K.R. MANGALAM UNIVERSITY

K.R. MANGALAM UNIVERSITY
THE COMPLETE WORLD OF EDUCATION

SCHOOL OF ENGINEERING AND TECHNOLOGY

MCA (Master in Computer Application)Programme Code: 56

Program Handbook

Program Structure & Evaluation Scheme

Two Year Post Graduate Program

(with effect from 2024-25)

Approved in the 34th Meeting of AcademicCouncil Held on 29 June 2024

Contents

Preamble

Welcome to the School of Engineering and Technology at K. R. Mangalam University. It is with great enthusiasm that we introduce you to an institution dedicated to nurturing future leaders in engineering and technology.

Established in 2013, our School has rapidly evolved into a premier center forinnovation, quality education, and skill development. With a focus on imparting advanced knowledge and fostering creativity, we are committed toproviding a transformative educational experience. Our state-of-the-art infrastructure, cutting-edge laboratories, and a distinguished team of faculty members collectively create an environment where academic and professional excellence thrives.

Our diverse programs encompass undergraduate degrees (B.Tech, BCA, B.Sc), postgraduate studies (M.Tech, MCA), and doctoral research across all engineering disciplines. Notably, we offer specialized B.Tech programs in areas such as Artificial Intelligence & Machine Learning, Data Science, Cyber Security, Full Stack Development, and UI/UX Development. These programs are designed to equip students with both technical proficiency and a deep understanding of emerging technologies.

At the heart of our mission is a commitment to a curriculum that integrates the best practices from leading global institutions while also incorporating insights from the Open-Source Society University.

Our emphasis on industry integration is reflected in our collaborations with renowned organizations such as IBM, Samatrix, Xebia, E.C Council, and ImaginXP. These partnerships ensure that our students gain practical experience and insights that are directly applicable to industry demands. Elective options across diverse domains, including AI, Cloud Computing, Cyber Security, and Full Stack Development, offer students the flexibility to tailor their educational experience to their career aspirations.

We are also dedicated to fostering a culture of innovation and entrepreneurship through our Entrepreneurship and Incubation Center and initiatives like 'MindBenders,' 'Hack-KRMU,' and participation in the 'Smart India Hackathon.' These programs are designed to inspire and prepare students to become forward-thinking leaders in the technology sector.

Our modern computing facilities and comprehensive infrastructure support advanced research, simulations, and hands-on projects, ensuring that our students are well-prepared for the challenges of the professional world. K. R. Mangalam University is recognized for its commitment to providing quality education, and our alumni have made notable contributions across various sectors, from multinational corporations to public sector enterprises.

We are excited to accompany you on this journey and look forward to supporting your academic and professional growth. Welcome to acommunity where excellence and innovation are at the core of everything wedo.

School of Engineering & Technology
K.R Mangalam University

Categories of Courses

Major: The major would provide the opportunity for a student to pursue in-depth study of a particular subject or discipline.

Multidisciplinary (Open Elective): These courses are intended to broaden the intellectual experience and form part of liberal arts and science education. These introductory-level courses may be related to any of the broad disciplines given below:

- Natural and Physical Sciences
- Mathematics, Statistics, and Computer Applications
- Library, Information, and Media Sciences
- Commerce and Management
- Humanities and Social Sciences

A diverse array of Open Elective Courses, distributed across different semesters and aligned with the aforementioned categories, is offered to thestudents. These courses enable students to expand their perspectives and gain a holistic understanding of various disciplines. Students can choose courses based on their areas of interest.

Ability Enhancement Course (AEC): Students are required to achieve competency in a Modern Indian Language (MIL) and in the English languagewith special emphasis on language and communication skills. The courses aim at enabling the students to acquire and demonstrate the core linguistic skills, including critical reading and expository and academic writing skills, that help students articulate their arguments and present their thinking clearly and coherently and recognize the importance of language as a mediator of knowledge and identity.

Skills Enhancement Courses (SEC): These courses are aimed at imparting practical skills, hands-on training, soft skills, etc., to enhance theemployability of students.

Discipline Specific Electives (DSE): The purpose of offering discipline-specific electives is to provide students with the flexibility to specialize in emerging and high-demand domains such as Full Stack Development, Cloud Computing, AI & ML, and Cyber Security. These electives are designed to equip students with advanced knowledge and skills in their chosen fields, ensuring they are well-prepared for specialized roles and industry demands in these cutting-edge areas.

Industry project/Research Project: Students choosing a 2-Year PG degree (MCA) are required to take up research projects under the guidance of a faculty member. The students are expected to complete the Research Project in the final semester. The research outcomes of their project work may be published in peer-reviewed journals or may be presented in conferences /seminars or may be patented.

University Vision & Mission

Vision

K. R. Mangalam University aspires to become an internationally recognized institution of higher learning through excellence in inter-disciplinary education, research and innovation, preparing socially responsible life-longlearners contributing to nation building.

Mission

- Foster employability and entrepreneurship through futuristic curriculum and progressive pedagogy with cutting-edge technology.
- Instill notion of lifelong learning through stimulating research, Outcomes-based education and innovative thinking.
- Integrate global needs and expectations through collaborative programs with premier universities, research centers, industries andprofessional bodies.
- Enhance leadership qualities among the youth having understanding ofethical values and environmental realities.

About School

The School of Engineering and Technology at K. R. Mangalam University started in 2013 to create a niche of imparting quality education, innovation, entrepreneurship, skill development and creativity. It has excellent infrastructure, state of the art Labs, and a team of qualified and research- oriented faculty members.

The school is offering undergraduate programs (B.Tech, BCA, B.Sc), postgraduate programs (M.Tech, MCA) and Ph.D (all disciplines of Engineering). We are offering B.Tech programs in recent areas of specializations like AI & ML, Data Science, Cyber Security, Full stack development, UI/UX development etc.

Our strength lies in our highly qualified, research oriented, and committed teaching faculty. We believe in empowering minds through expert guidance, ensuring that our students receive a world-class education that prepares them for the challenges of the ever-evolving technological landscape.

The School of Engineering & Technology is committed to providing a cutting-edge curriculum by integrating the best practices from top global universities and leveraging the rich knowledge resources of the Open-Source Society University. The curriculum focuses on problem-solving, design, development, interdisciplinary learning, skill development, research opportunities and application of various emerging technologies with focus on innovative teaching learning methodologies.

We take pride in offering an industry-integrated curriculum that goes beyond traditional education. Collaborations and training led by industry experts, along with partnerships with renowned organizations such as IBM, Samatrix, Xebia, E.C Council, ImaginXP etc ensure that our students gain practical insights and skills that align with real-world industry demands.

With elective options across various domains, including AI, Cloud Computing, Cyber Security, and Full Stack Development, we empower students to customize their learning experience. Our goal is to provide the flexibility needed for each student to shape their academic and professional future.

We prioritize career growth by offering comprehensive training, placements, international internships, and preparation for further studies. Our commitment to nurturing globally competitive professionals is reflected inthe diverse pathways we pave for our students.

SOET aims at transforming the students into competitive engineers with adequate analytical skills, making them more acceptable to potential employers in the country. At our school, we emphasize learning through doing. Whether it's project-based learning, field projects, research projects, internships, or engaging in competitive coding, our students actively shape their futures by applying theoretical knowledge to practical scenarios. We provide opportunities for industrial projects, R&D projects, and start-upprojects in the final year, ensuring that our students engage in real-world innovation.

We are dedicated to fostering a culture of innovation and entrepreneurship, recognizing these as essential pillars for the success of our students in the rapidly evolving world of technology. We inspire innovation and entrepreneurship through our dynamic Entrepreneurship and Incubation

Center, engaging contests like 'MindBenders' ,'Hack-KRMU,' participation in ‘Smart India Hackathon’, International Conference ‘MRIE’ empowering students to become forward-thinking leaders in the ever-evolving realm of technology.

We pride ourselves on providing state-of-the-art computing facilities and infrastructure. Our modern labs and computing resources are equipped to support the diverse needs of our students, enabling them to engage in advanced research, simulations, and hands-on projects.

K.R. Mangalam University has marked its presence in Delhi NCR as a value- based university, successfully imparting quality education in all domains. Ouralumni are working across all sectors of technology, from MNCs to PSUs.

School Vision & Mission

Vision

To excel in scientific and technical education through integrated teaching,research, and innovation.

Mission

- **Creating** a unique and innovative learning experience to enhance quality in the domain of Engineering & Technology.
- **Promoting** Curricular, co-curricular and extracurricular activities that support overall personality development and lifelong learning, emphasizing character building and ethical behavior.
- **Focusing** on employability through research, innovation and entrepreneurial mindset development.
- **Enhancing** collaborations with National and International organizations and institutions to develop cross-cultural understanding to adapt and thrive in the 21st century.

About the Program

The Master of Computer Applications (MCA) program is a comprehensive postgraduate degree designed to provide students with advanced knowledge and skills in computer science and applications. The program typically spans two to three years and is structured to encompass a blend of theoretical and practical learning. It covers a wide range of subjects, including programming languages, database management, software engineering, computer networks, artificial intelligence, and data analytics. The curriculum is meticulously crafted to keep pace with the ever-evolving technology landscape, ensuring that graduates are well-equipped to tackle contemporary challenges in the IT industry.

In addition to core computer science topics, the MCA program often includes courses on business management, mathematics, and communication skills, which are crucial for holistic professional development. Students engage in hands-on projects, internships, and practical labs that foster experiential learning and problem-solving abilities. These activities are designed to simulate real-world scenarios, preparing students for the dynamic and demanding environment of the tech industry.One of the distinguishing features of the MCA program is its emphasis on emerging technologies such as cloud computing, Internet of Things (IoT),cybersecurity, and machine learning. This forward-looking approach ensures that graduates are not only proficient in current technologies but are also capable of adapting to future advancements. Moreover, the program encourages research and innovation, often culminating in a significant capstone project or thesis, which allows students to explore and contribute to cutting-edge developments in their chosen areas of interest. Overall, the MCA program aims to produce highly skilled and adaptable professionals ready to excel in various roles within the IT sector.

Programme Outcomes (POs)

Programme Outcomes are statements that describe what the students are expected to know and would be able to do upon the graduation. These relate to the skills, knowledge, and behavior that students acquire through the Programme.

Programme Specific Outcomes (PSOs)

Programme Specific Outcomes define what the students should be able to do at the time of graduation and they are Programme specific. There are two to four PSOs for a Programme.

Programme Educational Objectives (PEOs)

Programme Educational Objectives of a degree Programme are the statements that describe the expected achievements of graduates in their career, and what the graduates are expected to perform and achieve during the first few years after graduation.

- **Credit**

  Credit refers to a unit that measures the amount of academic work required for a course. It typically reflects the number of instructional hours per week.

Program Educational Objectives (PEO)

PEO1: Achieve success in industry, government, academia, research, entrepreneurship, or consulting roles related to computer applications.

PEO2: Apply advanced computer application principles and methodologies to solve real-world problems, integrating theoretical knowledge with practical skills.

PEO3: Uphold professional ethics and social responsibilities, contributing to societal well-being and sustainability through the effective use oftechnology.

PEO4: Exhibit leadership and teamwork skills to effectively manage and collaborate in diverse, multidisciplinary projects within the field of computer applications.

Program Outcome (PO)

1. **PO1. Core Competencies in Computer Applications:** Demonstrate a solid foundation in computer application principles, critical problem analysis, and solution design, with the ability to conduct thorough investigations and address complex technical challenges.

2. **PO2. Modern Tool Usage:** Create, select, and apply advanced techniques, resources, and IT tools for complex computer application tasks, with an understanding of their limitations and applications.

3. **PO3. Societal and Environmental Responsibility:** Evaluate and address societal, health, safety, legal, and cultural issues related to computing, considering the environmental impact of solutions andadvocating for sustainable practices.

4. **PO4. Ethics:** Apply ethical principles and adhere to professional ethicsand norms in the practice of computer applications.

5. **PO5. Effective Communication and Team Collaboration:** Excel in both individual and team roles in diverse and multidisciplinary environments, effectively communicating complex computer application concepts through reports, presentations, and interactions.

6. **PO6. Project Management:** Utilize management principles to lead and manage projects effectively within computer applications contexts.

7. **PO7. Life-Long Learning:** Engage in continuous learning to stay updated with technological advancements and evolving practices in computer applications.

PSOs (Program Specific Outcomes)

1. **PSO1:** Understanding advanced concepts, theories, tools, techniques,and methodologies in computer applications.

2. **PSO2:** Applying advanced knowledge and techniques to address andsolve real-world challenges in computer applications.

3. **PSO3:** Analyzing and evaluating methodologies, problems, and issues within various contexts of computer applications.

4. **PSO4:** Assessing alternative solutions and making informed decisions to address complex problems in computer applications.

5. **PSO5:** Designing and developing innovative solutions to tackle sophisticated challenges in computer applications.

Career Avenues

Diverse career avenues available to A master’s in computer application(MCA) program are as follows:

1. Software Development: Graduates can pursue careers as softwaredevelopers, working on designing, coding, testing, and maintaining software applications and systems. They can specialize in areas such as web development, mobile app development, game development, or enterprise software development.

2. Systems Analyst: A systems analyst analyzes an organization's computer systems and procedures to improve efficiency and effectiveness. They work on designing and implementing new systems, conducting feasibility studies, and identifying areas for improvement inexisting systems.

3. Data Scientist: With the increasing volume of data in various industries, data scientists are in high demand. They utilize their skills in data analysis, statistics, and machine learning to extract insights from large datasets, make data-driven decisions, and develop predictive models.

4. Artificial Intelligence Engineer: As AI technology continues to advance, there is a growing demand for professionals skilled in developing AI algorithms and systems. AI engineers work on creating intelligent machines, developing natural language processing systems, computer vision applications, and other AI-driven solutions.

5. Cyber security Analyst: In an era of heightened cyber security threats, organizations require experts who can protect their systems and data. Cyber security analysts identify vulnerabilities, implement security measures, conduct risk assessments, and respond to security incidentsto safeguard computer systems and networks.

6. Network Engineer: Network engineers are responsible for designing, implementing, and maintaining computer networks within organizations. They ensure network reliability, security, and performance, and troubleshoot network issues to ensure smooth operations.

7. IT Project Manager: IT project managers oversee the planning, execution, and delivery of technology projects within organizations.

They manage project teams, coordinate resources, track progress, and ensure projects are completed within budget and on time.

8. Database Administrator: Database administrators manage and maintain databases, ensuring data integrity, security, and performance. They design database structures, implement backup and recovery procedures, and optimize database systems for efficient data storage and retrieval.

9. Software Quality Assurance Engineer: QA engineers are responsible for ensuring the quality and reliability of software applications. They develop and execute test plans, identify and report bugs and issues, and work closely with development teams to improve software quality.

10. Research and Development: Graduates can pursue careers in research and development, working on innovative projects, exploring new technologies, and pushing the boundaries of computer science and engineering. This can involve academic research, industry research labs, or research and development departments within companies.

Duration

2Years (4 Semesters) - Full-Time Program

Criteria for award of certificates and degree

Students must successfully complete required credits of 89 to be eligible for award of degree.

Student's Structured Learning Experience in the Programme

a. University Education Objective

Focus on Employability and Entrepreneurship through Holistic Education using Bloom's Taxonomy. By targeting all levels of Bloom's Taxonomy—remembering, understanding, applying, analysing, evaluating, and creating—students are equipped with the knowledge, skills, and attitudes necessary for the workforce and entrepreneurial success. At KRMU we emphasize on learners critical thinking, problem-solving, and innovation, ensuring application of theoretical knowledge in practical settings. This approach nurtures adaptability, creativity, and ethical decision-making, enabling graduates to excel in diverse professional environments and to innovate in entrepreneurial endeavor's, contributing to economic growth and societal well-being.

b. Importance of Structured Learning Experiences:

A structured learning experience (SLE) is crucial for effective education as it provides a clear and organized framework for acquiring knowledge and skills. By following a well-defined curriculum, teaching-learning methods and assessment strategies, learners can build on prior knowledge systematically, ensuring that foundational concepts are understood before moving on to more complex topics. This approach not only enhances comprehension but also fosters critical thinking by allowing learners to connect ideas and apply them in various contexts. Moreover, a structured learning experience helps in setting clear goals and benchmarks, enabling both educators and students to track progress and make necessary adjustments. Ultimately, it creates a conducive environment for sustained intellectual growth, encouraging learners to achieve their full potential.

At K.R. Mangalam University SLE is designed as rigorous activities that are integrated into the curriculum and provide students with opportunities for learning in two parts:

- **Inside the Classroom:**

Our educational approach within the classroom is designed to foster **cognitive development** and enhance **student-centric learning**. We prioritize active engagement and deep understanding by employing a variety of methods, tools, and techniques. These include **problem-based learning**, **case studies**, **interactive discussions**, and **technology-enhanced learning platforms**. Our faculty focuses on developing critical thinking, analytical reasoning, and problem-solving abilities, ensuring students achieve well-defined **cognitive outcomes**. Additionally, we integrate the use of **modern teaching tools**, such as Learning Management Systems (LMS), virtual labs, and multimedia resources, to enhance the learning experience and accommodate diverse learning styles. This comprehensive approach not only promotes academic excellence but also nurtures independent learning and lifelong intellectual curiosity.

- **Outside the Classroom:**

Beyond the classroom, our focus shifts to developing students' **people skills** and **psychomotor skills** through hands-on experiences in **industry, community, and laboratory settings**. We encourage participation in internships, industrial visits, community engagement projects, and research opportunities, which allow students to apply theoretical knowledge to real-world challenges. These activities build essential interpersonal skills such as **teamwork, leadership, communication**, and **professional networking**. Simultaneously, students engage in **lab-based learning** and technical workshops that refine their psychomotor abilities, including precision, technical expertise, and problem-solving under practical conditions. Through these outside-the-classroom experiences, students gain a holistic skill set that prepares them to excel in both professional and societal contexts, aligning their education with real-world expectations and industry needs.

c. Educational Planning and Execution

The Master of Computer Application (MCA) at K.R. Mangalam University is designed to foster a holistic educational experience, integrating both theoretical knowledge and practical skills. The program offers students a structured path from entry to exit, ensuring they develop technical expertise, problem-solving skills, and professional competencies.

- **Entry Phase**

Upon entering the MCA program, students are introduced to the foundational concepts of Artificial Intelligence, Machine Learning and programming. This phase is designed to strengthen their understanding of core scientific and technical principles. Courses such as AI & ML, Problem solving and advanced programming concept, Advanced DBMS and Data Structure and Algorithm provide a strong foundation. Students also engage in hands-on laboratory sessions to complement theoretical learning, which helps them connect classroom knowledge with real-world applications.

Orientation Program: The university conducts a **one-day orientation program** for first-year students to familiarize them with the university's environment and key aspects. During the program, students are introduced to the university’s highlights, important procedures, key functionaries, and the code of conduct. This orientation serves to ensure that students are well-informed and prepared for a smooth transition into university life.

In the first year, students are exposed to critical problem-solving approaches, basic programming, and ethics in engineering, laying the groundwork for their technical and professional growth.

Induction program: The School organizes a **5-day induction program** for first-year students, aimed at providing them with a comprehensive understanding of the school's various aspects. During the program, students are introduced to learning resources, facilities, and opportunities available to them, along with the rules and regulations governing academic and campus life. The induction also includes faculty introductions, guidelines on academic conduct, and detailed information about examination and evaluation methods, ensuring students are well-prepared for their academic journey.

Core Learning

As students advance through the program, they delve deeper into core computer science subjects such as Data Structures, Algorithms, Object-Oriented Programming (C++), Operating Systems, and Database Management Systems. This phase emphasizes both theoretical concepts and their practical application through lab work. The learning is enhanced through exposure to industry-standard tools and techniques, including programming languages like Java and Python, and systems for data management and networking.

The structured academic schedule, with a well-distributed credit system over eight semesters, ensures students acquire deep technical knowledge and skills in software development, systems design, and computing technologies. The Summer Internship Programs and Minor Projects in the curriculum allow students to apply their learning in real-life projects, facilitating experiential learning.

Summer Internships: School offers 2-credit summer internships spanning 6 weeks, where students are encouraged to pursue internships in startups, industries, or premier institutions such as IITs, NITs, and IIITs. In addition, students have the opportunity to earn global certifications during this period. The School also organizes in-house summer schools in collaboration with industry partners, providing further avenues for students to gain hands-on experience and enhance their professional skills. These initiatives are designed to offer students practical exposure, helping them develop industry-relevant expertise.

Skill Development

Throughout the program, there is a significant emphasis on developing practical skills and ensuring students are industry-ready. Courses on Artificial Intelligence, Machine Learning, Cloud Computing, and Cyber security provide students with cutting-edge knowledge in emerging fields

.

Capstone and Exit Phase

In the final semesters, students undertake discipline-specific electives and capstone projects. These projects integrate the knowledge and skills they have acquired over the course of their studies. Electives such as Natural Language Processing, Generative AI, and Blockchain Technologies offer students the flexibility to specialize in areas of their interest.

The final Industrial Project or R&D Project in the eighth semester is a full-time engagement where students work on live industry problems, research projects, or start-up ideas. This project phase, combined with career readiness boot camps and placement preparation activities, ensures that students are equipped to enter the workforce with both technical competence and professional acumen.

Co-Curricular and Extra-Curricular Activities

Students are encouraged to participate in various clubs, societies, and extra-curricular activities. Engagement in activities such as hackathons, coding competitions, and leadership roles in clubs fosters teamwork, leadership, and creativity. These activities complement academic learning, contributing to the students' holistic development.

Ethics and Professional Values

The program places a strong emphasis on ethics and professionalism. Students are taught to incorporate ethical considerations in technological development and decision-making processes. This prepares them to not only be skilled engineers but also responsible professionals who contribute positively to society.

Career Counseling and Entrepreneurship

The university offers comprehensive career counselling services, providing students with expert guidance on job placements, internships, and skill development to help them effectively navigate their career paths. In addition, the university's incubation center plays a pivotal role in nurturing entrepreneurial and leadership skills, empowering students to explore innovative ideas and launch their own ventures. These initiatives are designed to equip students with the tools and resources necessary for professional success and entrepreneurial growth.

Course Registration

- Every student has to register at the beginning of each semester for the courses offered in the given semester. Major courses are registered centrally for the students. However, for other multidisciplinary courses (DSE, OE) the students have to register by themselves through ERP.

1. Student Support Services

Mentor-Mentee: At K.R. Mangalam University, the **Mentor-Mentee Program** plays a crucial role in fostering academic and personal growth. Each student is assigned a faculty mentor who serves as a guide throughout their academic journey. This program ensures continuous interaction, where mentors assist students with academic planning, help in resolving personal issues, and provide career guidance. The mentor-mentee relationship transcends the classroom and often involves personal development, professional growth, and overall well-being. The program aims to nurture a supportive environment that enhances the learning experience and helps students reach their full potential.

2. **Counselling and Wellness Services:** The university places a strong emphasis on the mental and emotional well-being of its students through its Counselling and Wellness Services. A dedicated team of trained counselors provides personalized sessions, workshops, and wellness programs to address the mental health needs of the student community. These services focus on holistic well-being, including stress management, emotional resilience, and coping strategies. Regular wellness programs,

meditation sessions, and mental health awareness campaigns are conducted to promote a balanced lifestyle and ensure that students can focus on their studies while maintaining their emotional health.

3. **Evaluation of Learning:**

At K.R. Mangalam University, assessment and evaluation are integral components of the teaching-learning process, designed to ensure continuous academic progress and holistic development of students. The university follows a Learning Outcome-Based Framework (LOCF), where assessments are aligned with the specific learning outcomes of each program. A variety of assessment methods, including assignments, presentations, quizzes, practical examinations, and project work, are used to gauge students' understanding. The examination system is 100% automated, ensuring timely and transparent evaluation processes. Results are processed efficiently, typically within 13 days, and complaints related to evaluation are minimal, reflecting the university's commitment to maintaining a high standard of academic integrity. This robust system of continuous assessment and feedback fosters a culture of academic excellence and skill development among students.

Evaluation Scheme (Theory Courses):

Evaluation Components	Weightage
A. Internal Assessments: **1. Continuous Assessment (30 Marks)** **(Minimum 5 components to be used and to be evenly spaced)** Project/ Quizzes/Test/ Assignments and Essays/ Presentations/ Class Participation/ Case Studies/ Reflective Journals	**30 Marks**
2. Mid Term Exam	**20 Marks**
B. External Assessments: End term Examination	**50 Marks**
Total	**100 Marks**

Evaluation Scheme (Laboratory/Practical Courses):

Evaluation Components	Weightage
Internal Assessments–	
1. Conduct of Experiment	10 Marks
2. Lab Records	10 Marks
3. Lab Participation	10 Marks
4. Lab Project	20 Marks
External Assessments- End term Practical Exam and Viva Voce	50 Marks
Total	**100 Marks**

5. Feedback and Continuous Improvement Mechanisms:

K.R. Mangalam University is deeply committed to academic excellence through a robust **feedback and continuous improvement system**. This system is designed to gather comprehensive input from a diverse range of stakeholders, including **students, faculty, alumni, employers, and academic peers**. Feedback is systematically collected and thoroughly analyzed to identify areas for enhancement in **curricula, teaching methodologies, and academic processes**. Based on the insights gained, actionable measures are formulated and communicated to the appropriate bodies for timely implementation.

This structured feedback mechanism ensures that the university's programs remain aligned with **industry trends and societal needs**, providing students with a cutting-edge education that prepares them for real-world challenges. Moreover, the university demonstrates its commitment to continuous improvement through **regular curriculum updates** and the integration of **innovative teaching strategies**, fostering an environment where both faculty and students can grow and excel. By maintaining this cycle of feedback and improvement, K.R. Mangalam University ensures the continuous advancement of its academic offerings and the overall learning experience.

6. Academic Integrity and Ethics:

K.R. Mangalam University upholds the highest standards of academic integrity and ethics as a core value of its educational philosophy. The university implements a zero-tolerance policy towards academic misconduct, including plagiarism and other unethical practices. To ensure transparency and honesty in academic work, plagiarism detection software like Drillbit is used to maintain the originality of student submissions and research outputs. Students and faculty are regularly sensitized on the importance of ethical behavior through workshops, seminars, and classroom discussions. The university also integrates ethics and professional values into its curriculum across various disciplines, ensuring that graduates not only excel academically but also demonstrate integrity and responsibility in their professional and personal lives.

Scheme of Studies

Program Name	Master of Computer Application
Total Credits	89
Total Semesters	4

Credit Distribution Summary

Program Name	I	II	III	IV	Total Credits
MCA	22	28	27	12	89

SEMESTER I

SNo	Category	Course Code	Course Title	L	T	P	C
1	Major-1	ENMM701	Problem Solving and Advanced Programming Concepts	3	1	0	4
2	Major-2	ENMM703	Data Structures and Algorithms	3	1	0	4
3	Major-3	ENMM705	Advanced Database Management Systems	3	1	0	4
4	Major-4	ENSP701	AI & Machine Learning	4	0	0	4
5	Major-5	ENMM751	Problem Solving Lab	0	0	2	1
6	Major-6	ENMM753	Data Structures and Algorithms Lab	0	0	2	1
7	DSE-1	ENSP751	AI & Machine Learning lab	0	0	2	1
8	AEC-1	AEC006	Verbal Ability	3	-	-	3
			TOTAL	16	3	6	22

SEMESTER II

SNo	Category	Course Code	Course Title	L	T	P	C
1	Major-7	ENMM702	Advanced Computer Organization & Architecture	3	1	0	4
2	DSE-2		Discipline Specific Elective -I	4	0	0	4
3	Major-8	ENMM704	Advanced Web Technologies	3	1	0	4
4	Major-9	ENMM706	Introduction of Neural Network and Deep Learning	3	1	0	4
5	DSE-3		Discipline Specific Elective -I Lab	0	0	2	1
6	Major-10	ENMM754	Advanced Web Technologies Lab	0	0	2	1
7	Major-11	ENMM756	Introduction to Neural Networks & Deep Learning Lab	0	0	2	1
8	AEC-2	AEC007	Communication & Personality Development	3	0	0	3
9	Open Elective-1		Students can choose one of the electives from the pool of open electives of university	3	0	0	3
10	Proj-1	ENSI152	Minor Project	-	-	-	2
11	CS-1	CS002	Community Service	1	-	-	1
TOTAL				19	3	6	28

Discipline Specific Elective-I (AI & Machine Learning)							
(i)	DSE	ENSP302	Natural Language Processing	4	-	-	4
	DSE	ENSP352	Natural Language Processing lab	-	-	2	1
(ii)	DSE	ENSP304	Image Processing & Computer Vision	4	-	-	4
	DSE	ENSP354	Image Processing & Computer Vision lab	-	-	2	1
(iii)	DSE	ENSP306	Introduction to Generative AI	4	-	-	4
	DSE	ENSP356	Generative AI lab	-	-	2	1
(iv)	DSE	ENSP308	Transfer Learning	4	-	-	4
	DSE	ENSP358	Transfer Learning lab	-	-	2	1

SEMESTER III

	Category	Course Code	Course Title	L	T	P	C
SNo	DSE-4		Discipline Specific Elective-II	4	0	0	4
1	DSE-5		Discipline Specific Elective-III	4	0	0	4
2	Major-12	ENMM816	Advance Concepts in Software Engineering & Project Management	3	1	0	4
3	SEC-1	SEC036	Competitive Coding Lab	-	-	4	2
4	DSE-6		Discipline Specific Elective-II Lab			2	1
5	DSE-7		Discipline Specific Elective-III lab			2	1
7	Open Elective-2		Students can choose one of the electives from the pool of open electives of University	3	0	0	3
8	INT-1	SIMM851	Summer Internship				2
9	MOOC-1		Applied Programming and Problem-Solving Skills for Campus Interviews	-	-	-	2
10	AEC-3	AEC008	Arithmetic and Reasoning Skills-I	3	-	-	3
11	**CS-2**	**CS001**	**Club/Society**	**1**	**-**	**-**	**1**
			TOTAL	14	1	8	27

Discipline Specific Elective - II (Cloud Computing)							
(i)	DSE	ENSP401	Computational Services in The Cloud	4	-	-	4
	DSE	ENSP451	Computational Services in The Cloud Lab	-	-	2	1
(ii)	DSE	ENSP403	Microsoft Azure Cloud Fundamentals	4	-	-	4
	DSE	ENSP453	Microsoft Azure Cloud Fundamentals Lab	-	-	2	1
(iii)	DSE	ENSP405	Storage and Databases on Cloud	4	-	-	4
	DSE	ENSP455	Storage and Databases on Cloud Lab	-	-	2	1
(iv)	DSE	ENSP407	Application Development and DevOps on Cloud	4	-	-	4
	DSE	ENSP457	Application Development and DevOps on Cloud Lab	-	-	2	1
Discipline Specific Elective - III (Full Stack Development)							
(i)	DSE	ENSP409	Mobile Application Development using iOS	4	-	-	4
	DSE	ENSP459	Mobile Application Development using iOS Lab	-	-	2	1
(ii)	DSE	ENSP411	DevOps & Automation	4	-	-	4
	DSE	ENSP461	DevOps & Automation Lab	-	-	2	1
(iii)	DSE	ENSP413	.Net FRAMEWORK	4	-	-	4
	DSE	ENSP463	.Net FRAMEWORK Lab	-	-	2	1
(iv)	DSE	ENSP415	New Age Programming languages	4	0	0	4
	DSE	ENSP465	New Age Programming languages Lab	0	0	2	1

SEMESTER IV

SNo		Course Code	Course Title	L	T	P	C
1	PROJ	SIMM852	Industry Project /Research Project	-	-	-	12
TOTAL				0	0	0	12

Total Credits: 89

Syllabus

Semester: 1

Problem Solving and Advanced Programming Concepts

Program Name	**MCA (Master in Computer Application)**		
Course Name: **Problem Solving and Advanced Programming Concepts**	**Course Code**	**L-T-P**	**Credits**
	ENMM701	3-1-0	4
Total Contact Hours	40		
Type of Course:	Major		
Pre-requisite(s):			

Course Perspective. This course is to familiarize students with techniquesin Basics of Python programming, Condition, Control Structures and Functions, Objects and Classes and Python libraries and Database connectivity their applications. It aims to equip students with standard concepts and tools from intermediate to advanced levels that will enable them to tackle more advanced mathematical and engineering problems relevant to their disciplines. The course is divided into 4 modules:

a) Basics of Python programming

b) Condition, Control Structures and Functions

c) Objects and Classes

d) Python libraries and Database connectivity

The Course Outcomes (COs). On completion of the course theparticipants will be:

COs	Statements
CO1	**Understanding** fundamental concepts of programing language such assyntax, working environment.
CO2	**Applying** control structures and functions to solve programmingproblems
CO3	**Developing and using** Python classes and objects to implement object-oriented programming concepts.
CO4	**Developing** Python applications using libraries and database connectivity, **applying** classification techniques to datasets.

CO = Course outcomes. A student is expected to have learnt concepts and demonstrated/developed abilities or skills related to strategic management at the end of the course.

Course Outline:

<table>
<tr><th>Unit Number: 1</th><th>Title: Basics of Python programmingand Control Flow & Functions</th><th>No. of hours: 8</th></tr>
<tr><td colspan="3"><b>Content:</b>
<b>Introduction to Python:</b> History, features, and applications; Setting up Python environment: Installing Python, IDEs (e.g., VSCode, Anaconda, PyCharm); Basic syntax and data types:Variables, numbers, strings, lists, tuples, dictionaries; Basic input/output operations.
<b>Conditional statements:</b> if, elif, else; Loops: for loop, while loop, nested loops; Control flow statements: break, continue; Functions: Defining functions, parameters, return statement; Scope and lifetime of variables.</td></tr>
</table>

<table>
<tr><th>Unit Number: 2</th><th>Title: Data Structures and File Handling</th><th>No. of hours: 10</th></tr>
<tr><td colspan="3"><b>Content:</b>
<b>Lists:</b> Operations, methods, slicing; Tuples and sets: Properties,operations; Dictionaries: Creating, accessing, modifying;
<b>File handling:</b> Opening, reading, writing, and closing files;Exception handling: try, except, finally blocks.</td></tr>
</table>

<table>
<tr><th>Unit
Number: 3</th><th>Object-Oriented Programming inPython, Modules</th><th>No. of
hours: 12</th></tr>
<tr><td colspan="3"><b>Content:</b>
<b>Introduction to object-oriented programming (OOP) concepts;</b> Classes and objects: Defining classes, creating objects; Encapsulation, inheritance, and polymorphism; Method overriding and overloading; Special methods (dunder methods):
___init___,____str____,____repr_____.

<b>Understanding modules and packages;</b> Importing modules and packages; Standard libraries: os, sys, math, random, datetime;</td></tr>
<tr><th>Unit
Number: 4</th><th>Title: Modules, Packages, Libraries andApplied Python</th><th>No. of
hours: 10</th></tr>
<tr><td colspan="3"><b>Content:</b>

<b>Third-party libraries:</b> NumPy, Pandas, Matplotlib; Exploringdocumentation and using external libraries.

<b>Regular expressions:</b> Syntax, patterns, re module; Lambdafunctions and map, filter, reduce functions;</td></tr>
</table>

Learning Experiences

Inside Classroom Learning:

1. Interactive Coding Exercises: Students practice writing Python programs in real-time with in-class coding challenges related to loops, data types, and control structures.
2. Live Code Reviews: Students submit code for peer review, receiving constructive feedback on maintaining clean code principles.
3. Hands-on File Handling Tasks: Students work on file manipulation tasks, exploring real-world scenarios using CSV, JSON, and regular expressions.
4. Mini-Projects: Small projects like building a basic text-based game to apply Python basics and data structures.
5. Web Scraping Practice: Implement web scraping in class to extract data from websites using Python libraries.
6. Error-Handling Workshops: Collaboratively debug code with exception-handling techniques, promoting troubleshooting skills.

7. Function Design Practice: Group exercises in designing clean, efficient functions using lambda, map, and filter functions.

Outside Classroom Learning:

1. Online Coding Competitions: Participation in coding challenges on platforms like LeetCode or HackerRank to practice clean code.
2. Group Web Scraping Projects: Students collaborate on web scraping projects using real-world data to extract insights from websites
3. Database Connectivity Assignments: Real-time project work with MySQL databases, creating systems to fetch and insert data
4. Open-Source Contributions: Students contribute to open-source Python projects, adhering to clean coding standards.
5. Hackathons: Participation in hackathons focused on building Python-based solutions for cybersecurity challenges.
6. Documentation Writing: Students practice writing clear, concise documentation for their Python projects, reinforcing clean coding habits.

Textbooks:

□ John V Guttag. “Introduction to Computation and Programming UsingPython”, Prentice Hall of India

Reference Books:

□ R. Nageswara Rao, “Core Python Programming”, Dreamtech

□ 2. Wesley J. Chun. “Core Python Programming, Second Edition”, Prentice Hall5.

Additional Readings:

1. Link to UDEMY course contents: https://www.udemy.com/topic/python/
2. Link to topics related to course: https://onlinecourses.nptel.ac.in/noc23_cs18/preview

DATA STRUCTURE AND ALGORITHMS

Program Name	**MCA (Master in Computer Application)**		
Course Name: Data Structure and Algorithms	**Course Code**	**L-T-P**	**Credits**
	ENMM703	3-1-0	4
Total Contact Hours	40		
Type of Course:	Major		
Pre-requisite(s), if any: Basics of Computer Programming			

Course Perspective:

This course introduces students to fundamental data structures, bridging theoretical concepts with practical applications in computer science. It covers essential topics such as arrays, stacks, queues, linked lists, trees, graphs, and sorting and searching algorithms. By exploring these core areas, students will gain the skills needed to design and implement efficient data structures and algorithms, crucial for solving real-world computational problems. The course is divided into four modules:

a. Introduction to Data Structures

b. Stacks, Queues, and Linked Lists

c. Trees and Graphs

d. Sorting, Searching, and File Structures.

The Course Outcomes (COs). On completion of the course the participants will be:

COs	Statements
CO1	**Understanding** the fundamental concepts of data structures and theirsignificance in computational problem-solving.
CO2	**Applying** the concepts of data structure types for given problem.
CO3	**Articulating** Data Structure concepts such as Stack, Queue, Linked list,Hashing, Graph and traversal techniques and their use in programs
CO4	**Designing & implementing** various sorting and searching algorithms andmanage file structures to improve data retrieval and processing efficiency.

CO = Course outcomes. A student is expected to have learnt concepts and demonstrated/developed abilities or skills related to strategic management at the end of the course.

Course Outline:

Unit Number: 1	**Title: Introduction to Data Structure**	**No. of hours: 10**
Content: **Introduction to Data Structures:** Definition of data structures and abstract data types, Static and Dynamic implementations, Examples and real life applications; Arrays: ordered lists, representation of arrays in memory **Basic Analysis:** Differences among best, average, and worst case behaviours of an algorithm, Asymptotic analysis of upper and expected complexity bounds, Big O notation: formal definition and use, big omega and big theta notation , Complexity classes, such as constant, logarithmic, linear, quadratic, and exponential, Time and space trade-offs in algorithms, Recurrence relations , Analysis of iterative and recursive algorithms		
Unit Number: 2	**Title: Stacks, Queues and Linked List**	**No. of hours: 10**

Content: **Stacks:** ADT Stack and its operation, Array based implementation of stacks, Examples: Infix, postfix, prefix representation, Conversions of an arithmetic expression from Infix to postfix, Applications: Tower of Hanoi Problem, Algorithms and their complexities **Queues:** ADT Queue and its operation, Array based implementation of linear Queues, Circular Queues, Priority queues **Linked List:** Definition, Components of linked list, Representation of linkedlist, Advantages and Disadvantages of linked list. Types of linked list: Singly linked list, Doubly linked list, Circular linked list and circular doubly linked list. Operations on different types of linked list : creation, insertion, deletion, search and display (based on the different position as specified by the user).Linked representation of Stacks & Queues, Algorithms and their Complexities		
Unit Number: 3	**Title: Trees and Graphs**	**No. of hours: 10**
Content: **Trees:** Basic Terminology, Binary Trees and their representation, expression evaluation, Complete Binary trees, Extended binary trees, traversing binary trees, Searching, Insertion and Deletion in binary search trees. AVL trees, Threaded trees, algorithms and their analysis. **Graphs:** Terminology and Representations, Graphs & Multigraphs, DirectedGraphs, Sequential representation of graphs, Adjacency matrices, Transversal Connected Component and Spanning trees, algorithms and their analysis		
Unit Number: 4	**Title: Sorting ,Searching andFile Structure**	**No. of hours: 10**

Content:

Sorting Algorithms: Introduction, Sorting by exchange, selection sort, insertion sort, Bubble sort, Selection sort, Efficiency of above algorithms, Shell sort, Performance of shell sort, Merge sort, Quick sort Algorithm analysis, Heap sort: Heap Construction, Heap sort, bottom – up, Top – down Heap sort approach.

Searching Algorithms: Straight Sequential Search, Binary Search (recursive & non–recursive Algorithm.

File structures: Physical storage media, File Organization, Linked organization of file, Inverted file, Organization records into blocks, Sequential blocks, Hash function, Indexing & Hashing

Learning Experiences

Classroom Learning Experience

1. **Interactive Lectures:** Introduce key concepts in data structures using PPTs and coding demonstrations.
2. **Conceptual Understanding**: Cover topics like arrays, linked lists, stacks, queues, trees, and graphs.
3. **Problem-Solving Sessions**: Conduct in-class exercises focused on implementing and using various data structures.
4. **Theory Assignments**: Assign theoretical problems that reinforce data structure concepts, discussed in class.
5. **Group Work**: Collaborate on projects that require designing and optimizing data structures.
6. **Case Studies**: Analyze real-world applications of data structures in software development.
7. **Continuous Feedback**: Implement quizzes and peer reviews to assess understanding and coding practices.

Outside Classroom Learning Experience

1. **Theory Assignments**: Assign take-home projects that apply data structure concepts to practical problems.
2. **Lab Projects**: Facilitate hands-on programming tasks using data structures in real-world scenarios.
3. **Question Bank**: Provide practice problems and resources for self-assessment on data structures.
4. **Online Forums**: Create platforms for discussing data structure challenges and solutions.

Text Book

□ "Schaum's Outline of Data Structures" by Seymour Lipschutz

□ "Algorithms and Data Structures: The Basic Toolbox" by Kurt Mehlhorn andPeter Sanders

Reference Book

□ E. Horowitz and S. Sahani, “Fundamentals of Data Structures”, Galgotia Book source Pvt. Ltd.

□ Data Structures & Algorithms in Python by John Canning, Alan Broder,Robert Lafore Addison-Wesley Professional.

□ "Introduction to Algorithms" by Thomas H. Cormen, Charles E.Leiserson, Ronald L. Rivest, and Clifford Stein.

Additional Readings:

Online Learning Resources for Data Structure And Algorithms

1. Interactive Platforms

- **LeetCode**: A platform offering coding challenges and problems with afocus on data structures and algorithms.
- **HackerRank**: Provides coding challenges and tutorials on algorithmsand data structures.
- **CodeSignal**: Offers coding challenges, practice problems, andassessments.

2. Online Courses

- **Coursera - Data Structures and Algorithm Specialization**:Offered by the

University of California, San Diego and National Research University Higher School of Economics.

- **edX - Algorithmic Design and Techniques**: Offered by theUniversity of California, San Diego.
- **Udacity - Data Structures and Algorithms Nanodegree**: A comprehensive program covering data structures, algorithms, andproblem-solving techniques.

3. **Video Tutorials**
   - **YouTube - Computer Science**: Channels like "mycodeschool," "Abdul Bari," and "MIT OpenCourseWare" offer in-depth tutorials ondata structures and algorithms.
   - **Khan Academy - Algorithms**: Provides beginner-friendlyexplanations and interactive exercises.
4. **Books and E-books**

- **“Introduction to Algorithms”** by Cormen, Leiserson, Rivest, andStein.
- **“Data Structures and Algorithm Analysis in C++”** by Weiss.
- **“Algorithms”** by Sedgewick and Wayne.

5. **Online Documentation and Tutorials**
   - **GeeksforGeeks - Data Structures**: Offers tutorials, articles, andproblems on various data structures and algorithms.
   - **TutorialsPoint - Data Structures**: Provides a comprehensiveoverview of data structures and algorithms with examples.
6. **Practice and Problem-Solving Sites**
   - **Project Euler**: A collection of challenging mathematical/computerprogramming problems.
   - **TopCoder**: Features competitive programming challenges andcontests.

Advanced Database Management Systems

Program Name	**MCA (Master in Computer Application)**		
Course Name: Advanced Database Management Systems	**Course Code**	**L-T-P**	**Credits**
	ENMM705	3-1-0	4
Total Contact Hours	40		
Type of Course:	Major		
Pre-requisite(s), if any: Integration/Differentiation			

Course Perspective. This course provides a comprehensive introduction to the fundamental concepts and advanced techniques of database management systems (DBMS). It is designed to equip students with the knowledge and skills required to design, implement, and manage databases effectively. The course covers a broad range of topics, including database architecture, data models, SQL, transaction management, concurrency control, database recovery, and security. The course is divided into 4 modules:

a) Database Concurrent Mechanisms

b) Distributed DBMS

c) Object-Oriented DBMS

d) Database Security

The Course Outcomes (COs). On completion of the course theparticipants will be:

COs	Statements

CO 1	**Understanding** fundamental concepts, terminology, and principles of database management systems.
CO 2	**Examining** the architecture and features of object-oriented DBMS, such ascapturing semantics and supporting object-relational database models.
CO 3	**Evaluating and optimizing** buffering and caching strategies to improvedatabase performance.
CO 4	**Analyzing and evaluating** transaction processing, concurrency controland database recovery protocols in databases.
CO5	**Designing and implementing** privilege analysis and virtual privatedatabases to control access and ensure data privacy

CO = Course outcomes. A student is expected to have learnt concepts and demonstrated/developed abilities or skills related to strategic management at the end of the course.

Course Outline:

<table>
<tr><td><b>Unit Number: 1</b></td><td><b>Title: Database Concurrent Mechanisms</b></td><td><b>No. of hours: 10</b></td></tr>
<tr><td colspan="3"><b>Content Summary:</b>
ER Model, Indexing Dependencies, Transaction processing, Transaction management, Scheduling in Databases, Serial Scheduler, Non-Serial Schedulers, Concurrency Control, Time stamp ordering Protocol, Multi version concurrency control, Validation concurrency control, Recovery Techniques in Databases, Undoing, Deferred Update, Immediate Update, Buffering and Caching in Databases, Design Elements, Cache SQL ResultSet, Cache selected fields and values, Caching serialized applications, Shadow Paging, Data Backup Architecture, Transactionlogging, Paging Architecture in Databases, Full Backup, Differential Backup.</td></tr>
<tr><td><b>Unit Number: 2</b></td><td><b>Title Distributed DBMS</b></td><td><b>No. of hours: 10</b></td></tr>
<tr><td colspan="3"><b>Content Summary:</b>
Distributed DBMS, Architecture of a DDBMS, Distributed Design of Relational Database, Variant of the 2 Tier Model, Fragmentation and Replication, Update in DDBMS, Update Strategies, Eager update, Distributed Reliability, Phase commit protocol, Asynchronous Update, Distributed transaction management, Distributed concurrency control,Distributed deadlock management.</td></tr>
<tr><td><b>Unit Number: 3</b></td><td><b>Title: Object-Oriented DBMS</b></td><td><b>No. of hours: 10</b></td></tr>
</table>

<table>
<tr><td colspan="3"><b>Content Summary:</b>
Object-Oriented DBMS Architecture, Capturing Semantics, Message Passing, OODB persistence, Object relational database model, Object relational features, Object tables, Nested tables, Varying arrays, Support for large objects, Database Migration, Migration Architecture, Heterogeneous migration versus homogeneous migration, Migration consistency, Database Transformation, Differential Querying, Migration Consistency Verification, Order violation, Consistency violation, LocalTransactions</td></tr>
<tr><td><b>Unit Number: 4</b></td><td><b>Title: Database Security</b></td><td><b>No. of hours: 10</b></td></tr>
<tr><td colspan="3"><b>Content Summary:</b>
Flex Cluster Architecture, Policy-Based Cluster, Grid Infrastructure, Kernel Requirements, Data Redaction, Database Auditing, DatabaseDiagnostic Monitor, Global User Authentication, Label Security, Data Masking, Privilege Analysis, Virtual Private Databases.</td></tr>
</table>

Learning Experience

Inside Classroom Learning Experience

1. Advanced SQL Querying and Optimization
   - Master the use of complex SQL queries including subqueries, joins, and set operations.
   - Analyze and optimize query performance using tools like query planners and execution plans.
   - Understand and apply indexing strategies to enhance performance.
2. Transaction Management and Concurrency Control
   - Explain the principles of ACID (Atomicity, Consistency, Isolation, Durability) in database transactions.
   - Implement transaction management techniques using locking, timestamp ordering, and multi-version concurrency control (MVCC).
   - Handle deadlocks and conflicts in concurrent transactions.
3. Database Design and Normalization
   - Apply normalization techniques to design efficient database schemas (up to 5NF).
   - Use Entity-Relationship (ER) models and UML diagrams to design complex databases.
   - Balance trade-offs between normalization and denormalization based on real-world needs.
4. Database Security and Access Control
   - Implement access control techniques, including role-based access control (RBAC) and mandatory access control (MAC).
   - Understand database vulnerabilities and apply best practices in database security.
   - Secure sensitive data using encryption, authentication mechanisms, and auditing.

Outside Classroom Learning Experience

1. Practical Database Application Development

- Develop full-stack applications that integrate with a database backend.
- Implement data storage and retrieval operations within web and mobile applications.
- Explore real-world databases in industries like e-commerce, healthcare, and finance.

2. Hands-on Experience with Cloud Databases

    - Deploy and manage cloud-based databases using platforms like AWS RDS, Google Cloud SQL, or Azure SQL Database.
    - Understand the cost, scalability, and performance considerations of cloud-based database solutions.
    - Explore serverless databases and database as a service (DBaaS) options for scaling.

3. Research and Independent Study

    - Conduct independent research on emerging trends in database technologies, such as blockchain databases or quantum databases.
    - Investigate the future of data storage and data processing with focus areas like AI-driven databases.
    - Present research findings in the form of technical papers, presentations, or prototypes.

Text and Reference Book

1. R. Elmasri and S.B. Navathe, 2000, Fundamentals of DatabaseSystems, 3rd Ed, AW.
2. C.J. Date, 2000, An Introduction to Database Systems, 7th ED.,Addison-Wesley.
3. Database System Concepts", 6th Edition by Abraham Silberschatz, Henry F. Korth, S. Sudarshan, McGraw-Hill.
4. Pankaj B. Brahmankar, Advanced Database Management Systems (1ed.), Tech Neo Publications, 2019. ISBN 9788194154600.
5. Soudamini Patil, Narendra Joshi, Vrushali R. Sonar, Umesh M. Patil, Yogesh B. Patil and Rohit A. Kautkar, Advanced Database ManagementSystems (1 ed.), Technical Publications, 2020. ISBN 9789389180336.

Additional Readings:

Online Learning Resources for "Database Management Systems"

1. **NPTEL-Database Management System**

    □

2. **MIT OpenCourseWare - Database Systems (6.830)**

    □ **Advanced course materials from MIT covering database systeminternals and advanced topics.**

    □ **Link: MIT OpenCourseWare - Database Systems**

3. **Oracle - Database 2-Day Developer's Guide**

    □ **Official documentation and guide for Oracle databasedevelopers.**

- **Link: Oracle - Database 2-Day Developer's Guide**

4. SQLBolt - Learn SQL with interactive exercises

- **Interactive SQL tutorials and exercises to practice databasequerying.**
- **Link: SQLBolt - Learn SQL**

AI & Machine Learning

Program Name	**MCA (Master in Computer Application)**		
Course Name:AI & Machine Learning	**Course Code**	**L-T-P**	**Credits**
	ENSP701	4-0-0	4
Total Contact Hours	40		
Type of Course:	Major		
Pre-requisite(s), if any: None			

Course Perspective. This course introduces students to the foundational aspects Artificial Intelligence and Machine Learning is a branch of study or discipline which includes theories, standards, methods and innovations of various different domains like mathematics, cognitive science, electronics and embedded systems to make intelligent systems that mimic human behaviour. Artificial Intelligence (AI) and Machine Learning (ML) focus on collecting, categorizing, strategizing, analysing and interpreting data. It also incorporates the concepts of machine learning and deep learning model building for solving various computational and real-world business problems.The course is structured into four comprehensive modules:

a) Introduction to AI
b) Problem Solving, Knowledge and Reasoning
c) Introduction to ML
d) Classification and Regression

The Course Outcomes (COs). On completion of the course the participants will be:

COs	Statements
CO1	**Understanding** the fundamental concepts of Artificial Intelligence andMachine Learning**.**
CO2	**Analyzing and applying** problem-solving techniques, knowledgerepresentation, and reasoning methods

CO3	**Designing and evaluating** classification and regression models toaddress real-world business and computational problems
CO4	**Applying** the different Machine learning methods .

CO = Course outcomes. A student is expected to have learnt concepts and demonstrated/developed abilities or skills related to strategic management at the end of the course.

Course Outline:

Unit Number:1	**Title:** Introduction to AI	**No. of hours:10**
Content: Basic Definitions and Terminology, Foundation and History of AI, Overview ofAI problems, Evolution of AI - Applications of AI, Classification/Types of AI. Artificial Intelligence vs Machine learning. Intelligent Agent: Types of AI Agent, Concept of Rationality, nature ofenvironment, structure of agents. Turing Test in AI.		
Unit Number:2	**Title:** Problem Solving, Knowledgeand Reasoning	**No. of hours:10**
Content: **Search Algorithms in Artificial Intelligence:** Terminologies, Properties ofsearch Algorithms, Types of search algorithms: uninformed search and informed search, State Space search Heuristic Search Techniques: Generate-and-Test; Hill Climbing; Properties of A* algorithm, Best-first Search; Problem Reduction. **Constraint Satisfaction problem**: Interference in CSPs; Back trackingsearch for CSPs; Local Search for CSPs; structure of CSP Problem**.** **Knowledge-Based Agent in Artificial intelligence:** Architecture, approaches to designing a knowledge-based agent, knowledge representation: Techniques of knowledge representation, Propositional logic,Rules ofInference, First-Order Logic, Forward Chaining and backward chaining in AI, **Reasoning in Artificial intelligence:** Types of Reasoning and Probabilisticreasoning, Uncertainty.		
Unit Number:3	**Title:** Introduction to ML	**No. of hours:12**
Content: **Introduction to Machine Learning:** History of ML Examples of MachineLearning Applications, Learning Types, ML Life cycle, AI & ML, dataset for		

ML, Data Pre-processing, Training versus Testing, Positive and NegativeClass, Cross-validation. **Types of Learning:** Supervised, Unsupervised and Semi-Supervised Learning. Supervised: Learning a Class from Examples, Types of supervisedMachine learning Algorithms. **Unsupervised:** Types of Unsupervised Learning Algorithm, DimensionalityReduction: Introduction to Dimensionality Reduction, Subset Selection, and Introduction to Principal Component Analysis.		
Unit Number:4	**Title: Classification and Regression**	**No. of hours:8**
Content: **Binary and Multiclass Classification:** Assessing Classification Performance, Handling more than two classes, Multiclass Classification-Onevs One, One vs Rest. **Regression:** Assessing performance of Regression – Error measures,Overfitting and Underfitting, Catalysts for Overfitting, VC Dimensions**.**		

Text and Reference Book

1. 1. Russell, S. and Norvig, P. 2015. Artificial Intelligence - A ModernApproach, 3rd edition, Prentice Hall.
2. J. Gabriel, Artificial Intelligence: Artificial Intelligence for Humans (ArtificialIntelligence, Machine Learning), Create Space Independent Publishing Platform, First edition , 2016.
3. Introduction to Artificial Intelligence & Expert Systems, Dan W Patterson,PHI., 2010 2. S Kaushik, Artificial Intelligence, Cengage

Additional Readings:

R 1. https://www.tutorjoes.in/python_programming_tutorial/
R 2. https://www.udemy.com/course/100-days-of-code/
R 3. https://favtutor.com/blog-details/7-Python-Projects-For-Beginners
R 4. https://github.com/NaviRocker/100-days-of-python
R 5. https://hackr.io/blog/python-projects

Online Learning Resources

1. Coursera

- **Courses**:
  - *AI For Everyone* by Andrew Ng
  - *Machine Learning* by Andrew Ng
  - *Deep Learning Specialization* by Andrew Ng

- **Features**: University-level courses, hands-on projects, andspecializations.

2. edX

- **Courses**:
  - *Artificial Intelligence MicroMasters* by Columbia University
  - *Machine Learning Fundamentals* by UC San Diego
  - *AI for Business* by the University of London
- **Features**: Professional certificates, MicroMasters programs, anduniversity-level courses.

3. Udacity

- **Courses**:
  - *AI for Trading*
  - *Machine Learning Engineer Nanodegree*
  - *Deep Learning Nanodegree*
- **Features**: Nanodegrees with real-world projects, mentorship, andcareer services.
- **Courses**:
  - *Introduction to Machine Learning with Python*
  - *Deep Learning in Python*
  - *Machine Learning Scientist with Python*
- **Features**: Interactive coding exercises, data science focus, and skilltracks.

5. Khan Academy

- **Courses**:
  - *Introduction to Algorithms*
  - *Statistics and Probability*
- **Features**: Free educational resources, interactive exercises, andfundamental concepts.

6. Udemy

- **Courses**:
  - *Machine Learning A-Z™: Hands-On Python & R In Data Science*
  - *Deep Learning A-Z™: Hands-On Artificial Neural Networks*
  - *Python for Data Science and Machine Learning Bootcamp*
- **Features**: Wide range of courses, often with lifetime access, andhands-on projects.

7. MIT OpenCourseWare

- **Courses**:
  - *Introduction to Deep Learning*
  - *Machine Learning with Python*
- **Features**: Free course materials from MIT, including lecture notes andassignments.

8. Fast.ai

- **Courses**:
  - *Practical Deep Learning for Coders*
  - *Deep Learning from the Foundations*
- **Features**: Free, practical courses focused on deep learning andhands-on coding.

9. Google AI

- **Courses**:
  - *Machine Learning Crash Course*
  - *TensorFlow in Practice Specialization*
- **Features**: Free resources and tutorials, TensorFlow-related courses.

10. Codecademy

- **Courses**:
  - *Learn Data Science*
  - *Learn Machine Learning*

Program Name	**MCA (Master in Computer Application)**		
Course Name: Problem Solving Lab	**Course Code**	**L-T-P**	**Credits**
	ENMM751	0-0-2	1
Total Contact Hours	40		
Type of Course:	Major		

Defined Course Outcomes

CO1	**Demonstrating** proficiency in using Python's development environment and effectively utilize basic and advanced programming constructs to solve problems.
CO 2	**Developing** Python programs that incorporate fundamental data structures, control flows, and functions to process and manipulate data
CO 3	**Constructing** object-oriented Python applications demonstrating an understanding of classes, objects, inheritance, and polymorphism to solve complex problems
CO 4	**Implementing** Python libraries for data manipulation, visualization,and database interaction to perform data analysis and build data-driven applications.

Lab Experiments

Projects	Task	Mapped CO/COs
Project 1	**Project Title: Personal Expense Tracker** **Problem Statement:** Develop a personal expense tracker thatallows users to input their daily expenses, categorize them, and generate a summary report. The tracker should utilize Python basics, string manipulation, and input/output operations. This project will help users manage their finances by tracking their spending habits and categorizing their expenses. **Tasks:** • Implement user input functionality to enter dailyexpenses. • Categorize expenses (e.g., food, travel, entertainment). • Store expenses in a file or database. • Generate a summary report showing total expenses percategory and overall spending. • Provide options to view, update, or delete past entries**.**	CO1
Project 2	**Project Title: Inventory Management System** **Problem Statement:** Design an inventory management system for a small retail store. The system should allow store managers to add, remove, and update product details. It should also provide search functionality and generate inventoryreports. This project will help store managers efficiently manage their inventory and ensure product availability.	CO2
Project 3	**Project Title: Library Management System** **Problem Statement:** Create a library management system that allows librarians to manage books and users. The system should include functionalities to add, remove, and update bookdetails, issue and return books, and generate user and book reports. This project will help librarians efficiently manage library operations and ensure book availability.	CO3
Project 4	**Project: Project Title: Snake Game** **Problem Statement**: Develop a classic Snake game using	CO3

	Python. The game should allow users to control a snake thatmoves around the screen, eating food to grow longer while avoiding collisions with the walls or its own tail. Implement the following features: • A game loop that continuously updates the game state. • User input handling for controlling the snake's direction. • Random placement of food on the screen. • Collision detection for the snake's head with the walls andits own body. • A scoring system that tracks the player's score based on thenumber of food items eaten. • A game-over screen that displays the player's final scoreand offers an option to restart the game.	
Project 5	**Project Title: Data Visualization Dashboard using Flask** **Problem Statement:** Develop a data visualization dashboardusing Flask that allows users to upload datasets (e.g., CSV files), process the data, and generate various visualizations. The dashboard should be interactive, providing options forusers to select different types of charts and filter data. **Implement the following features:** • A web interface using Flask that allows users to uploaddatasets. • Data processing and cleaning functionality. • Integration with visualization libraries like Matplotlib, Plotly,or Seaborn to create charts (e.g., bar charts, line charts, scatter plots). • Interactive features allowing users to filter data and updatevisualizations in real-time. • Options to download generated charts as images.	CO4
Project6	**Project Title: Blog Web Application using Flask** **Problem Statement:** Create a blog web application using Flask that allows users to create, read, update, and delete blogposts. Implement user authentication to manage access to certain features. Implement the following features: • A web interface using Flask for users to view blog posts. • User authentication and authorization to manage user	CO4

	accounts and restrict access to certain features. • CRUD (Create, Read, Update, Delete) operations for blogposts. • A rich text editor for writing and editing blog posts. • A commenting system where users can leave comments onblog posts. • Basic search functionality to find blog posts by keywords.	
Project 7	**Project 7: Weather Forecast Application** Problem Statement: Develop a weather forecast application that fetches weather data from an API and displays the forecast for a specified location. Users should be able to searchfor weather information based on city names and view current and future weather conditions.	Co4
Project 8	**Project 8: Student Management System** **Problem Statement:** Create a student management system that allows administrators to manage student records, including adding, updating, and deleting student information. The system should generate reports and support searching forstudents based on various criteria	Co4

Online learning resources

□ **Codecademy**

- Interactive coding platform that offers hands-on Python courses,teaching both the basics and more advanced topics in Python. Ideal for practicing specific programming tasks.
- Link: Codecademy Python Course

□ **HackerRank**

- Provides a vast range of programming problems across variousdomains of computer science, along with a dedicated Python domain. Great for practicing coding skills and understanding algorithms.
- Link: HackerRank Python

□ **LeetCode**

- Known for its extensive array of programming challenges that can help improve your understanding of data structures and algorithms. It's particularly good for preparing for technical jobinterviews.
- Link: LeetCode

- **GitHub**
    - Not just a code repository, GitHub offers collaborative features and a wealth of open-source projects where students can engagein real-world software development and contribute to ongoing projects.
    - Link: GitHub

ALGORITHMS LAB

Program Name	**MCA (Master in Computer Application)**		
Course Name: **DATA STRUCTURES AND ALGORITHMS LAB**	**Course Code**	**L-T-P**	**Credits**
	ENMM753	0-0-2	1
Total Contact Hours	40		
Type of Course:	Major		
Pre-requisite(s), if any: Basics of Computer Programming			

Lab Experiments

Defined Course Outcomes

COs	
CO 1	**Analyzing** and evaluate the time and space complexity of algorithmsfor various scenarios, demonstrating an understanding of asymptotic notations.
CO 2	**Implementing** and manipulate single-dimensional and multi-dimensional arrays, including operations like insertion,deletion, and traversal.
CO 3	**Developing** and perform operations on linked lists (single, doubly, and circularly linked), stacks, and queues using both array and linked list representations.
CO 4	**Designing** and analyze the efficiency of different sorting andsearching algorithms, as well as implement and compare

	advanced data structures like binary search trees, AVL trees,and graph algorithms.

S.N	Experiment Title	Mapped CO/COs
1	**La**b 1: **Array Operations Project** **Objective:** Implement basic array operations includinginsertion, deletion, search, and display	CO1
2	**Lab 2: Stack Implementation** • **Objective:** Implement a stack using both array andlinked list approaches.	CO1
3	**Lab 3: Linked List Operations** • **Objective:** Implement singly linked list, doubly linkedlist, and circular linked list. • **Features:** Operations for creation, insertion, deletion,search, and display	CO1
4	**Lab 4: Binary Search Tree (BST) Project** • **Objective:** Implement binary search tree with operationsfor insertion, deletion, and search.	CO2
5	**Lab 5: Stack Applications** **Problem Statement** You are developing a calculator application that evaluates postfix expressions. Implement a stack-based approach to handle postfix expressions. The calculator should supportbasic arithmetic operations (+, -, *, /).	CO2
6	**Lab 6: Queue Operations** **Problem Statement** You are designing a print job scheduler for a printer. Implement a circular queue to manage print jobs. Thequeue should support the following operations: 1. **Create Queue**: Initialize an empty print job queue. 2. **Add Job**: Add a print job to the queue. 3. **Delete Job**: Remove the next print job from the	CO2

	queue. 4. **Display Queue**: Display all pending print jobs.	
7	**Lab Assignment 7: Sequential Search EfficiencyAnalysis** **Title**: “Search Efficiency Investigation” **Problem Statement**: You are given an array of integers. Implement a sequential search algorithm to find a specific target value within the array. Analyze the time complexityof your solution.	CO2
8	**Lab Assignment 8: Sorting Algorithms ComparisonTitle**: “Sorting Showdown” **Problem Statement**: Compare the efficiency of various sorting algorithms (e.g., insertion sort, selection sort, bubble sort, quick sort, merge sort, heap sort) on different input sizes.	CO3
9	**Lab Assignment 9: Hash Table Collision Handling** **Problem Statement**: Create a hash table with collisionresolution strategies (e.g., chaining, open addressing). Test its efficiency and handle collisions appropriately.	CO3
10	**Project Title: University Course Registration System Problem Statement**: Create a university course registration system in C++ that allows students to registerfor courses, and manage course schedules and professor assignments. Use graphs to represent course prerequisitesand their relationships. Implement graph traversal algorithms (BFS and DFS) to check for valid course sequences and detect cycles in prerequisites. Use Dijkstra’s Algorithm and Floyd Warshall’s Algorithm to findthe shortest path for course completion and schedule optimization. Implement minimum spanning tree algorithms (Kruskal’s and Prim’s) to optimize resource allocation for classrooms and faculty.	CO4

AI & Machine Learning lab

Program Name:	**MCA (Master in Computer Application)**		
Course Name: **AI & Machine Learning lab**	**Course Code**	**L-T-P**	**Credits**
	ENSP751	0-0-2	1
Total Contact Hours	40		
Type of Course:	Major		

Proposed Lab Experiments

Defined Course Outcomes

COs	Statement
CO 1	**Understanding** the foundational concepts, history, and classification of Artificial Intelligence, differentiating between AIand Machine Learning.
CO 2	**Applying** various search algorithms and techniques to solve problems and understand the architecture and approaches toknowledge-based agents**.**
CO 3	**Demonstrating** knowledge of Machine Learning fundamentals,including types of learning, the ML lifecycle, and data preprocessing techniques.
CO 4	**Implementing** classification and regression techniques in ML, assessing their performance and understanding the concepts of overfitting, underfitting, and VC dimensions.

Ex. No	**Experiment Title**	**Mapped CO/COs**
P1	**Project 1: Intelligent Agent Simulation** **Problem Statement**: Develop a simulation of an intelligent agent that interacts with its environment toachieve a specific goal. The agent should demonstrate rational behavior and adapt to changes in the	CO1

	environment. - **Tasks**: - Implement different types of agents (simple reflex, model-based, goal-based,utility-based). - Simulate different environments (deterministic, stochastic, fully observable,partially observable). - Implement a rational agent that optimizesits actions based on the environment and goal	
P2	**Project 2: Implementing Search Algorithms Problem Statement:** Implement and compare various search algorithms to solve a given problem(e.g., maze navigation, pathfinding in a grid). - **Tasks:** - Implement uninformed search algorithms(BFS, DFS). - Implement informed search algorithms(A*, Best-first search). - Compare the performance of these algorithms in terms of time, space, andoptimality.	CO2
P3	**Project 3: Constraint Satisfaction Problem SolverProblem Statement:** ***Create a solver for constraint satisfaction problems (CSP) using different search techniques.*** - **Tasks:** - Implement a backtracking search algorithm for CSPs. - Integrate heuristics to improve the efficiency of the solver. - Apply the solver to real-world CSPs (e.g.,Sudoku, scheduling problems).	CO3
P4	**Project 4: Knowledge-Based Agent** **Problem Statement: Design and implement a knowledge-based agent that uses propositional**	CO4

	and first-order logic to make decisions. - **Tasks:** ○ Develop a knowledge base using propositional logic. ○ Implement inference rules for the knowledge-based agent. ○ Use forward and backward chaining to derive conclusions from the knowledgebase.	
P5	**Supervised and Unsupervised Learning Analysis Problem Statement:** Develop a comprehensive machine learning project that involves both supervisedand unsupervised learning techniques to analyze a given dataset. The project should include building and evaluating a supervised learning classifier as well as implementing clustering algorithms to uncover patterns without predefined labels**.**	Co4
P6	**Project 8: Regression Analysis** **Problem Statement**: Implement and evaluate regression models to predict a continuous targetvariable. - **Tasks:** ○ Implement linear and multiple linear regression models. ○ Assess the performance of the regression models using error measures (MSE, RMSE,MAE). ○ Analyze overfitting and underfitting, andapply techniques to mitigate them.	CO4

Verbal Ability

Program Name	**MCA (Master in Computer Application)**		
Course Name: **Verbal Ability**	**Course Code**	**L-T-P**	**Credits**
	AEC006	3-0-0	3
Total Contact Hours	36		
Type of Course:	AEC		
Pre-requisite(s), if any:			

Course Perspective. The course aims to improve language proficiency in three key areas: grammar, vocabulary and identification of grammatical errors in writing. Language proficiency enables students to comprehend lectures, understand course materials and enhances students' ability to express themselves clearly and effectively. In many professions, strong language skills are a prerequisite. Whether in business, medicine, law, or science, being able to communicate fluently and accurately is essential for collaboration, negotiation, and advancement. A strong command of verbal abilities can significantly impact job interviews. It allows candidates to answer questions confidently, demonstrate their qualifications effectively and leave a positive impression on potential employers.

The Course Outcomes (COs). On completion of the course theparticipants will be:

COs	Statements
CO1	Understanding the grammar rules and word meaning (Vocabulary).
CO 2	Applying grammar rules and vocabulary in different context & purpose

CO3	Analyzing situations/ context of communication and selecting appropriate grammar and words.
CO4	Developing sentences and paragraphs to describe and narrate a situation

CO = Course outcomes. A student is expected to have learnt concepts and demonstrated/developed abilities or skills related to strategic management at the end of the course.

Course Outline:

<table>
<tr><td><b>Unit Number: 1</b></td><td><b>Title: Vocabulary Development and Application</b></td><td><b>No. of hours: 10</b></td></tr>
<tr><td colspan="3">Content:
Understanding the concept of root words, Prefix and suffix, Ways to enhance Vocabulary, Crosswords and word quizzes, Confusing words, One word substitution, Odd one out, Synonyms and Antonyms, Commonly misspelt words, Idioms and Phrases</td></tr>
<tr><td><b>Unit Number: 2</b></td><td><b>Title: Fundamentals of Grammar and Sentence Structure</b></td><td><b>No. of hours: 8</b></td></tr>
<tr><td colspan="3"><b>Content:</b>
Content Summary: Introduction to Parts of Speech, Tenses and its ‘rules, Sentences (Simple, Compound and Complex), Subject Verb Agreement, Pronoun Antecedent agreement, Phrases and Clauses</td></tr>
<tr><td><b>Unit Number: 3</b></td><td><b>Title: Basic number system</b></td><td><b>No. of hours: 12</b></td></tr>
<tr><td colspan="3"><b>Content:</b>
Content Summary: Spot the error (grammatical errors in a sentence), Sentence Correction (Improvement of sentences based on Grammar rules), Sentence Completion, Cloze Tests</td></tr>
<tr><td><b>Unit Number: 4</b></td><td><b>Title: Number system</b></td><td><b>No. of hours: 6</b></td></tr>
<tr><td colspan="3"><b>Content:</b>
Logical Arrangement of Sentences, Comprehending passages, Contextual questions, Anagrams, Analogies</td></tr>
</table>

Additional Readings:

https://www.indiabix.com/online-test/aptitude-test/

https://www.geeksforgeeks.org/aptitude-questions-and-answers/
https://www.hitbullseye.com/

R1. Norman Lewis – Word Power Made Easy
R2. Wren & Martin – High School English Grammar & Composition
R3. R.S. Agarwal & Vikas Agarwal – Quick Learning Objective General English
R4. S.P. Bakshi - Objective General English
R 5. Praxis Groups -Campus Recruitment Complete Reference

Semester: 2

ADVANCED COMPUTER ORGANIZATION & ARCHITECTURE

Program Name:	**MCA (Master in Computer Application)**		
Course Name:	**Course Code**	**L-T-P**	**Credits**
Advanced Computer Organization & Architecture	**ENMM702**	3-1-0	4
Total Contact Hours	40		
Type of Course:	Major		
Pre-requisite(s), if any: Computer Organization and Architecture,Microprocessor			

Course Perspective. Advanced Computer Organization & Architecture (ACOA) delves into advanced topics in computer architecture and organization, concentrating on multicore processors, graphics processing units (GPUs), and heterogeneous system-on-chip (SOC) multiprocessor architectures. It examinesimplementation challenges from an architect's viewpoint. The course aims to offer thorough coverage of current and emerging trends in computer organization and architecture, with a focus on performance and thehardware/software interface. Emphasis is placed on analyzing key architectural design issues and their effects on application performance. The course isdivided into 4 units:

1. Theory of Parallelism
2. System Interconnect Architectures, Advanced processors and Pipelining
3. Memory Hierarchy Design, Multiprocessor architectures
4. Parallel Models and Languages

The Course Outcomes (COs). On completion of the course the participantswill be able to:

COs	Statements

CO 1	**Understanding** the various architectural concepts that may be applied to optimize and enhance the classical Von Neumann architecture intohigh performance computing hardware systems
CO 2	**Describing** the design issues relating to the architectural options.
CO 3	**Determining** the challenges faced in the implementation ofthese high-performance systems
CO 4	**Understanding** pipelining, instruction set architectures, memory addressing.
CO 5	**Identifying** the various techniques to enhance a processors abilityto exploit Instruction-level parallelism (ILP), and its challenges.
CO 6	**Understanding** the various models to achieve memory consistency.

CO = Course outcomes. A student is expected to have learnt concepts and demonstrated/developed abilities or skills related to strategic management at the end of the course.

Course Outline:

Unit Number:1	**Title: Theory of Parallelism**	**No. of hours: 10**
Content Summary: Parallel Computer Models, The State of Computing, Multiprocessors andMulticomputer, Multi-vector and SIMD Computers, PRAM and VLSI Models, Program and Network Properties, Conditions of Parallelism, Program Partitioning and Scheduling, Program Flow Mechanisms, System Interconnect Architectures, Principles of Scalable Performance, Parallel Processing Applications, Speedup Performance Laws, Scalability Analysis and Approaches.		
Unit Number: 2	**Title: System Interconnect Architectures, Advanced processors and Pipelining**	**No. of hours: 10**

<table>
<tr><td colspan="3"><b>Content Summary:</b>

Network properties and routing, Static interconnection Networks, Dynamic interconnection Networks, Multiprocessor system Interconnects, Hierarchical bus systems, Crossbar switch and multiport memory, Multistage and combining network.
Advanced processor technology, Instruction-set Architectures, CISC Scalar Processors, RISC Scalar Processors, Superscalar Processors.
Linear pipeline processor, nonlinear pipeline processor, Instruction pipeline Design, Mechanisms for instruction pipelining, Instruction Pipeline Principles, Pipeline Hazards Analysis, Control Hazards & Branch Prediction, MIPS
Pipeline for Multi-Cycle Operations, Dynamic instruction scheduling, Branch Handling techniques,branch prediction.</td></tr>
<tr><td><b>Unit Number:3</b></td><td><b>Title:Memory Hierarchy Design, Multiprocessor architectures</b></td><td><b>No. of hours: 10</b></td></tr>
<tr><td colspan="3"><b>Content Summary:</b>

Cache basics & cache performance, reducing miss rate and miss penalty, multilevel cache hierarchies, main memory organizations, design of memory hierarchies.
Symmetric shared memory architectures, distributed shared memory architectures, models of memory consistency, cache coherence protocols (MSI, MESI, and MOESI), scalable cache coherence, overview of directory- based approaches, design challenges of directory protocols, memory-based directory protocols, cache based directory protocols, protocol design trade-offs, synchronization.</td></tr>
<tr><td><b>Unit Number:4</b></td><td><b>Title:Parallel Models andLanguages</b></td><td><b>No. of hours: 10</b></td></tr>
<tr><td colspan="3"><b>Content Summary:</b>

Parallel Programming Models (Shared-Variable, Message passing, Data- Parallel, Object-Oriented), Parallel languages & Compilers (language features for parallelism, Parallel Program Development and Environments,Synchronization and Multiprocessing Modes, Parallel language constructs, optimizing compilers for parallelism), Code optimization & partitioning (Scalar optimization, Local & Global optimization, Vectorization, code generation & scheduling, Trace scheduling compilation).</td></tr>
</table>

Text book

I) David A. Patterson and John L. Hennessy ,“Computer Organization andDesign: The Hardware/Software Interface”,5thEdition, Elsevier

References

II) Mano M. Morris, “Computer System Architecture”, Pearson.

III) CarlHamache, “Computer Organization and Embedded Systems”, 6thEdition, McGraw Hill Higher Education

IV) “Computer Architecture and Organization”, 3rd Edition by John P. Hayes,WCB/McGraw-Hill

V) William Stallings “Computer Organization and Architecture: Designing forPerformance”, 10th Edition, Pearson Education

Additional Readings:

Online Learning References

a) **Udemy - CUDA GPU Programming Beginner to Advanced**

a. **Link**: Udemy

b. **Description**: Through this course students will be able to learn simple to advanced level concepts of Parallel Computing and GPU Programming withCUDA

b) **Coursera – Introduction to Parallel Programming with CUDA**

a. **Link**: Coursera

b. **Description:** This course covers the utilization of CUDA framework towrite C/C++ softwares that run on CPUs and Nvidia GPUs.

c) **NPTEL – Advanced Computer Architecture**

a. **Link:** NPTEL

b. **Description:** This course covers advanced processor design topics, with aspecific emphasis on GPUs, out-of-order pipelines, and compiler approaches to improve ILP

ADVANCED WEB TECHNOLOGIES

Program Name:	**MCA (Master in Computer Application)**		
Course Name: Advanced Web Technologies	**Course Code**	**L-T-P**	**Credits**
	ENMM704	3-1-0	4
Total Contact Hours	40		
Type of Course:	Major		
Pre-requisite(s), if any:			

Course Perspective. This course introduces web development, focusing on the essential technologies and tools for designing andcreating dynamic, interactive web pages. Students will learn key concepts including the World Wide Web, HTML, Cascading Style Sheets (CSS), and XML.The course is divided into 4 units:

1. JavaScript, CSS and Angular JS
2. Introduction to NodeJS
3. Basics of PHP
4. JSON, AJAX and Web Services

The Course Outcomes (COs). On completion of the course the participants willbeable to:

COs	Statements
CO 1	**Designing** and Develop Dynamic Web Pages
CO 2	**Demonstrating** comprehensive knowledge and proficiency in web technologies, including HTML, CSS, and client-side scripting

	languages.
CO 3	**Gaining** proficiency in utilizing GIT and GitHub for effective project management, version control, and collaboration in webdevelopment projects.
CO 4	**Applying** the Django web framework to design and develop websites, showcasing proficiency in creating dynamic androbust web applications.
CO 5	**Creating** interactive and responsive website using Javascript

CO = Course outcomes. A student is expected to have learnt concepts and demonstrated/developed abilities or skills related to strategic management at the end of the course.

Course Outline:

Unit Number: 1	**Title: JavaScript, CSS and Angular JS**	**No. of hours: 8**
Content Summary: **Java Script and CSS** CSS syntax, benefits, Responsive design, Bootstrap introduction, Javascript syntax, Java script inbuilt objects, Error handling and event handling, DOM, Asynchronous Programming **Introduction to Angular JS** Basics and Syntax of Angular JS, Features, Advantages, ApplicationStructure, Basics of routes and navigation, MVC with Angular JS, Services **Angular JS in Detail** Modules, Directives, Routes, Angular JS Forms and Validations, Data binding,Creating single page website using Angular JS		
Unit Number: 2	**Title: Introduction to NodeJS**	**No. of hours: 12**
Content Summary: **Introduction to Node JS** Setup Node JS Environment, Package Manager, Features, Console Object,		

<table>
<tr><td colspan="3">Concept of Callbacks
<b>Node JS in detail</b>
Events and Event Loop, timers, Error Handling, Buffers, Streams, Work with File System, Networking with Node (TCP, UDP and HTTP clients and servers),Web Module, Debugging, Node JS REST API, Sessions and Cookies, Design patterns, caching, scalability</td></tr>
<tr><td><b>Unit Number: 3</b></td><td><b>Title:Basics of PHP</b></td><td><b>No. of hours: 12</b></td></tr>
<tr><td colspan="3"><b>Content Summary:</b>
<b>PHP:</b> Starting to script on server side, Arrays, function and forms, advance PHP. Introduction to PHP, server-side scripting, php comments, variables, echo and print, PHP operators, data types, branching statements, loops.
Arrays, PHP functions, working with forms, $_GET, $_POST, $_REQUEST, String functions, include and require, session and cookie, error handling in PHP.
<b>Databases:</b> Basic command with PHP examples, Connection to server,creating database, selecting a database, listing database, listing table names creating a table, inserting data, altering tables, queries, deletingdatabase, deleting data and tables, PHP myadmin and database bugs.</td></tr>
<tr><td><b>Unit Number: 4</b></td><td><b>Title:JSON, AJAX and Web Services</b></td><td><b>No. of hours: 8</b></td></tr>
<tr><td colspan="3"><b>JSON:</b> What is JSON? JSON Syntax, JSON Data Types, JSON Objects,JSON Schema, JSON versus XML, JSON with PHP, JSON with AJAX.
<b>AJAX:</b> Introduction, AJAX from User’s Perspective, AJAX from Developer’s
Perspective, How AJAX works? Applications of AJAX
AJAX web application model, Performing AJAX validation, Handling XMLdata using PHP and AJAX, Connecting database using PHP and AJAX.
<b>Web Services:</b> Introduction, Characteristics of web services, Web services – Architecture, Web services - Components: XML-RPC, SOAP, WSDL, UDDI</td></tr>
</table>

References

I) Web Technologies, Uttam K. Roy, Oxford University Press

II) HTML Black Book, Stephen Holzner, Wiley Dreamtech.

III) Web Technology, Rajkamal, Tata McGraw-Hill.

IV) Web Technologies: A Computer Science Perspective, Jeffrey C.Jackson, Pearson.

V) XML: How to Program, Deitel & Deitel Nieto

Additional Readings:

Online Learning References

a) Khan Academy – Intro to HTML/CSS: Making webpages

- Description - Through this course students will understand how to create andstructure content, gain skills in applying styles to HTML elements and learn how to enhance the visual appeal of web pages.
- Link: Khan Academy

b) w3schools -JavaScript Tutorial

- Description - This course enables students to create interactive elements on web pages, live updates, animations, and interactive maps and validate formson client-side
- Link: w3schools

c) GeeksforGeeks – Introduction to AngularJS

- Description - Through this course students will be able to create interactive single-page applications, transform static HTML into dynamic content that adapts to user interactions and grasp command over features like data bindingand dependency injection streamline development.
- Link - GeeksforGeeks

d) Udemy – The Complete Node.js Developer Course (3rd Edition)

- Description - This course enable students to build, test, and launch Nodeapps, create Express web servers , APIs, real-time web apps with SocketIO and lastly deploy Node apps to production
- Link - NodeJS

e) Udemy – Complete JSON AJAX API Code Course Beginner to Professional

- Description - Students who complete this course will be able to create more dynamic and interactive apps, leverage AJAX requests over the web, and use JSONdata within web applications.
- Link - AJAX & JSON

f) W3Schools- PHP Tutorial

- Description - Students completing this course will be able to incorporate dynamic content with static webpages, execute server-side scripting, andintegrate databases to build robust data-driven applications.
- Link - W3Schools

ADVANCED WEB TECHNOLOGIES LAB

Program Name	**MCA (Master in Computer Application)**		
Course Name: **ADVANCED** **Web Technologieslab**	**Course Code** **ENCS754**	**L-T-P** 0-0-2	**Credits** 1
Type of Course:	Major		

Defined Course Outcomes

CO1	Designing simple webpages using HTML and CSS.
CO 2	Using GIT and GIT HUB for project management.
CO 3	Applying Django web framework to create websites.
CO 4	Creating interactive and responsive website using JavaScript.
CO5	Testing and deploying application web applications.

Proposed Lab Experiments

Projects	**Task**	**Mapped CO/COs**
Project1 and Project2	**Project Title: Personal Portfolio Website Problem Statement:** Develop a personal portfoliowebsite using HTML and CSS. The website shouldshowcase the user's skills, projects, and contactinformation. Implement basic web page structureand styling to create a professional-looking portfolio. **Project Title: Basic Company Landing Page Problem Statement:** Create a landing page for adummy company using HTML and CSS. The landingpage should include sections such as anintroduction, services,testimonials, and contact information. Ensure proper use of HTML tags andCSS for layout and design.	CO1
Project3 and Project4	**Project Title: To-Do List App** **Problem Statement**: Develop a simple to-do list and maintain its versions on GitHub. Use GitHub to track progress and manage code reviews. **Project Title: Expense Tracker** **Problem Statement:** Develop an application to track personal expenses and categorize them. Make sure to utilize GitHub to manage development and track issues.	CO2
Project 5	**Project 5: Fitness Tracker** **Problem Statement**: Design a personal fitness tracker using Django framework to keep a check on workouts, nutrition, and progress. The home page should include user authentication. After successful login, the user can log workouts, including exercise type, duration, and intensity. For Nutrition Tracking, the user can log meals and track calorie intake. The user can set fitness goals and track his/herachievement.	CO3
Project 6	**Project 6: Productive Habit Tracker Problem Statement**: Develop a web application to track personal productive habits and visualize the amount of hours spent inproductive activities. Utilize HTML to structure	CO4

	the application, CSS to style the application, Javascript to handle the application logic & interactions and Chart.js to visualize the progress with charts and graphs. Following modules should be included in the tracker: • User Authentication • Habit Management - Add, edit, and delete habits. • Progress Visualization - Display daily,weekly, and monthly statistics.	
Project 7	**Project 7: Personal Portfolio Website Testingand Deployment Problem Statement**: For the already designed project, perform the following: Unit Testing: Test individual components like contactforms and project displays. End-to-End Testing: Verify the overall user experience, including navigation and responsiveness.Accessibility Testing: Ensure the site is accessible to all users. *Deployment***:** Use GitHub Pages for deployment. Set up continuous deployment with GitHub Actions.	CO5

Online learning resources

- **Codecademy**
  - Interactive course that teaches HTML through hands-on exercises and projects, perfect for beginners.
  - Link : Codecademy - Learn HTML
- **W3Schools**
  - Interactive CSS tutorials with practical examples and exercises forlearning CSS styling and layout techniques.
  - Link: w3schools
- **Treehouse**
  - In this course, you'll learn the fundamental programming concepts andsyntax of the JavaScript programming language.
  - Link: Treehouse
- **GitHub**

- Not just a code repository, GitHub offers collaborative features and a wealth of open-source projects where students can engage in real-world software development and contribute to ongoing projects.
- Link: GitHub

LEARNING

Program Name:	**MCA (Master in Computer Application)**		
Course Name:	**Course Code**	**L-T-P**	**Credits**
Introduction of Neural Networkand Deep Learning	**ENMM706**	4-0-0	4
Total Contact Hours	40		
Type of Course:	Major		
Pre-requisite(s), if any:			

Course Perspective. The course begins by introducing the essential conceptsof neural networks, such as feed-forward neural networks andbackpropagation. Students will have the chance to learn programming languages like TensorFlow to train deep learning models. The course also covers the underlying principles of deep learning algorithms and their practical applications. The course is divided into 4 units:

1.Introduction to Neural Network
2.Feedforward Neural Network
3. Deep Learning
4. Probabilistic Neural Network and Deep Learning Research

The Course Outcomes (COs). On completion of the course the participants willbe able to:

COs	Statements
CO 1	Understanding the fundamental concepts and significance of neuralnetworks and their applications.
CO 2	Utilizing feedforward neural networks and training techniques todevelop and optimize neural network models.
CO 3	Analyzing deep learning architectures such as CNNs, RNNs, and DBNsto address overfitting and improve model performance.
CO 4	Assessing the use of probabilistic neural networks and advanced deep learning techniques in real-world applications like object recognition and natural language processing.

CO = Course outcomes. A student is expected to have learnt concepts and demonstrated/developed abilities or skills related to strategic management at the end of the course.

Course Outline:

Unit Number: 1	**Title: Basics of Neural Network**	**No. of hours: 10**
Content Summary: Basic ideas behind the Neural Network, Social Engineering about the data, Importance, and applications of Neural network Neural network concepts, Information flow in a neural network, understanding the basic structure of biological Neural Networks and ANN. Activation functions and their uses.(Sigmoid, Relu, Tanh etc.)		
Unit Number:2	**Title: Feedforward neural network**	**No. of hours: 10**
Content Summary: Linear Models, Training a Neural network, how to determine hidden layers,recurrent neural, multi-layer neural network, Risk minimization, regularization, model selection, and practical optimization.		
Unit	**Title: Deep**	**No. of hours: 10**

Number:3	**Learning**	
Content Summary: Deep Feed Forward network, bias-variance dilemma, Overfitting, dropouts, Gradient decent algorithm, Convolutional Neural Network, Recurrent Neural Network, Deep Belief Network.		
Unit Number:4	**Title: Probabilistic Neural Network andDeep Learning Research**	**No. of hours: 10**
Content Summary: Hopfield Net, Boltzmann machine, RBMs, Need of Encoders and Autoencoders, Object recognition, computer vision, natural language processing.		

Test book and Reference books

I) Goodfellow, I., Bengio,Y., and Courville, A., Deep Learning, MIT Press,2016.

II) Yegnanarayana, B., Artificial Neural Networks PHI Learning Pvt.Ltd, 2009.

III) Golub, G., H. and Van Loan, C, F, Matrix Computations, JHUPress,2013.

IV) Satish Kumar, Neural Networks: A Classroom Approach, TataMcGraw-Hill Education, 2004.

Additional Readings:

Online Learning References

(a) **Coursera - Neural Networks and Deep Learning**

- **Description -** Through this course, students will develop, train, and apply fully connected deep neural networks; apply effective neural networks; and get familiar with the major technological developments driving the rise of deep learning. Students will be able to comprehend the potential, difficulties, and outcomes of deep learning through the Deep Learning Course. It offers aroute for the students to acquire the know-how and abilities needed to use machine learning.

- Link: Coursera

(b) Udemy – Deep Learning A-Z 2024: Neural Networks, AI & ChatGPTPrize

- **Description -** Through Deep Learning Course students will be able to understand the intuition behind Artificial Neural networks, Convolutional Neural networks and Recurrent Artificial Neural networks and apply the samein practice

Link: Udemy

Program Name:	**MCA (Master in Computer Application)**		
Course Name: Introduction to Neural Network andDeep learning Lab	**Course Code**	**L-T-P**	**Credits**
	ENMM756	0-0-2	1
Type of Course:	Major		
Pre-requisite(s), if any:			

Defined Course Outcomes

COs	
CO 1	**Acquiring** a practical understanding of neural networks anddeep learning algorithms through hands-on lab experiments.
CO 2	**Developing** proficiency in implementing feedforward neuralnetworks and understanding their underlying principles.
CO 3	**Demonstrating** the ability to create and manipulate deepneural networks for solving complex real-world problems.

CO 4	**Analyzing** and evaluate the performance of neural networkmodels using appropriate evaluation metrics and techniques.

Lab Experiments

Projects	**Task**	**Mapped CO/COs**
P1	**Project 1: Neural Network-Based Medical Diagnosis System Problem Statement:** Develop a neural network-based system to diagnose diseases from patient data. **Objectives:** **1. Data Collection and Preprocessing:** ○ **Input:** Patient medical records and test results. ○ **Output:** Cleaned and pre-processed dataset ready fortraining. **2. Model Training:** ○ **Input:** Pre-processed dataset. ○ **Output:** Trained neural network model for diseasediagnosis. **3. Model Evaluation:** ○ **Input:** Test data. ○ **Output:** Evaluation metrics (accuracy, precision, recall)for the model. **4. Deployment:** ○ **Input:** Trained model. ○ **Output:** Deployed system for real-time diseasediagnosis.	CO 1
P2	**Project 2: Image Classification Using Deep Learning Problem Statement:** Develop a deep learning model to classify images of different objects for an e-commerce platform.	CO 2

	Objectives: 1. **Data Collection and Augmentation:** ○ **Input:** Image dataset of various objects. ○ **Output:** Augmented dataset for training. 2. **Model Training:** ○ **Input:** Augmented dataset. ○ **Output:** Trained deep learning model for image classification. 3. **Model Evaluation:** ○ **Input:** Test images. ○ **Output:** Evaluation metrics (accuracy, confusionmatrix) for the model. 4. **Integration:** ○ **Input:** Trained model. ○ **Output:** Integrated image classification system for thee-commerce platform.	
P3	**Project 3: Speech Recognition System Using NeuralNetworks** **Problem Statement:** Develop a speech recognition system usingneural networks for automated customer support. **Objectives:** 1. **Data Collection and Preprocessing:** ○ **Input:** Audio recordings of customer support calls. ○ **Output:** Preprocessed audio data for training. 2. **Model Training:** ○ **Input:** Preprocessed audio data. ○ **Output:** Trained neural network model for speech	CO 3

	recognition. **3. Model Evaluation:** ○ **Input:** Test audio data. ○ **Output:** Evaluation metrics (accuracy, word error rate)for the model. **4. Deployment:** ○ **Input:** Trained model. ○ **Output:** Deployed speech recognition system for real-time customer support.	
P4	**Project 4: Predictive Maintenance Using Deep Learning Problem Statement:** Develop a predictive maintenance systemusing deep learning to predict equipment failures in a manufacturing plant. **Objectives:** **1. Data Collection and Preprocessing:** ○ **Input:** Sensor data from manufacturing equipment. ○ **Output:** Preprocessed data for training. **2. Model Training:** ○ **Input:** Preprocessed sensor data. ○ **Output:** Trained deep learning model for failureprediction. **3. Model Evaluation:** ○ **Input:** Test sensor data. ○ **Output:** Evaluation metrics (precision, recall, F1-score)for the model. **4. Deployment:**	CO 4

	○ **Input:** Trained model. ○ **Output:** Deployed predictive maintenance system forreal-time monitoring and failure prediction.	
P5	**Project 5: Real-Time Object Detection System forAutonomous Vehicles Problem Statement: Develop a real-time object detectionsystem using neural networks for autonomous vehicles. Objectives:** **1. Data Collection and Preprocessing:** ○ **Input: Video data from autonomous vehiclecameras.** ○ **Output: Preprocessed video frames for training.** **2. Model Training:** ○ **Input: Preprocessed video frames.** ○ **Output: Trained neural network model for objectdetection.** **3. Model Evaluation:** ○ **Input: Test video frames.** ○ **Output: Evaluation metrics (mean averageprecision, recall) for the model.** **4. Integration:** ○ **Input: Trained model.** ○ **Output: Integrated object detection system forreal-time processing in autonomous vehicles.**	CO4

Online Learning References

1. **Deep Learning Specialization by Andrew Ng on Coursera**
    - **Deep Learning Specialization**
2. **Neural Networks and Deep Learning by Michael Nielsen**
    - **Neural Networks and Deep Learning**
3. **CS231n: Convolutional Neural Networks for Visual Recognitionby Stanford University**
    - **CS231n Course**
4. **Deep Learning by Ian Goodfellow, Yoshua Bengio, and AaronCourville**
    - **Deep Learning Book**
5. **MIT OpenCourseWare - Introduction to Deep Learning**
    - **MIT OpenCourseWare**

Program Name:	**MCA (Master in Computer Application)**		
Course Name: **Communication and Personality Development**	**Course Code**	**L-T-P**	**Credits**
	AEC007	3-0-0	3
Total Contact Hours	36		
Type of Course:	AEC		
Pre-requisite(s), if any:			

Course Perspective: The course enhances public speaking and presentation skills, helps students confidently convey ideas, information & build self-reliance and competence needed for career advancement. Personality assessments like the Johari Window and Myers & Briggs Type Indicator (MBTI) provide frameworks to enhance self-understanding, helps people increase their self-awareness, understand and appreciate differences in others and apply personality insights to improve their personal and professional effectiveness. Interpersonal skills included in the course deal with important topics like communication, teamwork and leadership, vital for professional success.

The Course Outcomes(COs). On completion of the course the participants will be:

COs	Statements
CO1	Improving public speaking and presentation abilities to confidently convey ideas and information.
CO2	Understanding the framework of Communication to augment oratory skills and written English
CO3	Cultivate essential soft skills required at the different workplaces.

CO = Course outcomes. A student is expected to have learnt concepts and demonstrated/developed abilities or skills related to strategic management at the end of the course.

Course Outline:

Unit	**Title: Developing self and others**	**No. of hours:8**
Number: 1		
Content Summary: Self Awareness, Personality Concepts (Personality Assessments -Johari Window, Myers & Brigg), Self-Management, Self Esteem, Self-Efficacy, Interpersonal skills, mindset, grit and working in teams.		
Unit Number: 2	**Title: Enhancing Reading and Writing Skills**	**No. of hours:6**
Content Summary: Speed reading and its importance in competitive examinations, techniques for speed reading, note-taking, and critical analysis. Paragraph Writing, Essay and Summary writing, Business Letter, Email writing		
Unit Number: 3	**Title: Effective Communication and Public Speaking**	**No. of hours:7**
Content Summary: Communication Framework, barriers & overcoming these barriers, Group Discussions, Extempore & Public Speaking drills, to manage stage fright and anxiety. Structuring and organizing a presentation (Oral & PPT), Etiquettes, Grooming, Body		

Language and Conversation starters, TMAY.		
Unit Number: 4	**Title: Career Guide and readiness**	**No. of hours:15**
Content Summary: Cover Letter, ATS friendly resume, Elevator Pitch, Video Resume (Visume), Networking, Group Discussion, Mock Interviews. Capstone Project		

References

R1 Talking to Strangers – Malcom Gladwell
R2 Fierce Conversation - Scot Susan
R3 Public Speaking - William S. Pfeiffer, Pearson
R4 Soft Skills for Everyone – Jeff Butterfield
R5 Business Communication – Rajendra Pal, J S Korlahalli
R6 The power of Positive Attitude -Roger Fritz
R7 Believe in Yourself – Dr. Joseph Murphy
J. Additional Readings

- **Websites & MOOCs**

  www.16personalities.com
  www.tonyrobbins.com
- **Specific Research Papers**

  GALLUP PRESS RESEARCH
  FRANKLIN COVEY LEADERSHIP CENTRE
- **Videos**

  The 7 Habits of Highly Effective People, Dr. Stephen R. Covey
  I Am Not Your Guru, Tony Robbins
- **Podcast**

  The Tim Ferriss Show
- **Magazines**

  SUCCESS Magazine
- **Journals**

  The IUP Journal of Soft Skills

DISCIPLINE SPECIFIC ELECTIVE - (AI & MACHIINELEARNING)

NATURAL LANGUAGE PROCESSING

Program Name:	**MCA (Master in Computer Application)**		
Course Name:	**Course Code**	**L-T-P**	**Credits**
NATURAL LANGUAGE PROCESSING	**ENSP302**	4-0-0	4
Total Contact Hours	**40**		
Type of Course:	Department Specific Elective I		
Pre-requisite(s), if any: Strong programming skills, particularly in Python.			

Course Perspective. The ultimate objective of NLP is to read, decipher,understand, and make sense of human languages in a manner that is valuable. It helps resolve ambiguity in language and adds useful numeric structure to the data for many downstream applications, such as speech recognition. The course is divided into 4 modules:

- Introduction to NLP
- Text Representation
- Information Extraction
- NLP for Social Media

The Course Outcomes (COs): On completion of the course the participants will be able to:

COs	Statements

CO 1	**Resolving** ambiguity in language and adds useful numeric structureto the data for many downstream applications, such as speech recognition or text analytics.
CO 2	**Building** systems that can make sense of text automatically.
CO 3	**Performing** tasks like translation, spell check, or topic classification

CO = Course outcomes. A student is expected to have learnt concepts and demonstrated/developed abilities or skills related to strategic management at the end of the course.

Course Outline:

<table>
<tr><td><b>Unit Number: 1</b></td><td><b>Title: Introduction to NLP</b></td><td><b>No. of hours: 8</b></td></tr>
<tr><td colspan="3"><b>Content Summary:</b> Natural Language Processing in real world, What islanguage, Approached to NLP
<b>Build NLP model:</b> Eights Steps for building NLP Model, Web Scrapping</td></tr>
<tr><td><b>Unit Number: 2</b></td><td><b>Title: Text Representation</b></td><td><b>No. of hours: 8</b></td></tr>
<tr><td colspan="3"><b>Content Summary:</b> Basic Vectorization, One-Hot Encoding, Bag of Words, Bag of N Grams, TF-IDF, Pre-trained Word Embedding, Custom Word Embeddings, Vector Representations via averaging, Doc2Vec Model,Visualizing Embeddings using TSNW and Tensor baord
<b>Text Classification:</b> Application of Text Classification, Steps for building text classification system, Text classification using Naïve Bayes Classifier, Logistic Regression, and Support Vector Machine, Neural embedding for Text Classification, text classification using deep learning, interpret text classification model</td></tr>
<tr><td><b>Unit Number: 3</b></td><td><b>Title: Information Extraction</b></td><td><b>No. of hours: 8</b></td></tr>
<tr><td colspan="3"><b>Content Summary:</b> Applications of Information Extraction, Processes for Information Extraction. Key phrase Extraction, Named Entity Recognition, Disambiguation and linking of named entity, Relationship extraction <b>Chatbot:</b> Real life applications of chatbot, Chatbot Taxonomy, Dialog Systems, Process of building a dialog, Components of Dialog System, End to End Approach, Rasa NLU</td></tr>
</table>

Unit Number: 4	Title: NLP for social media	No. of hours: 8
Content Summary: Application of NLP in social media, challenges with social media, Natural Language Processing for Social Data, UnderstandingTwitter Sentiments, Identifying memes and Fake News **NLP for E-Commerce**: E-commerce catalog, Search in E-Commerce, Howto build an e-commerce catalog, Review and Sentiment Analysis, Recommendations for E- Commerce		

Text Books:

1. Natural Language Processing with Python by Steven Bird, Ewan Kleinand Edward Loper
2. Foundations of Statistical Natural Language Processing by ChristopherManning and Hinrich Schütze

References

1. "Speech and Language Processing" by Dan Jurafsky and James H.Martin
2. "Natural Language Processing in Action: Understanding, Analyzing, andGenerating Text with Python" by Hobson Lane, Cole Howard, and Hannes Hapke
3. "Neural Network Methods for Natural Language Processing" by YoavGoldberg
4. "Introduction to Information Retrieval" by Christopher D. Manning,Prabhakar Raghavan, and Hinrich Schütze
5. "Practical Natural Language Processing: A Comprehensive Guide toBuilding Real-World NLP Systems" by Sowmya Vajjala, BodhisattwaMajumder, Anuj Gupta, and Harshit Surana

Additional Readings:

Online Learning Resources:

1. **Stanford University - Natural Language Processing with DeepLearning (CS224N)**

- Offered by Stanford University, this course covers the foundational concepts of NLP along with the application of deep learning models in NLP tasks. The lectures are available for free.
- Link: Stanford CS224N

2. **NLTK Book - Natural Language Processing with Python**

- An excellent resource for beginners, this book provides a practical introduction to NLP with Python, using the NaturalLanguage Toolkit (NLTK). It's available for free online.
- Link: NLTK Book

3. **fast.ai - Practical Deep Learning for Coders**

- Though not exclusively focused on NLP, this course includes several lessons on applying deep learning to natural language processing. It is practical, with a focus on coding and real-worldapplications.
- Link: fast.ai Course

Open-Source Society University (OSSU)

I) **OSSU Computer Science**

a. OSSU provides an open-source curriculum for learning computerscience. While it covers a broad range of topics in computer science, it includes resources for learning about artificial intelligence and machine learning, which are relevant to studentsinterested in NLP.

b. Link: OSSU Computer Science

Program Name:	**MCA (Master in Computer Application)**		
Course Name:	**Course Code**	**L-T-P**	**Credits**
NATURAL LANGUAGE Processing Lab	**ENSP352**	0-0-2	1
Type of Course:	Department Specific Elective I		
Pre-requisite(s), if any: Strong programming skills, particularly in Python.			

Defined Course Outcomes

COs	
CO 1	**Demonstrating** proficiency in implementing and utilizing basic and advanced natural language processing (NLP) techniques for textanalysis and processing.
CO 2	**Developing,** fine-tune, and evaluate machine learning and deep learning models for various NLP tasks, including text classification,named entity recognition, and sentiment analysis.
CO 3	**Implementing** and assess advanced text and speech processingmodels, focusing on performance improvement and practical applications in real-world scenarios.
CO 4	**Analyzing** and address ethical considerations, fairness, and privacyimplications in NLP applications, ensuring responsible and unbiased technology deployment.

Lab Experiments

Projects	**Task**	**Mapped CO/COs**
P1	**Project 1: Sentiment Analysis for Social MediaMonitoring** **Problem Statement:** Develop a sentiment analysistool to monitor and analyze sentiments on social media platforms in real-time. **Objectives:** 1. **Text Processing and Analysis:** ○ **Input:** Social media posts dataset. ○ **Output:** Cleaned and preprocessed textdata ready for analysis. 2. **Language Modeling:** ○ **Input:** Preprocessed text data.	CO 1

	o **Output:** Trained N-gram language modelfor sentiment prediction. **3. Model Training and Evaluation:** o **Input:** Sentiment-labeled dataset. o **Output:** Sentiment analysis model withhigh accuracy and evaluation metrics. **4. Real-Time Monitoring:** o **Input:** Live social media feed. o **Output:** Real-time sentiment analysis dashboard. o	
P2	**Project 2: Named Entity Recognition for LegalDocument Processing** **Problem Statement:** Implement a named entityrecognition (NER) system to identify and classify entities in legal documents. **Objectives:** **1. Data Preparation:** o **Input:** Legal documents dataset. o **Output:** Annotated text data with entitylabels. **2. Model Training:** o **Input:** Annotated dataset. o **Output:** Trained NER model using a pre-trained transformer. **3. Evaluation and Fine-Tuning:** o **Input:** Evaluation metrics and validationdata. o **Output:** Fine-tuned NER model with improved accuracy. **4. Deployment:** o **Input:** Trained NER model. o **Output:** Deployed NER system for processing new legal documents. o	CO 2
P3	**Project 3: Multilingual Machine TranslationSystem** **Problem Statement:** Develop a multilingual machine translation system to translate text betweenmultiple languages. **Objectives:** 1. **Data Collection and Preprocessing:**	CO 3

	o **Input:** Multilingual text corpus. o **Output:** Preprocessed parallel text datafor training. **2. Model Training:** o **Input:** Preprocessed text data. o **Output:** Trained sequence-to-sequencemodel with attention mechanism. **3. Evaluation and Tuning:** o **Input:** Translation evaluation metrics(e.g., BLEU score). o **Output:** Improved translation modelwith high accuracy. **4. Application Development:** o **Input:** Trained translation model. o **Output:** Multilingual translation application for end-users. o	
P4	**Project 4: Speech Recognition for AutomatedCustomer Support** **Problem Statement:** Implement a speech recognition system to transcribe customer supportcalls and provide automated responses. **Objectives:** **1. Data Collection and Preparation:** o **Input:** Customer support call recordings. o **Output:** Transcribed and labeled audiodata for training. **2. Model Training:** o **Input:** Preprocessed audio data. o **Output:** Trained speech-to-text modelwith high accuracy. **3. Evaluation and Fine-Tuning:** o **Input:** Evaluation metrics and validationdata. o **Output:** Fine-tuned speech recognitionmodel. **4. Integration:** o **Input:** Trained model and customersupport system. o **Output:** Integrated speech recognitionsystem for real-time call transcription and response.	CO 4

	o	
P5	**Project 5: Ethical Text Generation for CreativeWriting** **Problem Statement:** Develop an ethical text generation model for creative writing applications,ensuring bias and fairness considerations. **Objectives:** **1. Data Collection and Preprocessing:** ○ **Input:** Creative writing corpus. ○ **Output:** Preprocessed text data formodel training. **2. Model Training:** ○ **Input:** Preprocessed text data. ○ **Output:** Trained language model for text generation. **3. Bias and Fairness Analysis:** ○ **Input:** Generated text samples. ○ **Output:** Analysis of bias and fairness ingenerated text. **4. Ethical Text Generation:** ○ **Input:** Bias mitigation techniques. ○ **Output:** Improved text generation modelwith ethical considerations.	CO4

IMAGE PROCESSING & COMPUTER VISION

MCA	**Master of Computer Application**		
Course Name: **Image Processing & Computer Vision**	**Course Code**	**L-T-P**	**Credits**
	ENSP304	4-0-0	4
Total Contact Hours	40		
Type of Course:	Department Specific Elective I		
Pre-requisite(s), if any: (1) Linear Algebra and (2) programming inpython			

Course Perspective. This course introduces students to the fundamental concepts and techniques of image processing and computervision. It combines theoretical knowledge with practical applications, allowing students to understand and implement various algorithms for image enhancement, restoration, segmentation, and object recognition. The course aims to equip students with the skills needed to address real- world challenges in areas such as robotics, medical imaging, surveillance, and multimedia applications. The course is divided into 4 modules:

a) Basic Concepts of Image Formation
b) Image Restoration and Coloring
c) Image Compression and Segmentation
d) Object Representation and Computer Vision Techniques

The Course Outcomes (COs). On completion of the course the participants will be:

COs	Statements
CO1	**Remembering** key concepts, definitions, and algorithms in image processing and computer vision.
CO2	**Understanding** the principles and applications of essential techniqueslike edge detection and feature extraction.

CO3	**Applying** Use image processing methods to enhance digital images andanalyze the effects.
CO4	**Evaluating** the effectiveness of various computer vision models indifferent contexts
CO5	**Design** and implement projects that integrate multiple imageprocessing algorithms to solve complex problems.

CO = Course outcomes. A student is expected to have learnt concepts and demonstrated/developed abilities or skills related to strategic management at the end of the course.

Course Outline:

<table>
<tr><td><b>Unit Number: 1</b></td><td><b>Title: Title: Introduction to Basic Concepts of Image Formation</b></td><td><b>No. of hours: 8</b></td></tr>
<tr><td colspan="3"><b>Content Summary:</b>
Fundamentals and Applications of image processing, Image processing system components, Image sensing and acquisition, Sampling and quantization, Neighbors of pixel adjacency connectivity, regions and boundaries, Distance measures. Image Enhancement: Frequency and Spatial Domain, Contrast Stretching, Histogram Equalization, Low pass andHigh pass filtering</td></tr>
<tr><td><b>Unit Number: 2</b></td><td><b>Title: Image Restoration and coloring</b></td><td><b>No. of hours: 10</b></td></tr>
<tr><td colspan="3"><b>Content Summary:</b>
Model of The Image Degradation Restoration Process, Noise Models, Restoration in the presence of Noise Only Spatial Filtering, Periodic Noise Reduction by Frequency Domain Filtering, Linear Position Invariant Degradations, Estimation of Degradation Function, Inverse filtering, Wienerfiltering, Constrained Least Square Filtering, Geometric Mean Filter, Geometric Transformations.
Colour Image Processing, Image Segmentation, Texture Descriptors, ColourFeatures, Edges/Boundaries, Object Boundary and Shape Representations, Interest or Corner Point Detectors, Speeded up Robust Features, Saliency.</td></tr>
<tr><td><b>Unit Number: 3</b></td><td><b>Title: Image Compression and Segmentation</b></td><td><b>No. of hours: 8</b></td></tr>
<tr><td colspan="3"><b>Content Summary:</b>
Data Redundancies, Image Compression models, Elements of InformationTheory, Lossless and Lossy compression, Huffman Coding, Shanon-Fano</td></tr>
</table>

Coding, Arithmetic Coding, Golomb Coding, LZW Coding, Run Length Coding, Loss less predictive Coding, Bit Plane Coding, Image compressionstandards. Image Segmentation and Morphological Image Processing: Discontinuity based segmentation, similarity-based segmentation, Edge linking and boundary detection, Threshold, Region based Segmentation Introduction toMorphology, Dilation, Erosion, Some basic Morphological Algorithms Object		
Unit Number: 4	**Title: Object Representation and Computer Vision Techniques**	**No. of hours: 12**
Content Summary: Representation and description and Computer Vision Techniques: Introduction to Morphology, Some basic Morphological Algorithms, Representation, BoundaryDescriptors, Regional Descriptors, Chain Code, and Structural Methods. Review of Computer Vision applications; Artificial Neural Network for Pattern Classification, Convolutional Neural Networks, Machine Learning Algorithms and their Applicationsin Image Segmentation, Motion Estimation and Object Tracking, Gesture Recognition, Face and FacialExpression Recognition, Image Fusion		

References

1. Computer Vision: Algorithms and Applications by Richard Szeliski
2. Computer Vision and Image Processing: Fundamentals andApplications by Manas Kamal Bhuyan
3. Gonzalez Rafael C. and Woods Richard E., Digital Image Processing,New Delhi: Prentice– Hall of India.
4. M.K. Bhuyan, Computer Vision and Image Processing: Fundamentalsand Applications, CRC Press, USA.

Additional Readings:

Online Learning References:

I) **Fast.ai: Practical Deep Learning for Coders**

a. Practical course on deep learning, including applications in imageprocessing and computer vision.

b. Link: Fast.ai - Practical Deep Learning for Coders

II) **Deep Learning for Computer Vision with Python by AdrianRosebrock**

a. A comprehensive book on deep learning techniques for computervision applications.

b. Link: PyImageSearch - Deep Learning for Computer Vision withPython

III) **GitHub: OpenCV Projects and Tutorials**

a. Repository of projects and tutorials on OpenCV, a popular libraryfor computer vision.

b. Link: GitHub - OpenCV

IV) **Towards Data Science: Image Processing Tutorials**

a. A collection of tutorials on various image processing techniquesand applications.

b. Link: Towards Data Science - Image Processing

V) **IEEE Xplore Digital Library: Image Processing and ComputerVision Papers**

a. Access to research papers and articles on the latest developments in image processing and computer vision.

b. Link: IEEE Xplore Digital Library

IMAGE PROCESSING & COMPUTER VISION LAB

Program Name:	**MCA (Master in Computer Application)**		
Course Name: **Image Processing & ComputerVision Lab**	**Course Code**	**L-T-P**	**Credits**
	ENSP354	0-0-2	1
Type of Course:	Department Specific Elective-I		
Pre-requisite(s), if any: (1) Linear Algebra and (2) programming inpython			

Defined Course Outcomes

COs	Statements
CO 1	**Applying** image processing techniques using Python libraries.
CO 2	**Analyzing** and evaluate the effectiveness of different imageenhancement algorithms.
CO 3	**Implementing** image restoration algorithms and evaluate theirperformance in the presence of noise.
CO 4	**Developing** image compression algorithms and analyze their impacton image quality.
CO 5	**Formulating** computer vision techniques such as object detectionand tracking, gesture recognition, and facial expression recognition using Python.

Proposed Lab Experiments

Ex. No	Experiment Title	Mapped CO/COs
P1	**Project Title: Comprehensive Image Enhancement and Analysis** **Problem Statement:** Develop a comprehensiveimage enhancement and analysis pipeline. This project will involve basic image acquisition, enhancement techniques, and pixel connectivity analysis.	CO 1

	Objectives: 1. Implement image acquisition and display techniques. 2. Perform sampling and quantization on imagesand analyze the effects. 3. Develop a program for pixel connectivity analysis. 4. Implement image enhancement using histogram equalization and contraststretching. 5. Apply low pass and high pass filtering techniques in both frequency and spatial domains.	
P2	**Project Title: Advanced Image Restoration andColor Processing** **Problem Statement:** Develop advanced image restoration techniques and color processing methods. The project will involve simulating imagedegradation, applying restoration techniques, and implementing color image processing methods. **Objectives:** 1. Simulate image degradation using differentnoise models. 2. Implement image restoration techniques such as inverse filtering and Wiener filtering. 3. Estimate degradation functions and restore images using constrained least square filtering. 4. Develop techniques for color imageenhancement. 5. Implement edge detection and boundary detection algorithms for color images.	CO 1
P3	**Project Title: Efficient Image Compression andSegmentation Techniques** **Problem Statement:** Develop efficient image compression and segmentation techniques. Theproject will involve implementing various compression methods, performing image segmentation, and applying morphological operations. **Objectives:** 1. Implement Huffman coding for image	CO 1

	compression and compare it with other compression methods. 2. Develop lossless predictive coding techniquesfor image compression. 3. Implement bit plane coding for image compression and analyze the results. 4. Perform image segmentation using thresholding and region-based segmentationtechniques. 5. Apply morphological operations such asdilation and erosion on binary images.	
P4	**Project Title: Advanced Object Representationand Computer Vision Applications** **Problem Statement:** Develop advanced techniques for object representation and computervision applications. The project will involve extracting object descriptors, performing object tracking, and implementing machine learning algorithms for image classification and recognition.**Objectives:** 1. Extract boundary and regional descriptors from images. 2. Implement object tracking using motion estimation techniques. 3. Develop a simple Convolutional Neural Network (CNN) for image classification. 4. Implement gesture recognition using machinelearning algorithms. 5. Develop a face and facial expression recognition system using artificial neural networks.	CO 1

INTRODUCTION TO GENERATIVE AI

Program Name:	**MCA (Master in Computer Application)**		
Course Name: **Introduction to Generative AI**	**Course Code**	**L-T-P**	**Credits**
	ENSP306	4-0-0	4
Total Contact Hours	40		
Type of Course:	Department Specific Elective -I		
Pre-requisite(s), if any:			

Course Perspective: This course provides an in-depth introduction to the principles, techniques, and applications of Generative Artificial Intelligence (AI). Generative AI is a rapidly evolving field that has the potential to transform numerous industries by enabling machines to create content, predict outcomes, and enhance decision-making processes. This course will cover foundational concepts, the latest advancements in generative models, and practical applications, ensuring students gain a comprehensiveunderstanding of the subject. The course is divided into 4 modules:

a) Foundations of Generative AI
b) Deep Learning
c) Generative Adversarial Networks
d) Applications and Future Directions

The Course Outcomes (COs). On completion of the course the participants will be:

COs	Statements
CO1	**Understanding** generative AI principles, recent advancements, andapplications.
CO2	**Applying** probability theory and statistics in generative AI tasks.
CO3	**Designing** deep learning models for generative tasks.
CO4	**Evaluating** generative models for specific applications.

CO5	**Critically** assess ethical implications and propose solutions forgenerative AI.

CO = Course outcomes. A student is expected to have learnt concepts and demonstrated/developed abilities or skills related to strategic management at the end of the course.

Course Outline:

Unit Number: 1	**Title: Foundations of Generative AI**	**No. of hours: 12**
Content: **Introduction to Generative AI:** Definition, working and applications ofgenerative AI, Historical overview and recent advancements, Ethical considerations and societal impact. **Probability and Statistics for Generative AI:** Probability distributions andrandom variables, Maximum likelihood estimation, Bayesian inference and generative models. Fundamentals of Deep Learning: Neural networks and their architectures, Backpropagation and optimization algorithms, Transfer learning and pre-trained models.		
Unit Number: 2	**Title:Deep Learning**	**No. of hours: 8**
Content: Overview of generative models: Gaussian Mixture Models, Hidden Markov Models; Representation learning and latent variables; Autoencoders: Basics ofautoencoders and their applications, Encoder and decoder architectures, Reconstruction loss and latent space representation; Variational autoencoders (VAEs): Introduction to VAEs, reparameterization.		
Unit Number: 3	**Title: Generative Adversarial Networksand Flow-based Models**	**No. of hours: 8**
Content Summary: Generative Adversarial Networks (GANs): Introduction, Architecture of GANs, Training GANs and understanding the loss functions; Autoregressive Models (including information-theoretic foundations) Flow-based generative models and their advantages, Normalizing flows and invertible transformations, Training and sampling from flow-based models,		

Evaluation of Generative Models: Metrics for evaluating generative models(log-likelihood, Inception Score)		
Unit Number: 4	**Title:Applications and Future Directions**	**No. of hours: 12**
Content Summary: **Real-World Applications of Generative AI:** Image synthesis and editing, Data augmentation and data generation, Generative AI in healthcare, gaming,and art; Ethical Considerations and Challenges: Bias and fairness in generative models, Deepfakes and misinformation, Responsible AI practices; Emerging Trends and Future Directions: Reinforcement learning and generative models, Meta-learning and few-shot generation, OpenAI's DALL-E.		

Textbooks

1. Generative Deep Learning, by David Foster, 2nd Edition, O'ReillyMedia, Inc.
2. Deep Learning by Ian Goodfellow, Yoshua Bengio and Aaron Courville , The MIT Press
3. PATTERN RECOGNITION AND MACHINE LEARNING by Christopher M.Bishop
4. Natural Language Processing with Python" by Steven Bird, EwanKlein, and Edward Loper

Online Learning References:

I) **Deep Learning Specialization by Andrew Ng (Coursera)**:

   a. Link

II) **Generative Adversarial Networks (GANs) Specialization (Coursera)**:

   a. Link

III) **The GANs Playground (TensorFlow)**:

   a. Link

IV) **MIT OpenCourseWare - Introduction to Deep Learning**:

   a. Link

GENERATIVE AI LAB

Program Name:	**MCA (Master in Computer Application)**		
Course Name: **Generative AI Lab**	**Course Code**	**L-T-P**	**Credits**
	ENSP356	0-0-2	1
Type of Course:	Department Specific Elective I		
Pre-requisite(s), if any: NA			

Lab Experiments

Defined Course Outcomes

COs	
CO 1	**Designing,** developing, and evaluating deep neural networks for complex image classification tasks using advanced model improvementtechniques.
CO 2	**Implementing** and analyze variational autoencoders (VAEs) for anomaly detection in network traffic data, focusing on security threat identification.
CO 3	**Developing** and enhance generative adversarial networks (GANs) forrealistic image synthesis, emphasizing model architecture and performance improvement.
CO 4	**Utilizing** generative AI techniques for data augmentation in healthcare and other creative fields, assessing the impact onmodel performance and addressing ethical considerations.

Project No.	**Project Detail**	**Mapped CO/COs**
P1	**Project 1: Random Data Analysis**	CO1

	Generate random samples from various probability distributions using Python and analyze the generated data. - Choose different probability distributions - (e.g., normal distribution, uniform distribution, exponential distribution). - Generate random samples from each distribution using appropriate libraries or functions. - Visualize the generated samples for each distribution by plotting histograms or density plots - Compute and display summary statistics - (e.g., mean, standard deviation) - Compare the characteristics of the generated samples from different distributions and analyzetheir differences.	
P2	**Project 2: Parameter Estimation for aProbability Distribution** **Problem Statement:** In this project, you will implement the maximum likelihood estimation (MLE)algorithm to estimate the parameters of a selected probability distribution based on a given dataset. - Choose a probability distribution (e.g., normal distribution, exponential distribution) and specify its probability density function (PDF) or probability mass function (PMF). - Generate a synthetic dataset based on the chosen distribution with known parameters. - Implement the MLE algorithm using Python to estimate the parameters of the distribution from the synthetic dataset. - Compare the estimated parameters with the known true parameters and calculate the estimation error. - Repeat the process for multiple iterations with	CO2

	different dataset sizes to analyze the performanceof the MLE algorithm with varying sample sizes.	
P3	**Project 3: Exploring Pre-trained Generative AIModels** **Problem Statement:** Develop and train a GenerativeAdversarial Network (GAN) to generate realistic images from the MNIST dataset of handwritten digits.**Objectives:** 1. **Building the GAN Architecture:** • **Input:** MNIST dataset. • **Output:** Defined GAN architecture including the discriminator and generator. 2. **Training the GAN:** • **Input:** Training data and GAN training algorithms. • **Output:** Trained GAN capable of generatingrealistic handwritten digit images. 3. **Evaluating the GAN:** • **Input:** Generated images and evaluation metrics. • **Output:** Qualitative and quantitative evaluationof GAN performance. 4. **Improving GAN Performance:** • **Input:** Advanced GAN techniques (e.g., Wasserstein GAN, WGAN-GP). • **Output:** Improved GAN with enhanced imagequality and stability.	CO3
P4	**Project 4: Generative AI for Data Augmentationin Healthcare** **Problem Statement:** Use generative AI techniquesto augment a medical image dataset, enhancing the training process for a disease detection model. **Objectives:**	CO4

	1. **Data Augmentation Using VAEs:** • **Input:** Medical image dataset. • **Output:** Augmented dataset using a trained VAE to generate additional medical images. 2. **Training the Disease Detection Model:** • **Input:** Augmented dataset. • **Output:** Enhanced disease detection model with improved accuracy due to augmented training data. 3. **Evaluating the Augmented Model:** • **Input:** Evaluation metrics and test data. • **Output:** Detailed evaluation of model performance with and without data augmentation. 4. **Ethical Considerations:** • **Input:** Data augmentation techniques andgenerated images. • **Output:** Analysis of ethical considerations and potential biases introduced by data augmentation.	
P5	**Project 5: Future Applications of Generative AIin Art and Entertainment** **Problem Statement:** Explore and implement generative AI techniques to create digital art and interactive content for the entertainment industry.**Objectives:** 1. **Image Synthesis for Digital Art:** • **Input:** Art datasets and generative AI models. • **Output:** Generated digital art pieces using GANs and VAEs.	CO4

	2. **Interactive Content Generation:** • **Input:** Game development tools and generativemodels. • **Output:** Interactive game content generated using AI techniques. 3. **Ethical and Fairness Considerations:** • **Input:** Generative models and generated content. • **Output:** Analysis of bias, fairness, and ethical implications in generated art and entertainmentcontent. 4. **Exploring Emerging Trends:** • **Input:** Recent advancements in generative AI. • **Output:** Research and implementation of emerging trends like reinforcement learning and meta-learning in creative applications.	

TRANSFER LEARNING

Program Name:	**MCA (Master in Computer Application)**		
Course Name:	**Course Code**	**L-T-P**	**Credits**
Transfer Learning	**ENSP308**	4-0-0	4
Total Contact Hours	40		
Type of Course:	Department Specific Elective I		
Pre-requisite(s), if any:			

Course Perspective. This course introduces students to the key concepts and advanced techniques of transfer learning, a critical area in machine learning and deep learning. Transfer learning involves utilizing knowledge acquired in one domain to enhance learning in another domain. The course covers theoretical foundations, practical implementations, and applications across various fields, equipping students with the skills needed to apply transfer learning to real-world problems .The course is divided into 4 modules:

a) Foundations of Transfer Learning
b) Advanced Transfer Learning Techniques
c) Multi-task and Theoretical Aspects of Transfer Learning
d) Applications of Transfer Learning

The Course Outcomes (COs). On completion of the course the participants will be:

COs	Statements
CO1	**Understanding** transfer learning concepts and principles.
CO2	**Applying** transfer learning techniques to real-world problems.
CO3	**Analyzing** and evaluate the performance of transfer learning algorithms

CO4	**Synthesizing** and adapt pre-trained models for specific tasks.
CO5	**Critically** think, propose innovative approaches, and effectivelycommunicate transfer learning concepts

CO = Course outcomes. A student is expected to have learnt concepts and demonstrated/developed abilities or skills related to strategic management at the end of the course.

Course Outline:

Unit Number: 1	**Title:Introduction to TransferLearning**	**No. of hours: 8**
Content Summary: Fundamentals of transfer learning, motivation, and applications. Pretrained models and their usage. Fine-tuning and feature extraction. Transfer learning frameworks and libraries. Transfer learning strategies and techniques. Evaluationand performance metrics for transfer learning models. Case studies and real-world applications.		
Unit Number: 2	**Title:Transfer Learning Techniques andAlgorithms**	**No. of hours: 8**
Content Summary: Deep transfer learning methods, including domain adaptation, instance transfer, and multi-task learning. Transfer learning in convolutional neural networks (CNNs), recurrent neural networks (RNNs), and generative models. Ensemble- based transfer learning approaches. Handling domain shift and dataset bias. Advanced transfer learning algorithms and architectures.		
Unit Number: 3	**Title:Practical ImplementationofTransfer Learning**	**No. of hours: 12**
Content Summary: Data preprocessing and feature extraction for transfer learning. Fine-tuning andmodel adaptation techniques. Transferring knowledge across different domains and tasks. Implementing transfer learning in popular deep learning frameworks. Experimentation, analysis, and fine-tuning of transfer learning models.		
Unit Number: 4	**Title:Transfer Learning Applicationsand Challenges**	**No. of hours: 12**

Content Summary:

Transfer learning in computer vision, natural language processing (NLP), and speech recognition. Transfer learning for specific domains like healthcare, finance, and socialmedia analysis. Challenges in transfer learning, such as negative transfer and overfitting. Ethical considerations and fairness in transfer learning. Latest research trends and future directions in transfer learning.

Textbook

1. "Transfer Learning: Algorithms and Applications" by B. K. Tripathyand Sandipan Roy (Springer)
2. "Transfer Learning and Domain Adaptation in NLP" by Yoshua Bengio,Jian-YunNie, and Geoffrey J. Gordon (Morgan & Claypool Publishers)
3. Transfer Learning: Methods, Applications, and Challenges" by R.Sathya and K. S. Devi (CRC Press)

Reference Books:

1. "Transfer Learning" by Sinno Jialin Pan and Qiang Yang (Morgan & ClaypoolPublishers)
2. "Transfer Learning for Natural Language Processing" by ShervinMinaee andAmirali Abdolrashidi (Springer)
3. "Domain Adaptation in Computer Vision Applications" by GabrielaCsurka (Morgan & Claypool Publishers)
4. "Transfer Learning in Reinforcement Learning" by Panpan Cai, YangYu, andXuewen Yao (Springer)

Additional Readings:

I) **Online Resources:**

a. Papers with Code: Transfer Learning Section

b. TensorFlow Hub: Transfer Learning Models

II) **Open Source Projects:**

a. TensorFlow Hub: GitHub Repository

b. PyTorch Hub: GitHub Repository

III) **Community Forums and Discussions:**

a. Reddit: r/MachineLearning

b. Stack Overflow: Transfer Learning Tag

IV) OSSU Link:

Open Source Society University - Data Science Curriculum

TRANSFER LEARNING LAB

Program Name:	**MCA (Master in Computer Application)**		
Course Name: **Transfer LearningLab**	**Course Code**	**L-T-P**	**Credits**
	ENSP358	0-0-2	1
Type of Course:	Department Specific Elective I		
Pre-requisite(s), if any: NA			

Lab Experiments

Defined Course Outcomes

COs	Statement
CO 1	Understanding the fundamentals of transfer learning, including itsmotivation and applications.
CO 2	Gaining proficiency in implementing transfer learning techniques and algorithms using Python.
CO 3	Applying fine-tuning and model adaptation techniques to transferknowledge across different domains and tasks
CO 4	Evaluating and measure the performance of transfer learningmodels using appropriate metrics in Python.
CO5	Analyzing case studies and real-world applications of transferlearning to understand its practical implications

Project	Task	Mapped

		CO/COs
P1	**Project 1: Transfer Learning for Medical ImageClassification** **Problem Statement:** Develop a transfer learning model to classify different types of skin lesions using alimited dataset of medical images. **Objectives:** 1. **Instance-Based Transfer Learning:** ○ **Input:** Pre-trained CNN model and a small dataset of skin lesion images. ○ **Output:** Fine-tuned model with high classification accuracy. 2. **Feature-Based Transfer Learning:** ○ **Input:** Extracted features from a pre-trained model. ○ **Output:** New classifier trained on these features to improve performance on skin lesion classification. 3. **Model-Based Transfer Learning:** ○ **Input:** Model regularization techniques. ○ **Output:** Improved model generalization on unseen medical images.	CO1
P2	**Project 2: Adversarial Transfer Learning for FakeNews Detection** **Problem Statement:** Utilize adversarial transfer learning to improve the detection of fake news articlesusing limited labeled data. **Objectives:** 1. **Adversarial Transfer Learning:** ○ **Input:** Pre-trained adversarial network. ○ **Output:** Enhanced fake news detection	CO2

	model with high accuracy. 2. **Heterogeneous Transfer Learning:** ○ **Input:** Text and metadata features. ○ **Output:** Integrated model that leveragesboth types of data for better prediction. 3. **Transfer Learning in NLP:** ○ **Input:** Pre-trained language models. ○ **Output:** Fine-tuned model for specific fakenews detection tasks.	
P3	**Project 3: Multi-Task Learning for SentimentAnalysis and Topic Classification** **Problem Statement:** Create a multi-task learning model that can simultaneously perform sentiment analysis and topic classification on social media posts.**Objectives:** 1. **Multi-Task Supervised Learning:** ○ **Input:** Dataset with sentiment labels andtopic categories. ○ **Output:** Model capable of accurately predicting both sentiments and topics. 2. **Parallel and Distributed Multi-Task Learning:** ○ **Input:** Distributed computing resources. ○ **Output:** Scalable model that performs wellon large datasets. 3. **Transfer Learning Theory:** ○ **Input:** Theoretical understanding of multi-task learning. ○ **Output:** Application of theory to improve	CO3

	model performance.	
P4	**Project 4: Few-Shot Learning for Wildlife SpeciesRecognition** **Problem Statement:** Develop a few-shot learning model to recognize different wildlife species using a limited number of labeled images. **Objectives:** 1. **Few-Shot Learning Implementation:** ○ **Input:** Small dataset of wildlife images. ○ **Output:** Model that can accurately identifyspecies with few examples. 2. **Transfer Learning in Computer Vision:** ○ **Input:** Pre-trained computer vision models. ○ **Output:** Enhanced model for species recognition. 3. **Bayesian Program Learning:** ○ **Input:** Bayesian learning techniques. ○ **Output:** Improved model accuracy usingprobabilistic methods.	CO4

MINOR PROJECT

Program Name:	**MCA (Master in Computer Application)**		
Course Name: Minor Project	**Course Code**	**L-T-P**	**Credits**
	ENSI152	---	2
Type of Course:	Proj-I		

Pre-requisite(s), if any: NA

Duration:

The minor project will last for three months.

Project Requirements:

1. **Problem Identification and Analysis:**
   - Identify a relevant problem in society or industry.
   - Conduct a thorough analysis of the problem, consideringvarious perspectives and implications.
2. **Implementation:**
   - Develop and implement a solution to address the identified problem.
3. **Data Visualization:**
   - Utilize appropriate data visualization techniques to represent the problem, solution, and outcomes effectively.
4. **Presentation of Solutions:**
   - Prepare a comprehensive presentation of the implemented solution, including its development process, outcomes, and impact.
5. **Case Studies:**
   - Conduct case studies related to the problem and solution, analyzing existing examples and drawing relevant insights.

Guidelines:

1. **Project Selection:**
   - Choose a societal or industrial problem relevant to the fieldof computer science and engineering.
   - Ensure the problem is specific and well-defined.
2. **Literature Review:**
   - Conduct a thorough review of existing literature and solutions related to the problem.

- Identify gaps in existing solutions and potential areas for further investigation.

3. **Implementation:**

   - Develop a detailed plan for implementing the solution.
   - Execute the implementation using appropriate tools, technologies, and methodologies.

4. **Data Visualization:**

   - Collect relevant data and use visualization techniques to represent the problem, solution, and outcomes.
   - Ensure the visualizations are clear, accurate, and effectively communicate the information.

5. **Documentation:**

   - Document the entire process, including problem identification, literature review, implementation, data visualization, and case studies.
   - Use appropriate formats and standards for documentation.

6. **Presentation:**

   - Prepare a presentation summarizing the problem, existing solutions, implementation process, data visualization, andcase studies.
   - Ensure the presentation is clear, concise, and well- structured.

Evaluation Criteria for Minor Project (Out of 100 Marks):

1. **Problem Identification and Analysis (15 Marks):**

   - Comprehensive identification and analysis of the problem: 15marks
   - Good identification and analysis of the problem: 12 marks
   - Basic identification and analysis of the problem: 9 marks
   - Poor identification and analysis of the problem: 5 marks
   - No identification and analysis of the problem: 0 marks

2. **Implementation (30 Marks):**
    - Successful and thorough implementation: 30 marks
    - Good implementation: 25 marks
    - Moderate implementation: 20 marks
    - Basic implementation: 15 marks
    - Poor implementation: 10 marks
    - No implementation: 0 marks
3. **Data Visualization (20 Marks):**
    - Effective and clear data visualization: 20 marks
    - Good data visualization: 15 marks
    - Moderate data visualization: 10 marks
    - Basic data visualization: 5 marks
    - Poor data visualization: 0 marks
4. **Presentation of Solutions (15 Marks):**
    - Clear, concise, and engaging presentation: 15 marks
    - Clear but less engaging presentation: 12 marks
    - Somewhat clear and engaging presentation: 9 marks
    - Unclear and disengaging presentation: 5 marks
    - No presentation: 0 marks
5. **Case Studies (20 Marks):**
    - Comprehensive and insightful case studies: 20 marks
    - Good case studies: 15 marks
    - Moderate case studies: 10 marks
    - Basic case studies: 5 marks
    - Poor case studies: 0 marks

Total: 100 Marks

Course Outcomes:

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

1. **Identify and Analyze Problems:**
    - Identify relevant societal or industrial problems and conduct a thorough analysis of these problems.
2. **Implement Solutions:**
    - Develop and implement effective solutions to address identified problems using appropriate tools and technologies.
3. **Visualize Data:**
    - Utilize data visualization techniques to represent problems, solutions, and outcomes clearly and effectively.
4. **Present Solutions:**
    - Prepare and deliver comprehensive presentations summarizing the implementation process, outcomes, and impact of their solutions.
5. **Conduct Case Studies:**
    - Conduct case studies related to the problem and solution, analyzing existing examples and drawing relevant insights.
6. **Literature Review:**
    - Conduct comprehensive literature reviews to identify gaps in existing solutions and potential areas for furtherinvestigation.
7. **Documentation:**
    - Document the entire process, including problem identification, literature review, implementation, data visualization, and case studies, using appropriate formats and standards.
8. **Professional Development:**

- Develop skills in research, analysis, implementation, data visualization, documentation, and presentation, contributing to overall professional growth.

Semester 3

Advance Concepts in Software Engineering & ProjectManagement

Program Name	**MCA (Master in Computer Application)**		
Course Name: Advance Concepts in Software Engineering & Project Management	**Course Code**	**L-T-P**	**Credits**
	ENMM816	3-1-0	4
Total Contact Hours	40		
Type of Course:	Major		
Pre-requisite(s), if any: Fundamentals of software and computer			

Course Perspective: The course aims to teach the fundamentals of data structures, their design, implementation and effective use in problem solving approach. With the knowledge of data structures and practical experience in implementing them, students can become much more effective designer and developer. The course will start with the basic introduction of SDLC(Software Development Life Cycle Model) i.e Linear model, Prototype model, Iterative model, Spiral model etc. They will further proceed to evaluate the size estimation of the software and cost estimation of the software.

The Course Outcomes (COs).

COs	Statements
CO 1	**Understanding** various software development methodologies and paradigms, comparing their strengths and weaknesses in different projectscenarios

CO 2	**Applying** Understand advanced software design principles, including designpatterns and component-based development, and their application in solving complex software challenges.
CO 3	**Evaluating** Articulate the benefits and challenges of emerging technologiesin software development and project management, including cloud computing, microservices, and DevOps practices
CO 4	**Designing** Understand the principles of software architecture.
CO 5	**Analyzing and solving** After understanding the principle of software architecture then apply them to design and implement modular andmaintainable software systems.

CO = Course outcomes. A student is expected to have learnt concepts and demonstrated/developed abilities or skills related to software engineering/ management of the software at the end of the course.

Course Outline:

<table>
<tr><td><b>Unit Number: 1</b></td><td><b>Title: Software Engineering Principlesand Paradigms</b></td><td><b>No. of hours: 10</b></td></tr>
<tr><td colspan="3">Overview of software engineering paradigms: Waterfall, Agile, DevOps, etc. Software process models: Rational Unified Process (RUP), Scrum, Kanban, etc. Requirements engineering: Elicitation, analysis, specification, validation.
Software architecture and design principles. Software quality assurance and testing methodologies. Case study: Comparing software development methodologies.</td></tr>
<tr><td><b>Unit Number: 2</b></td><td><b>Title: Advanced Software Design andDevelopment</b></td><td><b>No. of hours: 12</b></td></tr>
<tr><td colspan="3">Object-oriented design principles and design patterns. Component-based development and service-oriented architecture. Designing for scalability, performance, and maintainability. Software reusability and code quality metrics.
Emerging technologies and their impact on software development (e.g., micro services, server less). Case study: Designing and developing a complex software system.</td></tr>
<tr><td><b>Unit Number: 3</b></td><td><b>Title: Software Project ManagementStrategies</b></td><td><b>No. of hours: 12</b></td></tr>
</table>

Project planning and estimation techniques (PERT, CPM, agile estimation). Risk management and mitigation in software projects. Resource allocation, team management, and conflict resolution. Change management and handling scope creep. Agile project management tools (e.g., JIRA, Trello) and techniques. Case study: Managing a software project using Agile methodologies		
Unit Number: 4	**Title: Software Quality Managementand Verification**	**No. of hours: 10**
Software testing strategies: Unit, integration, system, acceptance testing. Test automation and continuous integration/continuous deployment (CI/CD). Code reviews, static analysis, and dynamic analysis. Ensuring software security and addressing vulnerabilities. Monitoring and improving software quality over the development lifecycle. Case study: Implementing a comprehensive software testing and quality assurance process.		
Unit Number: 5	**Emerging Trends and Project ManagementChallenges**	**No. of hours: 10**
Cloud computing and its impact on software engineering. Big data and analytics in software development and project management. DevOps practices and their integration with project management. Ethical considerations in software engineering (privacy, bias, etc.). Handling distributed teams and global project management challenges. Case study: Adapting to new technologies and managing global software teams.		

Text book

Software Engineering: A Practitioner's Approach" by Roger S. Pressman or "Agile Estimating and Planning" by Mike Cohn for an in-depth understandingof software engineering and Agile

Reference Book

Software Engineering :New Age International Publication ,K.KAggarwal, Yogesh Singh.

Additional Readings:

Online Learning Resources forSoftware

Engineering

1. Students should explore Platforms like Coursera, edX, and Udemy offer

courses on software engineering principles and Agile practices.

2. Choose an open-source project and analyze its software development process. Compare the methodologies used and their impact on the project's success.

(DEPARTMENT ELECTIVE-II)

COMPUTATIONAL SERVICES IN THE CLOUD

Program Name	**MCA (Master in Computer Application)**		
Course Name: COMPUTATIONAL SERVICES IN THE CLOUD	**Course Code**	**L-T-P**	**Credits**
	ENSP 401	4-0-0	4
Total Contact Hours	40		
Type of Course:	Major		
Pre-requisite(s), if any: Department Specific Elective II			

Course Perspective: This course covers evolutionary computing paradigms from multi-processor systems to Cloud, Edge, and Fog Computing. The course design follows cloud deployment models, service models, virtualization techniques and cloud architectural solutions. It also elaborates cloud compliances and security at fine- grained level by following a shared responsibility model. It reveals a design pattern to the students, enabling them to think through the process of designing and implementing cloud infrastructure and optimal IT solutions

The Course Outcomes (COs).

COs	Statements
CO 1	**Understanding:** Explain the core concepts of the cloud computing paradigm: how and why this paradigm shift came about, the characteristics, advantagesand challenges brought about by the various models and services in cloud computing.

CO 2	**Applying:** Apply the fundamental concepts in datacenters to understand the tradeoff sin power, efficiency and cost.
CO 3	**Evaluating:** Identify resource management fundamentals, i.e. resource abstraction, sharing and sandboxing and outline their role in managinginfrastructure in cloud computing.
CO 4	**Designing:** Design various cloud programming models
CO 5	**Analyzing and sol**ving: After designing various cloud programming andapply them to solve problems on the cloud.

CO = Course outcomes. A student is expected to have learnt concepts and demonstrated/developed abilities or skills related to software engineering/ management of the softwareat the end of the course.

Course Outline:

Unit Number: 1	**Title: Introduction to Cloud Computing**	**No. of hours: 10**
Cloud Computing, Adoption of cloud-based IT resources, Service Models: Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), Software-as-a- Service(SaaS), Deployment models: Public Cloud, Private Cloud, Hybrid Cloud, Community Cloud, Cloud Computing Characteristics, Challenges of cloud computing, Virtualization concept, Types of virtualizations, Demo of virtualization, Virtualization Merits, Role of virtualization in cloud computing, Virtualization Demerits, VMPlacement, VM Migration, VM Migration Demo, VM clustering, Design Issues in VM Clustering, Need of Dockers and Containers, Docker Eco-System, Hypervisor vs Docker.		
Unit Number: 2	**Title: Microservices**	**No. of hours: 10**
Microservices, Service-Oriented Architecture, REST API, IP Addressing, Subnetting, Supernetting, Designing of Virtual Private Cloud, Demo of VPC, VPC Peering,VPC Case Study, Cloud Storage, Server less Computing, Cloud API Gateway, Cloud Databases, Resource Provisioning, Time shared and space shared, Efficient VMConsolidation on cloud server, Task/DAG Scheduling Algorithms, Min- Min, Max- Min, MET, B-level Demo, T-level Demo, Task-VM Mapping, Auto Scaling, Load Balancing.		
Unit Number: 3	**Title: Case Study**	**No. of hours: 10**

<table>
<tr><td colspan="3">Case Study: Cloud Market analysis, Security and Compliances, Shared security model in IAAS/PAAS/SAAS, Shared technology issues, Data loss or leakage, Accountor service hijacking, Implementation of cloud security, Security Groups, Network Access Control Lists, Cloud databases, Parallel Query Execution withNoSQLDatabase, Big Data, Handling Big Data on Cloud Platform, Map-Reduce framework for largeclusters using Hadoop, Design of data applications based on
Map Reducein Apache Hadoop.</td></tr>
<tr><td><b>Unit Number: 4</b></td><td><b>Title: Comparative study/analysis ofpublic clouds</b></td><td><b>No. of hours: 10</b></td></tr>
<tr><td colspan="3">Comparative study/analysis of public clouds, Edge Computing, Fog Computing, Data Offloading, Cloud-Based DevOps Tools, Task Partitioning, Data Partitioning, Data Synchronization, Distributed File System, Data center, Ongoing Research
Topics.</td></tr>
</table>

Text and Reference Book:

1. Lizhe Wang, Rajiv Ranjan, Jinjun Chen and Boualem Benatallah, CloudComputing (1 ed.), CRC Press, 2017. ISBN 978-1351833097.

2. Judith S. Hurwitz and Daniel Kirsch, Cloud Computing For Dummies(2 ed.),Hoboken: John Wiley & Sons, 2020. ISBN 978-1119546658.

3. Prerna Sharma, Moolchand Sharma and Mohamed Elhoseny,Applications ofCloud Computing (1 ed.), CRC Press, 2020. ISBN9780367904128.

4. Software Engineering: A Practitioner's Approach" by Roger S. Pressmanor "Agile Estimating and Planning" by Mike Cohn for an in-depth understanding of software engineering and Agile

5. Software Engineering :New Age International Publication ,K.K Aggarwal,Yogesh Singh.

Additional Readings:

Online Learning Resources for computational services in the cloud:

1. Lizhe Wang, Rajiv Ranjan, Jinjun Chen and Boualem Benatallah, CloudComputing (1 ed.), CRC Press, 2017. ISBN 978-1351833097.
2. Judith S. Hurwitz and Daniel Kirsch, Cloud Computing For Dummies (2ed.),Hoboken: John Wiley & Sons, 2020. ISBN 978-1119546658.
3. Prerna Sharma, Moolchand Sharma and Mohamed Elhoseny, ApplicationsofCloud Computing (1 ed.), CRC Press, 2020. ISBN 9780367904128.

DepartmentalElective III)

COMPUTATIONAL SERVICES IN THE CLOUD LAB

Program Name:	**MCA (Master in Computer Application)**		
Course Name: **Computational Services in The Cloud Lab**	**Course Code**	**L-T-P**	**Credits**
	ENSP451	0-0-2	1
Type of Course:	Department Specific Elective II		
Pre-requisite(s), if any: Nil			

Defined Course Outcomes

COs	
CO 1	Explaining the principles and methodologies for optimizing resource management and scheduling in cloud environments.
CO 2	Implementing a comprehensive security and compliance framework toprotect cloud infrastructures against security threats.
CO 3	Developing and deploying cloud-based solutions for high-performance scientific research, focusing on scalable and efficient data processing.
CO 4	Assessing the effectiveness of a cloud-based social computing platform infacilitating digital content sharing and social interaction.

List of Programs

Project No.	**Project Detail**	**Mapped CO/COs**

P1	**Project 1: Optimizing Resource Management and Scheduling in Cloud Environments** **Problem Statement:** Develop an advanced resource management and scheduling system for a cloud environment to enhance efficiency and performance of virtualized resources.	CO1
P2	**Project 2: Implementing Robust Cloud Security and Compliance Framework** **Problem Statement:** Establish a comprehensive security and compliance framework for a cloud infrastructure to protect against security threats and ensure compliance with regulations.	CO2
P3	**Project 3: Enhancing Data Privacy and Compliance in Multi-Regional Cloud Deployments** **Problem Statement:** Develop strategies to enhance data privacy and ensure compliance across multi-regional cloud deployments for a global enterprise.	CO3
P4	**Project 4: Leveraging Cloud Computing for High- Performance Scientific Research** **Problem Statement:** Utilize cloud computing resources to perform high-performance scientific research, enabling scalable and efficient data processing.	CO4
P5	**Project 5: Developing a Cloud-Based Social ComputingPlatform** **Problem Statement: Create a cloud-based social computing platform to facilitate digital content sharing and social technologies. interaction, leveraging the latest cloud**	CO4

MICROSOFT AZURE CLOUD FUNDAMENTALS

Program Name	**MCA (Master in Computer Application)**		
Course Name: Microsoft Azure CLOUD FUNDAMENTALS	**Course Code**	**L-T-P**	**Credits**
	ENSP403	4-0-0	4
Total Contact Hours	40		
Type of Course:	Major		
Pre-requisite(s), if any:			

Course Perspective. The Microsoft Azure Cloud Fundamentals course introduces the concepts of cloud computing and the Azure platform. It covers Azure services such as Virtual Machines, Storage, Networking, Identity, App Services, and Databases. The course focuses on security, monitoring, and management in Azure. Real-world case studies and hands- on labs enable practical application. The subject equips students to design trustworthy intrusion detection systems and enhances security in IoT networks usingAzure.

The Course Outcomes (COs).

COs	Statements
CO 1	Explaining the fundamental concepts, characteristics, benefits, and challengesof cloud computing, including various service and deployment models.
CO 2	Utilizing Microsoft Azure services to create and manage virtual machines andstorage solutions, tailored to specific performance requirements.
CO 3	Evaluating the suitability of different Azure regions, availability zones, andpricing models for various cloud deployment scenarios.
CO 4	Designing and configure virtual networks, load balancers, and VPNs in Azureto optimize network traffic distribution and secure communications.

CO5	Assessing the effectiveness of Azure Active Directory and other identity andaccess management solutions in ensuring secure authentication and authorization in cloud environments.

CO = Course outcomes. A student is expected to have learnt concepts and demonstrated/developed abilities or skills related to software engineering/management of the software at the end of the course.

Course Outline:

<table>
<tr><td><b>Unit Number: 1</b></td><td><b>Title: Introduction to Cloud Computing</b></td><td><b>No. of hours: 1 0</b></td></tr>
<tr><td colspan="3">Introduction to Cloud Computing and its Characteristics, Benefit, and Challenges of cloud computing. Cloud service models: Infrastructure asa Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS). Cloud Deployment Models, for example, Public, private, hybrid, and community clouds. shared responsibility model. Identify appropriate use cases for each cloud model. Consumption-based model. Comparison of cloud pricing models.</td></tr>
<tr><td><b>Unit Number: 2</b></td><td><b>Title: Introduction to MicrosoftAzure</b></td><td><b>No. of hours: 1 0</b></td></tr>
<tr><td colspan="3">Microsoft Azure cloud platform and its significance in the industry, Azure regions, and availability zones for understanding global data center distribution. Introduction of various Azure services and solutions available for different scenarios.</td></tr>
<tr><td><b>Unit Number: 3</b></td><td><b>Title: Azure Virtual Machines (VMs)and Storage</b></td><td><b>No. of hours: 10</b></td></tr>
<tr><td colspan="3">Create and manage virtual machines using Azure. Different VM sizes and types based on performance requirements. VM scaling and load balancing for optimizing application performance. Azure storage services: Blob Storage, TableStorage, File Storage, and Disk Storage.</td></tr>
<tr><td><b>Unit Number: 4</b></td><td><b>Title:Azure Networking, Identityand Access Management</b></td><td><b>No. of hours: 1 0</b></td></tr>
<tr><td colspan="3">creation and configuration of virtual networks and subnets in Azure. Azure Load Balancer for distributing incoming network traffic, VPN Azure Active Directory (Azure AD) for managing identities and authentication. Gateway for secure communication between on- premises networks, and Azure. Azure Active Directory (Azure AD) for managing identities and authentication. Database Services, Azure Storage Account for data storage and retrieval.</td></tr>
</table>

Text Book:

a) "Cloud Computing: Theory and Practice" by Dan C. Marinescu
b) "Exam Ref AZ-900 Microsoft Azure Fundamentals" by Jim Cheshire.

Reference Book

a) "Exam Ref AZ-900 Microsoft Azure Fundamentals" by Jim Cheshire.

b) "Microsoft Azure Essentials: Fundamentals of Azure" by Michael Collier andRobin Shahan.

c) "Azure for Architects: Implementing cloud design, DevOps, IoT, andserverless solutions on your public cloud" by Ritesh Modi.

d) "Azure Security Center: Protecting your cloud workloads" by Yuri Diogenes,Tom Shinder, and Debra Shinder.

e) "Azure Cost Management and Billing" by Sjoukje Zaal

Additional Readings:

Online Learning Resources for Software Engineering

I) Microsoft Learn:

Microsoft's official learning platform offers a wide range of Azure courses, modules, and learning paths for beginners to advanced users. Explore topics such as cloud computing basics, Azure architecture components, compute and networking services,storage services, identity management, cost management, governance, and monitoring. Microsoft Learn

R 1. Coursera:

Coursera provides Azure courses offered by top universities and organizations. Topics include cloud computing basics, Azure architecture, services, security, governance, and more. Coursera Azure Courses

2. Open-Source Society University (OSSU):

OSSU offers a structured, open-source curriculum for self-learning various topics, including cloud computing and Azure fundamentals. You can follow their curriculum to gain a comprehensive understanding of Azure concepts and services. OSSU CloudComputing Curriculum

MICROSOFT AZURE CLOUDFUNDAMENTALS LAB

Program Name:	**MCA (Master in Computer Application)**		
Course Name:	**Course Code**	**L-T-P**	**Credits**
Microsoft Azure Cloud FundamentalsLab	**ENSP453**	0-0-2	1
Type of Course:	Department Specific Elective II		
Pre-requisite(s), if any: Nil			

Proposed Lab Experiments Defined

Course Outcomes

COs	
CO 1	**Acquiring** a practical understanding of cloud computing through hands-onlab experiments.
CO 2	**Developing** proficiency in using global data centers.
CO 3	**Demonstrating** the ability to create virtual machines and handle datastorage
CO 4	**Analyzing** and maintain the security and access management system.

Project No.	Project Detail	Mapped CO/COs
P1	**Project 1: Deploying a Scalable Web Application on Microsoft Azure** **Problem Statement:** Develop and deploy a scalable web application using Azure's compute and networking services to ensure high availability and optimal performance.	CO1
P2	**Project 2: Implementing Azure Storage Solutions for a Data-Intensive Application** **Problem Statement:** Create and manage Azure storage solutions for a data-intensive application, ensuring efficient data storage, retrieval, and performance optimization.	CO2
P3	**Project 3: Establishing a Secure and Compliant Azure Environment** **Problem Statement:** Set up a secure and compliant Azure environment for a financial application, ensuring governance, cost management, and continuous monitoring.	CO3
P4	**Project 4: Migrating an On-Premise Application to Azure** **Problem Statement:** Migrate an on-premise application to Azure, ensuring minimal downtime and optimized performance post-migration.	CO4
P5	**Project 5: Implementing a Comprehensive Monitoring and Alerting System on Azure** **Problem Statement:** Develop a comprehensive monitoring and alerting system for a mission-critical application hosted onAzure,ensuring proactive management and issue resolution.	CO4

Arithmetic and Reasoning Skills-I

Program Name	**MCA (Master in Computer Application)**		
Course Name: **Verbal Ability**	**Course Code**	**L-T-P**	**Credits**
	AEC008	3-0-0	3
Total Contact Hours	45		
Type of Course:	AEC		
Pre-requisite(s), if any:			

Course Perspective. The course aims to improve basic arithmetic skills, speed, and accuracy in mental calculations, and logical reasoning. These abilities are essential for a strong math foundation, helping students succeed in academics and various practical fields.

The Course Outcomes (COs). On completion of the course theparticipants will be:

COs	Statements
CO1	Understanding arithmetic algorithms required for solving mathematical problems.
CO 2	Applying arithmetic algorithms to improve proficiency in calculations**.**

CO3	Analyzing cases, scenarios, contexts and variables, and understanding their inter-connections in a given problem.
CO4	Evaluating & deciding approaches and algorithms to solve mathematical & reasoning problems.

CO = Course outcomes. A student is expected to have learnt concepts and demonstrated/developed abilities or skills related to strategic management at the end of the course.

Course Outline:

Unit Number: 1	**Title:** Mathematical Essentials	**No. of hours: 15**
Content: Vedic Maths, Classification of Numbers and Divisibility Rule, Percentage, Ratio and Proportion		
Unit Number: 2	**Title:** Fundamentals of Logical Reasoning	**No. of hours: 6**
Content: Content Summary: Blood Relations, Direction Sense, Coding Decoding		
Unit Number: 3	**Title:** Elementary Quantitative Skills	**No. of hours: 18**
Content: Content Summary: Simple and Compound Interest, Average, Partnership, Time and Work, Time Speed & Distance		
Unit Number: 4	**Title:** Advanced Quantitative Skills	**No. of hours: 6**
Content: Permutation & Combination, Probability		

R1. Guha Abhijit: Quantitative Aptitude for Competitive Examinations, Tata McGraw Hill Publication
R2. Quantitative Aptitude by R.S. Aggarwal
R3. Verbal & Non-Verbal Reasoning by R.S. Aggarwal

Additional Readings:
https://www.indiabix.com/online-test/aptitude-test/
https://www.geeksforgeeks.org/aptitude-questions-and-answers/
https://www.hitbullseye.com/

STORAGE AND DATABASES ON CLOUD

Program Name	**MCA (Master in Computer Application)**		
Course Name: STORAGE AND DATABASE ON CLOUD	**Course Code**	**L-T-P**	**Credits**
	Ensp405	4-0-0	4
Total Contact Hours	40		
Type of Course:	Department Specific Elective II		
Pre-requisite(s), if any:			

Course Perspective. The course aims to teach the fundamentals of data structures, their design, implementation and effective use in problem solving approach. With the knowledge of data structures and practical experience in implementing them, students can become much more effective designer and developer. The course will start with the basic introduction of SDLC(Software Development Life Cycle Model) i.e Linear model, Prototype model, Iterativemodel, Spiral model etc. They will further proceed to evaluate the size estimationof the software and cost estimation of the software.

Course Outcomes (COs).

COs	Statements
CO 1	**Understanding:** Recall the fundamental concepts and principles of cloud storage and databases. Interpret the security considerations and bestpractices for cloud storage and databases.
CO 2	**Applying:** ideas and solutions for optimizing performance and improving efficiency in cloud storage and databases through indexing, caching, andquery optimization techniques.
CO 3	**Evaluating:** the requirements of an application or system and determine the most suitable cloud storage and database solutions to meet those requirements.
CO 4	**Designing:** different types of cloud storage and database services available.

CO 5	**Analyzing and solving:** the best practices and considerations for designingscalable,reliable, and secure cloud storage and database architecture anddata storage architecture for cloud-based applications.

CO = Course outcomes. A student is expected to have learnt concepts and demonstrated/developed abilities or skills related to software engineering/ management of the software at the end of the course.

Course Outline:

<table>
<tr><td><b>Unit Number: 1</b></td><td><b>Title: Introduction to Storage oncloud</b></td><td><b>No. of hours: 10</b></td></tr>
<tr><td colspan="3">Introduction to Cloud Computing, Overview of cloud databases and cloud storages,types ofcloud storages(Object, block and file), different types of cloud database management systems, Gartner Magic Quadrant for Cloud Database Management Systems, Advantages of Working with Cloud Databases, Considerations for Cloud Databases, Top Cloud Database, Factors that help in choosing the right cloud database, Challenges involved in using cloud storages and databases methodologies.</td></tr>
<tr><td><b>Unit Number: 2</b></td><td><b>Title: Data Integration, Migration, Security and Performance on Cloud</b></td><td><b>No. of hours: 10</b></td></tr>
<tr><td colspan="3">Techniques, tool, methods and considerations for migrating from premise database to cloud databases. Backup, Recovery, and Disaster Planning including automated backups, point-in-time recovery and replication. Performance Optimization and Monitoring including query optimization, indexing,caching, and monitoring tools
Scalability and High Availability: load balancing, replication, sharding, and auto- scaling, Cloud Data Warehousing</td></tr>
<tr><td><b>Unit Number: 3</b></td><td><b>Title: AWS Cloud Storage</b></td><td><b>No. of hours: 10</b></td></tr>
<tr><td colspan="3">Introduction to AWS cloud storage, AWS management console, AWS Storage Services, Uploading files and images , Creating a web server, Overview of Amazon S3, Storage Classes, EC2 Instance Storage, network file system Amazon Elastic Block Store, Amazon Elastic file system, Amazon Cloud Front.
Brief introduction to Google Cloud Storage, and Azure Blob Storage.</td></tr>
<tr><td><b>Title: Case Study</b></td><td><b>No. of hours: 10</b></td><td></td></tr>
<tr><td colspan="3">Case Studies and Real-world Examples of Netflix, Airbnb, Pinterest, spotify, coca-cola etc. Analyzing real-world use cases of organizations using cloud storage and databases,discussing architecture decisions, challenges, andlessons learned.</td></tr>
</table>

Text Book

1) "Database Cloud Storage: The Essential Guide to Oracle Automatic StorageManagement"by Nitin Vengurlekar, 2013

Reference Book

2) "Cloud Database Development and Management" by Lee chao, 2013

3.) "Advancing Cloud Database Systems and Capacity Planning with DynamicApplications" by

Narendra Kumar Kamila, 2017

Additional Readings:

Online Learning Resources

1. **E**xplore open-source projects related to cloud databases and storage. discover open-source projects from GitHub, learn from the codebase, and contribute to thedevelopment of cloud-based storage and database solutions. (https://github.com/topics/cloud-database
2. Join the course : https://www.coursera.org/specializations/cloud-computing
3. Join Online Courses like

   a. https://www.udemy.com/topic/cloud-computing/
   b. https://www.coursera.org/courses?query=data%20storage
   c. https://www.mygreatlearning.com/academy/learn-for-free/courses/databases-and-files-systems-in-aws
   d. https://www.youtube.com/watch?v=EN4fEbcFZ_E
   e. https://www.codecademy.com/catalog/subject/cloud-computing
   f. https://www.simplilearn.com/tutorials/cloud-computing-tutorial

4. Read case studies of organizations that have successfully implemented cloudstorage and database solutions.

   a. **Case Study of Delloite:** https://www2.deloitte.com/us/en/pages/consulting/articles/cloud-computing-case-studies.html
   b. Case Study of Amazon:

5. https://aws.amazon.com/solutions/case-studies/amazon/
6. Students should explore Platforms like Coursera, edX, and Udemy offer courses on softwareengineering principles and Agile practices.
7. Choose an open-source project and analyze its software development process. Compare themethodologies used and their impact on the project's success.

STORAGE AND DATABASES ON CLOUD LAB

Program Name:	**MCA (Master in Computer Application)**		
Course Name:	**Course Code**	**L-T-P**	**Credits**
Storages and Databases on Cloud Lab	**ENSP455**	0-0-2	1
Type of Course:	Department Specific Elective II		
Pre-requisite(s), if any: Nil			

Lab Experiments

Defined Course Outcomes

COs	**Understanding** the principles and concepts of cloud storage and databases.
CO 1	**Gaining** hands-on experience in using cloud storage services for data storageand retrieval.
CO 2	**Implementing** backup and recovery strategies for cloud-based storage anddatabases.
CO 3	**Optimizing** database performance in the cloud through indexing and queryoptimization techniques.
CO 4	**Analyzing** real-world case studies to understand the challenges andsolutions related to cloud storage and databases.

Project No.	Project Detail	Mapped CO/COs
P1	**Project 1: Deploying a Scalable Web Application onMicrosoft Azure** **Problem Statement:** Develop and deploy a scalable web application using Azure's compute and networking services to ensure high availability and optimal performance.	CO1
P2	**Project 2: Implementing Azure Storage Solutions for a Data-Intensive Application** **Problem Statement:** Create and manage Azure storage solutions for a data-intensive application, ensuring efficient data storage, retrieval, and performance optimization.	CO2
P3	**Project 3: Establishing a Secure and Compliant Azure Environment** **Problem Statement:** Set up a secure and compliant Azure environment for a financial application, ensuring governance, cost management, and continuous monitoring.	CO3
P4	**Project 4: Migrating an On-Premise Application to Azure** **Problem Statement:** Migrate an on-premise application to Azure, ensuring minimal downtime and optimized performance post-migration.	CO4
P5	**Project 5: Implementing a Comprehensive Monitoring and Alerting System on Azure** **Problem Statement:** Develop a comprehensive monitoring and alerting system for a mission-critical application hosted on Azure, ensuring proactive management and issue resolution.	CO4

Application Development and Devops On Cloud Application

Program Name:	**MCA (Master in Computer Application)**		
Course Name: Application Development And Devops OnCloud	**Course Code**	**L-T-P**	**Credits**
	ENSP407	4-0-0	4
Total Contact Hours	40		
Type of Course:	Department Specific Elective II		
Pre-requisite(s), if any:			

Course Perspective. The syllabus aims to equip students with practical skills and theoretical knowledge to design, develop, and deploy applications in cloud environments while implementing DevOps practices to enhance software development, delivery, and operations on the cloud. It prepares them for a career in the dynamic and rapidly growing field of cloud computing and DevOps, where demand for skilled professionals is high due to the increasing adoptionof cloud technologies in various industries.

The Course Outcomes (COs).

COs	Statements
CO 1	**Understanding:** The fundamental concepts and principles of cloud computing, including different cloud service models and deployment models. Gain insights into various cloud providers and their offerings tomake informed decisions for application development and deployment
CO 2	**Acquiring** practical knowledge and hands-on experience in developing cloud-native applications, utilizing containerization with Docker and or chestration using Kubernetes. Demonstrate proficiency in designing andimplementing microservices-based architectures.
CO 3	**Master** the principles and practices of DevOps in a cloud environment.Learn how to set up Continuous Integration (CI) and Continuous Deployment (CD) pipelines, automate infrastructure provisioning with Infrastructure as Code (IaC), and manage application configurationseffectively.

CO 4	**Developing** the skills to monitor, log, and optimize the performance of cloud-based applications. Explore auto-scaling and load balancing techniques to ensure high availability and disaster recovery strategies for resilient applications.
CO 5	**Applying** cloud-based DevOps tools and best practices to collaborate effectively in distributed teams. Gain knowledge of cloud-based versioncontrol, testing, cost management, and performance optimization techniques to deliver efficient and cost-effective cloud solutions.

CO = Course outcomes. A student is expected to have learnt concepts and demonstrated/developed abilities or skills related to cloud computing/ implementation of cloud computing at the end of the course.

Course Outline:

Unit Number: 1	**Title: Introduction to Cloud Computing**	**No. of hours: 10**
Overview of Cloud Computing: Definition, Characteristics, Advantages, and Disadvantages. Cloud Service Models: Infrastructure as a Service (IaaS), Platform as a Service(PaaS), Software as a Service (SaaS). Cloud Deployment Models: Public Cloud, Private Cloud, Hybrid Cloud, CommunityCloud. Cloud Providers: Introduction to popular cloud service psroviders (e.g., AWS, Azure,Google Cloud).		
Unit Number: 2	**Title: Cloud-Based Application Development**	**No. of hours: 10**
Cloud Native Applications: Principles and architecture for building applications optimized for cloud environments. Containerization: Docker and Kubernetes for application deployment, scaling, and management. Microservices Architecture: Designing, developing, and deploying microservices-based applications. Serverless Computing: Introduction to serverless architecture and Function as a Service (FaaS). Application Security in the Cloud: Best practices for securing cloud- based applications. Cloud-Based Databases: Database options and considerations in cloudenvironments.		

Unit Number: 3	**Title: DevOps Practices inCloud**	**No. of hours: 10**
Continuous Integration (CI) and Continuous Deployment (CD) in Cloud: Setting upCI/CDpipelines. Infrastructure as Code (IaC): Automating infrastructure provisioning using tools likeTerraform and Cloud Formation. Configuration Management: Managing application configurations in a cloud-basedenvironment. Monitoring and Logging in Cloud: Tools and techniques for monitoring applicationperformance and gathering logs. Auto-scaling and Load Balancing: Scaling applications dynamically based ondemand.		
Unit Number: 4	**Title: Cloud-Based DevOps Toolsand BestPractices**	**No. of hours: 10**
Cloud-Based Version Control: Using Git and other version control tools in cloud-based development. Collaboration and Communication Tools: Utilizing cloud-based collaboration tools fordistributed teams. Cloud-Based Testing and Quality Assurance: Strategies for testing applications incloudenvironments. Cost Management in Cloud: Optimizing cloud resource usage and cost control. Performance Optimization in Cloud: Techniques for improving applicationperformance incloud environments. Case Studies and Real-World Projects: Analyzing successful cloud-based applicationdevelopment and DevOps projects.		

Text Book

1. Jez Humble and David Farley, Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation, Pearson Education, Inc., 2011.

Reference books

2. Thomas Erl, Ricardo Puttini, and Zaigham Mahmood, Cloud Computing: Concepts, Technology & Architecture, Prentice Hall, 2013.

3. Arun Eapen, Docker on Amazon Web Services: Build, deploy, and manage your container applications at scale on AWS, Packt Publishing, 2017
4. Sam Newman, Building Microservices: Designing Fine-Grained Systems, O'Reilly Media, Inc., 2015.

5. Mark Richards and Neal Ford, Fundamentals of Software Architecture: AnEngineering Approach, O'Reilly Media, Inc., 2020.

Additional Readings:

Online Learning Resources for Devops

Online Learning Resources for ApplicationDevelopment and DevOps onCloud

I) Microsoft Learn: Azure DevOps and Development

a. **Description:** Comprehensive learning paths and modules on Azure DevOps, including CI/CD, IaC, and cloud-based application development.

b. **Link:** Microsoft Learn - Azure DevOps and Development

II) AWS Training and Certification: DevOps on AWS

a. **Description:** Detailed courses and certifications for learning DevOps practices and application development on AWS, covering tools like AWS CodePipeline, CodeBuild, and more.

b. **Link:** AWS Training - DevOps on AWS

III) Google Cloud Training: Application Development

a. **Description:** Google Cloud provides courses on developing applications using Google Cloud services, including Kubernetes, App Engine, and Cloud Functions.

b. **Link:** Google Cloud Training - Application Development

Program Name:	**MCA (Master in Computer Application)**		
Course Name:	**Course Code**	**L-T-P**	**Credits**
APPLICATION DEVELOPMENT AND DEVOPS ON CLOUD LAB	**ENSP457**	0-0-2	1
Type of Course:	Department Specific Elective II		
Pre-requisite(s), if any: Nil			

Defined Course Outcomes

COs	**Course Outcomes (COs)**
CO 1	**Implementing** continuous integration (CI) pipelines to automate the build, test, and integration processes, ensuring smooth and efficient integrationof new code changes.
CO 2	**Developing** and implement automated deployment pipelines for microservices and mobile applications, ensuring reliable and efficientdeployment processes.
CO 3	**Integrating** comprehensive testing strategies, including acceptance and non-functional requirements testing, into CI/CD pipelines to ensure high code quality and performance standards.
CO 4	**Managing** and monitor cloud-based application infrastructure using automation tools, ensuring efficient provisioning, configuration, and continuous monitoring.

Project No.	Project Detail	Mapped COs
P1	**Project 1:** Setting up a cloud-based developmentenvironment **Problem Statement: Installing a cloud based application.**	CO1
P2	**Project 2: Deploying a web application on a cloudplatform.** **Problem Statement:** Develop a web application and deploying application on a cloud platform.	CO2
P3	**Project 3: Containerizing and deploying with Dockerand Kubernetes**: **Problem Statement:** Students containerize anexisting application using Docker and deploy it on a Kubernetes cluster. They configure Kubernetes services, pods, and replicas for the application.	CO3
P4	**Project 4: Infrastructure Management for Cloud-BasedApplication** **Problem Statement:** Develop a comprehensive infrastructure management system for a cloud-based application to ensure efficient provisioning, configuration, and monitoring.	CO4
P5	**Project 5: Continuous Delivery and Risk Management for Financial Application** **Problem Statement:** Implement a continuous delivery (CD)pipeline with a strong focus on risk management for afinancial application to ensure secure and compliant releases.	
P6	**Project 6: Project for Configuring auto-scaling for an application** **Problem Statement:** Students design a scalable architecture for a multi-tier application and implement auto- scaling based on CPU utilization. They monitor application performance during auto-scaling events.	

P7	**Project 7: Exercise for Implementing high availabilityand disaster recovery**: **Problem Statement:**	
P8	**Project 8: Project for Building a serverless application** **Problem Statement:** Students develop a serverless application using FaaS. They implement serverless functionsto handle specific application features and integrate them with event triggers.	
P9	**Project 9: Exercise for Implementing application monitoring and logging** **Problem Statement:** Students set up monitoring and logging solutions (e.g., CloudWatch, Stackdriver) for their applications to collect metrics, logs, and perform analysis.	
P10	**Project 10: Project for Creating and managing a cloud-based database** **Problem Statement:** Project: Students design a database schema and implement a cloud-based database (e.g., Amazon RDS, Azure SQL Database). They perform data migration and backup strategies.	
P11	**Project 11: Project for Integrating third-party APIswith an application** **Problem Statement:** Students integrate a popular third-party API (e.g., Google Maps API, Twitter API) into their application to enhance its functionality and showcase real-timedata retrieval.	

P12	**Project 12: Exercise for Implementing security bestpractices** **Problem Statement:** Students configure Identity and Access Management (IAM) roles and policies to enforce security for their cloud-based applications. They implement Network Security Groups andsecurity groups to control traffic flow.	
P13	**Project 13: Exercise for Managing application configurations**: **Problem Statement:** Students use cloud-native configuration management tools (e.g., AWS Systems Manager, Azure Configuration Management) to manage applicationconfigurations across multiple instances.	
P14	**Project 14: Project for Performing load testing on anapplication** **Problem Statement:** Students migrate an existing on-premises application to a cloud platform, considering factors like data transfer, application dependencies, and post-migration testing.	
P15	**Project 15: Project for Migrating an application to thecloud** **Problem Statement:** Students migrate an existing on-premises application to a cloud platform, considering factors like data transfer, application dependencies, and post-migration testing.	
P16	**Project 16: Project for Implementing DevOpspractices.** **Problem Statement:** Students collaborate in teams to develop a complete cloud-based application and implement DevOps practices,including version control, CI/CD, and automated testing.	

P17	**Project 17: Exercise for Deploying a multi-tier application architecture** **Problem Statement:** Students design and deploy a multi-tier application on the cloud, comprising frontend web servers, application servers, anddatabase servers.	
P18	**Project 18: Project for Integrating CI/CD pipeline with version control and issue tracking tools** **Problem Statement:** Students set up integration betweena CI/CD pipeline and version control (e.g., Git) and issue tracking tools (e.g., Jira) for automated build and issuetracking	
P19	**Project 19: Exercise for Implementing Canary deployment** **Problem Statement:** Students deploy a new feature of their application to a small subset of users using Canary deployment to gather feedback and monitor performance.	
P21	**Project 21: Project for Implementing feature flags** **Problem Statement:** Students implement feature flags (feature toggles) in their application to enable/disable features dynamically and gradually release new functionalities.	
P22	**Project 22: Project for Implementing cost optimization strategies** **Problem Statement:** Students analyze the cost of their cloud resources and implement costoptimization techniques,such as using reserved instances and rightsizing resources.	
P23	**Project 23: Project for Deploying a microservices-based application** **Problem Statement:** Students design and deploy a microservices-based application architectureon the cloud, using containers and orchestrators like Kubernetes.	

P24	**Project 24: Project for Real-world project showcasing cloud-based development** **Problem Statement:** Students work on a real-world cloud-based development project, applying the concepts learned throughout the course to design, develop, deploy, and manage an application in the cloud.	

(DEPARTMENT ELECTIVE-III)
MOBILE APPLICATION DEVELOPMENTUSING IOS

Program Name:	**MCA (Master in Computer Application)**		
Course Name:	**Course Code**	**L-T-P**	**Credits**
Mobile Application Development usingiOS	**ENSP409**	4-0-0	4
Total Contact Hours	40		
Type of Course:	Department Specific Elective III		
Pre-requisite(s), if any: Basics of Android			

Course Perspective. The objective of the course is to provide skills to develop applications for OS X and iOS. It includes an introduction to the development framework Xcode. Objective-C is used as a programming language to develop applications. Objective-C is the superset of the C programming language andprovides object-oriented capabilities and a dynamic runtime. Objective-C inherits thesyntax, primitive types, and flow control statements of C and adds syntax for definingclasses and methods. The course is divided into 4 modules:

1. Introduction to IDE and SDK of iOS App Development
2. Swift Programming
3. Encapsulating Data
4. Developing iOS Applications

The Course Outcomes (COs). On completion of the course the participants willbe:

COs	Statements
CO 1	**Understanding** the fundamental concepts of variables, constants, and basicdata types in SWIFT.

CO 2	**Analyzing** the use of control flow statements such as for, if, and switch invarious programming scenarios.
CO 3	**Applying** object-oriented concepts in SWIFT, including the use of classes,structures, and protocols.
CO 4	**Creating** functions, closures, and extensions to enhance code modularityand reuse.
CO5	**Evaluating** error handling techniques and type checking mechanisms todevelop robust SWIFT applications

CO = Course outcomes. A student is expected to have learnt concepts and demonstrated/developed abilities or skills related to strategic management at the end of the course.

Course Outline:

Unit Number: 1	**Title: Introduction to SWIFT Language**	**No. of hours:10**
Variables & Constants, Introduction to functions (methods), Arrays, Dictionaries, Data, Date and other basic data types, Enums, structures, closuresFor, If, switchstatement, Object oriented concepts with SWIFT, Type check, AnyObject, Any Protocols, Extensions, Error handling, Working with classes		
Unit Number: 2	**Title: Working with Xcode**	**No. of hours:8**
Introduction to XCODE, COCOA touch framework, iOS application architecture,Application lifecycle		
Unit Number: 3	**Title: Introduction to view controllersand Views**	**No. of hours:12**

View Controllers, view, view lifecycle, Basic Controls – Label, Buttons, Text field, image View, Table view with default cells and customized cells, Collection view with default cells and customized cells, Picker view, Date picker, scroll view, navigation and Tab bar controller, Understanding Interface builder, XIB files, Creating outlets and Actions, Handling touch and gesture events, Segment and Page control, switch view, UIAlertView		
Unit Number: 4	**Title: Integrating with Database**	**No. of hours:10**
Introduction to data storage methods in iOS, Using Core Data, SQLite database,User Defaults, Property List		

References

1. iOS 14 Programming for Beginners: Kickstart your iOS app development journey with the Swift programming language and Xcode 12, 6th Edition,Ahmad Sahar and Craig Clayton

2. Mastering iOS 14 Programming: Build professional-grade iOS applicationswith Swift 5 and Xcode 12

Additional Readings:

Online Learning Resources for Mobile Application Development Using iOS

- Apple Developer Documentation
  - **Description:** Comprehensive documentation and tutorials for iOS appdevelopment using Swift and Xcode.
  - **Link:** Apple Developer Documentation
- Ray Wenderlich: iOS and Swift Tutorials
  - **Description:** A collection of high-quality tutorials and courses on iOSapp development, covering Swift, Xcode, and various iOS frameworks.
  - **Link:** Ray Wenderlich iOS Tutorials
- GitHub: iOS Development Resources
  - **Description:** A curated list of open-source projects, libraries, andresources for learning and improving iOS development skills.

Link: GitHub - iOS Development Resources

MOBILE APPLICATION DEVELOPMENTUSING IOS LAB

Program Name:	**MCA (Master in Computer Application)**		
Course Name:	**Course Code**	**L-T-P**	**Credits**
Mobile Application Development usingiOS Lab	**ENSP459**	0-0-2	1
Type of Course:	Department Specific Elective III		
Pre-requisite(s), if any: Basics of Android			

Proposed Lab ExperimentsDefined

Course Outcomes

COs	
CO 1	**Understanding** and apply fundamental concepts of iOS development using Xcode and the Cocoa Touch framework to build robust and user-friendly applications.
CO 2	**Developing** interactive and dynamic user interfaces in iOS applicationsusing view controllers, views, and gesture recognizers.
CO 3	**Creating** and manage user interfaces and view controllers in iOS applications using Xcode, demonstrating proficiency in InterfaceBuilder and UIKit components.
CO 4	**Developing** interactive and dynamic user interfaces in iOS applications using view controllers, views, and gesture recognizers.

Project No.	**Project Detail**	**Mapped CO/COs**
P1	**Project 1: Task Management App with Xcode and Cocoa Touch Framework** **Problem Statement:** Develop a task management application to create, manage, and track daily tasks using Xcode and the Cocoa Touch framework.	**CO3**
P2	**Project 2: Personal Finance Tracker with View Controllers and Views** **Problem Statement:** Create a personal finance tracker app that allows users to record expenses, view spending categories, and analyze spending habits using view controllersand views.	**CO3**
P3	**Project 3: Recipe Organizer with Database Integration** **Problem Statement:** Develop a recipe organizer app tostore, retrieve, and manage recipes, utilizing Core Data for databaseintegration.	**CO3**
P4	**Project 4: Event Planner with Advanced UI Components** **Problem Statement:** Create an event planner app that helps users organize events, send invitations, and track RSVPs usingadvanced UI components and views.	**CO3**
P5	**Project 5: Fitness Tracker with Data Storage and Retrieval** **Problem Statement:** Develop a fitness tracker app to recordworkouts, track progress, and analyze fitness data using Core Data and SQLite database.	**CO4**

DEVOPS & AUTOMATION

Program Name:	**MCA (Master in Computer Application)**		
Course Name: **DevOps & Automation**	**Course Code**	**L-T-P**	**Credits**
	ENSP411	4-0-0	4
Total Contact Hours	40		
Type of Course:	Department Specific Elective III		
Pre-requisite(s), if any: Nil			

Course Perspective. Throughout the subject, students will engage in hands-on exercises and projects to gain practical experience with various DevOps tools and practices. By the end of the course, students will be well-equipped to embrace the DevOps culture and apply automation techniques to enhance software development,delivery, and operations processes. The course is divided into 4 modules:

a) Introduction to DevOps

b) Version Control and CI/CD

c) Containerization and Orchestration

d) Configuration Management and Monitoring

The Course Outcomes (COs). On completion of the course the participants willbe:

COs	Statements
CO 1	**Understanding** the principles and benefits of DevOps, and its role in enhancing collaboration and efficiency between development andoperations teams.

CO 2	**Acquiring** hands-on experience with popular DevOps tools such as Git, Jenkins, Docker, Kubernetes, and Ansible for implementing continuous integration, continuous delivery, and automated deployment processes.
CO 3	**Demonstrating** proficiency in containerization and orchestration techniquesusing Docker and Kubernetes for efficient and scalable application deployment and management.
CO 4	**Implementing** configuration management and Infrastructure as Code (IaC) using Ansible and Terraform to automate the provisioning andmanagement of infrastructure resources.
CO 5	**Developing** skills in monitoring, logging, and security practices in the context of DevOps, ensuring application performance, resilience, and adherence tosecurity best practices.

CO = Course outcomes. A student is expected to have learnt concepts and demonstrated/developed abilities or skills related to strategic management at the end of the course.

Course Outline:

Unit Number: 1	**Title: Introduction to DevOps**	**No. of hours: 12**
Content Summary: **DevOps Principles and Culture**: Understand the core principles of DevOps and its cultural impact. Collaboration, automation, continuous integration, continuous delivery, and continuous deployment. **DevOps Toolchain**: Overview of tools and technologies used in DevOps practices. Introduction to popular DevOps tools like Git, Jenkins, Docker, Kubernetes, and Ansible. **Version Control with Git**: Branching, merging, and collaborative development using Git. **Continuous Integration (CI):** Setting up CI pipelines with Jenkins for automated building and testing. **Continuous Delivery and Deployment:** Implementing CD pipelines for deploying.		
Unit Number: 2	**Title: Version Control and CI/CD**	**No. hours: of 8**

Content Summary: **Version Control with Git**: Version control concepts, Git workflows, and collaborationstrategies. **Continuous Integration with Jenkins**: Setting up Jenkins pipelines, automatedtesting, and deployment. **Maven Integration**: Integrate Maven for dependency management and buildingprojects.		
Unit Number: 3	**Title: Containerization and Orchestration**	**No. hours: of 8**
Content Summary: **Introduction to Docker:** Docker concepts, container management, and Docker filecreation. **Container Orchestration with Kubernetes:** Kubernetes architecture, deployment,scaling, and networking. **Docker Compose:** Managing multi-container applications with Docker Compose.		
Unit Number: 4	**Title: Configuration Management andMonitoring**	**No. of hours: 12**
Content Summary: **Configuration Management with Ansible**: Ansible playbooks, roles, andinfrastructure automation. **Infrastructure as Code (IaC)**: Terraform for provisioning and managinginfrastructure. **Monitoring and Logging**: Monitoring tools, log management, and applicationperformance monitoring in DevOps. **Security in DevOps:** Implementing security best practices in CI/CD pipelines and containerized environments.		

References

- Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation, Authors: Jez Humble and David Farley, Publisher: Pearson Education, Inc., Year: 2011
- The Kubernetes Book, Author: Nigel Poulton, Publisher: Independentlypublished, Year: 2018

- Building Microservices: Designing Fine-Grained Systems, Author: Sam Newman, Publisher: O'Reilly Media, Inc., Year: 2015
- Microservices Patterns: With examples in Java, Author: Eberhard Wolff,Publisher: Manning Publications, Year: 2018
- Terraform: Up & Running: Writing Infrastructure as Code, Author: YevgeniyBrikman, Publisher: O'Reilly Media, Inc., Year: 2017

Additional Readings:

Online Learning Resources for DevOps & Automation

I) Kubernetes Academy by VMware

a. **Description:** Free courses provided by VMware on Kubernetes, covering everything from basic concepts to advanced orchestration techniques.

b. **Link:** Kubernetes Academy by VMware

II) HashiCorp Learn: Terraform

a. **Description:** HashiCorp's official resource for learning Terraform, providing tutorials and hands-on labs for infrastructure as code.

b. **Link:** HashiCorp Learn - Terraform

III) Docker: Docker for Developers

a. **Description:** Docker's official training resources for developers, covering containerization, Docker Compose, and more.

Link: Docker - Docker for Developers

DEVOPS & AUTOMATION LAB

Program Name:	**MCA (Master in Computer Application)**		
Course Name: **DevOps & Automation Lab**	**Course Code**	**L-T-P**	**Credits**
	ENSP461	0-0-2	1
Type of Course:	Department Specific Elective III		
Pre-requisite(s), if any:			

Lab Experiments Defined

Course Outcomes

COs	Course Outcomes
CO 1	**Implementing** collaborative development and continuous integration using Git and Jenkins, demonstrating proficiency in version control, automated testing, and deployment processes.
CO 2	**Developing** and deploy microservices applications using Docker for containerization and Kubernetes for orchestration, managing multi- container applications efficiently.
CO 3	**Managing** automated infrastructure provisioning and configuration using Ansible and Terraform, demonstrating expertise in infrastructure as code and configuration management.
CO 4	**Implementing** continuous monitoring, logging, and security best practices in a DevOps environment, ensuring application performance, systemhealth, and data integrity.

Project No.	Project Detail	Mapped CO(s)
P1	**Project 1: Collaborative Development and Deployment Using Git and Jenkins** **Problem Statement:** Develop a collaborative project using Git for version control and Jenkins for continuous integration, automated testing, and deployment.	CO1
P2	**Project 2: Microservices Deployment with Docker and Kubernetes** **Problem Statement:** Create and deploy a microservices application using Docker for containerization and Kubernetes for orchestration.	CO2
P3	**Project 3: Automated Infrastructure Management with Ansible and Terraform** **Problem Statement:** Develop a system for automated infrastructure management using Ansible for configuration management and Terraform for infrastructure provisioning.	CO3
P4	**Project 4: Continuous Monitoring and Logging in a DevOps Environment** **Problem Statement:** Implement a continuous monitoringand logging system to track application performance andsystem health in a DevOps environment.	CO4
P5	**Project 5: Secure CI/CD Pipeline Implementation** **Problem Statement:** Develop and implement a secure CI/CD pipeline that ensures the integrity and security of the deployment process.	

.NET FRAMEWORK

Program Name:	**MCA**		
Course Name: **.NET Framework**	**Course Code**	**L-T-P**	**Credits**
	ENSP413	4-0-0	4
Total Contact Hours	40		
Type of Course:	Department Specific Elective III		
Pre-requisite(s), if any:			

Course Perspective. The ".NET Framework" syllabus covers introduction and components of .NET, programming languages, Visual Studio, OOP, exception handling, memory management, Windows Forms/WPF, ASP.NET, web services, .NET Core, Entity Framework, and WCF. Emphasis on practical application and development skills for building robust and secure applications. The course is divided into 4 modules:

a) .NET Framework

b) .NET Framework Fundamentals

c) Building Applications with .NET Framework

d) ASP.NET Framework

The Course Outcomes (COs). On completion of the course the participants willbe:

COs	Statements
CO 1	**Understanding** .NET Framework's architecture, CLR, and CTS for cross-language integration and platform independence.
CO 2	**Applying** OOP concepts in .NET for designing robust software solutions.

CO 3	**Utilizing** Visual Studio debugging for diagnosing and fixing errors in .NETapplications.
CO 4	**Demonstrating** proficiency in memory management and garbage collectionin .NET.
CO 5	**Designing** web applications using ASP.NET, incorporating best practices.

CO = Course outcomes. A student is expected to have learnt concepts and demonstrated/developed abilities or skills related to strategic management at the end of the course.

Course Outline:

Unit Number:1	**Title: .NETFramework**	**No. of hours: 8**
Content Summary: NET Framework - Architecture, Common Language Runtime, Common Type System, Namespaces, Assemblies, Memory Management, Process Management, Class Libraries		
Unit Number:2	**Title:.NET Framework Fundamentals**	**No. of hours: 8**
Content Summary: Object-Oriented Programming (OOP) in .NET, Classes, objects, and inheritance, Exception Handling and Debugging, Debugging techniques and tools in Visual Studio, Logging and error reporting in .NET applications, Memory Management and Garbage Collection, Automatic memory management in .NET, Garbage collection, Finalizers and the Dispose pattern		
Unit Number:3	**Title: Building Applications with .NET Framework**	**No. of hours:12**

<table>
<tr><td colspan="3"><b>Content Summary:</b>
.NET - Declaration, Expression, Control Structures, Function, String, Array, Encapsulation, Class, Property, Indexer, Delegate, Inheritance, Interface, Polymorphism, Exception Handling, Modules, Graphics, File handling and Data Access. .NET – Form- Event–Form Controls – Containers – Menus - Data controls -
Printing – Reporting – Dialogs – Components - Single and Multiple DocumentInterfaces.</td></tr>
<tr><td><b>Unit Number:4</b></td><td><b>Title: ASP. NETFramework</b></td><td><b>No. of hours:12</b></td></tr>
<tr><td colspan="3"><b>Content Summary:</b>
ASP.NET – Web Pages, Web Forms, Web Site Design, Data Controls, Validation Controls, HTML, Navigation Controls, Login Controls, Reports - Master Pages – WebService Architecture - Basic Web Services – Web Reference – Standards</td></tr>
</table>

References

1. Pro C# 8 with .NET Core: Foundational Principles and Practices in Programming by Andrew Troelsen and Philip Japikse, Apress, 9th Edition,2020
2. Pro ASP.NET Core 3 by Adam Freeman, Apress
3. ASP.NET Core in Action by Andrew Lock

Additional Readings:

Online Learning Resources:

I) Online Tutorials and Documentation: Direct students to the official Microsoft documentation for .NET Framework, which provides comprehensive guides and resources. Microsoft .NET Documentation

II) Hands-on Coding Exercises: Assign coding exercises from platforms like LeetCode or HackerRank that focus on implementing concepts of .NET Framework. LeetCode HackerRank

III) Project-Based Learning: Encourage students to work on small projects using different aspects of the .NET Framework. Provide examples of project ideas and resources like GitHub repositories for inspiration. GitHub

.NET FRAMEWORK LAB

Program Name:	**MCA (Master in Computer Application)**		
Course Name: **.NET Framework Lab**	**Course Code**	**L-T-P**	**Credits**
	ENSP463	0-0-2	1
Type of Course:	Department Specific Elective III		
Pre-requisite(s), if any: Nil			

Proposed Lab ExperimentsDefined

Course Outcomes

COs	Statements
CO 1	**Understanding** and apply object-oriented design principles, exception handling, memory management, and debugging techniques to develop robust .NET applications.
CO 2	**Developing** graphical user interfaces and handle events in .NET applications tocreate interactive and user-friendly software solutions.
CO 3	**Implementing** web development techniques in ASP.NET, including web forms, user authentication, master pages, and web services to build secure and dynamic web applications.
CO 4	**Analyzing** and utilize data handling, reporting, and visualization techniques to create comprehensive and functional software systems for various domains.

Proposed Lab Projects

Project. No	**Project Detail**	**Mapped CO/COs**
P1	**Project 1: Comprehensive Inventory Management System Using .NET** **Problem Statement:** Develop a comprehensive inventorymanagement system to track and manage stock levels, orders,and deliveries using .NET Framework Fundamentals.	CO3
P2	**Project 2: Real-Time Customer Feedback System Using .NET** **Problem Statement:** Create a real-time customer feedback system to collect, analyze, and display customer feedback for businesses.	CO3
P3	**Project 3: E-Commerce Website Using ASP.NET** **Problem Statement:** Develop an e-commerce website to manage product listings, user authentication, and online transactions using ASP.NET Framework.	CO4
P4	**Project 4: Health Monitoring Application Using .NET Framework** **Problem Statement:** Create a health monitoring application to track patient health metrics and generate reports for healthcare providers.	CO4
	Project 5: Online Learning Management System Using ASP.NET **Problem Statement:** Develop an online learning management system to manage courses, track student progress, and facilitate communication between students and instructors.	CO4

NEW AGE PROGRAMMING LANGUAGES

Program Name:	**MCA (Master in Computer Application)**		
Course Name: **New-Age programming languages**	**Course Code**	**L-T-P**	**Credits**
	ENSP415	4-0-0	4
Total Contact Hours	40		
Type of Course:	Department Specific Elective III		
Pre-requisite(s), if any: Nil			

Course Perspective. New-Age programming languages (GO, F#, Clojure, Kotlin) provides an introduction to the concepts and applications of modern programming languages. It explore the features and benefits of GO, F#, Clojure, and Kotlin, and develop practical skills in programming using these languages. The course will coverlanguage syntax, data types, control structures, functional programming concepts, concurrency, and integration with other technologies. The course is divided into 4 modules:

1. GO Programming Language
2. F# Programming Language
3. Clojure Programming Language
4. Kotlin Programming Language

The Course Outcomes (COs). On completion of the course the participants willbe:

COs	Statements
CO 1	**Understanding** principles and paradigms of modern programming languages.
CO 2	**Developing** proficiency in syntax, data structures, and control flow of eachlanguage.
CO 3	**Exploring** unique features and strengths of each language.
CO 4	**Applying** development tools to improve code quality and productivity.
CO 5	**Designing** and implement projects integrating multiple programming languages.

CO = Course outcomes. A student is expected to have learnt concepts and demonstrated/developed abilities or skills related to strategic management at the end of the course.

Course Outline:

Unit Number: 1	**Title: GO programming Language**	**No. of hours: 10**

Content Summary:

Overview and Comparison: Overview of GO, F#, Clojure, and Kotlin, Comparison with traditional programming languages, Installation and setup of development environment,

GO Programming Basics: Introduction to GO syntax and data types, Control structures in GO, Functions and packages, Arrays, slices, and maps, Structs and custom data types,Pointers and memory management

Unit Number: 2	**Title: F# Programming Language**	**No. of hours: 10**

Content Summary:

Introduction to F# syntax and functional programming concepts, Data Types, Variables, Operators, Decision Making, Loops, Functions, Strings, Options, Immutable data types and pattern matching, Higher-order functions and currying, Asynchronous and parallel programming in F#, Object-Oriented Programming with F#, Database access with F#, Querying and manipulating data using F#, Integration with relational and NoSQL databases

Unit Number: 3	**Title: Introduction to Clojure Programming**	**No. of hours: 10**

Content Summary:

Introduction to Clojure: Overview of Clojure and its features, Setting up thedevelopment environment,

Basic Syntax and Functional Programming, Basic syntax and data structures, Functional programming concepts, Immutable data and pure functions, Higher-order functions and recursion, Collections and sequence operations, Restructuring and pattern matching

Error Handling and Testing: Exception handling and error management in Clojure,Testing strategies and frameworks in Clojure,

Data Manipulation and Transformation: Data manipulation with Clojure's sequence functions, Data transformation with transducers, Data-driven developmentwith data literals and data readers

Unit Number: 4	**Title: Introduction to Kotlin Programming**	**No. of hours: 10**

Content Summary:

Overview of Kotlin and its advantages, Setting up the development environment, Basic syntax and data types in Kotlin, Conditional statements and loops, Function declarations and parameters, Lambda expressions and higher-order functions,

Object-Oriented Programming in Kotlin: Classes, objects, and inheritance, Properties and access modifiers, Interfaces and abstract classes, Understanding nullable and non-nullable types, Safe calls and the Elvis operator, Type inference and smart casting,

Collections and Functional Programming: Working with lists, sets, and maps inKotlin, Collection operations and transformations, Introduction to functional programming concepts in Kotlin, Creating extension functions in Kotlin, Using DSLsfor domain-specific problems, Builder pattern and DSL implementation.

Text Books:

1. The Go Programming Language, Alan A. A. Donovan and Brian W. Kernighan,Addison-Wesley Professional.
2. An Introduction to Programming in Go, Caleb Doxsey, CreateSpaceIndependent Publishing.
3. Real-World Functional Programming: With Examples in F# and C#, TomasPetricek and Jon Skeet, Manning.
4. Programming F# 3.0: A Comprehensive Guide for Writing Simple Code toSolve Complex Problems, Chris Smith, O'Reilly Media.
5. Getting Clojure: Build Your Functional Skills One Idea at a Time, Russ Olsen, O′Reilly.
6. The Joy of Clojure, Michael Fogus and Chris Houser, Manning Publication.
7. Atomic Kotlin, Bruce Eckel and Svetlana Isakova, Mindview LLC.
8. Kotlin in Action, Dmitry Jemerov and Svetlana Isakova, Manning Publication.

Additional Readings:

Online Learning Resources for New-Age Programming Languages

a) Go (Golang)

1. **Coursera: Programming with Google Go**

    1. **Description:** An introductory course to Go programming, covering language syntax, data structures, and more.

    2. **Link:** Coursera - Programming with Google Go

2. Go by Example

    1. **Description:** A hands-on introduction to Go using annotatedexample programs.

    2. **Link:** Go by Example

b) F#

1. **Microsoft Learn: Introduction to F#**

    1. **Description:** A series of modules introducing the F# language,its syntax, and functional programming concepts.

    2. **Link:** Microsoft Learn - Introduction to F#

c) Clojure

1. **ClojureBridge**

    1. **Description:** Free Clojure workshops for beginners, includingresources and exercises.

    2. **Link:** ClojureBridge

2. Learn Clojure: Clojure for the Brave and True

    1. **Description:** A beginner-friendly book that teaches Clojurethrough real-world projects and examples.

    2. **Link:** Clojure for the Brave and True

d) Kotlin

1. **Kotlin Lang: Kotlin Documentation**

    1. **Description:** Official Kotlin documentation and tutorials byJetBrains.

2. **Link:** Kotlin Documentation

2. Udacity: Kotlin for Android Developers

1. **Description:** A course by Udacity focusing on Kotlin for Androiddevelopment.
   Link: Udacity - Kotlin for Android Developers

Program Name:	**MCA (Master in Computer Application)**		
Course Name:	**Course Code**	**L-T-P**	**Credits**
New Age Programming languages Lab	**ENSP465**	0-0-2	1
Type of Course:	Department Specific Elective III		
Pre-requisite(s), if any: Nil			

Course Outcomes (CO)

COs	Statements
CO1	**Understanding** the fundamental principles and paradigms of modernprogramming languages.
CO2	**Developing** proficiency in using the syntax, data structures, and control flowconstructs of each language.
CO3	**Exploring** the unique features and strengths of each language, such as Go's focus on concurrency, F#'s functional programming capabilities, Clojure's emphasis on immutability and simplicity, and Kotlin'sinteroperability with existing Java code.

CO4	**Applying** the languages' respective development tools and best practices.
CO5	**Designing and implementing** projects that utilize the strengths of eachlanguage to tackle complex problems or tasks.

Lab Experiments

	Experiment Title	
P1	**Project Title:** Development of a Library Management System Using GO **Problem Statement:** Develop a Library Management System to manage book inventory, track book loans, and handle member information using GO programming basics.	PO1
P2	**Project Title:** Development of a Financial Portfolio Management SystemUsing F# **Problem Statement:** Create a Financial Portfolio Management System to track investments, analyze financial data, and provide insights using F# programming language features.	PO2
P3	**Project Title:** Development of a Real-Time Chat Application Using Clojure **Problem Statement:** Create a Real-Time Chat Application to facilitate instant messaging, manage user profiles, and ensure data security using Clojure programming language features.	PO3
P4	**Project Title:** Development of an E-Commerce Platform Using Kotlin **Problem Statement:** Create an E-Commerce Platform to manage products, handle customer orders, and process payments using Kotlinprogramming language features.	PO4

Applied Programming & Problem-Solving Skills for Campus Interviews

Program Name:	**Master of Computer Application (MCA)**		
Course Name: Applied Programming & Problem-Solving Skills for Campus Interviews	**Course Code**	**L-T-P**	**Credits**
		2 - -	2
Type of Course:	MOOC I		
Pre-requisite(s), if any: NA			

Objective: The course "Applied Programming and Problem-Solving Skills for Campus Interviews" is designed to equip students with the essential technical and interpersonal skills required to excel in campus interviews and recruitment processes. By focusing on key areas such as Data Structures and Algorithms, Object-Oriented Programming, Java, and Database Management Systems, the course fosters problem-solving abilities critical for software development and technical roles. In addition to strengthening technical expertise, the course also prepares students for aptitude exams like AMCAT, enhances their soft skills, and simulates real-world interview scenarios through mock interviews. With certifications from Infosys Springboard, this comprehensive program ensures students are well-prepared for the demands of the modern job market.

All students are required to pass the individual components to receive the final marks out of 100 and earn 2 credits for the Practical Training Module in their course structure. Students must obtain specific **free certifications** from Infosys Springboard (https://infytq.onwingspan.com/web/en/page/home).

Course Outcomes (CO)

COs	Statements

CO1	**Understanding** and apply fundamental and advanced concepts in data structures, algorithms, object-oriented programming, and database management systems to solve complex problems efficiently
CO2	**Developing** and enhance programming skills using Java, demonstrating the ability to write, debug, and optimize code for real-world applications
CO3	**Analyzing** and solve aptitude and soft skills assessments, demonstrating proficiency in logical reasoning, quantitative ability, and effective communication
CO4	**Demonstrating** readiness for campus placements through successful participation in mock interviews, AMCAT assessments, and obtaining certifications from Infosys Springboard, showcasing industry-relevant skills and knowledge

Detailed Plan

Module	Link	Hrs
Data Structures and Algorithms - Part 1	https://infytq.onwingspan.com/web/en/app/toc/lex_auth_0125409699132620801065_shared/overview	30
Data Structures and Algorithms - Part 2	https://infytq.onwingspan.com/web/en/app/toc/lex_auth_0127667384693882883448_shared/overview	38
Object Oriented Programming	https://infytq.onwingspan.com/web/en/app/toc/lex_auth_0125409722749255681063_shared/overview	46
Programming using Java	https://infytq.onwingspan.com/web/en/app/toc/lex_auth_012880464547618816347_shared/overview	113
Database management Systems (part I)	https://infytq.onwingspan.com/web/en/app/toc/lex_auth_01275806667282022456_shared/overview	64
Database management Systems (Part II)	https://infytq.onwingspan.com/web/en/app/toc/lex_auth_0127673005629194241_shared/overview	40
Aptitude Exam	Online Sessions to be conducted by KRMU	NA

AMCAT (Aspiring Minds Computer Adaptive Test)	To be organized by KRMU through External agency	NA
Softs Skills	Online Sessions to be conducted by KRMU	NA
MOCK Interview	To be organized by KRMU	

Important Notes

1. **Training on Soft Skills and Aptitude:** KRMU will organize online training sessions on Soft Skills and Aptitude components. Assessments will be conducted by the CDC.
2. **Certification Submission:** Full marks for certifications will only be awarded upon submission of the final certificate.
3. **AMCAT Exam Requirement:** Students are required to appear for the AMCAT exams. The AMCAT is a widely recognized employability assessment used by many companies to evaluate candidates' skills and job readiness. For the syllabus, please explore https://www.myamcat.com.
4. **MOCK Interviews:** These will focus on your preparation for placements, testing your coding skills, and preparation for technical and HR interview rounds.
5. Hard copies of the certificates for the specified certifications must be submitted to the Dean’s office by the designated deadline. Certificates submitted after the deadline will not be accepted.

Learning Experiences

Classroom Learning Experience

1. **Interactive Workshops**: Engage in hands-on workshops focused on essential career skills.
2. **Problem-Based Assignments**: Tackle real-world challenges to enhance job readiness and critical thinking.
3. **Mock Interviews**: Participate in simulated interviews to practice communication and presentation skills.
4. **Collaborative Group Work**: Work in teams on projects that mimic workplace scenarios and dynamics.
5. **Continuous Feedback**: Receive regular assessments and constructive feedback from instructors.

Outside Classroom Learning Experience

1. **Moodle Access**: Access course materials anytime via Moodle for flexible learning.
2. **Question Bank**: Utilize a question bank and resources for job application preparation.
3. **Networking Opportunities**: Participate in events to connect with industry professionals and expand your network.

Semester 4

Program	**Master of Computer Application (MCA)**		
Course Name: Industry Project / Research Project	Course Code	L-T-P	Credits
	SIMM852		12
Type of Course:	PROJ		
Pre-requisite(s), if any:			

Preface:

The **MCA Full-Time Project Work** is a culmination of the academic journey for engineering students at the School of Engineering & Technology, K.R. Mangalam University. This detailed Standard Operating Procedure (SOP) is designed to guidestudents through their project, ensuring a comprehensive, practical, and outcome-driven approach.

The SOP provides a framework for students to choose from three types of projects—**Industrial Projects**, **Research & Development (R&D) Projects**, and**Start-up Projects**. It emphasizes experiential learning, real-world problem- solving, and interdisciplinary collaborations focusing on holistic development, innovation, and entrepreneurship. Students will work under the mentorship of both internal faculty and external experts, ensuring they are equipped with the skills and knowledge required to excel in industry, research, or entrepreneurship.

This document outlines each stage of the project work, from proposal submission to final evaluation, and offers clear guidelines for successful completion. By adhering to this SOP, students will not only demonstrate their technical proficiencybut also contribute meaningfully to industry, academia, and society.

Standard Operating Procedure (SOP) for MCA Final Semester Full-Time Project Work

1. Introduction

The **MCA Final Semester Full-Time Project Work** is an essential academic requirement aimed at providing students with the opportunity to apply theoretical knowledge to practical challenges. The project is designed to foster critical thinking, problem-solving, innovation, and research-orientedlearning, with a focus on real-world industrial, research, and entrepreneurial domains. Students may choose from:

- **Industrial Project**: Solving real industrial problems in collaborationwith an industry partner.
- **Research & Development (R&D) Project**: Contributing to academic and applied research, with external guidance from academic/research institutions.
- **Start-up Project**: Developing and launching innovative start-up ideas with entrepreneurial mentors.

---

2. Objectives

The primary objectives of the full-time project are:

- **Application of Theoretical Knowledge**: Enabling students to applytheir academic learning to practical problems.
- **Holistic Development**: Promoting interdisciplinary learning, criticalthinking, creativity, and problem-solving.
- **Research and Innovation**: Encouraging innovative solutions, leadingto publications, patents, or prototypes.
- **Industry Collaboration**: Fostering partnerships with industries forreal-world problem-solving.
- **Entrepreneurship Development**: Developing entrepreneurial skillsand creating viable start-ups.

- **Global Competency**: Ensuring students develop the skills required to excel in global environments through research, innovation, and collaboration.

---

3. Types of Projects

a) Industrial Project

Students working on **Industrial Projects** will:

- Collaborate with an industry partner.
- Identify specific, real-world challenges faced by the company.
- Propose and implement a solution that provides value to the industry.
- Develop a final product or prototype that can be implemented in theindustrial setting.

Project Proposal:

- Problem Statement and Objectives: Identify the industrial problem andoutline the objectives.
- Proposed Solution: Present a detailed methodology for solving theproblem.
- Deliverables: Define tangible deliverables, including prototypes, software, or hardware.
- Expected Impact: Outline the expected impact on the industry.

Evaluation Criteria:

- Practical implementation and solution viability (40%)
- Project innovation (20%)
- Industrial applicability and impact (20%)
- Final presentation and report quality (20%)

b) Research & Development (R&D) Project

The **R&D Project** focuses on creating innovative research outcomes throughcollaborations with academic or research institutions. This can result in publications, research reports, or new discoveries.

Project Proposal:

- Literature Review: Detailed research on existing work related to the chosen topic.
- Hypothesis/Research Questions: Define the specific research problem or question.
- Methodology: Include data collection, experimental design, and analysis techniques.
- Research Timeline: Step-by-step phases of research with milestones.

External Mentor: Collaboration with an **external academic expert** is mandatory for research projects. The external mentor must be a researchprofessional with expertise in the specific field of study.

Internal Mentor: Each student will also be assigned an **internal faculty member** who will supervise the project. The internal mentor will ensure thatthe research meets academic standards and deadlines.

Evaluation Criteria:

- Quality of Research and Novelty (30%)
- Research Methodology (25%)
- Contributions to the field (20%)
- Final Report, Presentation, and Publication (25%)

c) Start-up Project

The **Start-up Project** involves developing a business model or creating a start-up venture. Students work on a product/service idea that addresses asignificant market need or societal problem.

Project Proposal:

- Start-up Idea: Explain the business or product idea.
- Market Research: Detailed research on the market, target customers, competitors, and potential revenue streams.
- Business Plan: Define the steps needed to take the idea to market, including funding, development phases, marketing, and operational plans.
- Product Prototype: If applicable, develop a working prototype.

Mentorship:

- **External Mentor**: An industry/start-up expert will guide the studentin refining the idea, business model, and market strategy.
- **Internal Faculty Mentor**: An internal mentor will provide academicguidance and ensure the start-up idea is feasible and innovative.

Evaluation Criteria:

- Start-up viability and market potential (30%)
- Product or service innovation (30%)
- Prototype/Business Model Development (20%)
- Final Pitch/Presentation and Start-up Plan (20%)

4. Roles and Responsibilities

a) Student's Responsibilities:

- Select a suitable project topic based on interests (industrial, R&D, orstart-up).
- Draft and submit a detailed proposal with objectives, methodology,timelines, and deliverables.
- Coordinate with both external and internal mentors regularly forfeedback and guidance.
- Maintain a weekly progress report for both mentors.
- Submit a final comprehensive report and present the project.

b) Internal Supervisor:

- Guide the student throughout the project.
- Provide academic input and ensure that the project aligns with theprogram outcomes.
- Conduct progress reviews and ensure timelines are adhered to.
- Evaluate the project at the mid-term and final stages.

c) External Mentor:

- Offer specialized industrial, research, or entrepreneurial guidance.

- Provide real-world problem insights for industrial and start-up projects.
- Ensure the project is relevant to the chosen industry, research domain,or start-up ecosystem.
- Participate in the final evaluation of the project.

5. Project Phases

Phase 1: Proposal Submission and Approval

- Students will submit a project proposal during the first two weeks ofthe final semester.
- The proposal must include the problem statement, objectives, literature review (for R&D projects), methodology, and expected outcomes.
- The proposal is subject to review and approval by the internal supervisor and external mentor.

Phase 2: Planning and Resource Allocation

- Once approved, the student will develop a project plan that includes:
  - **Project Milestones**: Break down the project into smaller taskswith defined milestones.
  - **Resource Requirements**: Identify any software, hardware, labresources, or tools required for the project.
  - **Team Roles**: For group projects, define the roles of each teammember.
  - **Risk Assessment**: Highlight potential risks and the corresponding mitigation strategies.

Phase 3: Mid-term Review

- A mid-term review will be conducted halfway through the project to assess progress.
- Students will present their work to a committee consisting of the internal supervisor, external mentor, and department head.

- The review will assess the progress against the timeline and suggest course corrections if needed.

Phase 4: Final Execution and Evaluation

- **Industrial Projects**: Students must submit a prototype or industrial report, demonstrating the solution's applicability to the industry.
- **R&D Projects**: Students must submit a final research report or publish findings in academic journals.
- **Start-up Projects**: Students must present a business plan, along with a working prototype, market analysis, and revenue model.

Phase 5: Final Report Submission and Presentation

- **Final Report**: The project report should contain a title page, abstract, introduction, problem statement, objectives, methodology, results, discussion, conclusions, future scope, references, and appendices.
- **Presentation**: Students will deliver a final presentation to a panel of evaluators, showcasing their work, findings, or product.
- **Evaluation**: Based on the final report and presentation, students will be awarded marks in accordance with the evaluation rubrics.

6. Collaboration and Mentorship

For **Research Projects**, the mentorship will involve both:

- **External Mentor**: An academic expert outside the institution, preferably from a reputed university or research institute.
- **Internal Mentor**: A faculty member from the student's department to provide academic and administrative guidance.

For **Industrial Projects**:

- External mentorship will come from industry professionals, preferablyfrom the partnering company.

For **Start-up Projects**:

- External mentorship will involve experienced entrepreneurs, start-upfounders, or investors.

Mentors will:

- Provide critical inputs on the technical, business, or research aspectsof the project.
- Offer feedback and advice during each phase of the project.

7. Documentation and Submission Requirements

Students are required to:

- Submit their proposal, mid-term report, final report, and any supporting documents via the **Learning Management System (LMS)**.
- Maintain detailed project logs and weekly reports.