SOMAIYA
VIDYAVIHAR UNIVERSITY
K J Somaiya School of Engineering
(formerly K J Somaiya College of Engineering)

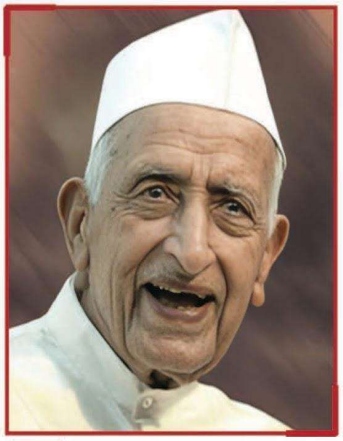
Syllabus
Third Year
B. Tech. Computer Engineering
(Programme commenced from 2023-24)
(Department of Computer Engineering)

From
Academic Year 2025-26
(SVU –KJSCE 2.0)
(Approved by BOS dated 17/04/2025,
Academic Council dated 28/04/2025, Item No. 14.04)

SOMAIYA VIDYAVIHAR UNIVERSITY

Somaiya TRUST

Our Vision



Padma Bhushan
Shri Karamshi Jethabhai Somaiya
Our Founder

Padma Bhushan Shri Karamshi Jethabhai Somaiya's first initiative in education was the founding of a school in 1942 in rural Maharashtra, to provide quality holistic education. It was founded on the belief that education is an important pillar of nation-building with the power to change lives and that it is the duty of the privileged to provide education to the needy.

He later founded the Girivanvasi Pragati Mandal, The K J Somaiya Medical Trust, the Girivanvasi Education Trust and sister institutions to make great citizens of India and the world. In the words of Swami Vivekananda, "We want that education by which character is formed, strength of mind is increased, the intellect expanded, and by which one can stand on one's own feet." We have now grown into a multi-disciplinary and multi-campus education institution with over 1500 faculty, and 39000+ students.

ज्ञानादेव तु कैवल्यम्।

KNOWLEDGE ALONE LIBERATES

Our motto is: ज्ञानादेव तु कैवल्यम्। Knowledge alone liberates. Liberates from poverty, from hunger. Also to liberate one from the attachments that bind us to small-mindedness. Knowledge also provides opportunity. To make the life lived more meaningful. In the service of one's family, one's community, one's समाज, country, and indeed the world. Bearing in mind that there is no religion other than the life lived in the service of humanity.

न मानुषात् परो धर्मः।

We will strive to provide access and opportunity to build a more inclusive society.

Our education in any subject will reflect its timeless fundamentals, its current context, and its applications. There is so much scientific discovery taking place, at the intersection of fields, in biology, computing, medicine, the social sciences and everywhere else. We will provide students and faculty with an environment to engage with this world, discover new truths, and make new applications to create and share knowledge.

Our education will also be experiential. With projects that are 'real' and those that complement the learning inside the classroom. Our students and faculty will be at the cutting edge of change, to incubate companies, to create NGOs, and pursue any field of their passion.

Our education will also be holistic. Sports and physical exercise must be a firm part of the curriculum. For students to develop a love for sports, for recreation, for health, for teamwork, and for competition. Our education will also instil an appreciation for art, culture, nature and biodiversity.

अभ्यासेन तु कौन्तेय वैराग्येण च गृह्यते।

In the Bhagavad Gita, Arjun asks Krishna how is one to control one's mind, which is as fleeting as the wind. Krishna responds that it can only be done through practice and discipline. अभ्यासेन तु कौन्तेय वैराग्येण च गृह्यते। We will strive to teach our students to learn to stay calm in our turbulent world. And our education will also include the ancient Indian tradition, its culture, its depth, and its knowledge. We must keep the connection with our mother tongue and our languages. Languages are storehouses of culture, and the loss of a language takes with it much learning, stored through it over the ages.

Finally, our education will help students lead a full life, and to fall in love with life. Our dream is to build a world-class research and teaching institution, that is global in the reach of its ideas, and universal in its service. Welcome to our community.

Our Mission

To nurture excellence and provide freedom of possibilities in education and research to foster a culture of creativity, innovation, leadership, responsible citizenship, service, and all-round growth.

From the Desk of Dean Faculty of Engineering and Technology:

In the era of technological revolution, engineering education must evolve to keep pace with the dynamic demands of industry and society. Our engineering institute is committed to fostering a learning environment that nurtures innovation, creativity, and a profound understanding of engineering principles. The **National Educational Policy 2020 (NEP 2020)** framed by the Government of India recommends a holistic, inclusive, and flexible approach to ensure equitable access to quality education across all levels, promote multidisciplinary research, and impart skill-based education with integration of technology.

Somaiya Vidyavihar, with its esteemed legacy in education, has consistently upheld the values of excellence, inclusivity, and innovation. Applicable for **Somaiya Vidyavihar University (SVU)'s** undergraduate engineering programs, the **SVU Scheme 2023** presented here is aligned with the transformative vision of Somaiya Vidyavihar as well as NEP 2020 to cultivate a holistic, experiential, and interdisciplinary approach to engineering education. The **salient features** of the scheme include:

Professional Core and Elective Courses: The curriculum includes state-of-the-art courses that cover both the fundamentals and emerging trends in respective branches of engineering. With an optimal balance between theoretical knowledge and practical application, core courses provide a strong foundation in essential engineering principles, while elective courses offer flexibility for students to explore and specialize in areas of interest.

Open Elective Courses: Recognizing the importance of interdisciplinary knowledge, the curriculum includes a diverse range of Open Electives categorized into four types: Open Elective Technology (OET), Open Elective Humanities, Open Elective Management (OEHM), and Open Elective Generic (OEG). These courses, offered at institute-level, enable students to expand their knowledge across various disciplines, fostering a versatile skill set and adaptability in an ever-evolving global landscape.

Innovation and Project-based Learning (PBL): The curriculum engages students in innovation and PBL through ideation, mini and major projects right from the first year to the final year of engineering. With diverse projects, collaboration, and field work/community engagement initiatives, students gain a profound understanding of engineering concepts and contribute through innovative solutions to the Sustainable Development Goals (SDGs), societal challenges and advancements.

Learning-by-Doing: The curriculum places emphasis on exposure courses through Skill-Based Learning (SBL) and Activity-Based Learning (ABL), focusing on responsibilities towards society, problem-solving abilities, leadership and teamwork, motivation for life-long learning, etc.

Elements of the Indian Knowledge System: The curriculum incorporates aspects of the Indian Knowledge System that emphasize on drawing insights from ancient wisdom and rich intellectual heritage of India to address modern challenges.

Internships and Research: Enabling students to gain industry insights and enhance their employability, the curriculum integrates flexible internship opportunities in Semester VII or VIII, allowing students to gain hands-on experience in industries, government sectors, NGOs, and MSMEs. Alternatively, they can opt for a specialized research project and courses in Semester VIII. Besides this Semester-long Internship, all the students are required to complete a mandatory 10-week internship over four years, with a maximum of 4 weeks dedicated to socially relevant internships and a minimum of 6 weeks in technical domains.

Learning through MOOCs: The curriculum leverages and promotes Massive Open Online Courses (MOOCs) to offer students flexible and diverse learning opportunities. Complementing on-campus education, students can learn through MOOCs for Open Electives – OET and OEHM during the Pre-final and Final Year, as well as Professional Core courses during their Internship.

Student Exchange Programs: The curriculum also offers student exchange programs that promote global exposure and cross-cultural learning, elevating academic and personal growth. Interested students can participate in the Student Exchange Programs as an alternative to the semester-long internship. Credits from the foreign university where they study will be transferred, providing them with an opportunity to experience different educational systems, cultures, and perspectives.

Minors Courses: Students can expand their academic horizons by pursuing minors in disciplines other than their major, earning additional credits. These minor courses provide an opportunity to acquire multidisciplinary knowledge, significantly enhancing their versatility and adaptability in the professional world.

Honors Courses: For high-achieving students, the SVU 2023 scheme offers Honors courses that delve deeper into specialized topics and gain additional credits for the same. These advanced courses align with high-end industry standards and provide an enriched learning experience, offering multiple opportunities to expand knowledge and expertise in areas of interest.

This forward-thinking SVU 2023 scheme is designed to equip our graduating engineers to emerge as innovative leaders, capable of addressing global challenges and contributing to the advancement of society. Our Boards of Studies, comprising experts in different disciplines, have meticulously designed syllabus for various programs under this SVU 2023 Scheme. We are confident that the joint efforts of the faculty, alumni, students, industry experts, and all the stakeholders will strengthen the academic, research, and entrepreneurial culture of our institution, reinforcing K. J. Somaiya College of Engineering's position as one of the premier engineering institutions in the nation and a top choice for engineering aspirants.

Dr. S. K. Ukarande

Dean – Faculty of Engineering and Technology
Somaiya Vidyavihar University, Mumbai

From the Desk of Chairperson, Board of Studies:

It is with immense pleasure that I present the meticulously designed syllabus for the Third year B.Tech in Computer Engineering. This curriculum has been thoughtfully crafted to provide students with a strong academic foundation, industry-relevant skills, interdisciplinary exposure, and continuous assessment mechanisms that foster lifelong learning and professional growth.

Recognizing the dynamic nature of the computing field, this syllabus is structured to ensure a balance between fundamental knowledge, specialization, practical skill development, and innovation-driven learning.

The salient features of this curriculum include:

- Programme Core Courses: Six essential courses are included in the third year of the programme, ensuring that students develop a solid understanding of core principles in computer engineering, spanning algorithms, programming paradigms, systems design, and emerging technologies.

- Programme Elective Courses: With two elective courses, students have the flexibility to tailor their learning experience and explore advanced topics that align with their interests and career aspirations. This approach nurtures specialization while encouraging adaptability to evolving industry trends.

- Open Electives: Recognizing the importance of holistic education, students are offered one course each in Humanities and Management. These electives facilitate broader learning experiences, helping students cultivate communication skills, ethical reasoning, leadership qualities, and an understanding of real-world socio-economic implications.

- Skill Development: Practical expertise is essential in today's technology-driven world. To enhance hands-on proficiency, this curriculum incorporates a dedicated Full Stack Development laboratory course, equipping students with crucial web development, database management, and system integration skills that are highly sought after in the industry.

- Project-Based Learning: Innovation and problem-solving take centre stage with the introduction of a Capstone project from Semester VI onwards. This initiative encourages students to apply their knowledge in real-world scenarios, engage in interdisciplinary collaboration, and develop solutions that address contemporary challenges in computing and technology.

- Continuous Assessment: To ensure consistent academic progress and skill enhancement, all theory and laboratory courses follow a rigorous continuous assessment framework. This approach emphasizes learning through frequent evaluations, feedback mechanisms, and iterative improvements, fostering a culture of academic excellence and accountability.

- Applications in Biological and Life Sciences: Recognizing the increasing intersection between computing and life sciences, relevant courses in the curriculum integrate applications and case studies from biological and healthcare domains. This exposure enables students to explore computational techniques in genomics, bioinformatics, medical imaging, and healthcare systems, preparing them for interdisciplinary innovation.

This comprehensive syllabus is aligned with the latest industry advancements, pedagogical best practices, and the evolving needs of the digital era. By integrating structured learning with innovation-driven strategies, the programme aims to shape graduates into competent professionals, researchers, and entrepreneurs, ready to lead and contribute meaningfully to the global tech landscape.

We look forward to an enriching and transformative academic journey for all our students and welcome them to this exciting learning experience.

Dr. Deepak H. Sharma
Chairperson
Board of Studies Computer Engineering

Board of Studies in Computer Engineering

Dr. Deepak H. Sharma	Chairperson
Dr. Parikshit Mahalle	Academician Member
Mr. Chetan Mistry	Industry Member
Dr. G. Sivakumar	Research Institute Member
Mr. Dhaval Chothani	Alumni Industry Member
Dr. Manish Potey	Faculty Member (Professor)
Dr. Bhakti Palkar	Faculty Member (Associate Professor)
Dr. Archana Gupta	Faculty Member (Assistant Professor)

Program Outcomes (PO):

After successful completion of the program Computer Engineering Graduate will be able to:

PO 1: ***Engineering knowledge:*** *Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineerin problems.*

PO 2: ***Problem analysis****: Identify, formulate, review research literature, and analyze comple engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.*

PO 3: ***Design/development of solutions:*** *Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, a environmental considerations.*

PO 4: ***Conduct investigations of complex problems:*** *Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.*

PO 5: ***Modern tool usage:*** *Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.*

PO 6: ***The engineer and society:*** *Apply reasoning informed by the contextual knowledge to assess societal, cultural, environmental, health, safety and legal issues relevant to the professional engineering practice; understanding the need of sustainable development*

PO 7: ***Multidisciplinary Competence:*** *Recognize/ study/ analyze/ provide solutions to real-li problems of multidisciplinary nature from diverse fields*

PO 8: ***Ethics:*** *Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice*

PO 9: ***Individual and team work:*** *Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings*

PO 10: ***Communication:*** *Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend a write effective reports and design documentation, make effective presentations, and giv and receive clear instructions.*

PO 11: ***Project management and finance:*** *Demonstrate knowledge and understanding of engineering and management principles and apply these to one's own work, as member and leader in a team, to manage projects and in multidisciplinary environmen*

PO 12: ***Life-long learning:*** *Recognize the need for, and have the preparation and ability engage in independent and life-long learning in the broadest context of technologi change.*

Program Specific Outcomes (PSO):

PSO 1:	Design, construct and implement hardware and software based modern Computin Information systems with varying complexities
PSO 2:	Demonstrate competence in designing, implementation and maintenance of compu based applications, computer-controlled equipment and networks of intelligent devic

Acronyms used:

1. **Acronyms for Category of course and Syllabus Document**

Acronym for category of courses		Acronyms used in syllabus document	
Acronym	**Definition**	**Acronym**	**Definition**
BS	Basic Science Courses	**CA**	Continuous Assessment
ES	Engineering Science	**ESE**	End Semester Exam
HS	Humanities and Social Sciences including Management Courses	**IA**	Internal Assessment
PC	Professional Core Courses	**LAB /TUT CA**	Continuous Assessment of Laboratory/ Tutorial
PE	Professional Elective courses	**TH**	Theory
OET	Open Elective - Technical	**TUT**	Tutorial
OEHM	Open Elective - Humanities and Management	**ISE**	In- Semester Examination
LC	Laboratory Courses	**CO**	Course Outcome
PR	Project	**PO**	Program Outcome
EX	Exposure Course	**PSO**	Program specific Outcome

2. **Type of Course**

Acronym used	Definition
C	Core Course
E	Elective Course
O	Open Elective Technical
H	Open Elective Humanities/Management/SWAYAM-NPTEL
P	Project
L	Laboratory Course
T	Tutorial
X	Exposure course
W	Workshop
V	Value Based Course

3. Eight Digit Course code e.g. 216U06C301

Acronym Serially as per code	Definition
2	SVU 2023 Second revision
16	College code
U	Alphabet code for type of programme
06	Programme code
C	Type of course
3	Semester number - Semester III
01	Course Number

Semester V
Credit and Examination Scheme

SEMESTER V (With Effect from 2025-26)

Credit Scheme

Course Code	Course Category	Name of the Course	Teaching Scheme TH-PR-TUT	Total (hrs.)	Credit Scheme TH-PR-TUT	Total Credits
216U01C501	PC	Software Engineering	3 – 0--0	03	3 – 0 – 0	03
216U01C502	PC	Computer Networks	3 – 0--0	03	3 – 0 – 0	03
216U01C503	PC	Theory of Automata	3 – 0--1	04	3 – 0 – 1	04
216U01E51x	PE	Departmental Elective-I	3 – 0--0	03	3 – 0 – 0	03
216U06H5xx	HS	OEHM 1 (College offered/ NPTEL/Coursera)	3 – 0– 0	03	3 – 0–0	03
216U01L501	PC	Software Engineering Lab	0 –2– 0	02	0 – 1 – 0	01
216U01L502	PC	Computer Networks Lab	0 – 2 – 0	02	0 – 1 – 0	01
216U01L51x	PE	Departmental Elective-I Lab	0 – 2 – 0	02	0 – 1 – 0	01
216U01L504	PR	Full Stack Development Lab	0 – 2 – 1	03	0 – 1 – 1	02
		Total	**15 – 08– 02**	**25**	**15 – 04 – 02**	**21**

Examination Scheme

Course Code	Course Category	Name of the Course	LAB/ TUT CA	CA		ESE	Total
				IA	ISE		
216U01C501	PC	Software Engineering	--	20	30	50	100
216U01C502	PC	Computer Networks	--	20	30	50	100
216U01C503	PC	Theory of Automata	25	20	30	50	125
216U01E51x	PE	Departmental Elective-I	–	20	30	50	100
216U06H5xx	HS	OEHM 1 (College offered/ NPTEL/Coursera)	--	20	30	50	100
D216U01L501	PC	Software Engineering Lab	50	--	--	--	50
216U01L502	PC	Computer Networks Lab	50	--	--	--	50
2116U01L51x	PE	Departmental Elective-I Lab	50	--	--	--	50
216U01L504	PR	Stack Development Lab	50	--	--	--	50
		Total	**225**	**100**	**150**	**250**	**725**

List of Departmental Elective - I for Semester-V:

Sr. No.	Course Code	Course Name	Remarks
1	216U01E511	Computer Graphics	
2	216U01E512	Modern Data Architectures	
3	216U01E513	Soft Computing	
4	216U01E514	Quantum Computing	
5	216U01E515	Software Testing and Quality Assurance	
6	216U01E516	Applied Cryptography	Not available to Honours in Cyber Security and Forensics
7	216U01E517	Data Mining and Business Intelligence	Not available to students with Honours in Data Science and Analytics

Semester VI

Credit and Examination Scheme

SEMESTER VI (With Effect from 2025-26)

Credit Scheme

Course Code	Course Category	Name of the Course	Teaching Scheme TH-PR-TUT	Total (hrs.)	Credit Scheme TH-PR-TUT	Total Credits
216U01C601	PC	Digital Signal & Image Processing	3 -0 -0	03	3 – 0 – 0	03
216U01C602	PC	Network & Information Security	3 - 0 - 0	03	3 – 0 – 0	03
216U01C603	PC	Artificial Intelligence	3 - 0 - 0	03	3 – 0 – 0	03
216U01E6xx	PE	Departmental Elective-II	3 - 0 - 0	03	3 – 0 – 0	03
216U06H6xx	HS	OEHM 2 (College offered/ NPTEL/Coursera)	3 - 0 - 0	03	3 – 0 – 0	03
216U01L601	PC	Digital Signal & Image Processing Lab	0 – 2 –0	02	0 – 1 –0	01
216U01L602	PC	Network & Information Security Lab	0 – 2 – 0	02	0 – 1 – 0	01
216U01L603	PC	Artificial Intelligence Lab	0 – 2 – 0	02	0 – 1 – 0	01
216U01L6xx	PE	Departmental Elective-II Lab	0 – 2 – 0	02	0 – 1 – 0	01
216U01P601	PR	Project 1	0 – 4 – 0	04	0 – 1 – 1	02
		Total	**15–12–0**	**27**	**15–05–01**	**21**

Examination Scheme

Course Code	Course Category	Name of the Course	LAB/ TUT CA	CA		ESE	Total
				IA	ISE		
216U01C601	PC	Digital Signal & Image Processing	--	20	30	50	100
216U01C602	PC	Network & Information Security	--	20	30	50	100
216U01C603	PC	Artificial Intelligence	--	20	30	50	100
216U01E6xx	PE	Departmental Elective-II	--	20	30	50	100
216U06H6xx	HS	OEHM 2 (College offered/ NPTEL/Coursera)	–	20	30	50	100
216U01L601	PC	Digital Signal & Image Processing Lab	50	--	--	--	50
216U01L602	PC	Network & Information Security Lab	50	--	--	--	50
216U01L603	PC	Artificial Intelligence Lab	50	--	--	--	50
216U01L6xx	PE	Departmental Elective-II Lab	50	–	–	–	50
216U01P601	PR	Project 1	50	--	--	--	50
	--	**Total**	**250**	**100**	**150**	**250**	**750**

List of Departmental Elective - II for Semester-VI:

Sr. No.	Course Code	Course Name	Remarks
1	216U01E611	Compiler Construction	
2	216U01E612	Virtual Reality & Augmented Reality	
3	216U01E613	Embedded System and IOT	
4	216U01E614	Mobile Communication and Mobile App Development	
5	216U01E615	Microservices Foundations	
6	216U01E616	Agile Project Management	
7	216U01E617	Cloud Computing and Virtualization	
8	216U01E618	Machine Learning	Not available to students with Honours in Data Science and Analytics
9	216U01E619	Introduction to Biotechnology & Bioinformatics	

Course Code	Name of the Course			
216U01C501	Software Engineering			
Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	03	--	--	03
Credits Assigned	03	--	--	03

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	--	20	30	50	100

Course prerequisites (if any):

-

Course Objectives:

The Course focuses at developing an understanding of software process models such as the waterfall and evolutionary models. It further provides an understanding of software requirements and the SRS documents. The course aims at enabling the students to prepare the system design and test cases for proper testing of the software.

Course Outcomes:

At the end of successful completion of the course the student will be able to

CO1	Understand the software development process and Estimate different types of resources for the given project.
CO2	Analyze the software requirements and Model the defined problem with the help of UML diagram.
CO3	Prepare the System Design and Model
CO4	Identify and manage configuration items and risks for the software
CO5	Test the given software for different test cases with proper test planning.

Module No.	Unit No.	Details	Hrs.	CO
1	**The Product and the Process:**		**09**	**CO 1**
	1.1	Software life cycle models: Waterfall, RAD, Spiral, Agile process.		
	1.2	Understanding software process, Process metric, CMM Levels		
	1.3	Planning & Estimation: Product metrics Estimation- LOC, FP, COCOMO models.		
	1.4	Project Management activities : Planning, Scheduling and Tracking		
2	**Requirement Engineering**		**09**	**CO 2**
	2.1	Introduction to OO Methodologies :Booch,Rambaug and Jacobson		
	2.2	Requirements Engineering Tasks, Requirement Elicitation Techniques, Software Requirements: Functional, Non- Functional		
	2.3	Requirements Characteristics, Requirement qualities, Requirement Specification, Requirement Traceability, System Analysis Model Generation, Documentation : Use Case Diagram, Acitvity Diagram		
	2.4	Categorizing classes: entity, boundary and control ,Modeling associations and collections-Class Diagram		
	2.5	Dynamic Analysis - Identifying Interaction – Sequence and Collaboration diagrams, State chart diagram		
3	**System Design Engineering**		**8**	**CO 3**
	3.1	Design quality, Classification of Design Activities, Design Concepts: Modularity and Layering, Introduction to Pattern-Based Software Design,		

	3.2	Software Architecture, Data Design, Object-Oriented versus Function-Oriented Design, Design of Software Objects, Methods, Cohesion and Coupling between Objects,		
	3.3	User Interface Design: Rules, User Interface Analysis and Steps in Interface Design, Design Evaluation		
	3.4	Software Reuse, Component-Based Software Engineering		
	3.5	Agile Software Development Methods, Introduction to Devops and MLops		
4	**System Implementation, Configuration Management & Risk Management**		**10**	**CO 4**
	4.1	Packages and interfaces: Distinguishing between classes/interfaces, Exposing class and package interfaces		
	4.2	Mapping model to code , Mapping Object Model to Database Schema		
	4.3	Component and deployment diagrams: Describing Dependencies		
	4.4	Managing and controlling Changes, Managing and controlling version		
	4.5	Categories of Risks, Nature Of Risk, Types of Risk, Risk Identification, Risk Assessment, Risk planning and control, Risk management, Evaluating risk to schedule, PERT technique.		
5	**Testing and Maintenance**		**9**	**CO 5**
	5.1	Testing Concepts: Purpose of Software Testing, Testing Principles, Goals of Testing, Testing aspects: Requirements, Test Scenarios, Test cases, Test scripts/procedures,		
	5.2	Strategies for Software Testing, Testing Activities: Planning Verification and Validation, Software		

		Inspections,FTR		
	5.3	Levels of Testing : unit testing, integration testing, regression testing, product testing, acceptance testing and White-Box Testing		
	5.4	Black-Box Testing: Test Case Design Criteria, Requirement Based Testing, Boundary Value Analysis, Equivalence Partitioning		
	5.5	Object Oriented Testing: Review of OOA and OOD models, class testing, integration testing, validation testing		
	5.6	Reverse and re-engineering, types of maintenance		
	#Self-Learning : Testing tools			
		Total	**45**	

#Students should prepare all Self Learning topics on their own. Self-learning topics will enable students to gain extended knowledge of the topic. Assessment of these topics may be included in IA and Laboratory Experiments.

Recommended Books:

Sr. No.	Name/s of Author/s	Title of Book	Name of Publisher with country	Edition and Year of Publication
1	Roger Pressman	Software Engineering	Tata McGraw Hill.	Ninth edition, 2023
2	Bernd Bruegge	Object oriented software engineering	Pearson Education.	Third Edition, 2009
3	Ian Sommerville	Software Engineering	Pearson Education	Tenth edition, 2015
4	John Nicholas, Herman Steyn	Project Management for Business Engineering and Technology	Routledge	Seventh Edition, 2025
5	Bob Hughes, Mike Cotterell, Rajib Mall	Software Project Management	Tata McGraw Hill	Sixth Edition, 2016

Course Code	Name of the Course
216U01C502	Computer Networks

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	03	--	--	03
Credits Assigned	03	--	--	03

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	--	20	30	50	100

Course prerequisites: NA

Course Objectives:

1. To introduce concepts and fundamentals of data communication and computer networks.
2. To explore the inter-working of various layers of OSI.
3. To understand and apply IP addressing concepts in network design.
4. To assess the strengths and weaknesses of various routing algorithms.
5. To understand the transport layer and various application layer protocols.

Course Outcomes (CO):

At the end of successful completion of the course the student will be able to

CO1.	Explain the fundamentals of the data communication networks, reference models, topologies, physical media, devices, simulators and identify their use in day-to-day networks.
CO2.	Demonstrate Data Link Layer, MAC layer technologies & protocols and implement the functionalities like error control, flow control.
CO3.	Demonstrate various network layer protocols and network design using IP addressing, forwarding, routing concepts.
CO4.	Demonstrate Transport layer concepts like socket, flow control, error control, congestion control, QoS.
CO5.	Describe various features and operations of application layer protocols such as Telnet, HTTP, DNS, SMTP.

Module No.	Unit No.	Contents	No of Hrs.	CO
1	**Introduction to networking**		**06**	**CO 1**
	1.1	Types of Networks: LAN, WAN, MAN. Network Topology (types)		
	1.2	Network Software: Protocol hierarchy, Design Issues for layers, Connection oriented and connectionless services, Reliable and Un-reliable services		
	1.3	OSI and TCP/IP reference model, Comparison of OSI and TCP/IP reference model		
	1.4	Overview of connecting devices, NIC, Repeater, Hub, Bridge, Router, Gateway		
		# Self-Learning: Guided and Un-guided transmission media		
2	**Data Link and MAC Layer**		**12**	**CO 2**
	2.1	Error Control: Types of Errors; Redundancy, Checksum, Hamming Code and CRC.		
	2.2	Framing, and Flow Control; Flow control Protocols: Stop-and-wait, Go-Back-N, Selective-Repeat, Piggybacking		
	2.3	MAC address; Random Access: ALOHA, slotted ALOHA, Efficiency; CSMA, CSMA/CD, CSMA/CA.		
	2.4	Controlled Access, Channelization, IEEE standards, different Ethernets		
		# Self-Learning: Modular Arithmetic		
3	**Network Layer**		**10**	**CO 3**
	3.1	Network layer services, IPv4, strategies to bridge the limitations (IP sub netting, CIDR, NAT, Addressing, Options, Extension headers, Packet forwarding, Congestion Control)		
	3.2	ARP, RARP, DHCP and ICMP		
	3.3	IPV6 Addressing		
	3.4	Shortest Path routing, DV, Link state Routing. Unicast protocols: OSPF, BGP.		
	3.5	Multicast routing protocols: IGMP; Hierarchical Routing, DVMRP		

4	**Transport Layer: Protocols**		**08**	**CO 4**
	4.1	Services, Transport layer protocols, UDP, TCP: State Transition diagram, flow control, error control, TCP Timers, Queuing disciplines		
	4.2	TCP Congestion control, SCTP		
	4.3	Quality of Service		
5	**Application Protocols**		**09**	**CO 5**
	5.1	HTTP, WWW		
	5.2	DNS		
	5.3	FTP, Telnet		
	5.4	SMTP		
		# Self Learning: POP and IMAP		
		Total	45	--

Recommended Books:

Sr. No	Name/s of Author/s	Title of Book	Publisher	Edition/ Year
1	B.A.Forouzan	Data Communication and Networking	Tata McGraw Hill edition	Fourth Edition
2	A.S.Tanenbaum	Computer Networks	Pearson Education	Sixth Edition
3	B. A. Forouzan	TCP/IP Protocol Suite	Tata McGraw Hill edition	Fourth Edition
4	Prakash C Gupta	Data Communications and Computer Networks	PHI Learning	Second Edition
5	Michael W Lucas	Networking for Systems Administrators (IT Mastery)	Tilted Windmill Press	Second Edition
5	Doug Lowe	Networking All-in-One For Dummies	John Wiley & Sons (US)	Eighth Edition

*In addition to printed books, faculty can suggest (authentic) urls or e-books, e-contents etc.

Course Code	Name of the Course
216U01C503	Theory of Automata

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	3	-	1	4
Credits Assigned	3	-	1	4

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	25	20	30	50	125

Course prerequisites:-

Course Objectives:

1. To build up mathematical fundamentals required to understand the theory of computation
2. To formalize mathematical models of computation: basic machines, deterministic and non deterministic machines and pushdown machines and Turing Machines.
3. To learn the fundamentals of formal grammars and languages.
4. Develop understanding of different types of Turing machines, their use, capabilities & limitations.
5. Understand the concept of Undecidability

Course Outcomes (CO):

CO1. Describe regular languages using Regular Expressions, Finite Automata, Nondeterministic Finite Automata, Mealy Machines, Moore Machines and its applications

CO2. Write, simplify and normalize context free grammars and describe context free languages using context free grammar and push down automata

CO3. Design Turing Machines for various problems and its applications

CO4. Understand the concept of Un-decidability and Recursively Enumerable Languages

Module No.	Unit No.	Contents	No of Hrs.	CO
1	**Finite Automata**		10	CO1
	1.1	Introduction: Alphabets, String, Language, Basic Operations on language, Concatenation, Kleene Star, Chomsky hierarchy.		
	1.2	Finite. Automata (FA), acceptance of strings, and languages, DFA, Language of a DFA, Language recognizers, NFA, Language of an NFA, Conversion from NFA to DFA, Conversions and Equivalence between NFA with and without ε- transitions. Finite Automata with output- Moore and Mealy machines. Conversion from Mealy to Moore machine and Moore to Mealy machine. Minimization of FSM-Myhill-Nerode Theorem, Partition Method, Equivalence between two FSM's.		
2	**Regular Expression & Languages**		7	CO1
	2.1	FA and Regular Expressions, Conversion from RE to FA and FA to RE.		
	2.2	Closure properties of Regular languages -Closure under Boolean operations, reversal, homomorphism, inverse homomorphism, etc, Pumping lemma for Regular languages		
	2.3	Linear Grammar, Left Linear Grammar, Right Linear Grammar, Conversion of RLG to LLG and LLG to RLG.		
3	**Context Free Grammars**		8	CO2
	3.1	Context-free Grammars (CFGs), Acceptance of Language, Sentential forms, Leftmost and Rightmost derivations, Language of a CFG. Derivation tree or Parse tree, Ambiguity in Grammar and Language.		
	3.2	Simplification of CFGs -Removing useless symbols, epsilon-Productions, and unit productions. Normal forms -CNF and GNF, Pumping lemma for CFLs.		
	3.3	Closure properties of CFLs -Closure under union, concatenation, Kleene closure, substitution, Inverse homomorphism, reversal, intersection with regular set, etc.		

4	**Push Down Automata**		8	CO2
	4.1	Definition and languages of PDA, acceptance by final state and empty stack, Instantaneous descriptions (Ids),		
	4.2	Equivalence & conversion of CFG's and PDA's,		
	4.3	Types of PDA-Deterministic PDA, Non-Deterministic PDA.		
5	**Turing Machine**		8	CO3
	5.1	Turing Machines- Formal definition and behavior, Transition diagrams, Language of a TM, TM as acceptors deciders and generators. TM as a computer of integer functions. Design and Simulation of TMs,		
	5.2	Types of TMs –Deterministic Turing Machine, Multi-tape TMs, Nondeterministic TMs. TMs with semi-infinite tapes,		
6	**Un-decidability and Recursively Enumerable Languages**		4	CO4
	6.1	Recursive and Recursively Enumerable Languages. Properties of Recursive and Recursive Enumerable Languages.		
	6.2	Decidability and Undecidability, Halting Problem, Post Correspondence Problem, Context Sensitivity and Linear Bounded Automata.		
		Total	45	--

Recommended Books:

Sr. No.	Name /s of Auth or/s	Title of Book	Publisher	Edition / Year
1.	John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman	"Introduction to Automata Theory, Languages and Computation"	Pearson Education	Third Edition, 2006
2.	J.C.Martin,	"Introduction to languages and the Theory of Computation"	TMH	Fourth Edition, 2010
3.	Michael Sipser	"Theory of Computation"	Cengage Learning	Third Edition, 2012
4.	O.G.Kakde	"Theory of Computation"	Laxmi Publications	First Edition, 2007

*In addition to printed books, faculty can suggest (authentic) urls or e-books, e-contents etc.

Course Code	Name of the Course
216U01E511	Computer Graphics

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	03	–	--	03
Credits Assigned	03	–	--	03

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	--	20	30	50	100

Course prerequisites: Knowledge of C++ Programming and Basic Mathematics.

Course Objectives:

1. To equip students with the fundamental knowledge and basic technical competence in the field of Computer Graphics.
2. To emphasize the implementation aspect of Computer Graphics Algorithms.
3. To prepare the student for advance areas and professional avenues in the field of Computer Graphics

Course Outcomes (CO):

At the end of successful completion of the course the student will be able to

CO1. Understand and apply fundamental computer graphics concepts, including drawing algorithms, anti-aliasing, filled area primitives, and basic OpenGL functions.

CO2. Apply and analyze two-dimensional geometric transformations, including translation, scaling, rotation, reflection, and shear, using matrix representation and OpenGL functions.

CO3. Analyze and implement two-dimensional and three-dimensional viewing transformations, geometric transformations, curve generation, fractal geometry, and clipping operations using appropriate algorithms and OpenGL functions.

CO4. Apply visible surface detection, illumination models, and animation techniques, and explore basic VR and AR concepts using OpenGL.

Module No.	Unit No.	Contents	No of Hrs.	CO
1	Introduction and Overview of Graphics		10	CO1
	1.1	Basics of computer graphics, Application of Computer Graphics, Video Display Devices: Random Scan and Raster Scan displays		
	1.2	Line drawing algorithms(DDA, Bresenham's and Mid-Point), Circle generation algorithm (Bresenham's, Mid-point), midpoint algorithm for ellipse drawing		
	1.3	Aliasing, Antialiasing techniques like Pre and post filtering, super sampling, and pixel phasing		
	1.4	Filled Area Primitive: Scan line Polygon Fill algorithm, inside outside tests, Boundary Fill and Flood fill algorithm.		
	1.5	Introduction to Graphics software OpenGL, coordinate reference frames, specifying two-dimensional world coordinate reference frames in OpenGL, OpenGL point functions, OpenGL line functions, point attributes, line attributes, curve attributes, OpenGL point attribute functions, OpenGL line attribute Functions, OpenGL functions on fill area primitives.		
	1.6	Self-learning: Colour models- RGB, HSV, CMYK and YIQ colour models		
2	Two Dimensional Geometric Transformations		08	CO2
	2.1	Basic transformations: Translation, Scaling, Rotation, Reflection and Shear		
	2.2	Matrix representation and Homogeneous Coordinates		
	2.3	Composite transformation		
	2.4	Self-learning OpenGL functions of 2D transformation		
3	Two-Dimensional Viewing, Curves and Fractal Generation and Clipping		09	CO3
	3.1	Viewing transformation pipeline and Window to Viewport coordinate transformation		
	3.2	Bezier Curve, B-Spline Curve, Fractal-Geometry: Fractal Dimension, Koch Curve.		
	3.3	Clipping operations: Point clipping, Line clipping algorithms: Cohen-Sutherland, Liang: Barsky		
	3.4	Polygon Clipping Algorithms: Sutherland-Hodgeman, Weiler-Atherton		

	3.5	Self-learning OpenGL function for 2D viewing and clipping		
4	Three Dimensional Geometric Transformations and viewing,		09	CO3
	4.1	3D Transformations: Translation, Rotation, Scaling and Reflection		
	4.2	Composite transformations: Rotation about an arbitrary axis, affine transformations		
	4.3	3DViewing:3D viewing concepts, 3D viewing pipeline, 3D viewing coordinate parameters , Transformation from world to viewing coordinates		
	4.4	Projection transformation, orthogonal projections, perspective projections, The viewport transformation and 3D screen coordinates. OpenGL 3D viewing functions, 3D Modeling for Anatomy		
	4.5	**Self-reading:** OpenGL functions for 3D transformations		
5	Visible Surface Detection and Animation		08	CO4
	5.1	Properties of light, Illumination Models: Light sources, basic illumination models-Ambient light, diffuse reflection, specular and phong model, Corresponding OpenGL functions.		
	5.2	Visible Surface Detection: Classification of Visible Surface Detection algorithm, Back Surface detection method, Depth Buffer method, Area Subdivision method, OpenGL visibility detection functions		
	5.3	Animation: Introduction to Animation, Traditional Animation Techniques, Principles of Animation, Key framing: Character and Facial Animation, Deformation, Motion capture, Warping- Mesh Warping		
	5.4	Introduction to VR and AR: Definition, Software and Hardware configurations, Applications in Healthcare		
	5.5	Case Study: 3D Visualization of Human Organs, AR/VR Surgical Training Simulator and Real-Time Tumor Visualization in MRI		
	Self-Learning Component: Vulkan API by the Khronos group (known for OpenGL)			
		Total	45	--

Recommended Books:

Sr. No	Name/s of Author/s	Title of Book	Publisher	Edition/Year
1	Roy A. Plastock, Zhigang Xiang	Computer Graphics	McGraw-Hill Companies	Second Edition/
2	Donald Hearn & Pauline Baker	Computer Graphics with OpenGL	Pearson Education	3rd / 4th Edition, 2011
3	Edward S. Angel.	Interactive Computer Graphics, A top-down approach with shaderbased OpenGL	Pearson Education	6th Edition,2011.
4	Dave Shreiner, Graham Sellers, John Kessenich, and Bill Licea-Kane	OpenGL Programming Guide: The Official Guide to Learning OpenGL	AddisonWesley	8th Edition, 2013
5	R. K Maurya	"Computer Graphics with Virtual Reality"	Wiley India	
6	https://www.classcentral.com/course/interactivegraphics-2067			
7	https://swayam.gov.in/nd2_ntr20_ed15/preview			
8	https://nptel.ac.in/courses/106/106/106106090/			
9	https://www.edx.org/course/computer-graphics-2			

*In addition to printed books, faculty can suggest (authentic) URLs or e-books, e-contents etc.

Course Code	Name of the Course			
216U01E512	Modern Data Architectures			
Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	03	--	--	03
Credits Assigned	03	--	--	03

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	--	20	30	50	100

Course prerequisites:
Database systems

Course Objectives:
The objectives of this course is to understand design, manage data in Distributed, Parallel systems. Object Relational Databases, Active, temporal, spatial, graph, deductive databases for managing different types of data. NOSQL system types to manage big data. Building and using data warehouse for Online Analytical Processing

Course Outcomes (CO):
At the end of successful completion of the course the student will be able to

CO1. Understand, design, analyze and process data in distributed, parallel, databases understand design
CO2. Understand the concepts and design of Active, temporal, spatial, deductive databases
CO3. Understand and use NOSQL system types.
CO4. Model and Build multidimensional data warehouse and apply ETL process to populate data to data warehouse.
CO5. Perform Online Analytical Processing on the warehouse data.

Detailed Curriculum

Module No.	Unit No.	Contents	No of Hrs.	CO
1	**Modern Data Processing Architectures and Parallel, Distributed Databases**		10	CO1
	1.1	Databases, data warehouses, and data lakes, Data platform architecture at a high level, Lambda versus Kappa architecture, Lakehouse and Delta architectures, Data mesh		
	1.2	Database system architectures- centralized, client server system, parallel system, distributed system, network types.		
	1.3	Parallel databases – Introduction, I/O parallelism, query parallelism, Design of parallel systems, parallelism on multicore processor.		
	1.4	Distributed databases– types, distributed transactions, commit protocols, concurrency control, query processing. Cloud based databases		
		# Self-learning: Apache Spark		
2	**Object based , Active, temporal, spatial and deductive databases**		10	CO2
	2.1	Object based Databases– Overview complex data types, inheritance, object identity, reference types, object oriented versus object relational, Biomedical data types and challenges		
	2.2	Active, temporal and deductive databases concepts.		
	2.3	Spatial database-Introduction, Definition of GIS, Evolution of GIS, components of GIS Vector Data Model: Topology, Non topological Vector models, Attribute Data in GIS, Attribute Data Entry, Vector Data Query, Manipulation of Fields and Attribute Data Raster Data Model: Elements of Raster Data Model, Types of Raster Data, Raster Data Structure, Raster Data Query, Data Compression, Data Conversion, Integration of Raster and Vector data		
		# Self-learning: QGIS, PostGIS,		
3	**Big data Storage system and NoSQL databases**		10	CO3
	3.1	Introduction to Big Data, Big Data characteristics, types of Big Data, Traditional vs. Big Data business approach, Case Study of Big Data Solutions,Genomic and Electronic Health Record (EHR) storage formats		
	3.2	NOSQL systems, CAP theorem, NOSQL systems- document store-Mongodb Key-value Stores, Column store, Graph store - neo4j		

		#Self-learning: GeoSpark R, GraphX tools of Apache Spark		
4	**Data ware house Modeling and ETL**		10	CO4
	4.1	Data Warehouse: The Building Blocks Defining Features, characteristics of DWH Data Warehouses and Data Marts , Top-Down Versus Bottom-Up Approach, A Practical Approach ,DWH architecture , Types of Metadata		
	4.2	Principles of Dimensional Modeling Dimensional Modeling Basics, ER Modeling Versus Dimensional Modeling, STAR Schema, Snow flake schema, Fact less Fact Table, Schema Keys, Star and Snowflake Schema for Clinical Data		
	4.3	ETL Overview , ETL Requirements and Steps Data Extraction Techniques, Data Transformation and Data Loading, ETL Processes in Biomedical, Case Study: Genomics England - 100,000 Genomes Project		
5	**Data and information visualization & OLAP**		05	CO5
	5.1	Data and information visualization		
	5.2	Online Analytical Processing,Major Features and Functions, OLAP Models		
	5.3	Case Study: Google DeepMind and NHS collaboration,, MIMIC-III Clinical Data Warehouse		
		#Self-Learning:-Power BI		
		Total	45	--

 # Students should prepare all Self Learning topics on their own. Self-learning topics will enable students to gain extended knowledge of the topic. Assessment of these topics may be included in IA and Laboratory Experiments.

Reference Books*

Sr.	Name/s of	Title of Book	Publisher	Edition/

No	Author/s			Year
1	Brian Lipp	Modern Data Architectures with Python	Packt Publishing Ltd	1st Edition 2023
2	Elmasri and Navathe	"Fundamentals of Database Systems"	Pearson Education	7th Edition, 2015
3	Paulraj Ponniah	"Data Warehousing Fundamentals: A Comprehensive Guide for IT Professionals",	Wiley India	2nd Edition, 2017
4	Raghu Ramakrishnan and Johannes Gehrke	"Database Management Systems"	McGraw Hill	3rd Edition, 2018
5	Korth,Silberchatz,Sudarshan	"Database System Concepts".,	McGraw Hill	7th Edition 2019
6	Reema Thareja	Data warehousing	Oxford	1st Edition, 2009
7	Michael J. de Smith, Michael F. Goodchild and Paul A. Longley	Geospatial Analysis: A Comprehensive Guide to Principles, Techniques, and Software Tools,	Wiley	2nd Edition 2019
8	Radha Shankarmani, M. Vijayalakshmi	Big Data Analytics	Wiley	2nd Edition, January 2016
9	Open source	Neo4j documentation	https://neo4j.com/docs/	
10	Open source	GIS Manual	https://docs.qgis.org/3.34/en/docs/training_manual	

Course Code	Name of the Course
216U01E513	Soft Computing

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	03	--	--	03
Credits Assigned	03	--	--	03

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	--	20	30	50	100

Course prerequisites: A strong mathematical background in Linear Algebra, Probability and Statistics, Calculus, and Discrete Mathematics., proficiency in algorithms and programming skills is also required.

Course Objectives:

1. Understand the core concepts and application of soft computing techniques- neural networks, fuzzy logic, and genetic algorithms
2. Explore Neural Network Architectures and Functions
3. Develop Fuzzy Logic Systems
4. Introduce genetic algorithms for solving single-objective optimization problems..

Course Outcomes (CO):

CO1. Identify and describe soft computing techniques and their application
CO2. Apply basic learning rules to simple neural networks
CO3. Analyze various neural network architectures
CO4. Design Fuzzy controller system
CO5. Solving single-objective optimization problems using Genetic Algorithms

Module No.	Unit No.	Contents	No of Hrs.	CO
1		**Introduction to Soft Computing**	03	CO1
	1.1	Concept of computing systems, "Soft" computing versus		

		"Hard" computing, Characteristics of Soft computing		
	1.2	Brief descriptions of different components of soft computing including Neural networks, fuzzy logic, Some applications of Soft computing techniques.		
2	**Introduction to Artificial Neural Networks**		08	CO2
	2.1	Biological neurons and its working, Artificial Neural Network (ANN) , Comparison between Biological Neuron and Artificial Neuron		
	2.2	Basic Models of Artificial Neural Network– connection , Learning, Activation Functions Important Terminologies of ANN's, McCulloch- Pitts Neuron , Linear Separability		
	2.3	Neural network learning rules- Hebbian Learning, Perceptron learning, Delta Learning, Widrow-Hoff Learning, Correlation Learning, Winner-Take-All Learning, Outstar Learning		
	Self-Learning Topic: Early neural network architectures- ADALINE, MADALINE network			
3	**Supervised and Unsupervised Neural Network**		12	CO3
	3.1	Backpropagation networks- architecture, Training, Learning factors		
	3.2	Kohonen Self-Organizing Feature Maps- architecture, training, Adaptive Resonance Theory Network		
	3.3	Associative Memory networks- Auto-associative Memory networks, Heteroassociative Memory Network, Bidirectional associative Memory Network, Hopfield Network Applications of ANNs to solve some real-life problems.		
	Self-Learning Topic:, Boltzman Machine, Cognitron Network, Neocognitron Network			
4	**Fuzzy Logic**		12	CO4
	4.1	Introduction to Fuzzy logic, Fuzzy sets and membership functions, Operations on Fuzzy sets, Fuzzy relations, rules, propositions, implications and inferences, Defuzzification techniques		
	4.2	Fuzzy logic controller design, Some applications of Fuzzy logic.		

	Self-Learning Topic: Neuro Fuzzy system			
5	**Genetic algorithm**		10	CO5
	5.1	Introduction, Biological background, Tradition optimization and search techniques, Genetic Algorithm (GA) and search space, basic terminologies, operators in GA – Encoding, Selection, Crossover, Mutation Stopping condition for GA flow		
	5.2	Constraints in GA, Problem solving using GA		
		Total	45	--

Recommended Books:

Sr. No	Name/s of Author/s	Title of Book	Publisher	Edition/Year
1.	S.N. Sivanandam, S. N. Deepa	*Principles of Soft Computing*	Wiley	3rd Edition 2019
2.	S. Rajasekaran, and G. A. Vijayalakshmi Pai	*Neural Networks, Fuzzy Logis and Genetic Algorithms : Synthesis, and Applications*	Prentice Hall of India	2007
3.	Timothy J. Ross	*Fuzzy Logic with Engineering Applications*	Wiley	3rd edition 2011
4.	David E. Goldberg	*Genetic Algorithms In Search, Optimization And Machine Learning*	Pearson Education	2002
5.	Simon Haykin	*Neural Networks and Learning Machines*	Pearson Education	2009
6.	Jacek. N. Zurada	Introduction to Artificial Neural Network	Jaico Publishing House	13th Edition 2016
7.	J.-S. R. Jang, C.-T. Sun, and E. Mizutani	Neuro-Fuzzy and soft Computing	Pearson Education ,	2015

Course Code	Name of the Course
216U01E514	Quantum Computing

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	03	--	--	03
Credits Assigned	03	--	--	03

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	--	20	30	50	100

Course prerequisites: Knowledge of Programming and Basic Mathematics - Linear Algebra, Calculus, Probability and Statistics, Matrices.

Course Objectives:

1. Introduce students to the fundamental concepts and principles of quantum computing.
2. Equip students with the mathematical tools necessary for understanding and working with quantum systems.
3. Explore various quantum algorithms and their applications in different domains.
4. Provide practical experience through self-learning exercises and code samples.

Course Outcomes (CO):

At the end of successful completion of the course the student will be able to

CO1. Understand the basic principles of quantum mechanics and their application to computing.
CO2. Apply mathematical tools to analyze and manipulate quantum systems.
CO3. Explore various quantum algorithms and their applications.
CO4. Develop practical skills in quantum computing.

Module No.	Unit No.	Contents	No of Hrs.	CO
1	Introduction to Quantum Computing		10	CO 1
	1.1	Quantum Solutions overview: Introduction, Real-Life Problems and Solution, Solution Benefits. Solutions: Cryptography, Optimization, Cybersecurity.		
	1.2	Mathematics Behind Quantum Computing: Introduction, Initial Setup, Quantum Operators, Sets, Vectors, Matrices, Tensors		
	1.3	Self-learning: (Code Sample, Command for Execution, Output).		
2	Quantum Computing Basics		08	CO 1 CO 2
	2.1	Quantum Subsystems and Properties: Introduction, Initial Setup, Single Qubit System, Multiple Qubit System, Quantum States, Quantum Protocols, Quantum Operations. Quantum Transformations: Kronecker Transformation, Measure Gate Transformation.		
	2.2	Quantum Information Processing Framework: Introduction, Quantum Circuits, Quantum Communication, Quantum Noise, Quantum Error Correction, Limitations of Quantum Computing.		
	2.3	Quantum Algorithms: Deutsch–Jozsa, Simon's Algorithm, Shor's Algorithm, Grover's Algorithm. Quantum Subroutines.		
	2.4	#Self-Learning: Quantum Simulators		
3	Quantum Optimization Algorithms		09	CO 3
	3.1	Introduction, Approximate Optimization Algorithms, Semidefinite Programming.		
	3.2	Transportation Planning: Factors, Combinatorial Optimization, Quantum NP (BQNP)		
	3.3	Quantum Algorithms: Quantum Least Squares Fitting, Quantum Sort, Quantum Eigen Solvers, Quantum Semidefinite Programming.		
4	Quantum Neural Network Algorithms		09	CO 3
	4.1	Quantum ANN, Classical Neural Networks, Quantum Analogies to Neural Networks		
	4.2	Quantum Associative Memory, Quantum Dots, Quantum Random Access Memory		
	4.3	**Self-reading:** Quantum GAN		

5	Quantum Classification Algorithms		08	CO 3 CO 4
	5.1	Classifiers, Quantum Classifiers, Variational Quantum Classifier, Classical SVM, Quantum Sparse Support Vector Machines.		
	5.2	Quantum Data Processing: Classical K-Means, Quantum K-Means, Classic K-Medians, Quantum K-Medians, Classical Clustering, Quantum Clustering, Quantum Manifold Embedding.		
		Total	45	--

Recommended Books:

Sr. No	Name/s of Author/s	Title of Book	Publisher	Edition/Year
1	Bhagvan Kommadi	Quantum Computing Solutions	Apress	1st edition, Dec-2020
2	Dr. Bob Sutor	Dancing with Qubits	Packt Publishing	Second Edition 2024
3	Robert Hundt	Quantum Computing for Programmers	Cambridge University Press	First Edition 2022
4	Tutorials By IBM https://www.ibm.com/quantum/qiskit#performance https://developer.ibm.com/depmodels/quantum-computing/tutorials/ https://learning.quantum.ibm.com/ https://research.ibm.com/topics/quantum-machine-learning			
5	https://thequantumlaend.de/tutorials/			
6	https://pythonprogramming.net/quantum-computer-programming-tutorial/			

*In addition to printed books, faculty can suggest (authentic) URLs or e-books, e-contents etc.

Course Code	Name of the Course
216U01E515	Software Testing & Quality Assurance

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	03	--	--	03
Credits Assigned	03	--	--	03

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	--	20	30	50	100

Course prerequisites (if any):

Background of Software Engineering, Programming Concepts, and Algorithms.

Course Objectives:

The objective of this course is to impart understanding of techniques for software testing and quality assurance. To help students to develop skills that will enable them to construct software of high quality - software that is reliable, and that is reasonably easy to understand, modify and maintain.

Course Outcomes:

At the end of successful completion of the course the student will be able to

CO1	Explore the fundamentals of testing.
CO2	Describe the various levels of testing and their use in designing of various test cases.
CO3	Model various test cases for real life applications.
CO4	Outline software quality concepts
CO5	Identify software quality assurance goals and standards.

Module No.	Unit No.	Details	Hrs.	CO
1	**Fundamentals of Testing**		06	CO 1
	1.1	Human and errors, Testing and Debugging, Software Quality, Requirement Behavior and Correctness, Fundamentals of Test Process, Psychology of Testing, General Principles of Testing, The Tester's Role in a Software Development Organization, Origins of Defects, Defect Classes, The Defect Repository and Test Design.		
		# Self Learning - Defect Examples		
2	**Levels of Testing**		10	CO 2
	2.1	The Need for Levels of Testing, Unit Test, Unit Test Planning, Designing the Unit Tests. The Class as a Testable Unit, The Test Harness, Running the Unit tests and Recording results, Integration tests, Designing Integration Tests, Integration Test Planning, System Test – The Different Types, Regression Testing, Alpha, Beta and Acceptance Tests.		
		#Self-Learning -Junit Tool		
3	**Test Case Design and Implementation:**		10	CO 3
	3.1	Introduction to Testing Design Strategies, Test Case Design Strategies, Using Black Box Approach to Test Case Design, Random Testing, Equivalence Class Partitioning, Boundary Value Analysis, , Using White-Box Approach to Test design, Coverage and Control Flow Graphs, Covering Code Logic, Additional White Box Test Design, Contrast testing, stress testing, load testing, performance testing.		
		#Self Learning – Other Black box & Whitebox Test Design Approaches		
4	**Quality Assurance**		08	CO 4
	4.1	Introduction The Software Quality Challenge., What is Software Quality? Software Quality Factors The Components of the Software Quality Assurance System - Overview.		
	4.2	Pre-Project Software Quality Components Contract Review Development and Quality Plans		

	4.3	SQA Components in the Project Life Cycle Integrating Quality Activities in the Project Life Cycle Reviews. Software Testing – Strategies Software Testing Implementation		
		Assuring The Quality of Software Maintenance. Assuring The Quality of External Participants Parts Case Tools and their Effect on Software Quality.		
5	**Software Quality Assurance**		**09**	**CO 5**
	5.1	Software Quality Infrastructure Components Procedures and Work Instructions. Supporting Quality Devices Staff Training, Instructing and Certification. Preventive and Corrective Actions. Configuration Management Documentation and Quality Records Controls.		
	5.2	Software Quality Management Components Project Progress Control Software Quality Metrics Software Quality Costs		
	5.3	Standards, Certification and Assessment SQA Standards ISO 9001 Certification Software Process Assessment		
		Total	**48**	

Recommended books:

Sr. No	Name/s of Author/s	Title of Book	Name of Publisher with country	Edition and Year of Publication
1	Kshirsagar Naik, Priyadarshi Tripathy	Software Testing & Quality Assurance	Wiley , India	1st Edition 2016
2	Naresh Chauhan	Software Testing Principles & Practices	Oxford University Press	2nd Edition,2016
3	Daniel Galin	Software Quality Assurance: From Theory to Implementation	Pearson Publishers	1e Paperback,1 January 2008
4	William E. Lewis	Software Testing and Continuous Quality Improvement	CRC press	Third Edition 2014

Course Code	Name of the Course
216U01E516	Applied Cryptography

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	03	--	--	03
Credits Assigned	03	--	--	03

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	--	20	30	50	100

Course pre-requisites (if any):
Some mathematical maturity, in terms of understanding and working with mathematical definitions, concepts, and proofs, and elementary notions of logic, set theory, number theory, probability and statistics

Course Objectives:
In the era of Digital Computers and internet ensuring confidentiality, authentication, integrity of data during communication is very critical. This course impart students the knowledge of cryptographic algorithms and techniques to achieve same. It also introduces students to the advances in the area of cryptography
.

Course Outcomes (CO):

At the end of successful completion of the course the student will be able to:

CO1: Discuss fundamentals of cryptography and classical attacks on security goals
CO2: Implement various Cryptographic arithmetic algorithms for securing systems
CO3: Analyze and Implement Symmetric and Symmetric Key Cryptography Algorithms
CO4: Understand Authentication Mechanisms and Evaluate Cryptographic Hash Functions
CO5: Explore and investigate advances in the field of cryptography

Module No.	Unit No.	Details	Hrs.	CO
1	**Introduction to Information Security & Cryptography**		**07**	**CO 1**
	1.1	Information Security and its goals, Vulnerability Threats and Attacks, Security services and security mechanisms		
	1.2	Encryption and Decryption, Symmetric and Asymmetric Key Cryptography, Types of keys, Cryptanalysis methods		
	1.3	Classical attacks on security and counter measures: Eavesdropping, Traffic Analysis attack, Replay attack, non-repudiation attack, Man-in-the-Middle attack, Data Tampering, Denial of Service (DoS) attack, Brute Force Attack, zero day exploit attack, Phishing and social engineering, Spoofing, Malware, session hijacking attack,		
		#Self Learning: Cryptanalysis of substitution ciphers, The Adventure of the Dancing Men - Short story by Sir Arthur Conan Doyle		
2	**Cryptographic Arithmetic and Key management**		**08**	
	2.1	Cryptographic Arithmetic: Modular arithmetic, additive and multiplicative inverse, set of residues, Extended Euclidean Algorithm, fast exponentiation, Chinese Remainder Theorem (CRT)		
	2.2	Mathematics for Asymmetric key cryptography: Prime generation, primality testing, prime factorization, Euler Totient function,		
	2.3	**Key management:** Generating Keys, Nonlinear Keyspaces, Transferring Keys, Verifying Keys, Using Keys, Updating Keys, Storing Keys, Backup Keys, Compromised Keys, Lifetime of Keys, Destroying Keys, Public-Key Key Management		
	2.4	**Key exchange algorithm**: Diffie Hellman Key exchange, Man-in Middle attack		
		#Self Learning : Key Management concepts, Key Exchange Algorithms: Shamir's Three-Pass Protocol, Conference Key Distribution and Secret Broadcasting		
3	**Symmetric Key Cryptography**		**08**	**CO2**
	3.1	Building blocks of modern and classical Block Ciphers: P box, S Box, EX-OR operations, circular shifts, swaps, split and combine, Rounds, Initialization vectors, Confusion, Diffusion, Fiestel Ciphers, Non-Fiestel ciphers		

	3.2	Introduction to classical block ciphers as means of confusion and diffusion techniques : Initialization vector, Electronic code book[ECB], Cipher feedback[CFB], cipher block chaining[CBC], output feedback[OFB], propagating cipher block chaining[PCBC], counter[CTR]		
	3.3	**DES**: DES Structure, DES Analysis: Properties, Design Criteria, DES Strength and Weaknesses, DES Security, Multiple DES, 3DES, AES vs DES		
		#Self Learning –RC5, Classical Block Cipher Modes		
4	**Asymmetric Key Cryptography**		**07**	**CO2**
	4.1	**Public key cryptography:** Principles of public key cryptosystems, The RSA algorithm, attacks on RSA, Cryptanalysis of RSA		
	4.2	Elliptic Curve Cryptosystems, ECC as discrete logarithmic problem, ECC encryption- decryption process, Elliptic Curve – Diffie Hellman Key exchange algorithm, Elliptic Curve Digital Signature Algorithm		
		#Self Learning : Rabin Cryptosystem		
5	**Message Authentication and Digital Signatures**		**07**	**CO3**
	5.1	Overview of Authentication mechanisms: Biometrics, challenge response systems, one time pads, passwords, multi-factor authentication, token based authentication, single sign-on, Kerberos*, PKI, CAPTCHA Using Symmetric and Asymmetric Encryption for : Authentication, confidentiality, non-repudiation		
	5.2	Hash : Cryptographic Hash Function, Hash Function Requirements, Hash function attacks, Birthday Paradox SHA-512, HMAC		
	5.3	PKI: Roles - responsibilities of Certification Authority and Registration Authority, Applications of PKI, Digital certificates. Using Public Key for Authentication, Digital Signatures, Authentication Applications: X.509 Authentication Service, Kerberos		
		#Self Learning : RSA and Schnorr Digital Signature MD5 for non cryptographic applications, Challenge Handshake Authentication Protocol (CHAP), Extensible Authentication Protocol (EAP)		
6	**Introduction to Advances in Cryptography**		**08**	**CO4**
	6.1	Quantum Cryptography, Quantum key distribution-QKD		
	6.2	Homomorphic Encryption		

	6.3	Secure Multi-Party Computation (MPC), Zero-Knowledge Proofs		
	6.4	Cryptographic Obfuscation		
	6.5	Post-Quantum Cryptography		
	6.6	Blockchain and Cryptography		
	6.7	Privacy-Preserving Cryptographic Techniques		
		Total	**45**	

 # Students should prepare all Self Learning topics on their own. Self-learning topics will enable students to gain extended knowledge of the topic. Assessment of these topics may be included in IA and Laboratory Experiments.

***- Introduction of the concept, details in 5.3**

Recommended Books:

Sr. No.	Name/s of Author/s	Title of Book	Name of Publisher with country	Edition and Year of Publication
1.	Behrouz A. Forouzan	Cryptography and Network Security	Mc Graw Hill	3rd Edition, 2017
2.	William Stallings	Computer Security Principles and Practice	Pearson Education	2016. 5th Edition
3.	Bruce Schneier	Applied Cryptography	Wiley	2015, Second Edition
4.	Mark stamp	Information Security Principal and Practice	Wiley	2008, 3rd Edition
5.	Jaydip Sen	Theory and practice of cryptography and network security protocols and technologies	Intech Publishers, Croatia, Europe	2013. First Edition
6.	Oded Goldreich	Foundations of Cryptography – A Primer	Foundations and Trends® in Theoretical Computer Science: Vol. 1: No. 1, pp 1-116	2005

Course Code	Name of the Course

216U01E517	Data Mining and Business Intelligence

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	03	--	--	03
Credits Assigned	03	--	--	03

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	--	20	30	50	100

Course pre-requisites: Basic understanding of Data Structures and Algorithms. Familiarity with Databases and SQL and basic knowledge of Statistics and Probability.

Course Objectives: This course introduces the essential concepts of Data Mining (DM) and Business Intelligence (BI), covering the techniques, algorithms, and real-world applications that help organizations make data-driven decisions. Students will explore data mining techniques such as classification, clustering, and association rule mining and learn how these techniques are integrated into BI systems to enhance business decisions.

Course Outcomes (CO):

1. Understand key concepts in Data Mining and Business Intelligence.
2. Apply data pre-processing techniques to prepare data for analysis using different Power BI tools.
3. Analyze and evaluate classification and prediction models using ML algorithms and measure performance using different metrics.
4. Apply clustering techniques and association rule mining to discover patterns in datasets for various tasks.
5. Evaluate and apply Business Intelligence tools for data visualization, dashboard creation, and real-time analytics in business domains.

Detailed Curriculum

Module No.	Unit No.	Contents	No of Hrs.	CO
1	**Introduction to Data Mining and Business Intelligence**		09	CO1
	1.1	What is Data Mining? Knowledge Discovery in Databases (KDD), types of data to be mined, Data Mining techniques (Classification, Clustering, Association Rule Mining) and applications and challenges in Data Mining.		
	1.2	What is Business Intelligence? Business Intelligence Architectures, Decision Support Systems (DSS), ETL Processes (Extract, Transform, Load), Data Warehousing and OLAP.		
2	**Data Exploration and Pre-processing**		09	CO2
	2.1	Data Types and Descriptive Statistics: Nominal, Ordinal, Interval, Measures of Central Tendency (Mean, Median, Mode), Variance, and Standard Deviation,		
	2.2	Data Pre-processing Techniques: Data Cleaning, Integration, and Normalization, handling missing data and outliers, data transformation (Binning, Discretization) and Concept Hierarchy Generation, Normalization and feature extraction from genomic or imaging data.		
	2.3	Data Visualization: Techniques for visualizing distributions and relationships. Tools for Data Visualization (Power BI, Tableau, Matplotlib.)		
3	**Classification and Prediction Techniques**		09	CO3
	3.1	Supervised Learning Overview: Classification vs. Prediction tasks. Introduction to Decision Trees, Naïve Bayes, and Logistic Regression, Random forest and ensemble methods		
	3.2	Evaluation of Classification Models: Accuracy, Precision, Recall, F1-Score. Confusion Matrix, ROC Curve, and AUC		
4	**Clustering and Association Rule Mining**		09	CO4
	4.1	Clustering Techniques: K-Means and K-Medoids, Hierarchical Clustering (Agglomerative)		
	4.2	Association Rule Mining: Market Basket Analysis and Apriori Algorithm. Frequent Itemsets, Closed Itemsets, and Association Rules. Lift, Support, and Confidence for rule evaluation		
	4.3	Applications of Clustering and Association Mining: Customer Segmentation, Market Basket Analysis, Fraud Detection, Time-series mining for wearable/ICU data		
5	**Business Intelligence Applications**		09	CO5
	5.1	Applications of Business Intelligence: BI in Healthcare,		

		Retail, Banking, and Telecommunications. Market Segmentation, Customer Relationship Management (CRM),		
	5.2	Ethical Issues in Data Mining and BI		
		Total	45	--

Recommended Books:

Sr. No.	Name/s of Author/s	Title of Book	Publisher	Edition/Year
1.	Galit Shmueli, Nitin Patel, Peter Bruce	Data mining For Business intelligence	Wiley Student Edition	
2.	Han, Kamber	Data Mining Concepts and Techniques	Elsevier	2nd edition
3.	Alex berson & Stephen J Smith	Data Warehousing, Data Mining & OLAP	Tata McGraw Hill	
4.	M.H. Dunham	Data Mining Introductory and Advanced Topics	Pearson Education	
5.	Rajiv Sabherwal, Irma Becerra-Fernandez	Business Intelligence: Practices, Technologies and Management	Wiley	1 edition

Course Code	Name of the Course
216U01L501	Software Engineering Lab

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	--	02	--	02
Credits Assigned	--	01	--	01

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	50	--	--	--	50

Course prerequisites (if any):

-

Course Objectives:

The Course focuses at developing an understanding of software process models such as the waterfall and evolutionary models. It further provides an understanding of software requirements and the SRS documents. The course aims at enabling the students to prepare the system design and test cases for proper testing of the software.

Course Outcomes:

At the end of successful completion of the course the student will be able to

CO1	Understand the software development process and Estimate different types of resources for the given project.
CO2	Analyze the software requirements and Model the defined problem with the help of UML diagram.
CO3	Prepare the System Design and Model
CO4	Identify and manage configuration items and risks for the software
CO5	Test the given software for different test cases with proper test planning.

Laboratory will consist of experiments/ tutorials covering entire syllabus of the course 216U01C501, ‘Software Engineering’. Students will be graded based on continuous assessment of laboratory work.

Course Code	Name of the Course
216U01L502	Computer Networks Lab

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	--	02	--	02
Credits Assigned	--	01	--	01

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	50	--	--	--	50

Course prerequisites: NA

Course Objectives:

1. To introduce concepts and fundamentals of data communication and computer networks.
2. To explore the inter-working of various layers of OSI.
3. To understand and apply IP addressing concepts in network design.
4. To assess the strengths and weaknesses of various routing algorithms.
5. To understand the transport layer and various application layer protocols.

Course Outcomes (CO):

At the end of successful completion of the course the student will be able to

CO1	Explain the fundamentals of the data communication networks, reference models, topologies, physical media, devices, simulators and identify their use in day-to-day networks.
CO2	Demonstrate Data Link Layer, MAC layer technologies & protocols and implement the functionalities like error control, flow control.
CO3	Demonstrate various network layer protocols and network design using IP addressing, forwarding, routing concepts.
CO4	Demonstrate Transport layer concepts like socket, flow control, error control, congestion control, QoS.
CO5	Describe various features and operations of application layer protocols such as Telnet, HTTP, DNS, SMTP.

Laboratory will consist of experiments/ tutorials covering entire syllabus of the course 216U01C502, ‘Computer Networks’. Students will be graded based on continuous assessment of laboratory work.

Course Code	Name of the Course
216U01L511	Computer Graphics Lab

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	--	02	--	02
Credits Assigned	--	01	--	01

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	50	--	--	--	50

Course prerequisites: Knowledge of C++ Programming and Basic Mathematics.

Course Objectives:

1. To equip students with the fundamental knowledge and basic technical competence in the field of Computer Graphics.
2. To emphasize the implementation aspect of Computer Graphics Algorithms.
3. To prepare the student for advance areas and professional avenues in the field of Computer Graphics

Course Outcomes (CO):

At the end of successful completion of the course the student will be able to

CO1. Understand and apply fundamental computer graphics concepts, including drawing algorithms, anti-aliasing, filled area primitives, and basic OpenGL functions.

CO2. Apply and analyze two-dimensional geometric transformations, including translation, scaling, rotation, reflection, and shear, using matrix representation and OpenGL functions.

CO3. Analyze and implement two-dimensional and three-dimensional viewing transformations, geometric transformations, curve generation, fractal geometry, and clipping operations using appropriate algorithms and OpenGL functions.

CO4. Apply visible surface detection, illumination models, and animation techniques, and explore basic VR and AR concepts using OpenGL.

Laboratory will consist of experiments/ tutorials covering entire syllabus of the course 216U01E511, 'Computer Graphics'. Students will be graded based on continuous assessment of laboratory work.

Course Code	Name of the Course
216U01L512	Modern Data Architectures Lab

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	--	02	--	02
Credits Assigned	--	01	--	01

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	50	--	--	--	50

Course prerequisites:
Database systems

Course Objectives:
The objectives of this course is to understand design, manage data in Distributed, Parallel systems. Object Relational Databases,Active, temporal, spatial, graph, deductive databases for managing different types of data . NOSQL system types to manage big data. Building and using data warehouse for Online Analytical Processing

Course Outcomes (CO):
At the end of successful completion of the course the student will be able to

CO1. Understand, design, analyze and process data in distributed, parallel, databases understand design
CO2. Understand the concepts and design of Active, temporal, spatial, deductive databases
CO3. Understand and use NOSQL system types.
CO4. Model and Build multidimensional data warehouse and apply ETL process to populate data to data warehouse.
CO5. Perform Online Analytical Processing on the warehouse data.

Laboratory will consist of experiments/ tutorials covering entire syllabus of the course 216U01E512, 'Advance Database and Data Warehousing'. Students will be graded based on continuous assessment of laboratory work.

Course Code	Name of the Course

216U01L513	Soft Computing Lab

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	--	02	--	02
Credits Assigned	--	01	--	01

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	50	--	--	--	50

Course prerequisites: A strong mathematical background in Linear Algebra, Probability and Statistics, Calculus, and Discrete Mathematics., proficiency in algorithms and programming skills is also required.

Course Objectives:

1. Understand the core concepts and application of soft computing techniques- neural networks, fuzzy logic, and genetic algorithms
2. Explore Neural Network Architectures and Functions
3. Develop Fuzzy Logic Systems
4. Introduce genetic algorithms for solving single-objective optimization problems..

Course Outcomes (CO):

CO1. Identify and describe soft computing techniques and their application
CO2. Apply basic learning rules to simple neural networks
CO3. Analyze various neural network architectures
CO4. Design Fuzzy controller system
CO5. Solving single-objective optimization problems using Genetic Algorithms

Laboratory will consist of experiments/ tutorials covering entire syllabus of the course 216U01C513, 'Soft Computing'. Students will be graded based on continuous assessment of laboratory work.

Course Code	Name of the Course
216U01L514	Quantum Computing Lab

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	--	02	--	02
Credits Assigned	--	01	--	01

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	50	--	--	--	50

Course prerequisites: Knowledge of Programming and Basic Mathematics - Linear Algebra, Calculus, Probability and Statistics, Matrices.

Course Objectives:

1. Introduce students to the fundamental concepts and principles of quantum computing.
2. Equip students with the mathematical tools necessary for understanding and working with quantum systems.
3. Explore various quantum algorithms and their applications in different domains.
4. Provide practical experience through self-learning exercises and code samples.

Course Outcomes (CO):

At the end of successful completion of the course the student will be able to

CO1. Understand the basic principles of quantum mechanics and their application to computing.
CO2. Apply mathematical tools to analyze and manipulate quantum systems.
CO3. Explore various quantum algorithms and their applications.
CO4. Develop practical skills in quantum computing.

Laboratory will consist of experiments/ tutorials covering entire syllabus of the course 216U01E514, 'Quantum Computing'. Students will be graded based on continuous assessment of laboratory work.

Course Code	Name of the Course
216U01L515	Software Testing & Quality Assurance Lab

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	--	02	--	02
Credits Assigned	--	01	--	01

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	50	--	--	--	50

Course prerequisites (if any):

Background of Software Engineering, Programming Concepts, and Algorithms.

Course Objectives:

The objective of this course is to impart understanding of techniques for software testing and quality assurance. To help students to develop skills that will enable them to construct software of high quality - software that is reliable, and that is reasonably easy to understand, modify and maintain.

Course Outcomes:

At the end of successful completion of the course the student will be able to

CO1	Explore the fundamentals of testing.
CO2	Describe the various levels of testing and their use in designing of various test cases.
CO3	Model various test cases for real life applications.
CO4	Outline software quality concepts
CO5	Identify software quality assurance goals and standards.

Laboratory will consist of experiments/ tutorials covering entire syllabus of the course 216U01E515, 'Software Testing & Quality Assurance'. Students will be graded based on continuous assessment of laboratory work.

Course Code	Name of the Course

216U01L516	Applied Cryptography Lab

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	--	02	--	02
Credits Assigned	--	01	--	01

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	50	--	--	--	50

Course pre-requisites (if any):

Some mathematical maturity, in terms of understanding and working with mathematical definitions, concepts, and proofs, and elementary notions of logic, set theory, number theory, probability and statistics

Course Objectives:

In the era of Digital Computers and internet ensuring confidentiality, authentication, integrity of data during communication is very critical. This course impart students the knowledge of cryptographic algorithms and techniques to achieve same. It also introduces students to the advances in the area of cryptography

.

Course Outcomes (CO):

At the end of successful completion of the course the student will be able to:

CO1: Discuss fundamentals of cryptography and classical attacks on security goals
CO2: Implement various Cryptographic arithmetic algorithms for securing systems
CO3: Analyze and Implement Symmetric and Symmetric Key Cryptography Algorithms
CO4: Understand Authentication Mechanisms and Evaluate Cryptographic Hash Functions
CO5: Explore and investigate advances in the field of cryptography

Laboratory will consist of experiments/ tutorials covering entire syllabus of the course 216U01E516, 'Applied Cryptography'. Students will be graded based on continuous assessment of laboratory work.

Course Code	Name of the Course
216U01L517	Data Mining and Business Intelligence Lab

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	--	02	--	02
Credits Assigned	--	01	--	01

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	50	--	--	--	50

Course pre-requisites (if any):
Basic understanding of Data Structures and Algorithms. Familiarity with Databases and SQL and basic knowledge of Statistics and Probability.

Course Objectives:
This course introduces the essential concepts of Data Mining (DM) and Business Intelligence (BI), covering the techniques, algorithms, and real-world applications that help organizations make data-driven decisions. Students will explore data mining techniques such as classification, clustering, and association rule mining and learn how these techniques are integrated into BI systems to enhance business decisions.

Course Outcomes (CO):
At the end of successful completion of the course the student will be able to:
CO1: Discuss fundamentals of cryptography and classical attacks on security goalsUnderstand key concepts in Data Mining and Business Intelligence.
CO2: Apply data pre-processing techniques to prepare data for analysis using different Power BI tools.
CO3: Analyze and evaluate classification and prediction models using ML algorithms and measure performance using different metrics.
CO4: Apply clustering techniques and association rule mining to discover patterns in datasets for various tasks.
CO5: Evaluate and apply Business Intelligence tools for data visualization, dashboard creation, and real-time analytics in business domains.

Laboratory will consist of experiments/ tutorials covering entire syllabus of the course 216U01E517, 'Data Mining and Business Intelligence'. Students will be graded based on continuous assessment of laboratory work.

Course Code	Name of the Course
216U01L504	Full Stack Development Lab- JAVA

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	--	02	01	03
Credits Assigned	--	01	01	02

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		TW	Total
		IA	ISE		
	50	---	---	--	50

Course prerequisites:
Basics of object-oriented programming

Course Objectives:
Objective of this course is to provide the ability to design console based, GUI based and web based applications. Students will also be able to understand integrated development environment to create, debug and run multi-tier and enterprise-level applications

Course Outcomes:
On completion of the course students will be expected to:

CO1	Understand and implement front-end development concepts using React and JavaScript.
CO2	Develop and manage RESTful APIs using Java and Spring Boot.
CO3	Integrate front-end and back-end components using APIs and state management techniques.
CO4	Design, develop, and query relational and NoSQL databases for full-stack applications.
CO5	Implement authentication, authorization, and security best practices in web applications.
CO6	Deploy full-stack applications using cloud services, containerization, and CI/CD pipelines

Detailed Curriculum

Module No.	Unit No.	Contents	No of Hrs.	CO
1		**Introduction to Full-Stack Development**	5	1
	1.1	Overview of full-stack development Technology stack for full-stack development (React, Java, Spring Boot, Databases) Client-server architecture and REST API principles		
	1.2	**Introduction to Version Control system (Git/GitHub).** Overview of Version Control systems Setting up a basicGit server, Hosted Git Server, Branching and merging in Git, Git Server implementation, Working with Remote repository.		
2		**Frontend Development**	8	2
	2.1	**Basics of React** • Introduction to React and its ecosystem • Understanding JSX and Virtual DOM • React components: Functional vs. Class components • State and props in React • Event handling and conditional rendering		
	2.2	**Advanced React:** • React Hooks (useState, useEffect, useContext, useReducer) • React Router for navigation • API calls using Axios and Fetch API • State management using Context API and Redux • Performance optimization (React.memo, lazy loading, suspense) Form handling and validation		
3		**Backend Development with Java and Spring Boot**	11	3
	3.1	**Revision: Core Java Concepts**. Object-Oriented Programming (OOP) in Java,Exception handling and file handling, Multithreading and concurrency		
	3.2	**Spring Boot for Web Development:** Introduction to Spring Framework and Spring Boot Spring Boot project structure, Dependency Injection (DI) and Inversion of Control (IoC), Building RESTful APIs with Spring Boot, Spring Data JPA for database interaction		
4		**Database Management**	10	4
	4.1	· Relational databases (MySQL, PostgreSQL): · Introduction to NoSQL databases (MongoDB) · Writing SQL queries (joins, transactions, indexing)		

		· Spring Boot integration with databases.		
	4.2	**Authentication and Security** • JWT authentication and session management • Spring Security for role-based access control • OAuth 2.0 implementation.		
5	**Full-Stack Integration**		10	5
	5.1	**Connecting Frontend and Backend** • Making API calls from React to Spring Boot • Handling authentication and user sessions. • State management for API data in React. • GraphQL with React and Java • Integrating with external APIs (Payment Gateway, Maps, etc.).		
	5.2	**Local Deployment Setup Overview:** • **Database Server** → Stores application data (MySQL, PostgreSQL, MongoDB, etc.) • **UI Server** → The front-end interface (React, Angular, or JSP/Servlets) • **Backend (Application Server)** → Java-based API (Spring Boot, Java EE, etc.)		
	5.2	• **Swagger: Introduction to** Swagger **and** OpenAPI Specification (OAS), Understanding Swagger UI, Swagger Editor, and Swagger Codegen. Installation and setup of Swagger tools. Setting up Swagger with Spring Boot, Adding Swagger dependencies (Maven/Gradle), Configuring application. Properties for Swagger UI,		
Total			45	--

Reference Books*

Sr. No	Name/s of Author/s	Title of Book	Publisher	Edition/Year
1	Juha Hinkula	Full Stack Development with Spring Boot and React	Packt Publishing	2nd Edition, 2022
2	Alex Banks, Eve Porcello	Learning React: Modern Patterns for Developing React Apps	O'Reilly Media	3rd Edition, 2023
3	Stoyan Stefanov	React Up & Running: Building Web Applications	O'Reilly Media	2nd Edition, 2021
4	Cay S. Horstmann	Core Java Volume I–Fundamentals	Prentice Hall	12th Edition, December 2021
5	Gene Kim, Jez Humble	The DevOps Handbook	IT Revolution Press	2nd Edition, 2021

*In addition to printed books, faculty can suggest (authentic) urls or e-books, e-contents etc.

Course Code	Name of Course				
216U01L504	Full Stack Development Lab-Python				
Teaching Scheme (Hrs./Week)	TH	P	TUT	Total	
	--	2	1	3	
Credits Assigned	--	1	1	2	
Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	50	-	-	-	50

Course pre-requisites:
Basics of HTML,CSS, JavaScript and Concept of Database, Basics of Python.

Course Objectives:
This course is tailored to develop proficiency in Python programming and the creation of real-world web applications using Django. Students will gain both foundational and advanced knowledge, including writing Python scripts, interacting with databases, and creating and managing views, templates, forms, and models. The course also covers building and integrating REST APIs in Django, preparing students to apply these skills in professional engineering contexts.

Course Outcomes (CO):

CO1 – Understand integrating JavaScript with Django to enhance the frontend, configure the Django project, and manage version control using GitHub.

CO2 – Develop web applications using Django's Model-View-Template (MVT) paradigm while leveraging GitHub for collaboration, branching, and version management.

CO3 – Implement CRUD operations using Django's migrations, generate dynamic reports and visualizations with Python's CSV module, and maintain project repositories using GitHub.

CO4 – Configure authentication strategies in Django, customize the admin interface for application-specific needs, and develop a secure database-integrated admin panel with GitHub for code tracking and deployment.

CO5 – Build RESTful APIs, manage static and media files, and apply best practices for testing, CI/CD, and deployment using GitHub Actions in the Django stack.

Detailed Curriculum

Module No.	Unit No.	Contents	No of Hrs.	CO
1	**Introduction to Django**		07	1
	1.1	Overview of Django and Its Features,Setting Up the Development Environment (Python, Pip, Virtual Environments),Installing Django: An Overview of Versions and Dependencies, Creating the First Django Project, Understanding Django's Project Structure, Running the Development Server		
	1.2	**Using a Frontend JavaScript Library with Django -** JavaScript frameworks, JSX – a JavaScript syntax extension, The verbatim template tag		
	1.3	**Introduction to Version Control system (Git/GitHub).** Overview of Version Control systems Setting up a basic Git server, Hosted Git Server, Branching and merging in Git, Git Server implementation, Working with Remote repository.		
2	**Django URL Mapping, Views, Templates, Static Files**		12	2
	2.1	**URL Mapping** - Understanding function-based views, class-based views, URL configuration, Working with Django templates		
	2.2	**Working with Templates –** Injecting the data from the view to template, creating dynamic templates, Integrating variables in templates, Using Filters		
	2.3	Serving Statics File - Introduction to Static Files Finder,Generating static URLs with the static template tag,FileSystemFinder,The findstatic command		
3	**Forms, CSV Files, Models and Migrations**		10	3
	3.1	**Forms -** The Django Forms library, Validating forms and retrieving Python values, Custom field validation and cleaning, Adding placeholders and initial values		
	3.2	**CSV Files -** Working with CSV files inside Python, Working with Python's csv module,Playing with graphs in Python,Integrating visualizations with Django #Self-learning Learning – working with PDF, and Other Binary Files		
	3.3	**Models and Migration -** Understanding CRUD operations using SQL,Exploring Django ORM,Creating Django models and migrations, Django's database CRUD operations, Bulk create and bulk update operations		
4	**User Authentication, Authorization, and Admin Interface**		09	4

	4.1	**Implementing Authentication and Authorization -** Setting Up User Authentication, User Registration, Login, and Logout, Password Management and Reset, Managing Permissions and Groups, Custom User Models		
	4.2	**Customizing the Django Admin Interface -** Overview of the Django Admin Interface, Customizing the Admin Panel, Registering Models with the Admin, Using Admin Actions and Filters,Customizing the Django Admin UI introduced in Django		
5	**Building Test APIs and Testing**		7	5
	5.1	**Introduction to FastAPI and REST API Concepts** -What is FastAPI? Why FastAPI over Django REST?, RESTful API fundamentals: HTTP verbs, status codes, JSON, FastAPI core features: async support, automatic docs **Building REST APIs with FastAPI** - Path and query parameters, Serialization & Validation with Pydantic		
	5.2	**Testing FastAPI Applications** - Writing tests for endpoints: status codes, response data, Deployment		
		Total	45	--

Reference Books*

Sr. No	Name/s of Author/s	Title of Book	Publisher	Edition/ Year
1	ByBen Shaw	Web Development with Django:	Packt Publishing	2nd

	,Saurabh Badhwar ,Chris Guest ,Bharath Chandra K S	A definitive guide to building modern Python web applications using Django 4		Edition, 2023 ISBN-13 :9781803230603
2	Leonardo Luis Lazzaro	Ultimate Django for Web App Development Using Python: Build Modern, Reliable and Scalable Production-Grade Web Applications with Django and Python	Paperback – Import	22 January 2024
3	Jonathan Chaffer , Karl Swedberg	Learning jQuery - Fourth Edition	Packt Publishing	June 2013 ISBN 9781782163145
4	Rupesh Kumar Tipu , Vandna Batra , Suman Punia	Full-Stack Foundations: Django and JavaScript Integration	Paperback – Import,	20 March 2024
5	Bill Lubanovic	FastAPI	O'Reilly Media, Inc.	ISBN: 9781098135508November 2023
6	https://docs.djangoproject.com/en/5.1/, last received Sept 4, 2024			
7	https://tutorial.djangogirls.org/en/, , last received Sept 4, 2024			
8	https://fastapi.tiangolo.com/, last received April, 2025.			

Course Code	Name of the Course			
216U01L504	Full Stack Development - MERN			
Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	01	02	--	03
Credits Assigned	01	01		02
Evaluation Scheme	Marks			
	LAB/TUT CA	CA (TH)		TW	Total
		IA	ISE		
	--	--	--	*50	

Course pre-requisites:
Basics of HTML,CSS, JavaScript and Concept of Database.

Course Objectives:

- Learn to create interactive and dynamic user interfaces using React, including components, state, props, and hooks.
- Develop server-side applications, handle HTTP requests, and create RESTful APIs using Node.js and Express.
- Understand NoSQL databases, perform CRUD operations, and manage data using MongoDB.
- Integrate the entire MERN stack (MongoDB, Express, React, and Node.js) to develop complete web applications with a seamless client-server connection.
- Learn to deploy full-stack applications on cloud platforms (like Heroku, AWS, or Azure) and maintain CI/CD pipelines for continuous integration and delivery.

Course Outcomes (CO):

- Ability to build, deploy, and manage full-stack web applications using Database, Express.js, Angular, and Node.js and version control system
- Illustrate the React for frontend development and Node.js/Express.js for backend API development.
- Build Skill to database design, query, and manage NoSQL and SQL based databases.
- Illustrate the concepts of various front-end, back-end web application development technologies & frameworks using different web development tools.
- Deploy MERN applications to cloud platforms like Heroku or AWS and collaborate using GitHub version control.

Detailed Curriculum

Module No.	Unit No.	Contents	No of Hrs.	CO
1	**Introduction To MERN**		08	2
	1.1	**Introduction to MERN**, Architecture of MERN, Benefits of MERN, Application of MERN. **Introduction to Version Control system (Git/GitHub).** Overview of Version Control systems Setting up a basic Git server, Hosted Git Server, Branching and merging in Git, Git Server implementation, Working with Remote repository.		
	1.2	**Overview of Fullstack Development**: Client-Server Architecture, Single Page Applications (SPA) **Introduction to MERN Stack**: MongoDB, Express.js, React.js, Node.js - Role of each in the stack		
	1.3	**Environment Setup**: Installing Node.js, npm, MongoDB, and IDE (VS Code) **Version Control**: Basics of Git & GitHub (branching, merging, pull requests) **JavaScript Refresher**: ES6+ features (let, const, arrow functions, destructuring, promises, async/await)		
2	**Frontend Development with React**		10	2
	2.1	**React Basics**: Components, Props, State, Event Handling **React Hooks**: `useState`, `useEffect`, `useContext`, and custom hooks **React Router**: Navigation, Route Parameters, Nested Routes		
	2.2	**Form Handling**: Controlled and Uncontrolled Components, Form Validation **State Management**: Context API, Redux (Optional)		
3	**Backend Development with Node.js and Express.js**		12	3
	3.1	**Node.js Basics**: Introduction, Event Loop, Modules, File System, Buffers, Streams **Express.js**: Routing, Middleware, Handling Requests and Responses, Error Handling.		
	3.2	**REST API Development**: CRUD Operations (Create, Read, Update, Delete) **Authentication & Authorization**: JWT (JSON Web Token) implementation. **Swagger**: Introduction to Swagger and OpenAPI Specification (OAS), Understanding Swagger UI, Swagger Editor, and Swagger Codegen		

	3.3	**File Handling & Uploads**: Handling file uploads with multer. Creating a Mock API Server, Axios		
4	**Database Design**		10	3,4
	4.1	**MongoDB Basics:** NoSQL, Collections, Documents, Data Types **CRUD Operations:** Insert, Read, Update, Delete data in MongoDB **Mongoose ODM:** Schemas, Models, Validation, Relationships (One-to-Many, Many-to-Many)		
	4.2	**Database Security:** Securing sensitive data, Environment Variables (.env) **Aggregation Framework:** Filtering, Sorting, and Grouping Data		
	4.3	Introduction to Relational Databases PostgreSQL Installation & Setup Creating Tables & Writing SQL Queries Data Relationships & Normalization Using pgAdmin for Database Management		
5	**Fullstack Integration, Deployment**		5	5
	5.1	**Connecting Frontend with Backend**: Axios, Fetch API, Handling API Calls		
	5.2	**Deployment:** Deploying MERN application on Heroku, AWS, or Netlify		
		Total	45	--

Reference Books*

Sr. No	Name/s of Author/s	Title of Book	Publisher	Edition/Year
1	Greg Lim	Beginning MERN Stack: Build and Deploy a Full Stack MongoDB, Express, React, Node.js App	Independently Published	1st Edition, 2021
2	Shama Hoque	Full-Stack React Projects - Second Edition: Learn MERN Stack Development by Building Modern Web Apps Using MongoDB, Express, React, and Node.js	Packt Publishing	2nd Edition, 2020
3	Vasan Subramanian	Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node	Apress	2nd Edition, 2019

*In addition to printed books, faculty can suggest (authentic) urls or e-books, e-contents etc.

Course Code	Name of the Course				
216U01L504	Full Stack Development Lab- MEAN				
Teaching Scheme (Hrs./Week)	TH	P		TUT	Total
	01	02		--	03
Credits Assigned	01	01		--	02
Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		TW	Total
		IA	ISE		
	--	--	--	*50	50

Course pre-requisites:

Course Objectives:

- Learn to create dynamic, single-page web applications (SPAs) using Angular, including components, directives, services, and data binding.
- Understand NoSQL databases, perform CRUD operations, and manage data using MongoDB.
- Learn how to deploy MEAN applications on cloud platforms (like AWS, Heroku, or Azure) and set up CI/CD pipelines for continuous integration and deployment.

Course Outcomes (CO):

- Ability to build, deploy, and manage full-stack web applications using Database, Express.js, Angular, and Node.js and version control system
- Illustrate the Angular for frontend development and Node.js/Express.js for backend API development.
- Build Skill to database design, query, and manage NoSQL and SQL based databases.
- Illustrate the concepts of various front-end, back-end web application development technologies & frameworks using different web development tools.
- Deploy MEAN applications to cloud platforms like Heroku or AWS and collaborate using GitHub version control.

Detailed Curriculum

Module No.	Unit No.	Contents	No of Hrs.	CO
1	**Introduction to Mean stack**		5	1
	1.1	Introduction to MEAN, Architecture of MEAN, Benefits of MEAN, Application of MEAN,		
	1.2	Introduction to Version Control system (Git/GitHub). Overview of Version Control systems Setting up a basic Git server, Hosted Git Server, Branching and merging in Git, Git Server implementation, Working with Remote repository.		
2	Understanding of Database Design		5	2
	2.1	Scope of NoSQL, MongoDB: Structure and Applications, Commands, Mongo DB and Its connections, Getting started with Mongoshell, Interacting with data from the command line.		
	2.2	Introduction to Relational Databases PostgreSQL Installation & Setup Creating Tables & Writing SQL Queries Data Relationships & Normalization Using pgAdmin for Database Management		
3	**Nodejs and Express**		10	3
	3.1	Simple Server ii. Response Types – HTML, JSON iii. Routing iv. Express Intro v. Make a call from frontend to server		
	3.2	. Express Params and Query String ii. Express Middleware iii. API Authentication iv. JWT token, Passport.js v. Socket Programming		
4	**Mongoose, Schema and Validation**		10	4
	4.1	Mongoose Definition, connect MongoDB using Mongoose, Schema, Importing and exporting data , Connecting to a Node.js application, Querying the database from Node.js		
5	**Angular 2+ , MongoDB, Node.js**		15	5
	5.1	Introduction, Introduction to typescript, Environment Setup, Modules, Component, Template, Directives, Custom Directives, Pipes, Custom Pipes, Services, Routing, Dependency Injection, Change		

		Detection, Advanced Routing, Template Driven Form, Model Driven Form, Advanced HTTP, Animation, CRUD operations in MongoDB, REST API. **Swagger: Introduction to** Swagger **and** OpenAPI Specification (OAS), Understanding Swagger UI, Swagger Editor, and Swagger Codegen		
	5.2	**CODE REVIEW + DEPLOYMENT** i. Tools for code review ii. Standard coding conventions iii. Firebase iv. Deploy using Netlify v. Deploy using AWS Ec2		
Total			45	--

Reference Books*

Sr. No	Name/s of Author/s	Title of Book	Publisher	Edition/Year
2	David Gutman	Fullstack Node.js The Complete Guide to Building Production Apps with Node.js	Fullstack.io.	2019
3	Amit Phaltankar Liviu Nedov Michael Harrison Juned Ahsan	MongoDB Fundamentals: A hands-on guide to using MongoDB and Atlas in the real world	Packt Publication	1st EditionDec 2020

*In addition to printed books, faculty can suggest (authentic) urls or e-books, e-contents etc.

Semester VI

Course Code	Name of the Course
216U01C601	Digital Signal & Image Processing

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	03	--	--	03
Credits Assigned	03	--	--	03

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	--	20	30	50	100

Course prerequisites: Basic mathematical background of transforms, matrices, complex numbers and programming skills.

Course Objectives:

1. Understand the fundamental concepts of discrete-time signals, systems, and operations on signals.
2. Learn spatial and frequency domain enhancement techniques for processing 1-D and 2-D digital signals.
3. Explore various image transforms and their applications for analyzing digital signals in the frequency domain.
4. Examine segmentation, boundary detection, and compression techniques for representing and compressing images.
5. Develop practical skills in designing and implementing digital signal and image processing applications using appropriate software tools.

Course Outcomes (CO):

CO1. Interpret fundamentals of discrete time signals and systems and signal manipulation methods
CO2. Apply various spatial and frequency domain enhancement techniques for 1-D and 2-D signals
CO3. Analyze signals in frequency domain using various image transforms.
CO4. Evaluate extracted analyzed information for synthesis of digital signals.
CO5. Design and develop applications based on 1-D and 2-D digital signals.

Module	Unit	Contents	No of	CO

No.	No.		Hrs.	
1	**Discrete Time Signals and Systems**		09	CO1
	1.1	Introduction to digital signals and systems, Properties and operations on digital signals, Introduction to biosignals: ECG, EEG, EMG, PPG		
	1.2	Classification of signals, system, LTI system		
	1.3	Convolution in time domain (linear & circular), Correlation.		
2	**Fundamentals of Digital Image and Spatial Domain Enhancement**		09	CO2
	2.1	Digital image Representation, Elements of digital image processing systems, sampling and quantization, basic relationships between pixels, mathematical operations on images.		
	2.2	Spatial domain enhancement techniques: Point processing, Neighbourhood processing, spatial domain filtering, zooming.		
	2.3	Spatial enhancement: Global processing: Histogram Equalization.		
	Self-Learning Topic: Histogram specification and adaptive histogram equalization.			
3	**Image Transform: Frequency Domain Representation and Enhancement**		10	CO3
	3.1	Introduction , DFT and its properties, radix-2 algorithm(2-DFT), FFT algorithm: divide and conquer approach, Decimation in Time(DIT)-FFT		
	3.2	Discrete Cosine Transform, Walsh Transform, Hadamard Transform, Haar Transform, Principal component Analysis (PCA/ Hoteling Transform), Introduction to Wavelet Transform		
	3.3	Low Pass and High Pass Frequency domain filters: Ideal, Butterworth, Homomorphic filter		
	Self-Learning Topic: Discrete Sine Transform (DST)			
4	**Image Segmentation and Representation**		08	CO4
	4.1	Image segmentation based on discontinuities: point, line and edge detection (Laplacian, Cany), edge linking, Thresholding (Global, local, optimum), Region based segmentation, edge based segmentation: Hough Transform, Segmentation of medical images, Edge detection and enhancement for clearer imaging		

	4.2	Boundary descriptors: Signature, Chain code, Shape number, Moments		
5	**Introduction to Morphology and Image Compression**		10	CO5
	5.1	Morphological operations: Dilation, Erosion, Opening, Closing, Hit or Miss Transform, Boundary extraction		
	5.2	Introduction, redundancies: coding, inter-pixel, psycho-visual, compression ratio, fidelity criteria Lossless compression techniques: Run length coding, Arithmetic coding, Huffman coding, Differential PCM		
	5.3	Lossy Compression techniques: Improved grey scale quantization, Vector quantization, Transform coding, JPEG.		
	5.4	Case Study: ECG Signal Analysis and Arrhythmia Detection, Tumor Segmentation in MRI Images and Ultrasound Image Enhancement		
	Self-Learning Topic: Morphological operation - Thinning and Thickening			
		Total	45	--

Recommended Books:

Sr. No	Name/s of Author/s	Title of Book	Publisher	Edition/Year
1.	John G. Proakis and D.G. Manolakis	*Introduction to Digital signal processing*	Pearson	Fourth edition, 2015
2.	A. NagoorKani	*Digital Signal Processing*	McGraw Hill Publications	2nd edition
3.	R. C. Gonsales and R. E. Woods	*Digital Image Processing*	Pearson Education	Second edition
4.	A.K. Jain	*Fundamentals of Image processing*	Prentice Hall of India Publication	--
5.	S.Jayaraman, S Esakkirajan, T Veerakumar	*Digital Image Processing*	McGraw Hill	2018 Edition

Course Code	Name of the Course
116U01C602	Network & Information Security

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	03	--	--	03
Credits Assigned	03	--	--	03

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	--	20	30	50	100

Course prerequisites:
Basics of Operating System and Computer Network.

Course Objectives:
1. To understand fundamental principles of information security.
To gain knowledge on malicious & non-malicious programme errors & apply counter- measures.
3. To understand the various web attacks and its preventive mechanisms.
4. To apply different techniques to secure data in transit across data networks.
5. To study security laws and analyse the ethical issues.

Course Outcomes (CO):

At the end of successful completion of the course the student will be able to

CO1: Understand various security goals, threats, vulnerabilities and controls.
CO2: Apply various cryptographic algorithms for software security.
CO3: Identify, analyse and mitigate web attacks.
CO4: Apply network security mechanisms for threat management.
CO5: Understand investigation process, legal and ethical issues in security.

Detailed Curriculum

Module No.	Unit No.	Contents	No of Hrs.	CO
1	**Introduction**		08	CO 1
	1.1	Introduction to Information Security : Attacks, Vulnerability, Security Goals, Security Services and Mechanisms.		
	1.2	Conventional Cryptographic Techniques : Conventional substitution and transposition ciphers, One-time Pad, Block cipher and Stream Cipher, Steganography		
	1.3	Cryptography fundamentals: Encryption, decryption, keys, ciphers. Symmetric vs asymmetric encryption, AES: The Advanced Encryption Standard.		
	1.4	Hash functions, Public Key Cryptography, Digital Certificates, Digital Signatures, Public Key Infrastructure (PKI).		
2	**System & Software Security**		10	CO 2
	2.1	Unintentional (Non-malicious) Programming flaws: Oversights - Buffer Overflow, Incomplete Mediation, Time-of-Check to Time-of-Use, Undocumented Access Point Off-by-One, Error Integer Overflow, Unterminated Null-Terminated String, Parameter Length, Type, and Number, Unsafe Utility Program, Race Condition		
	2.2	Types of malware: Viruses, Worms, Trojans, Ransomware, Anti-malware strategies and tools.		
	2.3	Countermeasures: Countermeasures for Users, Countermeasures for Developers, Countermeasure Specifically for Security, Countermeasures that Don't Work, Secure Software Development Lifecycle (SSDLC), Privacy by Design (PbD).		
3	**Web & Application Security**		09	CO 3
	3.1	Web application vulnerabilities: SQL injection, XSS, CSRF.		
	3.2	Threat Modelling, Privilege Escalation, Access Control.		
	3.3	Email Security – Email based attacks, Phishing & its types, Protecting Against Email Attacks.		
	3.4	Web security standards: OWASP Top 10, Secure Code Review, Secure coding practices.		
4	**Network Security**		10	CO 4
	4.1	Network security principles and protocols, Threats to Network Communications - Interception, Modification,		

		Fabrication, Interruption.		
	4.2	Securing communication: SSL/TLS, IPSec, HTTPS protocol architecture & it's workflow, Authentication mechanisms: Passwords, biometrics, two-factor authentication Access control models: Discretionary Access Control (DAC), Mandatory Access Control (MAC), Role-Based Access Control (RBAC), Security policies and enforcement mechanisms.		
	4.3	Network based attacks – Denial of Service (DoS), Distributed DoS, Man-in-the-Middle Attack (MITM), DNS Spoofing, Importance of Firewalls, types of Firewalls, Next Generation Firewalls (NGFs), VPNs, IDS/IPS.		
	4.4	Wireless network security – WPA standards.		
5	**Incident Response, Forensics & Cyber Laws.**		08	CO 5
	5.1	Incident response lifecycle, role of CERT.		
	5.2	Tools and techniques for digital forensics, Evidence collection, Chain of Custody, Case Studies of Forensics.		
	5.3	Indian IT Act 2000, Digital Personal Data Protection Act (DPDP Act), 2023, International Cyber-Security laws/frameworks – GDPR, HIPAA.		
	5.4	Piracy, Copyrights, Trademarks, Trade-secrets, Patents, Geographical Indications.		
	5.5	Discussion on recent case studies on cyber-security and digital forensics.		
		Total	45	--

Reference Books

Sr. No	Name/s of Author/s	Title of Book	Publisher	Edition/ Year

1	Charles P. Pfleeger, Shari Lawrence Pfleeger, Jonathan Margulies.	Security in Computing	Prentice Hall	Sixth Edition
2	Behrouz.A. Fourouzan, Debdeep Mukhopadhyay	Cryptography and Network Security	McGraw Hill	Eight Edition
3	William Stallings	Cryptography and Network Security: Principles and Practice	Pearson	Eight Edition
4	Bernard Menezes	Network Security and Cryptography	Cengage Learning	Second Edition
5	Mark Stamp	Information Security: Principles and Practice	Wiley	Third Edition
6	Indian IT Act 2000	https://eprocure.gov.in/cppp/rulesandprocs/kbadqkdlcswfjdelrquehwuxcfmijmuixngudufgbuubgubfugbububjxcgfvsbdihbgfGhdfgFHytyhRtMjk4NzY=		
7	Digital Personal Data Protection Act 2023	https://www.meity.gov.in/writereaddata/files/Digital%20Personal%20Data%20Protection%20Act%202023.pdf		

Course Code	Name of the Course
116U01C603	Artificial Intelligence

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	03	--	--	03
Credits Assigned	03	--	--	03

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	--	20	30	50	100

Course prerequisites(if any):
Data structures, analysis of algorithms

Course Objectives:

1. To introduce the history, core concepts, and sub-areas of artificial intelligence, including current trends and ethical implications.
2. To explore the structure and functionality of intelligent agents, focusing on their environments, rationality, and problem-solving capabilities through search algorithms.
3. To understand knowledge representation and reasoning techniques, including first-order logic and uncertain reasoning using Bayesian networks.
4. To cover the basics of machine learning types, model evaluation, and practical applications through case studies in supervised and unsupervised learning.

Course Outcomes (CO):

At the end of successful completion of the course the student will be able to:

CO1	Design AI solution with appropriate choice of agent architecture
CO2	Analyse and solve problems for goal based agent architecture
CO3	Represent and formulate the knowledge to solve the problems using various reasoning techniques
CO4	Understand and differentiate between various types of machine learning

Module No.	Unit No.	Details	Hrs.	CO
1	**Introduction to Artificial Intelligence**		08	CO1
	1.1	History of Artificial Intelligence, Foundations of AI		
	1.2	Categorization of Intelligent System, Components of AI Program,		
	1.3	Sub-areas of AI: Machine Learning, Robotics, Neural Networks, Fuzzy Logic, Cognitive Computing		
	1.4	**Applications of AI,** **Current trends in AI:** concepts and applications of: Generative AI, Predictive AI, Discriminative AI, Descriptive AI, Prescriptive AI, Adaptive AI, Interactive AI, Cognitive AI, Explainable AI (XAI), Hybrid AI AI Ethics and Fairness		
		#Self Learning – Exploring and studying Emerging Trends and Research Directions in AI through CASE studies		
2	**Intelligent Agents**		08	CO1
		Agents and Environments, The concept of rationality, The Task environment and their properties, PEAS, The Wumpus World PEAS analysis, The structure of Agents, Types of Agents, Learning Agent.		
3	**Goal based agents for Problem Solving : Searching**		10	CO2
	3.1	**Solving problem by Searching :** Problem Solving Agent, Formulating Problems, Example Problems.		
		Problem Formulation in AI: State-space search, production systems/rules, Problem characteristics		
	3.2	**Uninformed Search Methods:** Breadth First Search Vs Depth First Search, Iterative Deepening depth first search		
	3.3	**Informed Search Methods:** Heuristic, properties of good heuristic, Greedy best first Search, A* Search		
	3.4	**Local Search Algorithms and Optimization Problems:** Hill climbing algorithms, Constraint Satisfaction problems-concept, inferences in CSP, CSP Backtracking algorithm, Genetic algorithms: The genetic algorithm process, solving problems with GA for optimization and learning		
	3.5	**Adversarial Search:** Games, Optimal strategies, The minimax algorithm , Alpha-Beta Pruning		
		#Self Learning – Online search algorithms, partially observable/imperfect information games		
4	**Goal based agents for Problem Solving : Planning**		04	
	4.1	**Planning :** The planning problem, Planning Vs Searching, ADL, Partial order planning, Hierarachical task planning		

		#Self Learning – Multiagent planning, classical planning and real-world planning		
5	**Knowledge representation and Reasoning**		08	CO3
	5.1	Knowledge based Agents, The Wumpus World, inference procedures, **First Order Logic:** Syntax and Semantic, Inference in FOL, Unification and lifting, Forward chaining, backward Chaining, Resolution, Knowledge Engineering process.		
	5.2	**Uncertain Knowledge and Reasoning:** Uncertainty, Representing knowledge in an uncertain domain, Inference in Bayesian network,		
		#Self Learning : Propositional Vs Predicate logic		
6	**Applications of AI and Current State of research in AI**		07	CO4
	6.1	**Natural Language Processing:** Steps in NLP, Parsing and semantic interpretation, discourse and pragmatic processing, implementation aspects of syntactic analysis		
	6.2	**Expert system:** components of expert system, Expert Systems vs Traditional System. Characteristics of expert systems, roles in ES implementation, ES implementation process, applications, advantages and limitations of ES		
	6.3	Case studies based on current research in AI **Prompt engineering:** types of prompts, Prompt Failures & Biases, applications **AI copilot:** GitHub copilot **Retrieval Augmented Generation**[RAG], Real world case studies of RAG **Biomedical and Life Sciences**		
		Total	45	

 # Students should prepare all Self Learning topics on their own. Self-learning topics will enable students to gain extended knowledge of the topic. Assessment of these topics may be included in IA and Laboratory Experiments.

Recommended Books:

Sr. No.	Name/s of Author/s	Title of Book	Name of Publisher with country	Edition and Year of Publication
1.	Stuart J. Russell and Peter Norvig	Artificial Intelligence : A Modern Approach	Pearson Education.	Fourth Edition
2.	Elaine Rich and Kevin Knight	Artificial Intelligence	The McGraw-Hill	Third Edition
3.	George F Luger	Artificial Intelligence	Pearson Education	Fourth Edition

Course Code	Name of the Course
216U01E611	Compiler Construction

	TH	P	TUT	Total
Teaching Scheme (Hrs./Week)	03	--	--	03
Credits Assigned	03	--	--	03

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	--	20	30	50	100

***Term Work will consist of Practical covering the entire syllabus of compiler construction. Students will be graded based on continuous assessment of their term work.**

Course prerequisites (if any):
Finite automata, pushdown automata etc. from Theory of Computer science.

Course Objectives
The course aims to give knowledge of the principal structure of a compiler and about the basic theories and methods used to implement the different phases of the compiler.

Course Outcomes
At the end of successful completion of the course the student will be able to

CO1	Study phases of compiler and illustrate different parsing techniques and semantic analysis.
CO2	Illustrate and analyze the different intermediate code generation techniques and run time storage allocation.
CO3	Apply optimization techniques
CO4	Analyze and interpret the different issues in code generation phase

Detailed Curriculum

Module No.	Unit No.	Details	Hrs.	CO
1	**Introduction to Compiler**		5	CO1
	1.1 1.2 1.3 1.4	Compilers: Introduction to Compilers, Phases of a compiler, Comparison of compilers and interpreters. Compiler-compilers : JAVA compiler environment, YACC compiler-compiler		
2	**Lexical Analysis**		3	CO1
	2.1 2.2 2.3 2.4 2.5 2.6	Role of a Lexical analyzer, input buffering, specification and recognition of tokens, Finite Automata, Designing a lexical analyzer generator, Pattern matching based on NFA's.		
3	**Syntax Analysis**		8	CO1
	3.1 3.2 3.3	Role of Parser, Top-down parsing: Recursive descent and predictive parsers (LL), Bottom-Up parsing: Operator precedence parsing, LR, SLR and LALR parsers.		
4	**Syntax Directed Translation**		5	CO1
	4.1 4.2 4.3 4.4 4.5	Syntax directed definitions, construction of syntax tree, Type checking Top-down translation and Bottom-up evaluation of inherited attributes, analysis of syntax directed definitions		
5	**Run Time storage**		6	CO2
	5.1 5.2 5.3 5.4 5.5	Activation record, handling recursive calls, management of variable length blocks, garbage collection and compaction, storage allocation strategies.		
6	**Intermediate Code Generation**		4	CO2

	6.1 6.2 6.3 6.4	Intermediate languages: graphical representations, DAGs, Three address code, Types of three address statements, Syntax directed translation into three address code, implementation of three address statements		
7	**Code Generation**		**8**	**CO4**
	7.1 7.2	Semantic stacks, attributed translations, evaluation of expressions, control structures, and procedure calls		
8	**Code Optimization.**		**6**	**CO3**
	8.1 8.2 8.3 8.4 8.5 8.6 8.7 8.8 8.9 8.10	Machine dependent and machine independent code optimization Sources of optimization Data flow analysis Tail call optimization and Tail Recursion Elimination, Procedure Integration, Inline Expansion Leaf Routine optimization and shrink wrapping Register allocation and assignment, Graph coloring, Unreachable Code Elimination, Straightening If simplifications, Loop Simplifications, Loop inversion, Un switching, Branch optimizations, Tail merging or cross jumping, Conditional moves, Dead code Elimination, Branch Prediction, Machine Idioms and Instruction combining		
Self Learning Component: Compilation of object-oriented languages				
		Total	**45**	

 # Students should prepare all Self Learning topics on their own. Self-learning topics will enable students to gain extended knowledge of the topic. Assessment of these topics may be included in IA and Laboratory Experiments.

Recommended books:

Sr. No.	Name/s of Author/s	Title of Book	Name of Publisher with country	Edition and Year of Publication
1.	A.V. Aho, and J.D.Ullman	Principles of compiler construction	Pearson Education	Second Edition, 2007
2.	Kenneth C. Louden	Compiler Construction, Principles and Practice	Cengage Learning	Fourth Edition, 2006
3.	Dick Grune, Koen G.L, Henri Bal	Modern Compiler Design	Wiley Publications	Second Edition, 2006
4.	D M Dhamdhere	System Programming	Tata McGraw Hill publication	First Edition, 2011

Course Code	Name of the Course
216U01E612	Virtual Reality & Augmented Reality

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	03	--	--	03
Credits Assigned	03	--	--	03

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	--	20	30	50	100

Course pre-requisites:
No programming experience is required. Knowledge about computer graphics can help in understanding real time 3D concept which means understanding coordinate systems and transforms. Information about OpenGL (a ubiquitous graphics API) and Fundamentals of C# programming can also be helpful.

Course Objectives:
1. Understand fundamental principles of information security.
2. Relate the ergonomics related to immersive technologies.
3. Use and comprehend VR Hardware and Software Components for AR-VR.
4. Analyse socio-legal aspects of immersive technologies.
5. Able to use tools and develop applications in the domain of AR-VR.

Course Outcomes (CO):
CO1. Understand Virtual, Mixed and Augmented Reality platforms and its associated technologies.
CO2. Identify ergonomic considerations in the design and use of AR/VR systems.
CO3. Understand Human Factors, Legal, and Social Considerations.
CO4. Use the tools and technologies used by professionals working in AR & VR.

Module No.	Unit No.	Contents	No of Hrs.	CO
1	**Introduction to Augmented and Virtual Reality**		08	CO1
	1.1	3D Computer Graphics: Introduction, The Virtual world space, positioning the virtual observer, the perspective projection, human vision, stereo perspective projection, 3D clipping, Colour theory, Simple 3D modelling, Illumination models, Reflection models, Shading algorithms, Radiosity, Hidden Surface Removal, Realism-Stereographic image.		
	1.2	Computer-Generated Worlds, What Is Augmented Reality? What Is Virtual Reality?		
	1.3	Understanding Virtual Space, Defining Visual Space and Content Defining Position and Orientation in Three Dimensions.		
	1.4	Historical evolution of AR-VR.		
2	**Ergonomics and Interaction Design for AR-VR.**		10	CO2
	2.1	The Mechanics of Sight., Component Technologies of Head-Mounted Displays, Augmenting Displays, Fully Immersive Displays		
	2.2	The Mechanics of Hearing, Audio Displays.		
	2.3	The Mechanics of Feeling, Tactile and Force Feedback Devices, Sensors for Tracking Position, Orientation & Motion, Devices to Enable Navigation and Interaction.		
3	**Sensors and Actuators in Augmented and Virtual Reality**		09	CO4
	3.1	Introduction to sensors, Augmented Reality, next-generation sensors, applications Sensors in Tablet/ Computer/ Smart phones, Bio Sensors.		
	3.2	Optical sensors, Inertial Measurement Unit (IMU), accelerometers, GPS, gyroscopes, solid-state compasses, Radio-frequency identification (RFID).		
	3.3	History of sensors in Virtual Reality, next-generation sensors, applications areas, Current Developments. Magnetometer, Gyroscope. Head tracking Mechanisms: Working Principle and Limitations.		
	3.4	History of sensors in Augmented Reality, Vibrator motors, Speaks systems in Augmented reality, AR headsets (Construction and working). Advanced sensors, Special Cameras: Time of Flight (ToF) Cameras, Binocular Depth Sensing, Structured-light sensors Heads-Up Display, Tethered headset-based sensors, Mobile		

		Headset-based sensors.		
4	**Legal and Regulatory Issues in AR-VR**		10	CO3
	4.1	Privacy concerns and data security in AR/VR applications.		
	4.2	Intellectual property (IP) and copyright issues in immersive media.		
	4.3	Regulatory frameworks for AR/VR (safety, accessibility, industry-specific guidelines).		
	4.4	Liability and accountability for AR/VR-related incidents (physical harm, content misuse).		
5	**AR-VR Tools & Applications.**		08	CO4
	5.1	AR-VR tools and softwares – Unity, ARCore, ARKit, ShapesXR, TiltBrush, Gravity Sketch, SCILab, Unreal Engine, SketchUp.		
	5.2	Application of Multidimensional systems in Industry Flight simulators, Space shuttle simulator, Heavy engineering simulators, Mining simulators, Bus/Truck simulators, Applications in Biomedical and Life Sciences		
	5.3	Future Augmented & Virtual Reality systems, Expectations from AR-VR technology, AR-VR research trends technology, Futures input, display technologies, Metaverse.		
		Total	45	--

Recommended Books:

Sr. No	Name/s of Author/s	Title of Book	Publisher	Edition /Year
1	Steve Aukstakalinis	*Practical Augmented Reality: A Guide to the Technologies, Applications, and Human Factors for AR and VR (Usability).*	O'Reilly Media	First Edition, 2016.
2	Paul Mealy	*Virtual & Augmented Reality For Dummies*	Wiley	First Edition, 2018.
3	Alan B. Craig	Understanding Augmented Reality : Concepts and Applications	Elsevier , USA	First Edition, 2013.
4	Nawaz Mohamudally	State of the Art Virtual Reality and Augmented Reality Knowhow	IntechOpen , USA	First Edition, 2018.
5	Paul T. Hayden	The Law of Virtual and Augmented Reality	University of Pennsylvania Law Review	2018
6	WilliamR. Sherman, Alan B. Craig	Understanding Virtual Reality: Interface, Application and Design.	Morgan Kaufmann	2008

Course Code	Name of the Course
216U01E613	Embedded System and IOT

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	03	--	--	03
Credits Assigned	03	--	--	03

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	–	20	30	50	100

Course prerequisites: Operating system, Data Networks/ Computer Networks

Course Objectives: To learn basic architecture of Wireless sensor networks and Internet of Things and understand WSN routing protocols and evaluate software, hardware platforms for IoT technology. Also create applications using IOT analytics.

Course Outcomes (CO):
At the end of successful completion of the course the student will be able to

CO1	Explain the basic architecture and working principle of wireless sensor networks and Internet of Things
CO2	Identify challenges and issues in WSN routing and suggest solutions.
CO3	To use different Operating system for Wirelese sensor networks and IoT
CO4	Evaluate the software and hardware platforms for IoT Technologies and design small IoT application.
CO5	Create IoT application data using IoT Analytics.

Detailed Curriculum

Module No.	Unit No.	Details	Hrs.	CO
1		**Introduction to Wireless Sensor networks and IOT**	**12**	**CO1**
	1.1 1.2 1.3.	**Characteristic requirements for WSN** - Challenges for WSNs – WSN vs Ad hoc Networks **Sensor network architecture** Commercially available sensor nodes –Imote, IRIS, Mica Mote, EYES nodes, BTnodes, TelosB– **IoT ARCHITECTURE:** Various architectures of the IoT middleware such as distributed, services oriented, centralized, M2M Domain model, Information model, functional model, communication model, IoT reference architecture **Self learning : Application Scenar ios for WSN and IOT-** Home Control - Building Automation - Industrial Automation, Agriculture, Medical, Environmental Monitoring.		
2		**Medium Access Control and Routing Protocols**	**12**	**CO2**
	2.1 2.2	**Medium Access Control Protocols:** Fundamentals of wireless MAC protocols, Contention-based protocols IEEE 802.11 - Schedule-based protocols; SMAC-BMAC - The IEEE 802.15.4 MAC protocol. **Routing Protocols :** Routing Challenges and Design Issues in Wireless Sensor Networks, Classification of Ad hoc Routing protocols, Flooding and gossiping - Data centric Routing – SPIN – Directed Diffusion – Energy aware routing - Gradient-based routing - Rumor Routing — Hierarchical Routing -– Location Based Routing – GF, GAF, GEAR, GPSR – Real Time routing Protocols		
3		**Operating system and Sensors in WSN and IOT**	**08**	**CO3**
	3.1 3.2	TinyOS, Raspbian, Debian Perception layer of the IoT: Various sensors such as light sensors, accelerometer, gyroscope, magnetometer, camera microphone, GPS, proximity sensors, moisture, pH sensors, Bio Sensors etc.		

4		**IoT Physical Devices**	**08**	**CO4**
	4.1 4.2 4.3	IoT Prototype design using microcontroller boards: Arduino, Raspberry PI/ Beagalbone, Introduction to Actuators in IoT applications. Case study: Home Automation/Industrial Automation, IoT in Agriculture, IoT in Healthcare.		
5		**IoT Analytics**	**05**	**CO5**
	6.1 6.2 6.3	Business Process in IoT IoT Analytics with cloud Edge analytics		
		Total	**45**	

Reference Books*

Sr. No.	Name/s of Author/s	Title of Book	Name of Publisher with country	Edition and Year of Publication
1.	Carlos De Morais Cordeiro, Dharma Prakash Agarwal	Adhoc and sensor networks:Theory and Applications	World Scientific Publishing	1st edition ,2006
2.	C.Siva Ram murthy,B.S.Manoj	Adhoc wireless networks	Pearson	1st edition,2006
3.	Arshdeep Bhaga and Vijay Madisetti	"Internet of Things (A Hands-on-Approach)",University Press	Tata McGraw-Hill ,India	4th edition ,2015
4.	Hakima Chaouchi	"The Internet of Things (Connecting objects to the web)"	Wiley publication	1st edition,2014
5.	Hakim Cassimally and Adrian McEwen	" Designing the Internet of things"	Wiley publication	1st edition,2013

Course Code	Name of the Course
216U01E614	Mobile Communication and Mobile App Development

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	03	--	--	03
Credits Assigned	03	--	--	03

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	--	20	30	50	100

Course pre-requisites:
Basic Knowledge of Computer Networks, Layered Architecture, Structure and working related Protocols.

Course Objectives:

1. To provide an overview of Mobile & Cellular Communication networks area and its applications in communication engineering.
2. To understand the various terminology, principles, concepts, Standards, algorithms and different methodologies used in Wireless Communication Networks specifically for Wireless Ad-Hoc Networks

Course Outcomes (CO):

CO1. Explain the basic concepts of various wireless networks and their working characteristics with respect to mobile network generations such as 2G, 3G and beyond.

CO2. Compare infrastructure based and Ad hoc networks, elaborating characteristics and features of Ad hoc Networks

CO3. Inspect designing of Wireless MAC protocols for Ad hoc networks; and the working principle of different WLAN IEEE standards.

CO4. Describe various Network Layer & Transport layer mechanisms and Routing Protocols for Wireless networks.

CO5. Explain basic Android App development concepts, Design and Develop Android Applications.

Detailed Curriculum

Modul	Unit	Contents	No of	CO

e No.	No.		Hrs.	
1	**Cellular Mobile Networks**			
	1.1	Cellular networks: Basic cellular system, Frequency allocation, Frequency re-use; CDMA Networks	10	CO1
	1.2	GSM System Architecture: GSM Radio subsystem, Interfaces, Network and switching subsystem, Operation subsystem		
	1.3	GSM channels: Traffic Channel multiframe, Control (Signaling) Channel Multiframe, Frames, Multi-frames, Super-frames and Hyper-frames		
	1.4	GSM Call Set up Procedure		
	1.5	Handoff: Hard and soft		
2	**2.5 G, 3 G Networks and beyond**			
	2.1	2.5G Networks: GPRS Architecture, GPRS Network Nodes: Mobile Station, Base Station System, GPRS Support Node, HLR and VLR, GPRS Interfaces	10	CO1
	2.2	3G Networks: The Universal Mobile Telecommunication System (UMTS) - UMTS Network Architecture, UMTS FDD and TDD		
	2.3	Next generation networks; 3GPP LTE, VoLTE		
	#Self Learning - 4G, 5G			
3	**Wireless Network Technologies and Protocols**			
	3.1	Infrastructure & Ad hoc Networks; Introduction to ad hoc networks – definition, characteristics features.	15	CO2 CO3 CO4
	3.2	MAC Protocols for Ad hoc wireless Networks: Issues in designing a MAC protocol for Ad hoc wireless Networks, Design goals and Classification of MAC protocols, Contention based protocols with reservation mechanisms.		
	3.3	IEEE standards: 802.11a/b/g/e/n/ac, features and comparisons.		
	3.4	Introduction to Mobile IP: Requirements, IP packet delivery, agent discovery, registration, tunneling and encapsulation.		
	3.5	Routing protocols for Ad hoc wireless Networks: Issues in designing a routing protocol for Ad-hoc wireless Networks, Classification: Proactive, Reactive routing protocols, Hybrid routing Algorithm. Unicast routing algorithms: DSR, AODV, ZRP		
	3.6	Classical TCP improvements – methods of mobile TCP: Indirect TCP, snooping TCP, mobile TCP		
	#Self Learning – Bluetooth, WLAN Security- WEP, WPA, WPA2			
4	**Core Android App Development**			
	4.1	Android Architecture & Development Environment		

		(Android Studio)	05	CO5
	4.2	Introduction; Android UI Design Basics - XML Layouts, Views, RecyclerView, ConstraintLayout		
	4.3	**Activity & Fragment Lifecycle**, Intents, Navigation Components		
	4.4	**User Input & Event Handling**, Toasts, Dialogs, Menus		
	4.5	**Storage:** SharedPreferences, Internal & External Storage, Room Database		
	#Self Learning:- WAP, WML			
5	**Advanced Android Concepts**		05	CO5
	5.1	**Networking & APIs:** Retrofit, Volley, JSON Parsing		
	5.2	**Multithreading & Background Tasks:** AsyncTask, Handlers, WorkManager, Coroutine		
	5.3	**Jetpack Components:** ViewModel, LiveData, DataBinding		
	5.4	**Publishing App:** App signing, build variants, Play Store deployment		
	#Self Learning:- Firebase Integration, Notifications, Maps SDK			
		Total	45	--

Reference Books*

Sr. No	Name/s of Author/s	Title of Book	Publisher	Edition/Year
1.	J. Schiller	Mobile Communications	Pearson Education	2nd Edition
2.	Kaveh Pahlavan, Prashant Krishnamurthy	Principles of Wireless Networks	Pearson Education	2003

3.	Mustafe Ergen	Mobile Broadband Including Wi Max and LTE	Springer	2009
4.	Savoy G. Glisic	Advanced Wireless Communication & Internet	Wiley Publication	3rd Edition
5.	Iti Saha Mishra	Wireless Communications and Networks: 3G and Beyond	McGraw Hill Education	2nd Edition, 2017
6.	John Horton	Android Programming for Beginners	Packt Publications	3rd Edition, 2021

*In addition to printed books, faculty can suggest (authentic) urls or e-books, e-contents etc.

Course Code	**Name of the Course**			
216U01E615	**Microservices Foundations**			
Teaching Scheme (Hrs./Week)	**TH**	**P**	**TUT**	**Total**
	03	--	--	03
Credits Assigned	03	--	--	03

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	–	20	30	50	100

Course pre-requisites:
Practical knowledge of Java

Course Objectives:
Microservices Foundations is an advance and modern subject which touches on many aspects of Microservices which are necessary to know before you start your micro-service journey or enhance your micro-service journey. This subject is designed as a deep dive into what is micro-service, how you think about it, and build, test, deploy and, breaking apart existing systems into a micro-service architecture. This subject creates your foundation for building Microservices using Java and Spring Boot which are the most popular programming language and framework used in the IT industry.

Course Outcomes (CO):
CO1. Understand the Fundamentals of Microservices
CO2. Analyze the Principles and Characteristics of Microservices
CO3. Design and Model Microservices Applications
CO4. Build Microservices Using Java and Spring Boot
CO5. Test and Document Microservices

Detailed Curriculum

Module No.	Unit No.	Contents	No of Hrs.	CO
1	**Microservices : Introduction**		03	CO1
	1.1	Introduction to Microservices, Definition		
	1.2	History of monolithic application: History of Microservices, Micro-service Early Adopters		
		#Self-Study: Understand monolithic vs Microservices applications		

2	**Microservices : Essentials**		**12**	**CO2**
	2.1	Monolithic application challenges		
	2.2	Microservices Principles, Microservices Characteristics, Microservices with Polyglot Architecture		
	2.3	Automation in Microservices environment, Microservices with a supporting ecosystem		
	2.4	Microservices & Distributed Architecture, Related Architecture styles		
3	**Microservices : Design**		**10**	**CO3**
	3.1	Introduction, Design a Microservices-Based Application, Best Practices, How micro-service applications communicate, The 12-Factor methodology		
	3.2	Data-Persistence Pattern for Microservices		
	3.3	Applying Domain Driven Design to Microservices Architecture		
		#Self-Learning –Understand 12-Factory principles & demonstrate it		
4	**Microservices: Development using Java and Spring Boot**		**12**	**CO4**
	4.1	Microservices Development Guidelines, The basic Spring Boot Application (Microservice)		
	4.2	A Real Three-Tier Spring Boot Application (Microservice), Starting with Microservices		
	4.3	The Microservices Journey through tools: Service Discovery & Load balancing, Routing with an API Gateway, Hands-on implementation		
5	**Microservices : Testing**		**8**	**CO 5**
	5.1	Testing Pyramid, Mapping Testing Pyramid to microservices		
	5.2	Types of Microservices Testing, Key Challenges for testing microservices, Performance testing (Locust), Capacity monitoring of microservices		
	5.3	How to overcome microservices testing challenges, Swagger based Spring Boot microservices documentation and testing		
	5.4	Microservices API testing using Postman tool , Hands-on Spring boot based microservices testing using Postman and Swagger		
		#Self-Learning – understand REST APIs / microservice testing and demonstrate it using Postman and Swagger		
		Total	45	--

Recommended Books:

Sr. No	Name/s of Author/s	Title of Book	Publisher	Edition/ Year
1	Aniket Mhala and Vinod Sashital	A Beginner's Guide Microservice Foundation	Modern Solutions Publications	July 2022
2	Moises Macero	Learn Microservices with Spring Boot	Apress Publications	February 2017
3	Morgan Bruce Paulo A. Pereira	Microservices in Action	2nd IBM limited edition	2019
4	Rajesh V	Spring Microservices	Packt Publication	June 2016
5	Eberhard Wolff	Microservices : Flexible Software Architecture	APress Publication	October 2016

Course Code	Name of the Course
216U01E616	Agile Project Management

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	03	--	--	03
Credits Assigned	03	--	--	03

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	-	20	30	50	100

Course pre-requisites:
Software Engineering

Course Objectives:
This course includes agile methodology, in different steps of project management like lifecycle, planning scheduling estimates etc.

Course Outcomes (CO):
CO1. Understand difference between traditional and agile methodology
CO2. Understand business case change in agile methodology
CO3. Apply planning and budget in agile development
CO4. Acquire skills for working in the team in agile development

Module	Unit	Contents	No of	CO

No.	No.		Hrs.	
1	**Introduction to Agile Project Management**		08	CO1
	1.1	What is Project? What is Project Management? Need for Project Management? Project Phases, Project stake holders, Key general Management Skills.		
	1.2	Traditional Lifecycle vs Agile Lifecycle		
	1.3	Project Process, Process Groups, Process Interactions, Customizing Process Interactions.		
	1.4	Mapping of Project Management Process.		
	1.5	Scaling for Enterprise Agile		
2	**The Agile Business Case**		09	CO2
	2.1	Planning steps, project execution plan, scope and statement of work, work definition.		
	2.2	Business case: An agile business case framework		
	2.3	Project Balance Sheet, Treacy-Wiersema Model		
	2.4	Business Value Models, Building the Business Case by Levels		
	2.5	Quality Values and Principles		
	2.6	Thought Leaders and Agile Quality		
	2.7	Sampling for Quality Validation		
		#Self Learning - Preparation of a business case		
3	**Developing the Scope and Requirements**		08	CO3
	3.1	Planning and scheduling charts		
	3.2	Network diagrams, critical path. Product Roadmap and Release Planning.		
	3.3	Kano charts, Story points, velocity, and burn down charts		
	3.4	Agile estimation techniques		
	3.5	The Black Box, Interfaces, and Connectivity		
	3.6	Change management		
	3.7	Risk management		
		#Self Learning : Application development by using Agile.		
4	**Planning and Scheduling**		10	CO3
	4.1	Planning in the enterprise context		
	4.2	Agile planning portfolio		
	4.3	Cone of uncertainty		
	4.4	Scheduling, Work stream planning		
	4.5	Agile estimates		

	4.6	Drivers on cost and schedule		
	4.7	Estimating process Delphi and Poker		
		#Self Learning : Preparation of cost and schedule for the model.		
5	**Agile Tools and Technologies**		10	CO4
	5.1	Overview of Agile Frameworks: Scrum, Kanban		
	5.2	Agile Project Management Tools (Jira, Trello, Azure DevOps)		
	5.3	Continuous Integration/Delivery (CI/CD) and DevOps Integration.		
	5.4	Agile Testing and Automation		
	5.5	Team management, Virtual Teams		
	5.6	Matrix Management in the Agile Space		
		Total	45	--

Recommended Books:

Sr. No	Name/s of Author/s	Title of Book	Publisher	Edition/Year
1	Project management for Engineers, Business and Technology	John M Nicholas and Herman Steyn	Routledge, Taylor and francis group	Fifth edition
2	Project management the Agile way	John C Goodpasture, PMP	J. Ross publication	USA, Second edition
3	Making Sense of Agile Project Management	Charles G. Cobb, PMP	2011 by John Wiley & Sons, I	First Edition
4	Agile project management A practical guide to using Agile, Scrum and Kanban	Rob Cole and Edward Scotcher	2015, Rob Cole and Edward Scotcher	First Edition
5	Agile for Project Managers	Denise Canty	2015, CRC Press	First Edition
6	Agile planning and estimation	Mike Cohn	Pearson publication	First Edition

*In addition to printed books, faculty can suggest (authentic) urls or e-books, e-contents etc.

Course Code	Name of the Course
216U01E617	Cloud Computing & Virtualization

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	03	--	-	03
Credits Assigned	03	--	-	03

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	-	20	30	50	100

Course pre-requisites:-

1. Fundamental knowledge on Operating system and Computer Networks
2. Basics of client/server programming and network protocols

Course Objectives:

Cloud computing has evolved as a very important computing model, which enables information, software, and other shared resources to be provisioned over the network as services in an on demand manner. Students will be exposed to the current practices in cloud computing. Topics may include distributed computing models and technologies, Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), Software-as-a-Service (SaaS), virtualization, performance and systems issues, capacity planning, federated clouds, challenges in implementing clouds, data centers, hypervisor CPU and memory management, cloud hosted applications, and other advanced and research topics in cloud computing.

Course Outcomes (CO):

CO1. Comprehend the issues related to cloud computing and its application
CO2. Investigate the system virtualization and outline its role in enabling the cloud computing System model
CO3. Analyse and apply cloud programming models to solve problems
CO4. Build cloud services and applications
CO5. Configure and experiment with advanced cloud technologies

Mod	Unit	Contents	No of	CO

ule No.	No.		Hrs.	
1	**Introduction**		8	CO1
	1.1	Cloud Computing at a Glance, Historical Developments, Building Cloud Computing Environments, Computing Platforms and Technologies - Microsoft Azure, Hadoop, Force.com and Salesforce.com, Manjrasoft Aneka		
2	**Virtualization**		10	CO1
	2.1	Introduction, Characteristics of Virtualized Environments , Taxonomy of Virtualization Techniques, Virtualization and Cloud Computing, Pros and Cons of Virtualization CO2		
	2.2	Technology Examples: Xen: Para virtualization, VMware: Full Virtualization, Microsoft Hyper-V		
	2.3	Cloud Computing Architecture : Cloud Reference Model, Types of Clouds, Economics of the Cloud, Open Challenges		
3	**Cloud Infrastructure and Platforms in Industry**		10	CO2
	3.1	Data-Center Design-Warehouse-Scale Data-Center, Modular Data Center. Interconnection Networks-Fat-Tree Interconnection Topology for Warehouse Scale Data Centers, Server-Centric BCube Network for a Modular Data Center.Data-Center Management Issues		
	3.2	Amazon Web Services – Compute Services, Storage Services, Communication Services, Additional Services		
	3.3	Google Cloud Platform, Google AppEngine: Architecture and Core concepts; Application Life Cycle		
4	**Cloud Applications**		8	CO2
	4.1	Scientific Applications – Healthcare: ECG analysis in Cloud, Biology: Protein Structure Prediction, Geoscience: Satellite Image Processing		
	4.2	Business and Consumer Applications – CRM and ERP, Productivity, Social Networking, Media Applications, Multiplayer Online Gaming		
		#Self-Learning – other Applications		
5	**Advanced Topics in Cloud Computing**		9	CO3
	5.1	Energy Efficiency in Clouds, Market Based Management of Clouds, Federated Clouds / Inter Cloud, Third Party		

		Cloud Services: MetaCDN, SpotCloud		
	5.2	Dockers and Containers, Micro Services, Cloud automation tools and DevOps concepts		
		Total	45	--

Recommended Books:

Sr. No.	Name/s of Author/s	Title of Book	Publisher	Edition / Year
1.	Rajkumar Buyya, Christian Vecchiola, S Thamarai Selvi	Mastering Cloud Computing	McGraw Hill Education Private Limited	2nd , 2013
2.	J.Vette, Toby J. Vette, Robert Elsenpeter	Cloud Computing: A Practical Approach	McGraw Hill Education Private Limited	1st , 2009
3.	Rajkumar Buyya, James Broberg, Andrzej Goscinski	Cloud Computing, Principles and Paradigms	Wiley	1st ,2013
4.	George Reese	Cloud Application Architectures: Building Applications and Infrastructure in the Cloud	O'Reilly Publication	1st , 2009

*In addition to printed books, faculty can suggest (authentic) urls or e-books, e-contents etc.

Course Code	Name of the Course
216U01E618	Machine Learning

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	03	--	--	03
Credits Assigned	03	--	--	03

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	--	20	30	50	100

Course pre-requisites: Knowledge of Python Programming and Basic Mathematics - Linear Algebra, Calculus, Probability and Statistics, Matrices.

Course Objectives:

1. Introduce students to the fundamental concepts of machine learning, types of learning, hypothesis space, and inductive bias
2. Enable students to apply regression techniques and optimization methods for model building and evaluation in machine learning.
3. Familiarize students with classification algorithms including decision trees, random forests, Naive Bayes, SVMs, and Bayesian belief networks.
4. Introduce students to unsupervised learning techniques like clustering (K-means, hierarchical, DBSCAN) and dimensionality reduction (PCA, ICA, SVD).
5. Explain reinforcement learning principles and paradigms in semi-supervised and self-supervised learning.

Course Outcomes (CO):

At the end of successful completion of the course the student will be able to

CO1. Understand and classify different types of machine learning and articulate the concept of hypothesis space and inductive bias.

CO2. Demonstrate proficiency in constructing and evaluating regression models to develop predictive models.

CO3. Apply and evaluate different classification models using appropriate evaluation techniques.

CO4. Apply clustering and dimension reduction algorithms to analyze and interpret data

patterns effectively.

CO5. Identify key components of reinforcement learning and differentiate between different paradigms in machine learning.

Module No.	Unit No.	Contents	No of Hrs.	CO
1	**Introduction to Machine Learning**		08	CO1
	1.1	What is Machine Learning? Types of learning, concept of hypothesis space and inductive bias.		
	1.2	Bias variance dichotomy, Evaluation, cross validation, Problem of overfitting, regularization		
	1.3	Feature Engineering: Data Collection, Data Exploration and Profiling, data cleaning for consistent data, Introduction to data pre-processing methods for improving data quality, Data Splitting for training and evaluation sets.		
2	**Supervised Machine learning: Regression**		09	CO2
	2.1	Simple Linear Regression, Multilinear regression, Building model, Model Diagnostic / Evaluation		
	2.2	Logistic Regression, Advanced Regression Techniques		
	2.3	Gradient descent learning algorithm through regression		
	2.4	Hyper-parameter Tuning		
3	**Supervised Machine learning: Classification**		10	CO3
	3.1	Decision Trees, Constructing Decision Trees, Classification and Regression Trees (CART), Random Forest, Ensemble Learning. Model Evaluation		
	3.2	Naive Bayes theorem, Bayes Classifier, Model Evaluation		
	3.3	Support Vector Machines: Maximum Margin Linear Separators, Quadratic Programming solution, Kernels for learning non-linear functions, Model Evaluation		
	3.4	Bayesian Belief networks, Hidden Markov Models. KNN, Model Evaluation		
		#Self-Learning: Applications of HMM, Applications of Bayesian Belief networks		
4	**Un-Supervised Machine learning and Dimension reduction techniques**		10	CO4
	4.1	Dimension reduction using Principle Component Analysis, Independent Component Analysis, Singular value decomposition, PCA for gene expression		
	4.2	K-means clustering, Hierarchical clustering, Agglomerative vs. divisive clustering & Density-based clustering (DBSCAN)		
	4.3	Introduction to Gaussian Mixture Models, Expectation-Maximization (EM) algorithm for clustering.		
	4.4	Radial Basis functions		

		#Self-Learning: Applications of RBF		
5	**Different paradigms in machine learning**		08	CO5
	5.1	What is Reinforcement Learning? Key components: Agent, Environment, Reward, Policy, Value Function.		
	5.2	Introduction to semi-supervised learning, self-supervised learning		
	5.3	Introduction Text processing, plain and simple, Text Classification		
	5.4	Case Study and Applications in Healthcare		
		Total	45	--

Recommended Books

Sr. No	Name/s of Author/s	Title of Book	Publisher	Edition/Year
1	Tom M. Mitchell	Machine Learning	McGraw Hill	2017
2	M. Gopal	Applied Machine Learning	McGraw Hill	2021, 2nd edition
3	Aurélien Géron	Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, 2nd Edition	O'Reilly	2019
4	Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani	An Introduction to Statistical Learning With Applications in R	Springer US	2021
5	Ian Goodfellow, Yoshua Bengio, Aaron Courville	Deep Learning	An MIT Press book	2016

Course Code	Name of the Course
216U01E619	Introduction to Biotechnology & Bioinformatics

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	03	--	--	03
Credits Assigned	03	--	--	03

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	--	20	30	50	100

Course prerequisites: Biology till 10th standard

Course Objectives:

This course aims to make student aware of the concept of Microbes and their potentials. It helps them to get familiarize with the fundamental make up of cells and the central dogma of life and make learners aware of the structure and properties of nucleic acids.
Further it provides core concept understanding with grasping the fundamental principles of bioinformatics, including its history, scope, and essential data formats. It supports students to explore Biological Databases with learning to navigate and utilize various primary, secondary, and specialized biological databases effectively

Course Outcomes (CO):

At the end of successful completion of the course the student will be able to
CO1: Identify and differentiate between major groups of microorganisms including bacteria, fungi, protists, and viruses and explain the concept of sterilization and disinfection
CO2: Examine the applications of microorganisms in different sectors of biotechnology.
CO3: Explain the structure and function of the cellular components in prokaryotic cell.
CO4: Illustrate structure and properties of DNA & RNA
CO5: Describe the fundamental principles of bioinformatics, including its history, scope, and essential data formats.
CO6: Explore Biological Databases to navigate and utilize various primary, secondary, and specialized biological databases effectively.

Module No.	Unit No.	Details	Hrs.	CO
1	**Microbiology I**		06	CO1
	1.1	Introduction to the concept of Biotechnology-Colors of Biotechnology		
	1.2	Introduction to the microbial world- Bacteria, Fungi, Protists, and Viruses		
	1.3	Microbial growth and control- cultivation of microorganisms, Sterilization, and disinfection		
	1.4	Effect of microbes on human health		
2	**Microbiology II**		06	CO2
	2.1	Applications of Microbes in commercial sectors – Food, Dairy, Beverages, and pharmaceuticals(Two examples each		
	2.2	Diagnostic Microbiology		
	2.3	Role of microorganisms in remediation of solid and liquid wastes-Bioremediation, Bioaugmentation, Phytoremediation		
	2.4	Wastewater treatment (Industrial wastewater and sewage treatment.)2L		
3	**Cell Biology- Prokaryotic Cell Biology**		07	CO3
	3.1	Cell Membrane: Structure and function of the prokaryotic cell membrane.		
	3.2	Cell Wall: Composition and differences between Gram-positive and Gram-negative bacteria.		
	3.3	Capsule and Slime Layer: Structure and roles in protection and pathogenicity.		
	3.5	Cytoplasm: Components and functions within the prokaryotic cytoplasm. Nucleoid: Organization and function of the prokaryotic chromosome.		
	3.6	Ribosomes: Structure and role in protein synthesis. Inclusion Bodies: Types and functions of storage granules and gas vesicles.		
	3.7	Flagella and Pili: Structure, function, and role in motility and conjugation.		
4	**Basic concepts of DNA & RNA**		08	CO4
	4.1	Introduction to central dogma of Molecular Biology, Features of genetic material,		
	4.2	Structure of DNA, properties of DNA, Types of DNA.		
	4.3	Plasmids- introduction, features, and functions. Structure of RNA, Properties of RNA		
	4.4	Types of RNA, Differences between DNA and RNA Functions of RNA.		

	4.5	Electrophoretic separation of nucleic acids.		
5	**Fundamentals of Bioinformatics – I**		**09**	CO5
	5.1	Introduction to Bioinformatics Overview of Bioinformatics: Definition, history, and scope. Human genome project and Biological data		
	5.2	Data Formats: FASTA, GenBank, and other common formats		
	5.3	Database file structures and Database Management Systems: Flat file, Relational, Object oriented		
	5.4	Biological Databases: Primary Databases: NCBI, EMBL, DDBJ Secondary Databases: UniProt, InterPro Specialized Databases: PDB, OMIM Database Examples and Usage: Practical examples and navigating biological databases. Introduction to Bioconductor package in R Biopython modules		
6	**Fundamentals of Bioinformatics – II**		**09**	**CO6**
	6.1	Sequence Analysis in bioinformatics: Understanding variation in biological organisms: Sources of variation: mutation/recombination. Concept of microevolution, synonymous and nonsynonymous mutations.		
	6.2	Sequence Alignment: Concepts of pairwise and multiple sequence alignment.		
	6.3	Dynamic Programming Algorithms: Needleman-Wunsch and Smith-Waterman algorithms.		
	6.4	Heuristic Methods: Introduction to BLAST and FASTA.BLAST and FASTA: Detailed usage and applications.		
	6.5	Multiple sequence alignment; algorithms, Progressive (CLUSTAL W) and Iterative algorithm.		
	6.6	Phylogenetics: Basics of phylogenetic tree construction. UPGMA, Neighbour Joining, Maximum parsimony		
		Total	**45**	

Recommended Books:

Sr. No.	Name/s of	Title of Book	Name of Publisher with	Edition and

	Author/s		country	Year of Publication
1.	*C B Powar, H F Daginawala*	General Microbiology, Vol. I and II	Himalaya Publishing House	Vol-I and II
2.	*Dr. P S Verma & Dr. V K Agarwal*	Cell Biology, Genetics, Molecular Biology, Evolution and Ecology (Library Editions)	S Chand And Company Ltd	July 2022 .
3.	*Xiong, J.*	Essential bioinformatics	Cambridge University Press	First Edition 2006
4.	*Strachan and Read*	Human molecular genetics	Garland Science	Fifth Edition, 2011
5.	*Pelczar Jr MJ, Chan ECS, and Krieg NR.*	Microbiology	Tata McGraw-Hill	5th edition 2004
6.	*Willey JM, Sherwood LM, and Woolverton CJ.*	Prescott's Microbiology	. McGraw-Hill Higher Education.	. 9th edition 2013

Course Code	Name of the Course
216U01L601	**Digital Signal & Image Processing Lab**

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	--	02	--	02
Credits Assigned	--	01	--	01

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	50	--	--	--	50

Course prerequisites: Basic mathematical background of transforms, matrices, complex numbers and programming skills.

Course Objectives:

1. Understand the fundamental concepts of discrete-time signals, systems, and operations on signals.
2. Learn spatial and frequency domain enhancement techniques for processing 1-D and 2-D digital signals.
3. Explore various image transforms and their applications for analyzing digital signals in the frequency domain.
4. Examine segmentation, boundary detection, and compression techniques for representing and compressing images.
5. Develop practical skills in designing and implementing digital signal and image processing applications using appropriate software tools.

Course Outcomes (CO):

CO1. Interpret fundamentals of discrete time signals and systems and signal manipulation methods
CO2. Apply various spatial and frequency domain enhancement techniques for 1-D and 2-D signals
CO3. Analyze signals in frequency domain using various image transforms.
CO4. Evaluate extracted analyzed information for synthesis of digital signals.
CO5. Design and develop applications based on 1-D and 2-D digital signals.

Laboratory experiments covering the entire syllabus of the course 216U01C601, 'Digital Signal & Image Processing'. Students will be graded based on continuous assessment during laboratory work.

Course Code	Name of the Course
116U01L602	Network & Information Security Lab

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	--	02	--	02
Credits Assigned	--	01	--	01

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	50	--	--	--	50

Course pre-requisites:
Basics of Operating System and Computer Network.

Course Objectives:
1. To understand fundamental principles of information security.
To gain knowledge on malicious & non-malicious programme errors & apply counter-measures.
3. To understand the various web attacks and its preventive mechanisms.
4. To apply different techniques to secure data in transit across data networks.
5. To study security laws and analyse the ethical issues.

Course Outcomes (CO):

At the end of successful completion of the course the student will be able to

CO1: Understand various security goals, threats, vulnerabilities and controls.
CO2: Apply various cryptographic algorithms for software security.
CO3: Identify, analyse and mitigate web attacks.
CO4: Apply network security mechanisms for threat management.
CO5: Understand investigation process, legal and ethical issues in security.

Laboratory experiments covering the entire syllabus of the course 116U01C602, 'Network & Information Security'. Students will be graded based on continuous assessment during laboratory work.

Course Code	Name of the Course
116U01L603	Artificial Intelligence Lab

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	--	02	--	02
Credits Assigned	--	01	--	01

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	50	--	--	--	50

Course prerequisites:
Basics of Operating System and Computer Network

Course Objectives:

1. To introduce the history, core concepts, and sub-areas of artificial intelligence, including current trends and ethical implications.
2. To explore the structure and functionality of intelligent agents, focusing on their environments, rationality, and problem-solving capabilities through search algorithms.
3. To understand knowledge representation and reasoning techniques, including first-order logic and uncertain reasoning using Bayesian networks.
4. To cover the basics of machine learning types, model evaluation, and practical applications through case studies in supervised and unsupervised learning.

Course Outcomes (CO):

At the end of successful completion of the course the student will be able to

CO1	Design AI solution with appropriate choice of agent architecture
CO2	Analyse and solve problems for goal based agent architecture
CO3	Represent and formulate the knowledge to solve the problems using various reasoning techniques
CO4	Understand and differentiate between various types of machine learning

Laboratory experiments covering the entire syllabus of the course 116U01C603, 'Artificial Intelligence'. Students will be graded based on continuous assessment during laboratory work.

Course Code	Name of the Course
216U01L611	Compiler Construction Lab

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	--	02	--	02
Credits Assigned	--	01	--	01

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	50	--	--	--	50

***Term Work will consist of Practical covering the entire syllabus of compiler construction. Students will be graded based on continuous assessment of their term work.**

Course prerequisites (if any):

Finite automata, pushdown automata etc. from Theory of Computer science.

Course Objectives

The course aims to give knowledge of the principal structure of a compiler and about the basic theories and methods used to implement the different phases of the compiler.

Course Outcomes

At the end of successful completion of the course the student will be able to

CO1	Study phases of compiler and illustrate different parsing techniques and semantic analysis.
CO2	Illustrate and analyze the different intermediate code generation techniques and run time storage allocation.
CO3	Apply optimization techniques
CO4	Analyze and interpret the different issues in code generation phase

Laboratory experiments covering the entire syllabus of the course 216U01E611, 'Compiler Construction'. Students will be graded based on continuous assessment during laboratory work.

Course Code	Name of the Course

216U01L612	Virtual Reality & Augmented Reality Lab

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	--	02	--	02
Credits Assigned	--	01	--	01

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	50	--	--	--	50

Course pre-requisites:
No programming experience is required. Knowledge about computer graphics can help in understanding real time 3D concept which means understanding coordinate systems and transforms. Information about OpenGL (a ubiquitous graphics API) and Fundamentals of C# programming can also be helpful.

Course Objectives:
1. Understand fundamental principles of information security.
2. Relate the ergonomics related to immersive technologies.
3. Use and comprehend VR Hardware and Software Components for AR-VR.
4. Analyse socio-legal aspects of immersive technologies.
5. Able to use tools and develop applications in the domain of AR-VR.

Course Outcomes (CO):
CO1. Understand Virtual, Mixed and Augmented Reality platforms and its associated technologies.
CO2. Identify ergonomic considerations in the design and use of AR/VR systems.
CO3. Understand Human Factors, Legal, and Social Considerations.
CO4. Use the tools and technologies used by professionals working in AR & VR.

Laboratory experiments covering the entire syllabus of the course 216U01E612, 'Virtual Reality & Augmented Reality'. Students will be graded based on continuous assessment during laboratory work.

Course Code	Name of the Course
216U01L613	Embedded System and IOT Lab

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	--	02	--	02
Credits Assigned	--	01	--	01

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	50	--	--	--	50

Course prerequisites: Operating system, Data Networks/ Computer Networks

Course Objectives: To learn basic architecture of Wireless sensor networks and Internet of Things and understand WSN routing protocols and evaluate software, hardware platforms for IoT technology. Also create applications using IOT analytics.

Course Outcomes (CO):
At the end of successful completion of the course the student will be able to

CO1	Explain the basic architecture and working principle of wireless sensor networks and Internet of Things
CO2	Identify challenges and issues in WSN routing and suggest solutions.
CO3	To use different Operating system for Wireless sensor networks and IoT
CO4	Evaluate the software and hardware platforms for IoT Technologies and design small IoT applications.
CO5	Create IoT application data using IoT Analytics.

Laboratory experiments covering the entire syllabus of the course 216U01E613, 'Embedded System and IOT'. Students will be graded based on continuous assessment during laboratory work.

Course Code	Name of the Course
216U01L614	Mobile Communication and Mobile App Development Lab

	TH	P	TUT	Total
Teaching Scheme (Hrs./Week)	--	02	--	02
Credits Assigned	--	01	--	01

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	50	--	--	--	50

Course pre-requisites:
Basic Knowledge of Computer Networks, Layered Architecture, Structure and working related Protocols.

Course Objectives:

3. To provide an overview of Mobile & Cellular Communication networks area and its applications in communication engineering.
4. To understand the various terminology, principles, concepts, Standards, algorithms and different methodologies used in Wireless Communication Networks specifically for Wireless Ad-Hoc Networks

Course Outcomes (CO):

CO1. Explain the basic concepts of various wireless networks and their working characteristics with respect to mobile network generations such as 2G, 3G and beyond.
CO2. Compare infrastructure based and Ad hoc networks, elaborating characteristics and features of Ad hoc Networks
CO3. Inspect designing of Wireless MAC protocols for Ad hoc networks; and the working principle of different WLAN IEEE standards.
CO4. Describe various Network Layer & Transport layer mechanisms and Routing Protocols for Wireless networks.
CO5. Explain various features and operations of Application Protocols of wireless Ad-hoc and Mesh Networks like sensor networks, VANETs etc.

Laboratory experiments covering the entire syllabus of the course 216U01E614, 'Mobile Communication and Mobile App Development'. Students will be graded based on continuous assessment during laboratory work.

Course Code	Name of the Course
216U01L615	Microservices Foundations Lab

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	--	02	--	02
Credits Assigned	--	01	--	01

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	50	--	--	--	50

Course pre-requisites:
Practical knowledge of Java

Course Objectives:
Microservices Foundations is an advance and modern subject which touches on many aspects of Microservices which are necessary to know before you start your micro-service journey or enhance your micro-service journey. This subject is designed as a deep dive into what is micro-service, how you think about it, and build, test, deploy and, breaking apart existing systems into a micro-service architecture. This subject creates your foundation for building Microservices using Java and Spring Boot which are the most popular programming language and framework used in the IT industry.

Course Outcomes (CO):
CO1. Understand the Fundamentals of Microservices
CO2. Analyze the Principles and Characteristics of Microservices
CO3. Design and Model Microservices Applications
CO4. Build Microservices Using Java and Spring Boot
CO5. Test and Document Microservices

Laboratory experiments covering the entire syllabus of the course 216U01E615, 'Microservices Foundations'. Students will be graded based on continuous assessment during laboratory work.

Course Code	Name of the Course
216U01L616	Agile Project Management Lab

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	--	02	--	02
Credits Assigned	--	01	--	01

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	50	--	--	--	50

Course pre-requisites:
Software Engineering

Course Objectives:
This course includes agile methodology, in different steps of project management like lifecycle, planning scheduling estimates etc.

Course Outcomes (CO):
CO1. Understand difference between traditional and agile methodology
CO2. Understand business case change in agile methodology
CO3. Apply planning and budget in agile development
CO4. Acquire skills for working in the team in agile development

Laboratory experiments covering the entire syllabus of the course 216U01E616, 'Agile Project Management'. Students will be graded based on continuous assessment during laboratory work.

Course Code	Name of the Course
216U01L617	Cloud Computing & Virtualization Lab

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	--	02	-	02
Credits Assigned	--	01	-	01

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	50	--	--	--	50

Course pre-requisites:-

3. Fundamental knowledge on Operating system and Computer Networks
4. Basics of client/server programming and network protocols

Course Objectives:

Cloud computing has evolved as a very important computing model, which enables information, software, and other shared resources to be provisioned over the network as services in an on demand manner. Students will be exposed to the current practices in cloud computing. Topics may include distributed computing models and technologies, Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), Software-as-a-Service (SaaS), virtualization, performance and systems issues, capacity planning, federated clouds, challenges in implementing clouds, data centers, hypervisor CPU and memory management, cloud hosted applications, and other advanced and research topics in cloud computing.

Course Outcomes (CO):

CO1. Comprehend the issues related to cloud computing and its application
CO2. Investigate the system virtualization and outline its role in enabling the cloud computing System model
CO3. Analyse and apply cloud programming models to solve problems
CO4. Build cloud services and applications
CO5. Configure and experiment with advanced cloud technologies

Laboratory experiments covering the entire syllabus of the course 216U01E617, 'Cloud Computing & Virtualization'. Students will be graded based on continuous assessment during laboratory work.

Course Code	Name of the Course			
216U01L618	Machine Learning Lab			
Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	--	02	--	02
Credits Assigned	--	01	--	01

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	50	--	--	--	50

Course pre-requisites (if any):

Knowledge of Python Programming and Basic Mathematics - Linear Algebra, Calculus, Probability and Statistics, Matrices.

Course Objectives:

- Introduce students to the fundamental concepts of machine learning, types of learning, hypothesis space, and inductive bias
- Enable students to apply regression techniques and optimization methods for model building and evaluation in machine learning.
- Familiarize students with classification algorithms including decision trees, random forests, Naive Bayes, SVMs, and Bayesian belief networks.
- Introduce students to unsupervised learning techniques like clustering (K-means, hierarchical, DBSCAN) and dimensionality reduction (PCA, ICA, SVD).
- Explain reinforcement learning principles and paradigms in semi-supervised and self-supervised learning.

Course Outcomes (CO):

At the end of successful completion of the course the student will be able to

- Understand and classify different types of machine learning and articulate the concept of hypothesis space and inductive bias.
- Demonstrate proficiency in constructing and evaluating regression models to develop predictive models.
- Apply and evaluate different classification models using appropriate evaluation

techniques.

- Apply clustering and dimension reduction algorithms to analyze and interpret data patterns effectively.
- Identify key components of reinforcement learning and differentiate between different paradigms in machine learning.

Laboratory will consist of experiments/ tutorials covering entire syllabus of the course 216U01E618, 'Machine Learning'. Students will be graded based on continuous assessment of laboratory work

Course Code	Name of the Course

216U01L619	Introduction to Biotechnology & Bioinformatics

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	--	02	--	02
Credits Assigned	--	01	--	01

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	50	--	--	--	50

Course pre-requisites (if any):
Biology till 10th standard

Course Objectives:
This course aims to make students aware of the concept of Microbes and their potential. It helps them to get familiarize with the fundamental makeup of cells and the central dogma of life and make learners aware of the structure and properties of nucleic acids.
Further it provides core concept understanding with grasping the fundamental principles of bioinformatics, including its history, scope, and essential data formats. It supports students to explore Biological Databases with learning to navigate and utilize various primary, secondary, and specialized biological databases effectively.

Course Outcomes (CO):

At the end of successful completion of the course the student will be able to
CO1: Identify and differentiate between major groups of microorganisms including bacteria, fungi, protists, and viruses and explain the concept of sterilization and disinfection
CO2: Examine the applications of microorganisms in different sectors of biotechnology.
CO3: Explain the structure and function of the cellular components in prokaryotic cell.
CO4: Illustrate structure and properties of DNA & RNA
CO5: Describe the fundamental principles of bioinformatics, including its history, scope, and essential data formats.
CO6: Explore Biological Databases to navigate and utilize various primary, secondary, and specialized biological databases effectively.

Laboratory will consist of experiments/ tutorials covering entire syllabus of the course 216U01E619, 'Introduction to Biotechnology & Bioinformatics'. Students will be graded based on continuous assessment of laboratory work.

Course Code	Name of the Course
216U01P601	Project 1

Teaching Scheme (Hrs./Week)	TH	P	TUT	Total
	–	04	–	04
Credits Assigned	–	01	01	02

Evaluation Scheme	Marks				
	LAB/TUT CA	CA (TH)		ESE	Total
		IA	ISE		
	50	-	-	-	50

Course pre-requisites:

Background of concepts of algorithms, database management systems, hardware and concepts in Artificial Intelligence and Machine learning. Development of Software Systems and knowledge of basic courses in Computer Engineering will be beneficial.

Course Objectives:

Objectives are:

- Develop a deep understanding of subject matter
- Develop critical thinking and problem-solving skills
- Work collaboratively with peers
- Build self-directed learning skills
- Engage with real-world problems and issues
- Develop an appreciation for the importance of lifelong learning.

Course Outcomes (CO):

After successful completion of the course the student will be able to :

CO1. Understand Project Life Cycle
CO2. Identify and formulate problem definition of real world Problems.
CO3. Design hardware and software components and systems to meet specifications of identified problem.
CO4. Apply project management principles to plan the execution of entire project work.
CO5. Use hardware and software tools for efficient implementation of project work.

The students have experienced project in Mini Project course in the previous semesters. For Project-1 in Semester VI, a group of maximum four students will be completing a comprehensive project work. Project work may be internally assigned or may be externally

assigned by the research institutes, industry etc. Each group will be assigned one faculty advisor / guide. This project work in final year may be extension of the Mini Project work done in pre-final year. The students may choose their problem statements in consultation with faculty members. The main intention of Project work is to enable students to apply the knowledge and skills learned out of courses studied to solve/implement predefined practical problem. The Project work may be beyond the scope of curriculum of courses taken or may be based on the courses but thrust should be

- Learning additional skills
- Ability to define, design, analyze and implement the solution to problems
- Learn the behavioral science by working in a group
- The project area may be selected in which the student intends to do further education and/or may be either intend to have employment or self-employment
- The topic of project should be different and / or may be advancement in the same topic of Mini Project
- The students may use this opportunity to learn different computational techniques / software tools / platforms etc. as well as some model development.

Students are expected to report to the faculty advisor about the progress of the work. A continuous assessment record of the progress of the project will be maintained by concerned faculty members The TW will be examined by approved internal faculty appointed by the head of the institute. A Report in prescribed format must be submitted. The oral examination is a presentation by the group members on the project along with demonstration of the work done. In the examination each individual student will be assessed for his/her contribution, understanding and knowledge gained. There will also be an intermediate evaluation at the mid of the semester by a panel of examiners.