VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

School of Computer Science and Engineering

CURRICULUM AND SYLLABI

(2025-2026)

M. Tech. (Artificial Intelligence and Machine Learning)

VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

VISION STATEMENT OF VELLORE INSTITUTE OF TECHNOLOGY

Transforming life through excellence in education and research.

MISSION STATEMENT OF VELLORE INSTITUTE OF TECHNOLOGY

- **World class Education**: Excellence in education, grounded in ethics and critical thinking, for improvement of life.
- **Cutting edge Research**: An innovation ecosystem to extend knowledge and solve critical problems.
- **Impactful People**: Happy, accountable, caring and effective workforce and students.
- **Rewarding Co-creations**: Active collaboration with national & international industries & universities for productivity and economic development.
- **Service to Society**: Service to the region and world through knowledge and compassion.

VISION STATEMENT OF THE SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

- To be a world-renowned centre of education, research and service in computing and allied domains.

MISSION STATEMENT OF THE SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

- To offer computing education programs with the goal that the students become technically competent and develop lifelong learning skill.
- To undertake path-breaking research that creates new computing technologies and solutions for industry and society at large.
- To foster vibrant outreach programs for industry, research organizations, academia and society.

VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

M. Tech. (Artificial Intelligence and Machine Learning)

PROGRAMME EDUCATIONAL OBJECTIVES (PEOs)

PEO 1: Graduates will acquire a core competency in Artificial Intelligence and Machine Learning theories, algorithms, and technologies, enabling them to design and deploy intelligent systems.

PEO 2: Graduates will pursue career as successful engineers, researchers or entrepreneurs by providing sustainable solutions across various industry domains.

PEO 3: Graduates will develop the ability to work in team adapting ethical behaviour and effective communication to succeed in their professional career.

PEO 4: Graduates will adopt a holistic approach to address the challenges in evolving technology trends.

VIT
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

M. Tech. (Artificial Intelligence and Machine Learning)

PROGRAMME OUTCOMES (POs)

PO1: Independently carry out research/investigation and development work to solve practical problems

PO2: Write and present a substantial technical report/document

PO3: Demonstrate a degree of mastery in applying the concepts of Artificial Intelligence and Machine Learning.

PO4: Analyse the complex engineering problems and design sustainable solutions

PO5: Communicate effectively at the work place and use engineering tools with ethical considerations.

PO6: Recognize the need for independent and life-long learning in the broadest context of technological change.

Category Credit Detail			
Sl.No.	**Description**	**Credits**	**Maximum Credit**
1	NC - Non Credit Course	3	3
2	UCC - University Core Courses	39	39
3	OEC - Open Elective Courses	3	3
4	PFCC - Professional Core Courses	24	24
5	PFEC - Professional Elective Courses	14	14
Total Credits		80	

Non Credit Course

sl.no	Course Code	Course Title	Course Type	Version	L	T	P	J	Credits
1	MAMAT500	Applied Statistics	Theory Only	1.0	2	1	0	0	3.0

University Core Courses

sl.no	Course Code	Course Title	Course Type	Version	L	T	P	J	Credits
1	MACSE698	Internship I/ Dissertation I	Project	1.0	0	0	0	0	10.0
2	MACSE699	Internship II/ Dissertation II	Project	1.0	0	0	0	0	10.0
3	MAENG501	Technical Report Writing	Embedded Theory and Lab	1.0	1	0	4	0	3.0
4	MASTS601	Competitive Coding I	Soft Skill	1.0	3	0	0	0	3.0
5	MASTS602	Competitive Coding II	Soft Skill	1.0	3	0	0	0	3.0

Open Elective Courses

sl.no	Course Code	Course Title	Course Type	Version	L	T	P	J	Credits
1	MASTS505	Professional Soft Skills	Soft Skill	1.0	3	0	0	0	3.0

Professional Core Courses

sl.no	Course Code	Course Title	Course Type	Version	L	T	P	J	Credits
1	MACSE501	Data Structures and Algorithm Analysis	Embedded Theory and Lab	1.0	3	0	2	0	4.0
2	MACSE502	Programming for Data Science	Embedded Theory and Lab	1.0	3	0	2	0	4.0
3	MACSE503	Data Engineering	Embedded Theory and Lab	1.0	3	0	2	0	4.0
4	MACSE504	Machine Learning	Embedded Theory and Lab	1.0	3	0	2	0	4.0

Professional Core Courses									
5	MACSE505	Deep Learning and Gen AI	Embedded Theory and Lab	1.0	3	0	2	0	4.0
6	MACSE506	Distributed Data Processing	Embedded Theory and Lab	1.0	3	0	2	0	4.0

Professional Elective Courses									
sl.no	**Course Code**	**Course Title**	**Course Type**	**Version**	**L**	**T**	**P**	**J**	**Credits**
1	MACSE515	Cybersecurity	Theory Only	1.0	3	1	0	0	4.0
2	MACSE601	Application Architecture and Deployment	Embedded Theory and Lab	1.0	3	0	2	0	4.0
3	MACSE603	Visual Analytics Lab	Lab Only	1.0	0	0	4	0	2.0
4	MACSE604	Edge Intelligence	Embedded Theory and Lab	1.0	3	0	2	0	4.0
5	MACSE605	Large Language Models	Theory Only	1.0	3	1	0	0	4.0

Course Code	Course Title	L	T	P	C
MAMAT500	Applied Statistics	2	1	0	3
Pre-requisite	NIL	Syllabus Version			
Nil		1.0			

Course Objectives

1. Understand and apply key concepts in linear algebra, probability and descriptive statistics for effective data analysis.
2. Learn the properties and applications of continuous random variables and their probability distributions.
3. Gain proficiency in common continuous distributions and hypothesis testing to make informed data-driven decisions.
4. Develop the skills to analyse and forecast time series data using various analytical techniques.
5. Apply statistical methods and perform data visualization using Excel and MATLAB for comprehensive data analysis.

Course Outcomes

1. Formulate and solve complex systems of linear equations and interpret eigenvalues and eigenvectors to analyse the behaviour of linear transformations and their applications.
2. Analyse and differentiate between various types of continuous random variables and distributions and apply probability density functions and cumulative distribution functions in real-world scenarios.
3. Design and conduct hypothesis tests using various statistical methods, evaluate statistical significance and errors, and make informed decisions based on test results.
4. Analyse time series data to identify components, transform non-stationary data to stationary, and apply advanced forecasting techniques such as ARIMA, SARIMA, and SARIMAX.
5. Synthesize and apply statistical techniques using Excel and MATLAB to perform descriptive and inferential statistics, create data visualizations, and forecast time series data effectively.

Module:1	Linear Algebra, Probability & Descriptive Statistics	5 hours

Systems of linear equations - Solving systems of linear equations - Vectors and linear transformation – Determinants and Eigenvectors - Probability Distribution Function, Probability mass function, Cumulative distribution function - Types of Probability distribution, Central limit theorem and normal distribution - Measure of Central Tendency (Mean, Median and Mode) – Co-variance and correlation, Pearson Correlation Coefficient, Spearman Rank Correlation - Measure of dispersion (variance and standard deviation);

Module:2	Continuous Random Variables	6 hours

Random Variable, Types of Random Variables: Discrete vs. Continuous - Properties and Characteristics of Continuous Random Variables - Common Continuous Probability Distributions: Uniform Distribution, Normal Distribution, Exponential Distribution - Probability Density Functions (PDFs): Definition, properties, and examples of PDFs for continuous random variables. - Cumulative Distribution Functions (CDFs): Definition, properties, and examples of CDFs for continuous random variables. - Continuous Uniform Distribution: Properties, probability density function, and applications of the continuous uniform distribution.

Module:3	Common Continuous Distributions & Hypothesis Testing	6 hours

Continuous Uniform Distribution - Normal Distribution - Log-normal Distribution - Student's T Distribution – Chi-square Distribution - z test, t test – Chi-square test - ANNOVA test - Null vs Alternate Hypothesis – Statistical Significance, P value - Type 1 and Type 2 error - Statistical tests, choosing the right test, assumptions for hypothesis testing		
Module:4	**Time Series Analysis**	**6 hours**
Time Series, Time Series vs Non-Time Series, Interpolation vs Extrapolation, Examples of time series – Components of Time Series: Trend, Season, Cycle, Noise - Moving Average, Types of Moving Average (SMA, CMA and EMA) - Stationary and Non-Stationary Time Series, Convert non-stationary to stationary test - Auto Correlation Function, Partial Correlation Function, Auto Regression - ARIMA, SARIMA, SARIMAX.		
Module:5	**Statistics Using Excel & MATLAB**	**5 hours**
Descriptive Statistics: Calculating mean, median, mode, and standard deviation using Excel formulas – Inferential Statistics: Hypothesis testing using Excel's T.TEST and F.TEST functions - Time Series Analysis: Calculating moving averages and exponential smoothing using Excel formulas - Creating time series plots and forecasting using Excel's TREND and FORECAST functions - Inferential Statistics: Hypothesis testing using MATLAB's ttest and ftest functions - Data Visualization: Creating histograms, box plots, and scatter plots using MATLAB's plotting tools (hist, boxplot, scatter).		
Module:6	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**30 hours**

Text Books	
1	J. D. Miller, Statistics for Data Science. Packt Publishing, 2017.
2	J. D. Hamilton, Time Series Analysis. Levant Books, 2012.
Reference Books	
1.	https://www.coursera.org/learn/machine-learning-linear algebra?specialization=mathematics-formachine-learning-and-data-science#modules
2.	https://www.coursera.org/learn/machine-learning-probability-and-statistics?specialization=mathematicsfor-machine-learning-and-data-science
3.	Deisenroth, M.P., Faisal, A.A. and Ong, C.S. Mathematics for machine learning. Cambridge University Press. (2020)
4.	https://www.coursera.org/learn/statistics-fundamentals-using-excel
5.	Bruce, P. and Bruce, A. Practical Statistics for data scientists: 50 Essential Concepts. 'O'Reilly Media, Inc.' (2017)

Mode of Evaluation: Quiz, Assignment, Design Project, Case Study, Seminar, CAT and FAT			
Recommended by Board of Studies			
Approved by Academic Council	**No.**	**Date**	

Course Code	Course Title	L	T	P	C
MACSE698	Internship I/ Dissertation I	0	0	20	10
Pre-requisite	NIL	**Syllabus Version**			
		1			

Course Objectives

To provide sufficient hands-on learning experience in the design, development, and analysis of appropriate products or processes.
To enhance technical skill sets in the student's chosen field of study.
To foster a research-oriented mindset through practical engagement.

Course Outcomes

Gain in-depth expertise in the major subject or field of study, including comprehensive understanding and critical insight into ongoing research and development trends.
Develop the ability to approach complex problems holistically, enabling critical, independent, and creative thinking in identifying, formulating, and addressing multifaceted challenges.
Cultivate a strong awareness of the ethical dimensions associated with research and development activities, promoting integrity and responsible conduct in scholarly and professional work.
Enhance research communication and dissemination skills, including the ability to effectively present findings through technical writing, oral presentations, and scholarly discussions.
Achieve recognition through publications in peer-reviewed journals and international conferences, which serve as valuable indicators of research quality and impact.

General Guidelines

(PROJECT DURATION - One Semester): - The dissertation may encompass a wide range of scholarly activities, including theoretical analysis, modeling and simulation, experimental investigation, prototype design, equipment fabrication, data correlation and analysis, software development, or applied research. It is expected to be an individual effort, demonstrating the student's independent capability in their area of specialization. The work may be conducted either within the university or externally, such as in a relevant industry or research institution. While not mandatory, publications resulting from the dissertation in peer-reviewed journals or international conferences will be considered a valuable addition, reflecting the quality and impact of the research undertaken.

Mode of Evaluation : Project Reviews, Project Reports, VivaVoce/Student Interactions

Recommended by Board of Studies :	02-06-2025
Approved by Academic Council : No. 78	12-06-2025

Course Code	Course Title	L	T	P	C
MACSE699	Internship II/ Dissertation II	0	0	20	10
Pre-requisite	NIL	**Syllabus Version**			
		1			

Course Objectives

To provide hands-on learning experience in the design of appropriate products or processes.
To develop practical skills related to the development and analysis of engineering solutions.
To enhance technical competencies in the student's chosen field through experiential learning.

Course Outcomes

Formulate well-defined problem statements for complex, real-world scenarios by applying appropriate assumptions and constraints.
Conduct comprehensive literature and patent searches to establish a strong foundation and context for the identified problem.
Design and implement experimental or analytical approaches, including iterative solution development, and systematically document findings.
Perform critical evaluations such as error analysis, bench marking, and cost estimation to assess the feasibility and efficiency of proposed solutions.
Synthesize results to derive meaningful scientific conclusions or practical solutions, and effectively communicate the outcomes through technical reports and presentations.

General Guidelines

(PROJECT DURATION - One Semester): - The dissertation may encompass a wide range of scholarly activities, including theoretical analysis, modeling and simulation, experimental investigation, prototype design, equipment fabrication, data correlation and analysis, software development, or applied research. It is expected to be an individual effort, demonstrating the student's independent capability in their area of specialization. The work may be conducted either within the university or externally, such as in a relevant industry or research institution. While not mandatory, publications resulting from the dissertation in peer-reviewed journals or international conferences will be considered a valuable addition, reflecting the quality and impact of the research undertaken.

Mode of Evaluation : Project Reviews, Project Reports, VivaVoce/Student Interactions

Recommended by Board of Studies :	02-06-2025
Approved by Academic Council : No. 78	12-06-2025

Course Code	Course Title	L	T	P	C
MAENG501	Technical Report Writing	1	0	4	3
Pre-requisite	NIL	**Syllabus Version**			
		1			

Course Objectives

To develop communicative competence in students.
To apply advanced technical communication principles in practical writing tasks for diverse professional contexts.
To execute research and analytical skills to produce coherent and evidence-based technical documents through practical exercises.

Course Outcomes

Apply the principles of effective report writing.
Demonstrate the ability to draft and present technical reports.
Produce key components of technical documents, including sections focusing on purpose, audience, and structure, through writing assignments.
Create technical summaries and abstracts by practicing techniques for summarizing and paraphrasing complex information.

Module:1	**Fundamentals of Technical Writing**	**3 hours**

Introduction to Technical Writing: Definition and typical forms (reports, instructions, proposals); Key Factors in Technical Writing: Purpose, Audience, and Tone; General Writing Basics: Clarity, fluency, effectiveness; The Process of Writing: Pre-writing, Writing, and Post-writing stages; Organization in Writing: Having an outline, using introductions, headings, lists, figures, and summaries.

Module:2	**Technical Grammar**	**2 hours**

Concord; Tense Shifts

Module:3	**Introduction to Reports and Report Structure Front Matter**	**2 hours**

Reports in Organizations: Role and importance of reporting in corporate and industrial segments.; Purpose of Reports: Conveying decisions, facts, and information accurately and up-to-date.; Report Structure Overview: Division into Front Matter, Main Body, and Back Matter; Components of the Front Matter

Module:4	**Report Structure Main Body and Back Matter**	**4 hours**

Components of the Main Body: Introduction, Discussion or Description, Conclusions, and Recommendations; Writing Introduction, Discussion or Description, Conclusions and Recommendations, Methods of Reporting: The Letter Method and The Schematic Method; Routine Reports: Nature, frequency, and function in organizations (often statistical, fixed intervals).		
Module:5	**Technical Proposals**	**2 hours**
Punctuation Right words and phrases; avoiding cliches, jargons, foreign words and phrases, ambiguity, redundancy, circumlocution - Developing hints		
Module:6	**Contemporary Issues**	**2 hours**
Guest Lecture		
	Total Lecture Hours:	**15 hours**
Text Book(s)		
Kumar. S & Pushplata. , **"Effective Communication Skills. "**, New Delhi OUP, **2018** Muralikrishna and Sunita Mishra, **"Communication Skills For Engineers"**, Pearson, 2nd Edition, **2011** Shirley Mathew., **"Effective Communication Skills."**, Nirali Prakashan, **2025**		
Reference Books		
Indicative Experiments		
1. **Introduction to the Technical Writing** Introduction to Technical Report Writing**;** Analyzing and identifying the characteristics of effective technical writing in sample documents.; Short writing exercises focusing on clarity, conciseness, and identifying purpose, audience, and tone in simple technical scenarios. **Activity:** Reviewing and providing feedback on short technical descriptions.		2 hours
2. **Analyzing Audience and Context for Practical Writing** Applying reader and stakeholder analysis techniques to specific engineering report scenarios; Drafting content segments tailored for different technical and non-technical audiences (e.g., writing an executive summary for managers vs. a technical description for fellow engineers); Simulating audience needs assessment based on given project descriptions. **Activity:** Developing a 'reader profile' for a major report project.		4 hours
3. **Technical Grammar and Style in Practice**		2 hours

Intensive practical exercises on complex grammatical structures, sentence syntax, and common errors in technical writing; Exercises in applying principles of clarity, conciseness, and precision to improve technical sentences and paragraphs; Practicing the appropriate use of active and passive voice in different report sections. **Activity:** Peer-editing session focusing on grammar, mechanics of writing, and technical style.	
4. **The Writing Process and Report Outlining Lab** Practicing prewriting techniques for complex technical topics: brainstorming, mind mapping, and systematic outlining; Developing detailed hierarchical outlines for a major technical report project, including main and sub-points; Planning content organization based on report type (e.g., feasibility, empirical research, lab report) and audience needs. **Activity:** Group exercise to compare and refine report outlines.	4 hours
5. **Writing the Report Front Matter and Introduction** Detailed practical guidance on drafting all components of the report Front Matter; Creating a professional Title Page, Forwarding Letter/Preface, Acknowledgements, and Table of Contents for the ongoing report project; Drafting effective Abstracts and Summaries (Executive Summaries) based on provided technical content, focusing on capturing the essence. **Activity:** Writing the Introduction section of the report project, including background, scope, and objectives.	4 hours
6. **Writing the Report Body: Data, Descriptions, and Discussion** Techniques for presenting data, technical descriptions, and analysis in the main body; Drafting sections of the report body focusing on presenting organized data and technical details; Writing the "Discussion" section, focusing on interpreting results and explaining findings based on provided or self-generated data. **Activity:** Integrating data and analysis points into the draft of the report body.	4 hours
7. **Writing the Report Body: Conclusions and Recommendations** Practical methods for drawing logical conclusions and formulating actionable recommendations; Drafting the "Conclusions" section based on the data and	4 hours

discussion from Module 6; Developing clear and practical recommendations based on the conclusions, considering the report's purpose and audience. **Activity:** Writing the "Recommendations" section of the report project.	
8. **Transcribing Visuals** Using charts, graphs and tables; Transcribing visuals that are clear, accurate, and effectively support the report's text; Integrating created visuals into the report draft, ensuring proper placement, captions, and referencing within the text. **Activity:** Peer review focusing on the effectiveness and integration of visuals.	2 hours
9. **Report Back Matter, Condensation, and Final Review** Practical session on creating the Back Matter: Appendices, Bibliography, Glossary, and Index (if applicable). Emphasis on consistent citation and referencing styles; Compiling Appendices and formatting a Bibliography for the report project; Practicing techniques for précis writing and summarization to condense longer texts. **Activity:** Final review of the complete report draft, focusing on overall structure, flow, formatting, and coherence.	2 hours
10. **Presentation Skills** Short Presentations, Formal Presentation with PPT Analytical Presentation of Charts, Graphs and Tables **Activity**: Presentations – Individual and Group	2 hours
Total Laboratory Hours:	**60 hours**
Mode of Evaluation : Continuous Assessment Test, Digital Assignment, Quiz, Final Assessment Test, Lab Continuous Assessment, Lab Final Assessment, Presentaion	
Recommended by Board of Studies :	16-05-2025
Approved by Academic Council : No. 78	12-06-2025

Course Code	Course Title	L	T	P	C
MASTS601	Competitive Coding I	3	0	0	3
Pre-requisite	NIL	Syllabus version			
		1.0			

Course Objectives

Equip learners with an in-depth understanding of linear data structures (linked lists, stacks, queues, deques, etc.). Enabling them to design, implement, and optimize these structures for various problem-solving scenarios. Emphasis is placed on both theory (complexities, operations) and hands-on coding to master data structure manipulation and usage.

Course Outcomes

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

● Confidently implement and utilize linked lists, stacks, queues, and deques

● Analyse and optimize the time/space complexities of linear data structure operations (insertion, deletion, traversal)

● Apply problem-solving strategies to real-world tasks and competitive coding challenges, leveraging linear data structures effectively

● Integrate debugging, testing, and best coding practices when working with linear data structures

Module:1	Linked List-1	6 hours

Introduction to Linked Lists: Definition, structure, advantages; Types of Linked Lists: Singly, Doubly, Circular.

Basic Operations: Creating a linked list; Insertion (at beginning, middle, end); Deletion (from beginning, middle, end); Traversal (iterative, recursive).

Module:2	Linked List-2	6 hours

Advanced Topics: Reversing a linked list (iterative, recursive); Detecting and removing loops; Merging and splitting linked lists; Intersection of two linked lists.

Module:3	Stack-1	5 hours

Introduction to Stack: Definition, LIFO property, real-world applications (e.g., undo functionality, browser history); Static vs dynamic stacks.

Operations on Stack: Push, pop, peek; Checking stack overflow and underflow.

Implementation: Using arrays; Using linked lists.

Module:4	Stack-2	4 hours

Applications of Stack: Balancing parentheses in expressions; Converting infix expressions to postfix; Evaluating postfix expressions; Solving backtracking problems (e.g., maze solving).

Module:5	Queue	7 hours

Introduction to Queue: Definition, FIFO property; Types of Queues: Simple, Circular, Double-Ended.

Operations on Queue: Enqueue (insertion), Dequeue (removal), Peek (front element); Checking queue overflow and underflow.

Implementation: Using arrays; Using linked lists.

Applications of Queue: Task scheduling (e.g., CPU scheduling, print queue); Breadth-First Search (BFS) implementation; Managing resources in real-time systems (e.g., job queues).

Module:6	Priority Queue	5 hours

Introduction to Priority Queue: Definition, importance in task prioritization; Differences between Priority Queue and Simple Queue.

Operations on Priority Queue: Enqueue and Dequeue operations based on priority.

Implementation: Using arrays; Using heaps (Binary Heaps: Min-Heap and Max-Heap).

Module:7	Hash map	9 hours
Introduction to Hash map: Definition, key-value pair structure, advantages over arrays. Hash Functions and Collision Handling: Direct addressing, simple hash functions, modulo operation; Collision resolution techniques: Chaining, Linear Probing, Quadratic Probing. Applications of Hash map: Frequency counting (e.g., counting occurrences of elements); Caching mechanisms (e.g., implementing LRU cache); Finding duplicates in arrays or strings; Solving anagram problems; Building indexes for fast lookup.		
Module:8	**Interview Prep - Networking**	**3 hours**
Networking Fundamentals: OSI model TCP/IP basics Network Protocols: HTTP/HTTPS DNS and DHCP FTP and SMTP - Practical Interview Questions: - Debugging network issues, Firewall, DNS, Ping and Trace route - Explaining client-server architecture - Designing scalable systems		
Total Lecture hours:		**45 hours**
Text Book		
1.	A Textbook of Data Structures and Algorithms 1: Mastering Linear Data Structures – 1st Edition by G A Vijayalakshmi Pai – Wiley Publication	
Reference Books		
1.	Data Structure and Algorithms Made Easy by Narasimha Karumanchi -5th edition	

Mode of Evaluation: Written assignment, Quiz, Project & FAT.			
Recommended by Board of Studies			
Approved by Academic Council		Date	

Course Code	Professional Soft skills	L	T	P	C
		3	0	0	3
Pre-requisite		Syllabus Version			
Nil		1.0			

Course Objectives

1. Develop a positive attitude, emotional intelligence and master the importance of body language to improve workplace interactions and develop effective stress management techniques.
2. Master the techniques to neutralize accents and improve clarity and enunciation for Enhancing presentation skills, listening and questioning skills.
3. Foster teamwork and ethical behaviour by understanding team dynamics and developing interpersonal skills to work effectively in diverse and virtual teams.
4. Understand the essentials of effective customer service and structured approaches to problem-solving and decision-making processes.
5. Gain proficiency in email etiquette and professional communication and acquire foundational knowledge of accounting systems and financial management principles.

Course Outcomes

1. Acquire positive attitude, emotional intelligence, and effective stress management, facilitating a smooth transition from campus to corporate life.
2. Exhibit clear and effective communication skills by neutralizing accents, improving clarity and enunciation, and mastering presentation, listening, and questioning skills.
3. Work effectively in diverse and virtual teams, understanding team dynamics and fostering ethical behaviour and inclusivity.
4. Apply structured problem-solving and decision-making approaches and deliver exceptional customer service in various scenarios.
5. Improve proficiency in email etiquette and professional communication, along with foundational knowledge of accounting systems and financial management principles.

Module:1	Positive Attitudeand Body Language	9 hours

Attitude- Campus to Corporate attitude change, Recognizing Negative Attitude, Campus to Corporate attitude change; Attitude at work- Impact of Negative Attitude in the Workplace, Overcoming Negative Attitude, positive attitude, thought process, Building self-confidence and Assertiveness; Toxic positivity; 3Es, Motivation-Intrinsic and Extrinsic Motivation, Inspiration vs motivation; Emotional Intelligence-Intro to EI, Four clusters. Transactional Analysis (TA), SWOT analysis - Professional analysis. Importance of Body Language, Five Cs of Body Language, Body language in different cultures, Positive Body Language; Voice Control- Pace. Pause and Pitch; Culture-Inclusivity and Proxemics across Global Cultures, Understanding POSH; Stress
Management-What is Stress, Eustress, Reasons of stress (work/ personal); Stress Management Techniques.

Module:2	Accent Neutralization,Presentation,Listening & Questioning skills	9 hours

Identifying and dealing with Mother Tongue Influence (MTI) – Pronunciation - Vowel Sounds and Consonant Sounds – Inflection – Pausing - Reducing rate of speech -
Volume and tone – Pitch – Clarity - and enunciation. Self-introduction – Exercises, Why Give Presentations; Craft your message-Plan the visuals, Manage the

Response; How to create an effective presentation - Virtual & Physical, Do's & Don'ts of Presentation Skills, Objection handling, Stage Fear – Causes and Cure, Practice the Delivery; Time Management-Common Time & Energy Wasters, Planning & Prioritizing Time Matrix & Analysis. Barriers to effective listening - how to overcome them; Exercises - Customer Call Flow – Role-play, Customer calls amongst the team; How to frame Questions, Different kinds of questions, asking appropriate questions; Spoken English - Introduction to Parts of Speech and its usage; Subject - Verb Agreement; Basic conversation skills-sentence construction – SVO.		
Module:3	**Teamwork**	**9 hours**
Teamwork and Ethics - Definition of TEAM - Team vs Groups. Difference b/w Healthy competition and cut throat competition, Importance of working in teams, Evolution of a TEAM, Benefits of team work; Virtual teams- Challenges and ways to overcome it, Diversity and Inclusion in a team; Development of Teams Stages of team development; Team dynamics-its importance & Interpersonal Skills Development Ethics- to enable students to identify and deal with ethical problems, develop their moral intuitions, which are implicit in everyday choices and actions; Conflict Management: Team building Activities- Predetermined/ Predesigned Indoor/ Outdoor activities to build a team, enhance language and inter personal skills		
Module:4	**Customer Service and Problem Solving and Decision Making**	**8 hours**
Customer Service - Different types of customers - Difference between customer service and customer experience - Telephone Etiquette - Handling difficult customers. Define a Problem - Define Decision Making- Blocks in problem solving - Stereotyping and unconscious biases – The process of Problem Solving and decision making - Problem Analysis- Decision Analysis - Potential Problem / Opportunity Analysis - Creative Thinking - Problem Solving process - Implementation of the solution		
Module: 5	**Business Email Etiquette and Chat, Financial Principles and Practices**	**8 hours**
Emails Etiquette: Share format/ signature - Emails etiquette - dos and don'ts. Accounting systems and how transactions are recorded - Financial statements: Profit & Loss account - balance sheet - cash flow statement - Fixed assets - depreciation and the capitalization of software development expense – Working capital and cash management - Using ratio analysis to assess corporate health and performance - Funding the business: equity - debt and other aspects - Budgeting & Forecasting – capex – apex - Designing a flexible budget - Capital expenditure appraisal and approval		
Module:6	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**45 hours**
Text Books		
1	Kerry Patterson, Joseph Grenny, Ron McMillan and Al Switzler, (2021).3rd Edition, Crucial Conversations: Tools for Talking when StakesareHigh. McGraw-Hill Contemporary, Bangalore	
2	Jon Gordon (2021) The Power of Positive Leadership: How and Why Positive Leaders Transform Teams and Organizations and Change the World"by Wiley 2021	
3.	English Phonetics and Phonology a Practical Course by Peter Roach · 2009, Cambridge University Press	
4.	Business Communication: Process and Product" by Mary Ellen Guffey and Dana Loewy (2020, 13th Edition)	
Reference Books		

1.	Dale Carnegie, (2016). How to Win Friends and Influence People. Gallery Books, New York.
2.	General Stanley McChrystal, David Silverman, Tantum Collins, Chris Fussell. (2015) Team of Teams: New Rules of Engagement for a Complex World Penguin Books Limited,
3.	Brené Brown (2018), Dare to Lead: Brave Work. Tough Conversations. Whole Hearts." By Random House.
4.	David Allen, (2015). Getting Things done: The Art of Stress-Free productivity. Penguin Books, New York City
5.	Philip Carr (2021) English Phonetics and Phonology: Theory and Practice" byWiley-Blackwell.
6.	ttps://www.udemy.com/course/understanding-accent-neutralization/
7.	https://www.coursera.org/learn/problem-solving
8.	https://1library.net/document/zxv5lv5n-recommendation-principles-good-practices-financial-education-awareness.html
4.	https://www.skillsyouneed.com/present/presentation-tips.html
5.	https://www.simplypsychology.org/transactional-analysis-eric-berne.html
6.	https://www.mindtools.com/pages/article/newTMM_30.htm
Mode of Evaluation: Quiz, Assignment, Design Project, Case Study, Seminar, CAT and FAT	

Recommended by Board of Studies			
Approved by Academic Council	**No.**	**Date**	

Course Code	Course Title	L	T	P	C
MACSE501	Data Structures and Algorithm Analysis	3	0	2	4
Pre-requisite	Nil	Syllabus Version			
		1			

Course Objectives

To familiarize the concepts of data structures and provide a mathematical framework for the design and analysis of algorithms.
To provide a deeper insight into linear and non-linear data structures.
To disseminate knowledge on how to develop algorithms using various design strategies to solve real-world problems.

Course Outcomes

Develop the ability to analyse the efficiency of algorithms using asymptotic notations and implement linear data structures.
Apply tree and graph-based data structures to design and implement efficient traversal, search, and pathfinding algorithms.
Demonstrate the ability to design and implement efficient solutions to complex problems using divide-and-conquer and greedy strategies.
Formulate optimal and exhaustive search-based solutions for computationally intensive problems using dynamic programming, backtracking, and branch-and-bound technique.
Critically analyse computational complexity, differentiate between P, NP, and NP-complete problems, and apply approximation algorithms to develop near-optimal solutions for intractable problems.

Module:1	Complexity Analysis and Elementary Data Structures	8 hours

Overview and importance of algorithms and data structures- Algorithm specification, Asymptotic Notation - The Big-O, Omega and Theta
notation, Recursion, Performance analysis, Array, Stack, Queue, Linked-list and its types, various representations, operations and applications of Linear Data Structures.

Module:2	Non Linear Data Structures	10 hours

Trees: Binary trees, Properties of binary trees, Binary Search Tree, AVL, Red-Black Trees, B-Trees, Segment Trees - Heaps: Binary heaps, Min-Max heap – Graphs: Representation, Breadth first Search, Depth First Search, Shortest path algorithm –Bellman Ford algorithm - Network flow algorithm: Ford-Fulkerson and Push-relabel algorithm.

Module:3	Divide and Conquer and Greedy Techniques	8 hours

Divide and Conquer: Merge Sort, Quick Sort, Karatsuba's fast multiplication method, Strassen's Matrix Multiplication - Greedy Strategy: Job Sequencing Problem with Deadlines, Minimum Spanning Tree: Prims algorithm, Kruskal's algorithm, Huffman coding.		
Module:4	**Dynamic Programming and Backtracking Branch and Bound Techniques**	**9 hours**
Dynamic programming: Matrix Chain Multiplication, Longest Common Subsequence - Backtracking: N-Queens problem, Subset Sum problem, Graph coloring problem Branch and Bound: Subset sum, A-Star, LIFO-BB and FIFO BB methods.		
Module:5	**Complexity Classes and Approximation Algorithms**	**8 hours**
Class P - Class NP - Reducibility and NP-completeness - Circuit Satisfiability problem-3CNF, Independent Set, Clique, Approximation Algorithms: Vertex Cover, Set Cover and Travelling salesman.		
Module:6	**Contemporary Issues**	**2 hours**
Total Lecture Hours:		**45 hours**
Text Book(s)		
Cormen, Thomas H., Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. , "**Introduction to Algorithms** ", MIT Press, **2022** Allen, Weiss Mark. , "**Data structures and algorithm analysis in C++**", Pearson Education India, **2007**		
Reference Books		
Skiena, Steven S. , "**The Algorithm Design Manual (Texts in Computer Science)**", Springer, 3rd Edition, **2020** Jon Kleinberg and EvaTardos, "**Algorithm Design**", Pearson Education, 1st Edition, **2014** Brass, Peter., "**Advanced data structures**", Cambridge University Press, **2008**		
Indicative Experiments		
1. Linear data structures: stack, queue, Linked lists In the situation where there are multiple users or a network computer system, you probably share a printer with other users. When you request to print a file, your request is added to the print queue. When your request reaches the front of the print queue, your file is printed. This ensures that only one person at a time has access to the printer and that this access is given on a first-come, first served basis. Design and implement the algorithm for this scenario.		4 hours

2. Non-linear data structures: Trees & Graphs You are making an iPod playlist to hear the songs. Assuming that shuffle functions are not applicable, choose an appropriate data structure that will add and delete songs onto you're your iPod in such a way that the recently inserted song will always be the first song currently on the iPod.	4 hours
3. Graph algorithms: BFS, DFS, network flow You are developing a diagnostic tool for a large, complex communication network (e.g., an internet service provider's infrastructure, or a peer-to-peer file-sharing network). The network shown below is an undirected, unweighted graph, where devices are nodes (routers, servers, end-points) which are connected directly by edges. Nodes: A, B, C, D, E, F, G, H, I, J Edges: A-B, A-C B-D, B-E C-E, C-F, D-G, E-H, F-I, G-J, H-J, I-J. Implement functionalities primarily using Breadth-First Search (BFS) with required assumptions to answer the following queries: • Given a source_device and a target_device, find the minimum number of hops (direct connections) required to establish a path between them. If no path exists, indicate so. • Given a starting_device and a max_hops limit, determine all devices that can be reached from the starting_device within max_hops. This could represent the reach of a network broadcast or a virus. • Given a specific pair of source_device and target_device, identify all "bridge devices" along every shortest path between them. A device is a "bridge device" in this context if its removal would increase the shortest path distance between the source_device and target_device or disconnect them entirely. This requires careful application or multiple runs of BFS. • Given a starting_device, find the diameter of the connected component to which it belongs. The diameter is the longest shortest path between any two nodes in that component.	4 hours
4. Trees: AVL, Red-Black tree You are tasked with designing the core data structure for a large e-commerce platform's product catalog. Products are organized in a strict multi-level hierarchy (e.g., Electronics -> Laptops -> Gaming Laptops -> BrandX Gaming Laptop Model Y). Each product has a unique product_ID, a price, and a	4 hours
quantity_in_stock. The catalog is highly dynamic, with frequent additions, removals, price changes, and stock updates. Implement a robust and efficient tree-based data structure (e.g., a balanced Binary Search Tree like an AVL Tree or Red-Black Tree, where nodes can represent categories or individual products) to manage this hierarchical product catalog. Your implementation must support the CRUD operations.	

5. Divide & Conquer: Merge sort, Quick sort, Karatsuba's fast multiplication method You have n coins, all of which are gold except one coin which appears to be a gold coin, but it is fake. All gold coins are of the same weight, the fake coin weighs less than the others. You have a balance scale, you can put any number of coins on each side of the scale at one time and it will tell you if the two sides weight the same, or which side is lighter if they don't weight the same. The problem is to identify the fake coin. Design an algorithm to find the fake coin in the given n coins.	4 hours
6.Dynamic programming: MCM, LCS A chain of matrices A is called as Add-multiply-chain(AM-chain) if all the matrices of the chain are sqaure matrices of the same size and both addition & multiplication operations are involved in the chain. For example, A = A1 ∗A2 ∗A3 +A4 ∗A5 ∗A6 ∗A7+A8 ∗A9 ∗A10 is an AM-chain. The above AM-chain A is written as (A, 10, 3, 7), which conveys that the chain A has 10 matrices, with two '+'operations, one '+'operation after A3, another '+' operation after A7, and multiplication is the operation in all the required places in the chain. Multiplication (or addition) between any two numbers is called a scalar operation. The number of scalar operations involved in the chain A1A2 + A3 is 12, where all three matrices are square matrices of size 2. (A, n, i, j) is a chain A with n matrices with a '+' operation after Ai, another '+' operation after Aj, and all the other operations are multiplication. Given an AM-chain (A, n, i, j), design a pseudocode and implement it to parenthesize the chain A such that the number of scalar operations to evaluate the chain A is minimum.	2 hours
7.Backtracking: N-Queens problem, Subset Sum, Graph Coloring You have a standard 8x8 chessboard (if your are not familiar with game of chess, please get to know), empty but for a single knight on some square. Your task is to generate a series of legal knight moves that result in the knight visiting every square on the chessboard exactly once. In addition, the knight	2 hours
must end on a square that is one knight's move from the beginning square. The output of your program should indicate the order in which the knight visits the squares, starting with the initial position. Generalize your program for an n x n board where n>8.	

8.Branch & Bound: A-Star, Subset Sum You are developing the pathfinding core for a logistics company using autonomous delivery vehicles in a dense urban environment. The city map is represented as a grid, where each cell has an associated "traffic cost" (representing time delay due to congestion, road conditions, etc.). Some cells may be temporarily designated as "restricted zones" or "accident sites" with extremely high or infinite costs. Vehicles can move horizontally, vertically, and diagonally. You need to find the most cost-effective path from a starting depot to a delivery destination. Implement the A* search algorithm to find the minimum-cost path from a start_cell (start_row, start_col) to a target_cell (target_row, target_col) within a given M x N integer grid.	2 hours
9.Approximation Algorithm: Vertex Cover, Set Cover You are a network security engineer responsible for monitoring a critical communication network. The network consists of various communication nodes (e.g., servers, routers, switches) and the direct communication links between them. To detect potential intrusions or failures, you need to install monitoring agents on a subset of these nodes. An agent on a node can monitor all direct communication links connected to that node. Your goal is to select the minimum number of nodes to place agents on, such that every communication link in the network is monitored. However, finding the absolute minimum is a known NP-hard problem for large networks, so an efficient approximation is required. Implement a 2-approximation algorithm for the Minimum Vertex Cover problem.	4 hours
Total Laboratory Hours:	**30 hours**
Mode of Evaluation : Continuous Assessment Test, Final Assessment Test	
Recommended by Board of Studies :	23-05-2025
Approved by Academic Council : No. 78	12-06-2025

Course Code	Course Title	L	T	P	C
MACSE502	Programming for Data Science	3	0	2	4
Pre-requisite	NIL	Syllabus Version			
		1.0			

Course Objectives

1. Master Python data structures and object-oriented programming for data analysis and web development.
2. Dive deep into the Pandas libraryfor data processing, modelling and visualization.
3. Understand Scala functional programming concepts, data structures, object-oriented programming and exception handling.
4. Explore Scala libraries, Apache Spark architecture and its functions for big data processing and analytics.
5. Learn criteria for selecting the appropriate programming language.

Course Outcomes

1. Demonstrate proficiency in implementing complex data structures and user-defined data structures in Python.
2. Analyse and manipulate data using advanced features of Pandas
3. Evaluate the use of Scala's case classes, companion objects, and traits in building robust applications.
4. Design and deploy scalable data processing pipelines using Scala libraries and Apache Spark.
5. Select the most suitable programming language based on the project requirements.
6. Apply Python and Scala programming skills to design, develop, and deploy real-world applications.

Module:1	Python Data Structures	8 hours

Condition and Branching Statements, Built- in data structures: List, Tuple, Dictionary, Set, User defined data structures: Stack, Queue, Priority Queue, String handling methods, Exception Handling, Object-Oriented Concepts, APIs and Data Collection, Simple API and REST APIs- HTTP Requests, File Handling- Read/Write Frameworks
and Libraries, NLTK, ChatterBot

Module:2	Python Libraries	9 hours

Pandas-Series, DataFrame, Handling Missing Values, Built-in functions, Data Operations, Filtering Data in DataFrame, Data Extraction, Working with Text Data, Merging DataFrames - Data Mining – Scrapy - Beautiful Soup - Data Processing and Modelling: NumPy - SciPy - Pandas - Keras - Scikit - Learn - PyTorch -TensorFlow,
XGBoost, Data Visualization: Matplotlib – Seaborn – Bokeh – Plotly - Folium.

Module:3	Scala Data Structures and Object-Oriented Programming	12 hours

Expanded Function Format, Variables and Strings, Getting user input, Numbers, Variable types, Operators, Booleans Data Structures-Arrays, Lists, Tuple, Sets, Hash set, Maps - Functional Combinators Map, Scala Object and Class, Anonymous object-Singleton, Companion Object-Case Classes, Objects-Constructors-Method Overloading - This Keyword – Inheritance - Field Overriding - Final, Abstract Class-
Trait, Trait Mixins, Access Modifiers, Scala Array-REPL

Module:4	Scala Libraries and Spark Basics	8 hours

Scala Libraries- Breeze, saddle, Exception Handling, Apache Spark Architecture - Spark Big Data - Apache Spark Components.		
Module:5	**Programming language selection criteria**	**6 hours**
Size of the Deployment: Data, Resource and Load - Security - Skill Set - The targeted platform - The elasticity of a language - The time to production - The performance - The support and community – Purpose – Programmer experience - Ease of Development and Maintenance - Efficiency - Availability of an IDE -Error Checking and Diagnosis		
Module:6	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Textbooks	
1	Alvaro Fuentes, Become a Python Data Analyst – By Packt Publishing (2018)
2	Bharti Motwani, Data Analytics using Python – By Wiley (2020)
3	Jules S. Damji, Learning Spark: Lightning-Fast Data Analytics, Second Edition – By Shroff/O'Reilly (2020)
4	Data Science and Machine Learning using Python – 10 August 2022, MGH, 2022
Reference Books	
1.	Tome, E., Bhattacharjee, R. and Radford, D. Data Engineering with Scala and Spark: Build streaming and batch pipelines that process massive amounts of data using Scala. Packt Publishing Ltd. (2024)
2.	Perrin, J.-G. Spark in Action, Second Edition: Covers Apache Spark 3 with Examples in Java, Python, and Scala. Manning. (2020)
3.	Wes McKinney, Python for Data Analysis: Data Wrangling with pandas, NumPy, and Jupyter, OReilly, (2022)
4.	https://www.coursera.org/learn/python-for-applied-data-science-ai
5.	https://www.datacamp.com/blog/top-python-libraries-for-data-science
6.	https://www.udemy.com/course/completescala3/?couponCode=IND21PM
7..	https://www.aalpha.net/blog/factors-to-consider-when-choosing-a-programming-language/
Mode of Evaluation: Quiz, Assignment, Design Project, Case Study, Seminar, CAT and FAT	
List of Experiments (Indicative)	
1.	Basic Data Manipulation with Pandas: Create a DataFrame with columns Name, Age, and City containing data for five individuals. Perform the following operations: select only the Name and Age columns, filter rows where Age is greater than 25, add a new column Country with a default value, and sort the DataFrame by Age in descending order. Finally, calculate the average age of the individuals.
2.	Data Cleaning and Preprocessing with Pandas: Create a DataFrame with some missing values in columns Name, Age, and City. Perform the following operations: fill missing values in the Age column with the mean age, drop rows where Name or City is missing, and convert the Age column to integer

	type. Finally, normalize the Age column using min-max scaling.
3.	Creating and Manipulating Arrays: Using NumPy, create a 2D array of shape (3, 4) with random integers between 0 and 10. Perform the following operations: calculate the mean and standard deviation of the entire array, slice the array to get the first two rows and last two columns, reshape the array to shape (4, 3), and perform element-wise multiplication with another array of the same shape.
4.	Feature Engineering, Exploratory Data Analysis: Create a DataFrame with columns Feature1, Feature2, and Target containing random data. Perform the following operations: create a new feature that is the logarithm of Feature1, bin Feature2 into three categories (low, medium, high), and calculate the correlation matrix of the DataFrame. Finally, create a scatter plot of Feature1 vs. Feature2 colored by Target, and interpret any visible patterns.
5.	Scrape data from a webpage and store it in a structured format like CSV or JSON.
6.	Interactive Bar Chart with Plotly. Create an interactive bar chart showing the population of different cities.
7.	Interactive Scatter Plot with Plotly. Create an interactive scatter plot showing the relationship between house size and price.
8.	Interactive Line Plot with Bokeh. Create an interactive line plot showing the daily temperatures over a week using Bokeh.
9.	Interactive Bar Chart with Bokeh: Problem Statement: Create an interactive bar chart showing the sales figures of different products using Bokeh.
10.	Interactive Scatter Plot with Bokeh: Problem Statement: Create an interactive scatter plot showing the relationship between petal length and petal width from the Iris dataset using Bokeh.
11.	Visualizing Clusters with scikit-learn and Plotly Problem Statement: Perform K-means clustering on the Iris dataset and visualize the clusters using an interactive 3D scatter plot in Plotly.
12.	Visualizing PCA with scikit-learn and Plotly: Problem Statement: Perform Principal Component Analysis (PCA) on the Iris dataset and visualize the first two principal components using an interactive 2D scatter plot in Plotly.
13.	Introduction to arrays in Scala: Create an array of integers with the values 2, 5, 9, 14, 20. Write a function that takes this array and returns the sum of its elements. Next, create an array of strings with the names of five different fruits. Write a function that concatenates all elements of this array, separated by commas. Finally, iterate over the array of integers and print each element to the consol.
14.	Understanding lists, sets, and tuples in Scala: Create a list of the first five prime numbers. Write a function that takes this list and returns a new list with each element squared. Then, create a set of unique characters from the string "hello world" and write a function that takes two sets and returns their intersection. Create a tuple with three elements: an integer, a string, and a boolean, then access and print each element.

15.	Handling collisions and resolving conflicts in HashMap's: Create a HashMap with keys representing student names and values representing their grades, then insert multiple entries including a duplicate key with a different grade. Write a function to merge two HashMap's, resolving conflicts by taking the higher grade. Finally, write a function to handle collisions by chaining, using lists to store multiple values for a single key.
16.	Exploring advanced functional combinators beyond the basic set: Create a list of integers from 1 to 10, then use filter to create a new list with only even numbers. Use map to create a new list where each element is multiplied by 3. Use flatMap to create a list of tuples where each integer is paired with its square. Finally, use foldLeft to calculate the product of all elements in the list.
17.	Case Classes: Create a case class Person with fields name, age, and city. Create a list of Person objects representing five different individuals. Write a function to filter out people older than 30. Next, write a function that groups people by their city. Then, write a function that transforms each Person object into a string in the format "Name is Age years old and lives in City". Finally, write a function to sort the list of people by age in ascending order.
18.	Use Saddle to load and manipulate a dataset of fruit prices and quantities, filtering for apples and calculating their average price. You will then visualize the price trend over time using Vegas. This exercise covers basic data manipulation with Saddle and interactive visualization with Vegas.
19.	Use Breeze library to calculate total revenue for each fruit type by multiplying price and quantity from a dataset. You will then create a bar chart to visualize these revenues using Plotly. Scala. This exercise introduces numerical computations with Breeze and dynamic visualizations with Plotly. Scala.
20.	Analyse fruit prices using Spire for precise statistics and Saddle for data manipulation. You will compute the mean and variance of prices for each fruit and visualize these trends over time using Vegas. This exercise demonstrates the use of Spire for numerical precision, advanced data manipulation with Saddle, and effective visualization with Vegas.

Total hours:			**30 hours**
Mode of Evaluation: Continuous Assessments and FAT			
Recommended by Board of Studies			
Approved by Academic Council	**No.**	**Date**	

Course Code	Course Title	L	T	P	C
MACSE503	Data Engineering	3	0	2	4
Pre-requisite	Nil	Syllabus Version			
		1.0			

Course Objectives

1. Understand and apply different data types and formats for efficient data management and analysis.
2. Develop the skills to implement data ingestion techniquesto ensure smooth data flow into data processing systems.
3. Learn exploratory data analysis and create visual representations for decision making.
4. Learn suitable storage and retrieval methods.
5. Understand and implement data lineage analysis.

Course Outcomes

1. Identify various data types and formats, and develop strategies for effective data type conversion, transformation, and serialization.
2. Analyse and implement data ingestion techniques and design solutions
3. Perform advanced data profiling and exploratory data analysis to uncover insights and support data-driven decision-making.
4. Design and evaluate efficient data storage and retrieval methods.
5. Develop comprehensive data lineage analysis frameworks to track data flow through ETL processes, data warehouses, and edge intelligence.
6. Apply data engineering knowledge to solve real-world data management challenges and optimize data-driven workflows.

Module:1	Data Types and Formats	4 hours

Data Types and Formats - Types of Data - Structured vs. Unstructured Data - Formats of Data - Semi-Structured Data - Data Type Conversion and Transformation
- Data Serialization - Choosing the Right Data Type and Format – Tools and Technologies for Data Types and Formats.

Module:2	Data Ingestion Techniques	10 hours

Data Ingestion - Streaming Data Ingestion - Batch Data Ingestion - Hybrid Data Ingestion - Data Ingestion vs. Data Integration - Data Ingestion Challenges - Tools and Solutions for Data Ingestion - StreamSets, DataOps Platform - Benefits of Data
Ingestion - Data Ingestion Framework.

Module:3	Data Profiling and Visual Representation	12 hours

Data Profiling and Visualization - Exploratory Data Analysis with Pandas - Data Analytics with Python - Business Intelligence Tools - Application of Data Analytics - Retrieving and Cleaning Data , Types of Missing Data and Handling Techniques - Feature Engineering - Inferential Statistics and Hypothesis Testing - Descriptive Statistics - Types of Descriptive Statistics - Concepts of Populations, Samples, and Variables - Statistical Methods for Describing Data Characteristics - Real-World Applications of Descriptive Statistics

Module:4	Storage and retrieval methods	9 hours

Storage and Retrieval - Types of Data and Storage Methods - Local vs. Distributed Storage & Retrieval – Hardware Aspects of Storage & Retrieval - Choosing Storage
Methods - Data Partitioning and Sharding - Data Replication and Redundancy - Data

Compression and Encoding - Data Archiving and Retrieval - Backup and Disaster Recovery – Data Lifecycle Management.		
Module:5	**Data Lineage Analysis**	**8 hours**
Data Lineage Analysis - Building a Data Flow - ETL (Extract, Transform, Load) Process - Usage of Data Warehouse - Edge Intelligence in Data Flow - Data Lineage - Benefits of Data Lineage - Data Lineage Tool Features.		
Module:6	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Books	
1	C. M. Judd, Data Analysis: A Model Comparison Approach To Regression, ANOVA, and Beyond, 3rd ed. New York, NY, USA: Routledge, 2017.
2	P.-Y. Bonnefoy, E. Chaize, R. Mansuy, and M. Tazi, The Definitive Guide to Data Integration, 1st ed. Birmingham, UK: Packt Publishing, 2024.
Reference Books	
1.	Esppenchutz, G., Data Ingestion with Python Cookbook: A practical guide to ingesting, monitoring, and identifying errors in the data ingestion process. Packt Publishing Ltd. (2023)
2.	Molin, S. Hands-On Data Analysis with Pandas. http://scholarvox.library.inseec-u.com/catalog/book/88872627. (2019)
3.	https://intellipaat.com/blog/tutorial/hadoop-tutorial/big-data-overview/#no2
4.	https://www.cuemath.com/data/descriptive-statistics/
5.	https://towardsdatascience.com/why-you-should-handle-missing-data-and-heres-how-to-do-it-270c321a4d6f
6.	Dr Reema Thareja, Data Science and Machine Learning using Python, McGraw Hill (2022).
Mode of Evaluation: Quiz, Assignment, Design Project, Case Study, Seminar, CAT and FAT	
List of Experiments (Indicative)	
1.	To create different file formats such as CSV, JSON, AVRO, Parquet, XML and HDF5 files with some data in it.
2.	Identify and classify different data types and formats.
3.	Migrating data from one db type to another db.
4.	Data Storage Challenge to design and implement a data storage and retrieval solution for a given scenario.
5.	Installation and setup of Pentaho.
6.	Data ingestion using Pentaho.
7.	Preprocessing data and batch data ingestion using Pentaho.
8.	Descriptive statistics and its basics using Excel.
9.	Visualization of categorical variables using python.
Total hours:	**30 hours**
Mode of Evaluation: Continuous Assessments and FAT	
Recommended by Board of Studies	
Approved by Academic Council	**No.** **Date**

Course Code	Course Title	L	T	P	C
MACSE504	Machine Learning	3	0	2	4
Pre-requisite	NIL	Syllabus Version			
		1.0			

Course Objectives

1. Develop a comprehensive understanding of supervised learning algorithms, their applications and limitations.
2. Gain proficiency in unsupervised and semi-supervised learning techniques,
3. Learn the fundamentals of reinforcement learning (RL and its applications
4. Understand and apply various dimensionality reduction techniques to improve data processing and visualization efficiency.
5. Master the skills of model evaluation, hyperparameter tuning and cross-validation to enhance model performance and generalization.

Course Outcomes

1. Explain the fundamental concepts of machine learningtechniques and their applications.
2. Analyse and synthesize unsupervised and semi-supervised learning techniques to solve complex data problems.
3. Evaluate different reinforcement learning algorithms, including policy-based, value-based, and model-based methods.
4. Develop applications that leverage dimensionality reduction methods to enhance data visualization and analysis.
5. Implement comprehensive model evaluation metrics.
6. Design end-to-end machine learning pipelines for real-world applications.

Module:1	Supervised Learning Algorithms	11 hours

Supervised Learning Algorithms - Classificationalgorithms:Logistic Regression, KNN, SVM, Decision Tree, Random Forest – Regression Algorithms: Linear Regression, Polynomial Regression, Multiple Linear Regression, Applications- Spam detection,
Image recognition, Challenges, limitations: Overfitting and imbalanced data

Module:2	Unsupervised and Semi-Supervised Learning Algorithms	10 hours

Unsupervised Learning: Types, Clustering Algorithms: K-Means Clustering, Hierarchical Clustering- Association Rule Learning: Apriori Algorithm - FP-Growth Algorithm -Evaluation metrics- Semi supervised Learning: Self-Training Algorithm - Label Spreading Algorithm - Label Propagation Algorithm – Naive Bayes - Support Vector Machines - Graph-Based Learning, Applications and Challenges of
unsupervised and semi-supervised learning algorithms

Module:3	Reinforcement Learning (RL)	8 hours

Policy-Based RL: Applications of Policy - Policy Gradient Methods - Policy Optimization Methods -Value-Based RL - Value Functions - Q-Learning - Model-Based RL - Model-based RL with model-free RL - Model-based RL with planning

Module:4	Dimensionality Reduction	6 hours

Overview of Dimensionality Reduction, Principal Component Analysis (PCA): Variants and applications- Locally Linear Embedding Algorithm (LLE): t-SNE,

Principles - Applications, limitations		
Module:5	**ML Model Evaluation and Hyperparameter Tuning**	**8 hours**
Classification and Regression Metrics and its Interpretation- Model Evaluation: regression models, binary classification models, multi-class classification models, Performance Metrics, Scikit-learn metrics, Cross Validation and its Types: schemes, Estimating the model generalization error with CV- Hyperparameter Tuning - Group Cross-Validation.		
Module:6	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Textbooks	
1	L. Goel, Artificial Intelligence: Concepts and Applications. Hoboken, NJ, USA: Wiley, 2021.
2	A. Muller, Introduction to Machine Learning with Python: A Guide for Data Scientists, 1st ed. Sebastopol, CA, USA: Shroff/O'Reilly, 2016.
Reference Books	
1.	Jo, T., Machine learning foundations: Supervised, Unsupervised, and Advanced Learning. Springer Nature. (2021)
2.	Dangeti, P. *Statistics for machine learning*. Packt Publishing Ltd. (2017)
2.	https://app.pluralsight.com/library/courses/building-classification-models-scikit-learn/table-of-contents
3.	https://www.udemy.com/course/artificial-intelligence-reinforcement-learning-in-python/
4.	https://www.udemy.com/course/data-science-in-python-regression/?couponCode=24T4FS22124
Mode of Evaluation: Quiz, Assignment, Design Project, Case Study, Seminar, CAT and FAT	
List of Experiments (Indicative)	
1.	Design a decision tree to classify cancer tumors (malignant/benign) and evaluate feature importance of transparent decision-making.
2.	Use a Random Forest model to handle class imbalance in credit card fraud detection, leveraging ensemble methods to enhance prediction performance.
3.	Demonstrates how linear regression and polynomial models with regularization predict housing prices and reduce prediction errors through hyper parameter tuning.
4.	Apply multiple linear regression to forecast demand by analyzing relationships between environmental and seasonal factors, with a focus on improving model accuracy.
5.	Demonstrates the application of K-Means to segment customers based on spending habits and income, with visualization techniques for better cluster interpretation.
6.	Group different iris flower species using hierarchical clustering and dendrograms, focusing on improving cluster interpretability.
7.	Apply the FP-Growth algorithm to discover frequent item sets and association rules in transactional data, aiding decision-making in retail analytics.

8.	Investigates the use of self-training and co-training algorithms to classify partially labeled text datasets, showcasing the power of semi-supervised learning.			
9.	Explore the efficacy of graph-based label propagation methods to classify nodes in citation networks, emphasizing its application in network data analysis.			
10.	Implement of Principal Component Analysis (PCA) to reduce the complexity of MNIST digit image data while retaining significant variance for better computational efficiency.			
11.	Use t-SNE to project complex high-dimensional data (Swiss Roll) into a 2D/3D space, revealing hidden structures and interpretable patterns.			
12.	Evaluating Regression models using regression metrics (MAE, RMSE) to assess model performance and the quality of predictions.			
13.	Implement nested and time-series cross-validation to ensure model robustness and minimize overfitting on temporal data.			
14.	Benchmark the performance of Reinforcement algorithms like Policy based and Model based, Q-Learning and Deep-Q on standard datasets, providing insights into computational efficiency, accuracy, and resource usage.			
Total hours:				**30 hours**
Mode of Evaluation: Continuous Assessments, FAT				
Recommended by Board of Studies				
Approved by Academic Council	**No.**	**Date**		

Course Code	Course Title	L	T	P	C
MACSE505	Deep Learning and Gen AI	3	0	2	4
Pre-requisite		Syllabus Version			
		1.0			

Course Objectives

1. Gain a comprehensive understanding of neural networks and deep learning techniques
2. Learn the architecture and applications of convolutional neural networks (CNNs)
3. Develop expertise in recurrent neural networks (RNNs) and their variants like LSTM and GRUs for applications in time series predictions
4. Implement transformer models and attention mechanisms for advanced applications Explore generative models such as VAEs and GANs and their applications

Course Outcomes

1. Explain the working mechanisms of deep learning models
2. Design and Evaluate CNN architectures for diverse practical applications.
3. Analyse and optimize RNNs for sequential data processing tasks.
4. Create and fine-tune transformer models, addressing scalability and ethical considerations.
5. Develop and optimize generative models for various creative and scientific applications.
6. Build and apply deep learning models to solve real-world problems

Module:1	Deep Learning	7 hours

Deep Learning Architecture, Forward Propagation, Back propagation, Loss Functions and Regularization Techniques, Applications: Computer Vision, NLP, Audio, Finance, Healthcare, Anomaly Detection Mathematics for Neural Networks, Deep Learning Frameworks: TensorFlow, PyTorch, Keras.

Module:2	Convolutional Neural Networks	10 hours

Convolutional Neural Network Operations, CNN Architecture, Overfitting in CNNs, variants of CNN Models: AlexNet, VGGNet, GoogleNet, ResNet, Training and Fine-tuning CNNs, Evaluation and Optimization of CNNs, Understanding Gradient Descent, Advanced Optimizers, Regularization Techniques, Hyperparameter Tuning, Applications: Medical Image Analysis, Autonomous Vehicles and Robotics, Video Analysis and Event Detection, Object Detection and Image Segmentation.

Module:3	Recurrent Neural Networks	8 hours

RNN Architecture, Types of RNNs: Vanilla RNNs, LSTM, GRUs, Bidirectional RNNs. Attention Mechanisms, GRUs Optimization, Regularization and Hyperparameter Tuning, Diagnosing Performance Issues, Applications: NLP, Time Series, Health Monitoring, Speech Recognition, Video Analysis.

Module:4	Transformers and Attention Models	8 hours

Attention Mechanism, Attention in Sequence-to-Sequence Models, Transformer Architecture, Training and Fine-Tuning Transformers, Transformers in Computer Vision and Audio Processing, Advanced Optimizers and Training Techniques, Quantization and Pruning, Multi-modal Transformers, Transformers for Reinforcement Learning and Generative Tasks, Scalability and Efficiency, Ethical

Considerations and Bias.		
Module:5	**Generative Models**	**10 hours**
Generative and Probabilistic Models, Variational autoencoders (VAE), Generative Adversarial Networks (GAN), Evaluation Metrics and Ethical Considerations, Training Techniques for GANs, Adversarial Training and Defence Mechanisms, Integration with Reinforcement Learning, Applications- Image/Video Generation, Text Synthesis, Music/Audio, Drug Discovery, Marketing.		
Module:6	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Books	
1	C. C. Aggarwal, Neural Networks and Deep Learning. Springer International Publishing AG, 2023.
2	J. Babcock and R. Bali, Generative AI with Python and TensorFlow 2. Packt Publishing, 2021.
Reference Books	
1.	https://app.pluralsight.com/library/courses/implement-image-recognition-convolutional-neuralnetwork/table-of-contents.
2.	https://www.udemy.com/course/deep-learning-convolutional-neural-networks-theanotensorflow/?couponCode=IND21PM.
3.	https://www.udemy.com/course/nlp-with transformers/?couponCode=24T4FS22124.
4.	Jo, T. Deep learning foundations. Springer Nature. (2023).
5.	Liang Zhao, Graph Neural Networks: foundations, frontiers, and applications. Springer, https://doi.org/10.1007/978-981-16-6054-2, (2022).
6.	Elgendy, M. Deep learning for vision systems. Manning. (2020).
Mode of Evaluation: Quiz, Assignment, Design Project, Case Study, Seminar, CAT and FAT	
List of Experiments (Indicative)	
1.	Implement a Feedforward neural network to classify handwritten digits, focusing on understanding the basic architecture and training process using MNIST Handwritten Digits Dataset (28x28 grayscale images).
2.	Compare the performance of Feedforward Neural Networks (FNN) and Backpropagation Neural Networks (BPNN) in terms of training efficiency and accuracy using MNIST Handwritten Digits Dataset.
3.	Develop a CNN to classify images of clothing items, emphasizing the role of convolutional layers, pooling, and feature extraction in image processing using Fashion-MNIST Dataset.
4.	Prepare text data using Shakespeare Text Corpus or song lyrics datasets, build RNN models like LSTM, BiSTM and GRU and train them. Generate sequences to demonstrate the ability of these models to model sequential data.
5.	Explore the key concepts of transformers, including attention mechanisms, by designing and training a transformer model for text generation using Custom text dataset or OpenWebText Corpus (like GPT training data).

6.	Fine-tune a pre-trained BERT model to perform sentiment analysis, focusing on transfer learning and domain adaptation techniques using IMDB Movie Review Dataset (50,000 labeled reviews).
7.	Design and train a transformer model from scratch for sentiment analysis, showcasing its flexibility and performance on text classification task by using IMDB Movie Review Dataset.
8.	Explore the use of large language models like GPT(Generative Pre-trained Transformer) for generating creative stories and summaries, experimenting with different prompt configurations and summarization techniques by considering Custom creative writing datasets or open datasets like CNN/DailyMail for summarization tasks.
9.	Create a chatbot trained on diverse topics and a conversational AI model for hotel booking, experimenting with pretrained models for text classification, sentiment analysis, and named entity recognition (NER) by considering MultiWOZ 2.2 Dataset (multi-domain conversational dataset).
10.	Explore AI-powered tools for generating, completing, and translating code, providing hands-on experience with state-of-the-art models using OpenAI Codex or custom code-related datasets.
11.	Design a recommendation system for movies based on user preferences and implement Retrieval-Augmented Generation (RAG) for legal document retrieval based on queries using MovieLens Dataset (for recommendations) and CaseLaw Dataset (for legal document retrieval).
12.	Explore various prompt types, including template prompts, cloze-type prompts, and query-based prompts, to enhance performance on NLP tasks such as language translation, topic modeling, and classification using IMDB Movie Review Dataset or other NLP benchmark datasets like WMT (for translation).

Total hours:			**30 hours**
Mode of Evaluation: Continuous Assessments and FAT			
Recommended by Board of Studies			
Approved by Academic Council	**No.**	**Date**	

Course Code	Course Title	L	T	P	C
MACSE506	Distributed Data Processing	3	0	2	4
Pre-requisite		Syllabus Version			
		1.0			

Course Objectives

1. Gain knowledge onthe architecture, task scheduling andfault tolerance, for deploying and managing Spark installations.
2. Gain proficiency in installing, configuring and troubleshooting PySpark and effectively use DataFrames and RDDs for data processing.
3. Master Spark SQL functions, joins and partitioning techniques to perform efficient data aggregation and optimization.
4. Impart skills to create and manage streaming DataFrames and DataSets, Explore Spark applications using Kubernetes, manage observability and implement continuous integration and deployment best practices.

Course Outcomes

1. Understand the architecture and lifecycle of Spark applications to optimize deployment, task scheduling, and fault tolerance across different deployment modes.
2. Create and configure efficient PySpark environments for data processing and analysis, integrating DataFrames and cluster management.
3. Evaluate advanced Spark SQL functions, joins, and partitioning techniques to optimize data processing and querying performance.
4. Develop real-time data processing applications using Structured Streaming, optimizing techniques.
5. Deploy and manage Spark applications on Kubernetes and other cluster managers.
6. Build and optimize Spark solutions for real-world data challenges.

Module:1	Spark Architecture	8 hours

Driver Program - Executor Nodes - Driver - Executor Communication - Task Scheduling - Executor Failures – Spark application lifecycle - Job scheduling - Task execution - Result aggregation - Fault tolerance - Overview of Spark deployment options - Local mode - Standalone mode - YARN mode - Mesos mode - Kubernetes mode - Cluster mode -Driver mode - Installing Spark on a cluster - Installing Spark on
a single machine - Best practices for deploying and managing Spark installations.

Module:2	PySpark Essentials	8 hours

Installing and Configuring PySpark - Setting up Spark Environment Variables - Configuring Spark Logging – Troubleshooting PySpark Installation - DataFrames - Pandas Integration - DataFrame Operations - DataFrames vs. RDDs- DataFrames and Data Warehousing– Cluster Mode– Driver Mode- spark - submit - Spark Cluster
Managers - Monitoring and Debugging.

Module:3	Spark SQL Hands-On	9 hours

Spark SQL functions - String functions - Numeric functions - Date and timestamp functions - Aggregate functions – User defined functions in Spark SQL - Joins in Spark SQL - Types of joins - Joining multiple tables - Joining tables with different data types - Handling null values in joins - Optimizing join performance in Spark SQL
- Aggregation in Spark SQL - Aggregate functions - Grouping data - Filtering aggregated data - Aggregate functions with windowing - Aggregating data with

DataFrames and DataSets - Partitioning in Spark SQL - Partitioning data - Partitioning schemes - Dynamic partitioning - Partition pruning and optimization - best practices for partitioning in Spark SQL.		
Module:4	**Structured Streaming**	**9 hours**
Creating Streaming DataFrames and DataSets, Joining Streaming Data, Handling Late and Out-of-Order Data, Basic Aggregation Deduplication - Reasons and Methods-Handling Duplicates - Windowing in StructuredStreaming-Specifications and Functions-Watermarks-Best Practices for Windowing and Watermarks-State Store Management-Using State Store for Aggregation, Handling State Timeouts and Expiration, Best Practices-Output Modes in Structured Streaming-Types of Output Modes and Sinks-Custom Sinks Best Practices for Choosing Output Modes and Sinks.		
Module:5	**Deploying and Managing Spark Applications**	**9 hours**
Kubernetes and containerization - Deploying Spark applications on Kubernetes - Spark Deployment Modes- Creating a Kubernetes cluster for Spark deployments - Configuring Spark applications for Kubernetes - Best practices for Kubernetes-based Spark deployments -Spark Connect - Using Spark Connect for data ingestion - Integrating Spark Connect with data sources - Advanced Spark Connect topics - Best practices for using Spark Connect - Observability in Spark applications - Monitoring Spark applications - Logging and log aggregation in Spark - Distributed tracing in Spark: Open Tracing and Jaeger - Alerting and notification systems for Spark applications - Best practices for observability in Spark applications - Change management in Spark applications - Handling code changes: spark-submit and spark-shell - Managing dependencies and versioning in Spark applications - Rolling back changes in Spark applications – Continuous Integration and Continuous Deployment for Spark applications - Best practices for handling changes in Spark applications.		
Module:6	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Books	
1.	J. S. Damji, Learning Spark: Lightning-Fast Data Analytics, Second Edition. Shroff/O'Reilly, 2020.
2.	G. Maas and F. Garillot, Stream Processing with Apache Spark. O'Reilly Media, 2019.
3.	A. Sarkar, Learning Spark SQL. Packt Publishing, 2017.
Reference Books	
1.	https://www.tutorialspoint.com/pyspark/index.htm
2.	https://spark.apache.org/sql/
3.	https://spark.apache.org/docs/latest/structured-streaming-programming-guide.html
4.	Perrin, J., Spark in Action, Second Edition: Covers Apache Spark 3 with Examples in Java, Python, and Scala. Manning. (2020).
5.	Kukreja, M., &Zburivsky, D. Data Engineering with Apache Spark, Delta Lake, and Lakehouse: Create scalable pipelines that ingest, curate, and aggregate complex data in a timely and secure way. Packt Publishing Ltd. (2021).

Mode of Evaluation: Quiz, Assignment, Design Project, Case Study, Seminar, CAT and FAT
List of Experiments (Indicative)

1.	Install, configure, and benchmark Apache Spark for optimal performance. Identify and troubleshoot common installation issues.			
2.	Run PySpark jobs, compare DataFramesvs RDDs, and submit a Spark job using spark-submit. Monitor and debug the execution.			
3.	Optimize Spark SQL queries by using joins, aggregations, and partitioning strategies to improve performance.			
4.	Integrate Apache Kafka with Spark Streaming to ingest and process real-time data efficiently.			
5.	Implement state management and different output modes (append, update, complete) in Spark Structured Streaming.			
6.	Enhance Spark performance using caching, broadcast variables, and check pointing. Measure and compare execution time before and after optimizations.			
7.	Train and evaluate machine learning models using Spark MLlib, specifically for regression and classification tasks.			
8.	Use GraphX to process graph data, implement PageRank, and find connected components in a dataset.			
9.	Package and deploy a Spark application in cluster mode. Apply best practices for monitoring and troubleshooting Spark jobs.			
10.	Design and implement an end-to-end data pipeline that integrates both batch and streaming workloads using Spark.			
Total hours:				**30 hours**
Mode of Evaluation: Continuous Assessment and FAT				
Recommended by Board of Studies				
Approved by Academic Council	**No.**	**Date**		

Course Code	Course Title	L	T	P	C
MACSE601	Application Architecture and Deployment	3	0	2	4
Pre-requisite	Nil	Syllabus Version			
		1.0			

Course Objectives

1. Understand different software architectures, focusing on monolithic and microservices.
2. Explore APIs, including web and local APIs, REST and SOAP, API testing and security mechanisms.
3. Learn about virtualization, containerization, Dockerarchitecture, Docker commands and orchestration tools.
4. Gain knowledge on Kubernetes, its components, resources and various objects like Config Maps, Secrets and YAML files.
5. Understand MLOps, different software development life cycle models, continuous integration/deployment and monitoring tools.

Course Outcomes

1. Evaluate the differences between monolithic and microservices architecture in AI applications.
2. Analyse the role of application programming interfaces in integrating AI models into larger systems.
3. Utilize Containerization and Virtualization to design, deploy, and manage secure scalable applications.
4. Deploy AI models in a production environment on Kubernetes
5. Create efficient MLOps pipelines to streamline the machine learning lifecycle.
6. Evaluate the stages of the Software Development Life Cycle and create effective strategies for model training and deployment.

Module:1	Architectures to build and organize Applications	6 hours

Software Architecture and its types - Monolithic Architecture and its Importance - Characteristics of Monolithic Architecture - Limitations of Monolithic Architecture - Microservices - Working of Microservices - Main Components of Microservices Architecture - Advantages of Microservices - Monolithic vs Microservices - Real World Example of Microservices - Challenges in Microservices. HLD AND LLD architectures.

Module:2	Application Programming Interface	7 hours

API - WEB APIs - LOCAL APIs - PROGRAM APIs - SOAP, REST API - REST APIs - HTTP methods: GET, POST, PUT, DELETE- Status Codes and URI structure - SOAP vs REST - API testing - Types of Testing - Tools for API Testing – Authentication Mechanisms - Authorization Mechanisms - Role Based Access Control, Flask.

Module:3	Containers	12 hours

Virtualization in Cloud Computing - Containerization - Container Lifecycle - Virtualization vs Containerization - Container Security - Serverless Containers – Docker and its Architecture - Components of Docker - Concept of Docker Images - Docker Commands - Advantages of Docker - Orchestration tools: Docker Swarm

Module:4	Kubernetes	12 hours

Kubernetes -Kubernetes and not only docker - Kubernetes Components - Node - Control Plane - Networking in Kubernetes - Kubernetes Resources - Pod, Deployment, Service, Volume, Namespace, node, cluster - Storage - Objects: Config

Maps and Secrets -Security - Monitoring, Logging, Scaling - Writing YAML files.		
Module:5	**ML Operations**	**6 hours**
ML Operations - Popular MLOps tools -SDLC - Stages of SDLC - Waterfall Model - Agile Model - Iterative Model - Importance of Each Models - Model Training - Model Deployment- Model Monitoring Data Drift-Continuous Integration and Continuous Deployment Prometheus - Grafana.		
Module:6	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Textbooks	
1	Scott Surovich& Marc Boorshtein, Kubernetes and Docker – By Packt Publishing (2021).
2	Mark Treveil, Nicolas Omont& Clément Stenac, Introducing MLOps: How to Scale Machine Learning in the Enterprise (Grayscale Indian Edition) – By Shroff/O'Reilly (2020).
Reference Books	
1.	Newman, S. Building microservices. 'O'Reilly Media, Inc.' (2021).
2.	Singh, P., Deploy machine learning models to production: With Flask, Streamlit, Docker, and Kubernetes on Google Cloud Platform. Apress. (2020).
3.	https://www.atlassian.com/microservices/microservices-architecture/microservices-vs-monolith
4..	https://blog.postman.com/intro-to-apis-history-of-apis/
5.	https://www.udemy.com/course/grafana-and-prometheus/?couponCode=24T4FS22124
Mode of Evaluation: Quiz, Assignment, Design Project, Case Study, Seminar, CAT and FAT	
List of Experiments (Indicative)	
1.	Creating a basic flask application and getting Welcome message through CMD curl command.
2.	Creating an application to receive input of 2 numbers from url,request body and provide mean and multiplication of those numbers as a response. use API Individual testing tools like Thunder client/Postman.
3.	Docker installation.
4.	Build a custom image for flask and publish it on a Container repository.
5.	Run and Test a Simple Flask Application with Docker (Run command and compose file).
6.	Deploy a Simple Nginx Web Server.
7.	Exposing web server using service - clusterIP, NodePort and LoadBalancer.
8.	DeployMySQL in Kubernetes with persistent storage to ensure data persistence and durability.
9.	Implement Pod Security Policies to enforce security best practices for the web server Pod running in the Kubernetes cluster.
10.	Deploying a Flask application with Statefulset, PVs and Ingress.

11.	Deploying one webserver as frontend and 2 flask application as microservice where 2 flask applications will communicate with each other using Kubernetes Individual services.			
12.	Developing a ML model and deploying it using docker and Kubernetes.			
13.	Using the setup from Experiment 13, deploy machine learning models as a third Flask application.			
14.	Create a ML Model deploy it using k8s and monitor it using prometheus and Grafana for drift and other parameters.			
Total hours:				**30 hours**
Mode of Evaluation: Continuous Assessments and FAT				
Recommended by Board of Studies				
Approved by Academic Council		**No.**	**Date**	

Course Code	Visual Analytics Lab	L	T	P	C
MACSE603		0	0	4	2
Pre-requisite	NIL	Syllabus version			
		1.0			

Course Objectives

1. Understand the fundamentals of data analysis and visualization using Power BI.
2. Impart the knowledge on working with large datasets to perform advanced manipulations and analysis.
3. Familiarize custom visualizations and dashboard development.
4. Deploy Applications using Docker and Kubernetes.

Course Outcomes

1. Create and manage Power BI projects, including data sources, datasets, and reports.
2. Implement advanced data analysis techniques using DAX and time intelligence functions.
3. Design interactive and informative dashboard using Power BI visualizations.
4. Develop web applications to integrate with Power BI dashboard.
5. Deploy applications using Docker and Kubernetes.
6. Apply critical thinking and problem-solving skills to data analysis and visualization

Indicative Experiments

1.	Setting up environment and connecting to data source : Install Power BI Desktop- SQL Server - Loading and previewing data.
2.	Data cleansing and transformation in Power BI : Apply transformations to data-load queries to data model with sales data set
3.	Designing a data model in Power BI : Creating model relationships- configuring table and column properties and creating hierarchies using sales data set.
4.	Calculated tables using Data analysis expressions:Creating calculated tables-calculated columns and measures (To answer questions like top sold product-average sales of product category.)
5.	Advanced DAX Calculations in Power BI : Use the CALCULATE () function to manipulate filter context - Use Time Intelligence functions for sales data (Example: calculate the total sales revenue for each product category- calculate YoY growth trend of a product category)
6.	Design a report in Power BI : Create and publish a report with sales data and publish the report with basic visualizations
7.	Design a report in Power BI : Enhancing the report with advanced visualizations and features interactive elements to the dashboard- such as filters- slicers- and drill-down capabilities- to allow users to explore the data in more detail.
8.	Create a dashboard in Power BI : Create a dashboard and share the dashboard through colExperimentoration features in Power BI-colExperimentorate with others on the dashboard- including adding comments and feedback- to improve its content and usability.
9.	Visual analytics using HTML- CSS : Create a basic structure for our dashboard-including a header navigation- and a main content area. Use CSS to style the frontend- including colours- fonts- and layout.
10.	Create Charts and graphs: Use a charting library eCharts to create interactive

	charts and graphs that display sales data. Include a variety of chart types- such as line charts- bar charts- and pie charts- to provide different perspectives on the data.			
11.	End point creation: Use Flask to create a backend API that connects to a data source Excel file and SQL database containing sales data. Create endpoints to expose data required for charts using pandas and dataframes.			
12.	Dashboard integration: Integrate the front end and backend to get data from backend. Add interactive elements to the dashboard- such as filters- slicers-and drill-down capabilities- to allow users to explore the data in more detail. Customize the dashboard's appearance- including colors- fonts- and layout- to make it visually appealing and easy to read.			
13.	Create a basic application using streamlit and also Creating multipage app for sales dashboard using streamlit.			
14.	Containerize Workload – Docker: Containerize the frontend and backend application using docker. Push the containerized images to a container registry.			
15.	Kubernetes deployment: Deploying the containers to Kubernetes cluster as a deployment.Using services to connect backend deployment to data source (SQL)			
Total Laboratory Hours				**60 hours**
Text Books				
1.	Definitive Guide to DAX, The: Business intelligence for Microsoft Power BI, SQL Server Analysis Services, and Excel, Marco Russo, Alberto Ferrari, Microsoft press, 2nd edition, 2019.			
Reference Books				
1.	Microsoft Power BI Cookbook - Third Edition: Convert raw data into business insights with updated techniques, use cases, and best practices Paperback -by Greg Deckler, Brett Powell, Packt Publishing; 3rd ed. edition (31 July 2024)			
2.	Introducing Microsoft Power Bi by Marco Russo and Alberto Ferrari, Microsoft Press (2016)			
Mode of assessment: **Continuous Assessments and FAT**				
Recommended by Board of Studies		DD-MM-YYYY		
Approved by Academic Council		No. xx	Date	DD-MM-YYYY

Course Code	Edge Intelligence	L	T	P	C
MACSE604		3	0	2	4
Pre-requisite		Syllabus Version			
		1.0			

Course Objectives

1. Examine the evolution and benefits of Edge Computing in contrast to Cloud Computing.
2. Understand the components and infrastructure of Edge AI.
3. Learn the architecture and integration of Multi Access Edge Computing (MEC) with mobile networks.
4. Explore real-time video analytics and edge-based solutions for video processing.
5. Analyse the benefits of edge-based log processing.

Course Outcomes

1. Analyse the implications of edge computing in various sectors.
2. Create efficient edge AI solutions by designing and deploying AI models on edge devices.
3. Evaluate the architecture and components of Multi Access Edge Computing.
4. Develop and optimize edge-based video analytics models for real-time applications.
5. Create edge-based log processing solutions.
6. Build edge computing solutions using AI, video analytics, and log processing to solve real-world problems.

Module:1	Edge Computing	8 hours

Contrasting Edge Computing with Cloud Computing - Historical Context and Evolution - Industry Trends and Adoption Patterns - Implications for Various Sectors: Healthcare, Manufacturing, Retail - Latency Reduction – Bandwidth Optimization - Data Sovereignty and Privacy - Advantages- Challenges, Standardization.

Module:2	Edge AI - Architectural Components	8 hours

Edge AI - Relationship with Machine Learning and Artificial Intelligence - On-device Processing - Edge Devices and Sensors - Edge Computing Infrastructure - AI Models and Algorithms - Types of Edge Devices:IoT Sensors, Embedded Systems, Smart Cameras - Sensor Fusion and Data Collection Techniques – Edge Device Management and Deployment Considerations.

Module:3	Multi Access Edge Computing	9 hours

Definition and Objectives - MEC Architecture Overview:Radio Access Network, MEC Platform- MEC Standards and Industry Initiatives - MEC Platform Components:MEC Server, MEC Application - Integration with Mobile Networks:4G LTE, 5G NR - MEC Service Orchestration and Management - Improved Latency and Quality of Experience - Edge-based Content Delivery and Augmented Reality - Use Cases in Smart Cities, Industrial IoT, and Telecommunications.

Module:4	Edge AI - Video Processing	9 hours

Real-time Video Analytics - Edge-based Video Processing vs. Cloud-based Solutions - Edge AI Models for Video Processing : Object Detection, Activity Recognition - Video Compression and Encoding Techniques - Edge-based Object Detection and Tracking - Video Analytics Applications : Surveillance, Smart Retail, Traffic Management– Deep Learning Architectures for Video Analysis - Model Optimization

Techniques for Edge Deployment - Case Studies and Practical Implementations in Video Surveillance and Smart Cities.		
Module:5	**Edge AI - Log Processing**	**9 hours**
Log Data Analysis for System Monitoring and Troubleshooting - Challenges of Centralized Log Processing – Benefits of Edge-based Log Processing - Log Data Collection and Aggregation Methods - Anomaly Detection and Pattern Recognition - Real-time Alerting and Incident Response - Edge-based Log Storage and Retention Policies - Log Data Visualization and Reporting Tools.		
Module:6	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Books	
1.	B. Scholl and H. Bai, Edge Computing and Capability-Oriented Architecture. CRC Press, 1st ed., 2021.
2.	D. Sabella, A. Reznik, and R. Frazao, Multi-Access Edge Computing in Action. Taylor & Francis Ltd, 1st ed., 2022.
Reference Books	
1.	https://www.udemy.com/course/learn-edge-computing-concepts-explained-from-thescratch/?couponCode=24T4FS22124
2.	https://developer.ibm.com/articles/edge-computing-architecture-and-use-cases/
3.	https://www.youtube.com/watch?v=SubQ6vJBXdk
4.	Situnayake, D., & Plunkett, J. AI at the Edge: Solving Real-World Problems with Embedded Machine Learning. "O'Reilly Media, Inc." (2023).
5.	Wang, X., Han, Y., Leung, V. C. M., Niyato, D., Yan, X., & Chen, X. Edge AI: Convergence of Edge Computing and Artificial Intelligence. Springer Nature. (2020).
Mode of Evaluation: Quiz, Assignment, Design Project, Case Study, Seminar, CAT and FAT	
List of Experiments (Indicative)	
1.	Set up a basic edge computing environment using a Raspberry Pi or a similar edge device. Configure necessary software and test connectivity.
2.	Deploy a simple machine learning model on an edge device, optimizing it for low-power consumption and efficient inference.
3.	Implement a Multi-access Edge Computing (MEC) application for low-latency services, ensuring efficient resource utilization and response time.
4.	Develop and deploy a real-time video analytics system on an edge device, utilizing AI models for object detection and tracking.
5.	Build a real-time log monitoring system for edge devices, analysing system logs and generating alerts for anomaly detection.
6.	Implementing AI-based Anomaly Detection for Edge Devices
7.	Integrating IoT Sensors with Edge Computing for Real-time Data Processing
8.	Developing an Edge-based Smart Surveillance System
9.	Optimizing Energy Efficiency in Edge Computing Devices

10.	Implementing Containerization for Edge Computing Applications			
Total hours:				**30 hours**
Mode of Evaluation: Continuous Assessments and FAT				
Recommended by Board of Studies				
Approved by Academic Council	**No.**	**Date**		

Course Code	Large Language Models	L	T	P	C
MACSE605		3	1	0	4
Pre-requisite		Syllabus Version			
		1.0			

Course Objectives

1. Gain an understanding of LLMs, their training, applications and underlying theoretical concepts.
2. Explore the architecture, mechanisms and variants of transformer models in detail.
3. Learn various real-world applications of LLMs.
4. Master the art of crafting effective prompts and understanding their influence on model behaviour.
5. Apply the LLM concepts for real world applications.

Course Outcomes

1. Explain the working mechanism of large language models using various frameworks.
2. Evaluate the self-attention mechanisms, transformer variants, and training procedures to optimize transformer models.
3. Create and implement LLM-based solutions for practical applications.
4. Develop and fine-tune prompts to optimize model performance.
5. Apply advanced LLMs to real-world scenarios.
6. Design and deploy LLM-based solutions using transformer models, prompt engineering.

Module:1	Fundamentals of Large Language Models	10 hours

Large Language Model - Deep Learning core concepts, LLM jargons and concepts - Training and fine-tuning of LLMs - Applications of LLM - Challenges, LLM hallucinations and Ethical considerations - Transformer Architecture - Variations of Transformer Models - RNN and LSTMs in Language Models - Training Infrastructure for LLMs - BERT/GPT Architecture case study - Development frameworks and Libraries :TensorFlow, PyTorch, Hugging face transformers, Data processing and management tools: data cleaning, tokenization and batch processing– Model training and fine-tuning platforms - Performance optimization and monitoring - Frameworks for developing LLM based applications - Deploying and scaling LLMs - Linear Algebra Fundamentals: Vectors, matrices, Eigen values and Eigen vectors, Singular Value decomposition - Probability and Statistics : Probability distribution, Statistical measures, Bayesian Statistics and Inference, Maximum Likelihood Estimation - Optimization techniques :Gradient descent, Convex optimization, regularization Methods - Information Theory : Entropy and Cross-Entropy, KL Divergence, Mutual
Information, Compression

Module:2	Transformer models	9 hours

Self-Attention mechanism - multi-head attention - Positional Encoding - Transformer Architecture – Feedforward Neural Networks - Masking Mechanism - Training transformer Models - Transformer Variants – Attention Mechanism Variants - Transformer Variants and Architectures - Cross-Modal Transformers Transfer learning with transformers - Efficient Transformer Models - Training procedure -
Evaluation and fine-tuning

Module:3	Practical Applications of LLMs	8 hours
Creative Writing, text summarization - Chatbots and dialog systems - Code generation and auto-completion - Sentiment Analysis and Opinion mining - Named Entity Recognition and Entity Linking - Text Classification:Spam detection and topic categorization - Language translation - Semantic Search and Knowledge graphs - Personalized recommendation systems - Retrieval Augmented Generation and Vector Stores – Information Retrieval using LLMs.		
Module:4	**Prompt Engineering**	**8 hours**
Prompt Engineering - Importance of prompts in fine-tuning - Template based prompts - Cloze-style prompts - Query-based prompts - Task and domain specific prompts - Multi-modal prompts - Hyperparameter optimizations - Tuning prompt parameters - Prompt Engineering vs Fine tuning - Prompt influences model behaviour – Evaluating prompt performance - Dynamic Prompt generation - Multi-modal prompt fusion - Mitigating Bias and fairness in prompt design - Security threat, vulnerabilities and Prompt Injection.		
Module:5	**Modern LLMs**	**8 hours**
Early Language models to GPT - OpenAI's ChatGPT - Google's Gemini - Anthropic Claude - Open-Source Models: Llama, Mistral, Applications of LLMs in Healthcare, Finance and Legal.		
Module:6	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Books	
1	Uday Kamath, Kevin Keenan, Garrett Somers Large Language Models: A Deep Dive: Bridging Theory and Practice, Springer, (2024)
2	E. R. Deforest, Prompt Engineering with Transformers and LLM. Kindle, 2024.
Reference Books	
1.	Transformers & LLMs: A Technical Deep Dive into RAG Architectures. Kindle, 2024.
1.	https://machinelearningmastery.com/prompt-engineering-for-effective-interaction-with-chatgpt/
2.	https://www.lakera.ai/blog/llm-security
3.	https://thesecmaster.com/5-security-challenges-in-llms-and-strategies-to-prevent-them/
4.	Tunstall, L., Von Werra, L., & Wolf, T. Natural Language Processing with Transformers, Revised Edition. “O’Reilly Media, Inc.” (2022).
5.	Kamath, U., Liu, J., & Whitaker, J. Deep learning for NLP and speech recognition. Springer. (2019).

Mode of Evaluation: Quiz, Assignment, Design Project, Case Study, Seminar, CAT and FAT			
Recommended by Board of Studies			
Approved by Academic Council	**No.**	**Date**	