basic programming in PHP**, including variables, control structures, arrays, functions, and interaction with HTML forms.

Practical's

Week 1

- Create a basic HTML document with paragraphs, headings, and lists.
- Add images, links, and tables to an HTML document.
- Create a simple form with various form controls (text inputs, radio buttons, checkboxes, etc.).

Week 2

- Create an HTML document using semantic HTML5 elements.
- Validate an XHTML document and correct any errors.
- Convert an HTML document to XHTML and ensure it follows proper syntax rules.

Week 3

- Create a basic CSS file and link it to an HTML document.
- Apply inline, internal, and external CSS styles to a webpage.
- Use CSS selectors to style different HTML elements.

Week 4

- Style lists, links, and tables using CSS.
- Implement the box model: padding, margin, and border.

Week 5

- Create a simple webpage layout using static, relative, absolute, and fixed positioning.

- Create a responsive webpage using media queries.

Week 6

- Implement CSS animations and transactions on a webpage.
- Write a simple JavaScript program that uses variables, operators, and expressions.

Week 7

- Create JavaScript functions and demonstrate scope.
- Implement control structures (loops and conditionals) in JavaScript.

Week 8

- Create and manipulate JavaScript objects and arrays.
- Use the Document Object Model (DOM) to interact with an HTML document.

Week 9

- Write a JavaScript program to handle events on a webpage.
- Write a JavaScript program that handles errors and debugging.

Week 10

- Write a basic PHP script to output "Hello, World!".
- Create a PHP script that uses variables, constants, and operators.
- Write PHP programs using control structures (conditionals and loops).

Week 11

- Create PHP functions and work with arrays.
- Develop a simple form in PHP and handle user input.
- Connect a PHP script to a MySQL database.

Week 12

- Perform CRUD (Create, read, Update, delete) operations in PHP.
- Implement session management in PHP.

Week 13

- Use cookies in PHP to store user preferences.
- Handle errors and debug a PHP application.

Week 14

- Develop the frontend using HTML, CSS and JavaScript.
- Create the backend using PHP and integrate it with MySQL database.

COURSE LEARNING OUTCOMES (CO):

By the end of this practical course, students will be able to:

CLO1: Students will be able to create well-structured documents using paragraphs, headings, lists, tables, and forms.
CLO2: Students will demonstrate the ability to use CSS for styling text, links, tables, and layouts using box model and positioning.
CLO3: Students will write JavaScript programs using variables, functions, control structures, and arrays.
CLO4: Students will write PHP scripts with form handling, session management, and database connectivity.

CLO-PLO Matrix for the Course

Unit-Wise CLOs	PLO 1	PLO 2	PLO 3	PLO 4	PLO 5	PLO 6	PLO 7	PLO 8	PLO 9	PLO 10	Average (CLO)
MMCALWP325.1	2	3	2	1	2	1	1	2	1	2	**1.7**
MMCALWP325.2	2	3	3	1	2	1	1	3	1	2	**1.9**
MMCALWP325.3	2	3	3	1	3	1	1	3	1	3	**2.1**
MMCALWP325.4	2	3	3	1	3	2	1	2	1	3	**2.1**
Average (PLO)	**2.00**	**3.00**	**2.8**	**1.0**	**2.5**	**1.3**	**1.0**	**2.5**	**1.0**	**2.5**	**1.96**

SEMESTER-IV

PROJECT:

PROBLEM IDENTIFICATION & ANALYSIS

PROJECT:

DISSERTATION

PROJECT:

SOFTWARE DEVELOPMENT

100

PROJECT:

RESEARCH COMPONENT

Table: CLOs-PLOs Mapping Matrix for all the courses

COURSE CODE	PLO 1	PLO 2	PLO 3	PLO 4	PLO 5	PLO 6	PLO 7	PLO 8	PLO 9	PLO 10	Average CLO
MMCACJP125	2.25	2.75	2.25	1.25	2.75	1	1.5	1.75	1.25	2.75	**1.95**
MMCACML125	3.0	2.75	2.75	2.0	2.75	1.0	2.0	2.0	2.0	2.75	**2.3**
MMCADAD125	3.0	3.0	2.5	1.0	3.0	0.25	2.0	2.5	1.5	3	**2.2**
MMCADCG125	1.5	1.67	1.5	1.0	1.5	1.5	1.5	1.5	0	0	**2.79**
MMCADMI125	2.25	2.25	2.25	2.25	2.25	2.25	1.5	2.0	1.75	2.5	**2.1**
MMCADSE125	3.0	3.0	3.0	2.0	2.75	1.0	2.0	2.25	2.0	3.0	**2.4**
MMCADDS125	3.0	2.5	2.3	1.0	2.8	1.3	1.0	1.8	1.5	2.5	**1.97**
MMCADAI125	2.75	2.00	2.25	1.25	2.25	1.00	1.25	2.00	1.25	2.25	**1.82**
MMCADBC125	3.0	2.5	2.25	2.0	3.0	1.75	2.25	2.5	2.5	2.75	**2.45**
MMCADCS125	3.0	2.5	2.8	1.0	3.0	2.75	2.0	2.25	1.75	3.0	**2.4**
MMCACRM125	3.0	2.5	2.5	2.5	3.0	2.5	3	2	3	2	**2.6**
MMCALJP125	2.5	3	2.75	1.25	2.75	0.5	1.25	2	2	1.5	**1.95**
MMCALML125	3.0	2.75	2.75	2.0	2.75	1.0	2.0	2.0	2.0	2.75	**2.3**
MMCACDA225	2.8	2.5	2.5	0.25	3.0	0.25	1.0	1.5	1.75	3.0	**1.85**
MMCACMA225	1.5	1.0	1.67	1.5	2.0	1.5	1.0	1.5	1.0	0	**2.63**
MMCADAO225	1.5	1.0	1.67	1.5	2.5	2.0	1.5	1.5	0	0	**2.54**
MMCADDI225	2.5	2.5	2.5	0.75	2.75	0.5	1.5	2	1	2.75	**1.8**
MMCADDS225	3.0	2.5	2.5	2.25	2.75	1.25	2.25	2	2	2.5	**2.3**
MMCADCN225	2.75	2.75	1.75	1.25	2.5	2.25	1.25	1.5	1.5	2.5	**2**
MMCADAC225	2.75	2.25	2.25	1.75	1.75	1.25	1.5	1.25	1.25	2.25	**1.38**
MMCADCC225	2.25	2.25	2.5	0.75	2.5	1	1.5	1.75	1	2.75	**1.8**
MMCADLP225	2.75	3.0	3.0	1.25	3.0	1.75	2.0	2.75	1.25	3.0	**2.38**
MMCADTC225	1.5	1.0	1.5	1.0	1.5	1.0	1.5	1.5	0	0	**2.88**
MMCACRP225	1.75	2.0	2.0	2.25	1.75	2.75	2.25	2.75	1.75	2.75	**2.2**
MMCALMA225	1.0	1.5	1.5	1.0	1.5	1.5	1.5	1.5	0	0	**2.75**
MMCACDS325	2.25	2.5	2.75	1.25	2.5	0.25	1.75	1.5	1.75	2.75	**1.93**
MMCACWP325	2.00	3.00	2.8	1.0	2.5	1.3	1.0	2.5	1.0	2.5	**1.96**
MMCADQC325	3.0	3.0	2.5	2.0	2.5	1.75	2.5	2.25	2.25	2.25	**2.4**
MMCADEH325	3.0	2.75	2.75	2.25	2.75	2.25	2.0	1.75	1.75	2.75	**2.4**
MMCADCV325	2.3	2.5	2.0	1.0	2.8	0.0	1.0	1.8	1.8	2.8	**1.78**
MMCADER325	2.75	2.50	3.00	2.25	2.75	1.50	1.75	2.25	1.75	2.75	**2.32**
MMCADNL325	3.0	2.75	2.5	2.0	2.5	1.75	2.0	2.25	1.75	2.25	**2.28**
MMCADSQ325	2.75	2.5	3.0	1.25	3.0	2.5	2.0	2.0	2.25	3.0	**2.43**
MMCADDL325	1.5	1.0	1.67	1.5	1.5	1.0	1.5	1.5	0	0	**2.63**
MMCADIT325	2.25	1.75	2.5	0.75	2.75	1.5	1.75	2	1.75	2.75	**1.97**
MMCACSP325	3.0	3.0	3.0	2.0	3.0	1.25	2.0	2.25	2.0	3.0	**2.5**
MMCALDS325	2.0	2.75	2.5	1.0	2.5	0.0	1.25	1.75	1.25	2.5	**1.75**
MMCALWP325	2.00	3.00	2.8	1.0	2.5	1.3	1.0	2.5	1.0	2.5	**1.96**
MMCAPPI425	-	-	-	-	-	-	-	-	-	-	-
MMCAPDI425	-	-	-	-	-	-	-	-	-	-	-
MMCAPSD425	-	-	-	-	-	-	-	-	-	-	-
MMCAPRC425	-	-	-	-	-	-	-	-	-	-	-
AVERAGE PLO	**2.45**	**2.38**	**2.39**	**1.45**	**2.52**	**1.35**	**1.67**	**1.96**	**1.46**	**2.21**	**1.642**