BE SCHEME & SYLLABUS

Third Year
(V and VI Semester)

With effect from 2022-23

Computer Science and Engineering

ST JOSEPH ENGINEERING COLLEGE

AN AUTONOMOUS INSTITUTION

Vamanjoor, Mangaluru - 575028

MOTTO

Service & Excellence

VISION

To be a global premier Institution of professional education and research.

MISSION

- Provide opportunities to deserving students of all communities, the Christian students in particular for quality professional education.
- Design and deliver curricula to meet the national and global changing needs through student-centric learning methodologies.
- Attract, nurture and retain the best faculty and technical manpower.
- Consolidate the state-of-art infrastructure and equipment for teaching and research activities.
- Promote all round personality development of the students through interaction with alumni, academia and industry.
- Strengthen the Educational Social Responsibilities (ESR) of the institution.

ST JOSEPH ENGINEERING COLLEGE

An Autonomous Institution
Vamanjoor, Mangaluru - 575028

Affiliated to VTU – Belagavi & Recognized by AICTE New Delhi
NBA – Accredited: B.E.(CSE,ECE,EEE, ME and CIV) & PG (MBA and MCA)
NAAC – Accredited with A+

B.E. SCHEME & SYLLABUS
(With effect from 2022-23)

Computer Science and Engineering

THIRD YEAR

(V and VI Semester)

AUTONOMY AND ACCREDITATION

St Joseph Engineering College (SJEC) is an Autonomous Institute under Visvesvaraya Technological University (VTU), Belagavi, Karnataka State, and is recognized by the All-India Council for Technical Education (AICTE), New Delhi. SJEC is registered under the trust "Diocese of Mangalore, Social Action Department".

The SJEC has been conferred Fresh Autonomous Status from the Academic Year 2021-22. The college was granted autonomy by the University Grants Commission (UGC) under the UGC Scheme for Autonomous Colleges 2018 and conferred by VTU. The UGC Expert Team had visited the college on 28-29 November 2021 and rigorously assessed the college on multiple parameters. The fact that only a handful of engineering colleges in the state have attained Autonomous Status adds to the college's credibility that has been on a constant upswing. Autonomy will make it convenient for the college to design curricula by recognizing the needs of the industry, offering elective courses of choice and conducting the continuous assessment of its students.

At SJEC, the Outcome-Based Education (OBE) system has been implemented since 2011. Owing to OBE practised at the college, SJEC has already been accredited by the National Board of Accreditation (NBA). Five of the UG programs, namely Computer Science & Engineering, Mechanical Engineering, Electronics and Communication Engineering, Electrical & Electronics Engineering and Civil Engineering and two of the PG programs, namely MBA and MCA programs, have accreditation from the NBA.

Also, SJEC has been awarded the prestigious A+ grade by the National Assessment and Accreditation Council (NAAC) for five years. With a Cumulative Grade Point Average (CGPA) of 3.39 on a 4-point scale, SJEC has joined the elite list of colleges accredited with an A+ grade by NAAC in its first cycle. The fact that only a small percentage of the Higher Education Institutions in India have bagged A+ or higher grades by NAAC adds to the college's credibility that has been on a constant upswing.

The college is committed to offering quality education to all its students, and the accreditation by NAAC and NBA reassures this fact. True to its motto of "Service and Excellence", the college's hard work has resulted in getting this recognition, which has endorsed the academic framework and policies that the college has been practicing since its inception. The college has been leveraging a flexible choice-based academic model that gives students the freedom to undergo learning in respective disciplines and a transparent and continuous evaluation process that helps in their holistic development.

CONTENTS

ABOUT THE DEPARTMENT

The Department of Computer Science and Engineering was setup during the inception of the college in 2002. The primary objective of this program is to prepare students for successful careers in Computer and Information technology industry that meet the needs of Indian and multinational organizations. The Department started Research program in the year 2011. Currently, the student intake is 180 at the UG level. The program involves wide variety of courses which enable the students to formulate, solve and analyze computer engineering problems, prepare them for graduate studies and develop the ability to synthesize data and technical concepts for application design & implementation of real time software products. The faculty of the Department are actively involved in teaching and research with specializations in Cloud Computing, Image Processing, Process Mining, Natural Language Processing and Soft & Evolutionary Computing. The Department was accredited by NBA for 2 years from June 2013 to May 2015, for six years from July 2016 to June 2022 and is reaccredited by NBA for three years from July 2022 to June 2025. The Department has received a grant of Rs. 19 lakhs from AICTE for setting up Center of Excellence in Augmented Reality and Virtual Reality (AR/VR) under MODROBS 2020-21 scheme.

DEPARTMENT VISION

- To be recognized as a centre of excellence in computer and allied areas with quality learning and research environment.

DEPARTMENT MISSION

- Prepare competent professionals in the field of computer and allied fields enriched with ethical values.
- Contribute to the socio-economic development of the country by imparting quality education in Computer and Information Technology.
- Enhance employability through skill development.

PROGRAM EDUCATIONAL OBJECTIVES (PEOs)

PEO 1: Apply Engineering Fundamentals: To impart to students a sound foundation and ability to Apply engineering fundamentals, mathematics, science and humanities necessary to formulate, analyze, design and implement engineering problems in the field of computer science.

PEO 2: Work in CS and allied fields: To develop in students the knowledge of fundamentals of computer science and engineering to work in various related fields such as network, data, web and system engineering.

PEO 3: Teamwork: To develop in students the ability to work as a part of team through effective communication on multidisciplinary projects.

PEO 4: Successful Career: To train students to have successful careers in computer and information technology industry that meets the needs of society enriched with professional ethics.

PEO 5: Higher Education: To develop in students the ability to pursue higher education and engage in research through continuous learning.

PROGRAM OUTCOMES (POs)

Engineering Graduates will be able to:

1. Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and engineering specialization to the solution of complex engineering problems.
2. Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
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, and environmental considerations.
4. Conduct investigations on complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and the synthesis of information to provide valid conclusions.
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.
6. The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
7. Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
8. Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
9. Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
10. Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
11. Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and a leader in a team, to manage projects and in multidisciplinary environments.
12. Life-long learning: Recognize the need for and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

PROGRAM SPECIFIC OUTCOMES (PSOs)

Graduates of the Computer Science and Engineering program are able to

PSO 1. Understand the principles underlying entrepreneurship, freelancing and the requirements to initiate a start-up in the IT or related domains.

PSO 2. Participate effectively in competitive examinations for career growth, higher studies and to pursue research.

V Semester (B.E. – Computer Science and Engineering)													
SI. No.	**Course and Course Code**		**Course Title**	**Teaching Department**	**Paper Setting Board**	**Teaching Hours/Week**			**Examination**				**Credits**
						Theory Lecture	**Tutorial**	**Practical/ Drawing**	**Duration in hours**	**CIE Marks**	**SEE Marks**	**Total Marks**	
						L	**T**	**P**					
1	HSMC	22CSE51	Software Engineering and Project Management	CSE	CSE	3	-	-	03	50	50	100	3
2	IPCC	22CSE52	Computer Networks (Integrated)	CSE	CSE	3	-	2	03	50	50	100	4
3	IPCC	22CSE53	Full Stack Development (Integrated)	CSE	CSE	3	-	2	03	50	50	100	4
4	PCC	22CSE54	Automata Theory and Computability	CSE	CSE	2	2	-	03	50	50	100	3
5	PCCL	22CSE55L	Mobile Application Development Laboratory	CSE	CSE	-	-	2	03	50	50	100	1
6	PEC	22CSE56X	Professional Elective - I	CSE	CSE	3	-	-	03	50	50	100	3
7	AEC/ SDC	22RMI57	Research Methodology and Intellectual Property Rights	CSE	CSE	2	-	-	03	50	50	100	2
8	AEC/ SDC	22ETP58	Emerging Technologies: A Primer	COM	COM	-	-	2	03	100	-	100	1
					Total	**16**	**2**	**8**	**24**	**450**	**350**	**800**	**21**

22CSE56X : Professional Elective I			
22CSE561	Big Data and Analytics	22CSE563	NoSQL Databases
22CSE562	Blockchain Technology	22CSE564	Salesforce Developer

VI Semester (B.E. - Computer Science and Engineering)													
SI. No.	**Course and Course Code**			**Teaching Department**	**Paper Setting Board**	**Teaching Hours/Week**			**Examination**				**Credits**
						Theory Lecture	**Tutorial**	**Practical/ Drawing**	**Duration in hours**	**CIE Marks**	**SEE Marks**	**Total Marks**	
						L	**T**	**P**					
1	IPCC	22CSE61	System Software & Compiler Design (Integrated)	CSE	CSE	2	2	2	03	50	50	100	4
2	IPCC	22CSE62	Machine Learning (Integrated)	CSE	CSE	3	-	2	03	50	50	100	4
3	PCC	22CSE63	UI/UX Design	CSE	CSE	3	-	-	03	50	50	100	3
4	PEC	22CSE64X	Professional Elective -II	CSE	CSE	3	-	-	03	50	50	100	3
5	OEC	22CSE65X	Open Elective -I	CSE	CSE	3	-	-	03	50	50	100	3
6	PRJ	22CSE66	Major Project Phase - I	CSE	CSE	-	-	4	03	100	-	100	2
7	HSMC	22CIV67	Environmental Studies	CIV	CIV	1	-	-	02	50	50	100	1
8	AEC/SDC	22IIP68	Innovation and Intellectual Property	COM	COM	-	-	2	03	100	-	100	1
Total						**15**	**2**	**10**	**23**	**500**	**300**	**800**	**21**

22CSE64X : Professional Elective II			
22CSE641	Robotic Process Automation	22CSE643	Go Programming
22CSE642	Agile and Scrum Technology	22CSE644	MuleSoft Anypoint Platform Fundamentals

22CSE65X : Open Elective I			
22CSE651	Introduction to Cyber Security	22CSE653	Python Programming
22CSE652	Object Oriented Programming with Java	-	-

V Semester

Software Engineering and Project Management			
Course Code	**22CSE51**	CIE Marks	50
Course Type (Theory/Practical/Integrated)	Theory	SEE Marks	50
		Total Marks	100
Teaching Hours/Week (L:T:P)	3:0:0	SEE	3 Hours
Total Hours	40 Hours	Credits	03

Course Learning Objectives: The objective of the course is to

- Outline software engineering principles and activities involved in building large software programs
- Describe the process of requirements gathering, requirements classification, requirements specification and requirements validation
- Analyze, design, implement, verify, validate, implement, apply, and maintain software systems or parts of software systems
- Describe the intricacies involved in software evolution and the need for software testing.

Module-1 Phases and Life Cycle Models of Software Development (8 hours)

Introduction: Professional software development; Software engineering: ethics and case studies **Software processes**: Software processes Models; The waterfall model, Incremental development, Integration and configuration.
Applications: Identifying phases and Life cycle models of Software Development
Text book 1: Chapter 1: 1.1-1.3 Chapter 2: 2.1

Module-2 Software Process activities, Agile software Development and Requirements Analysis (8 hours)

Process activities: Software specification, Software design and implementation, Software validation, Software evolution.
Agile Software development: Agile methods, Agile development techniques, Extreme Programming, Agile project Management
Software requirements: Functional and non-functional requirements; Requirements engineering Process, Requirement elicitation, Requirement specification, Requirement Validation and management
Applications: Systematically identifying the requirements and using agile method for software development
Text book 1: Chapter 2: 2.2 Chapter 3: 3.1-3.3, Chapter 4: 4.1-4.6

Module-3 System Modeling (8 hours)

System models: Context models; Interaction models; sequence diagram Structured methods; class diagram, generalization, Aggregation. Behavioral Models-Data driven modeling; Event driven modeling, Model driven engineering
Applications: Concept learning to develop system models based on software requirements.
Text book 1: Chapter 5: 5.1-5.5

Module-4 Software Design and Testing (8 hours)

Software Design: Object-oriented design using the UML, Design Patterns, Implementation Issues.
Software testing: Development testing, Test-driven development, Release testing, User testing.
Applications: Selecting the right method for testing software and maintaining the system.
Text book 1: Chapter 7: 7.1-7.3 Chapter 8: 8.1-8.4

Module-5 Evolution of Software and Project Management (8 hours)

Software evolution: Evolution processes, Legacy system management.
Project management: Risk management, Project planning: Software pricing, Plan-driven development, Quality management: Software quality, Software standards.
Applications: Managing a software by analyzing the risk involved and maintaining its quality while undergoing a change.

Text book 1: Chapter 9: 9.1-9.2 Chapter 22: 22.1 Chapter 23: 23.1-23.2 Chapter 24: 24.1-24.2
Additional Resources: Software development tools: Jira, Eclipse IDE, GitHub, Docker.

Course Outcomes: At the end of the course the student will be able to:	
22CSE51.1	Describe software process models and apply them in software development scenarios.
22CSE51.2	Describe software process activities and develop a plan for requirement engineering for agile method of software development
22CSE51.3	Apply object orientation and modeling constructs to design modeling diagrams for software systems.
22CSE51.4	Differentiate between various software testing methods and select the right method for testing a software.
22CSE51.5	Apply the principals involved in software evolution while maintaining software and describe the processes involved in project planning and quality management.
22CSE51.6	Function effectively in teams to develop software specification document, system models, and test cases while implementing a systematic approach to problem solving.

Sl. No.	**Title of the Book**	**Name of the Author/s**	**Name of the Publisher**	**Edition and Year**
Textbooks				
1	Software Engineering	Ian Somerville	Pearson Education Ltd	10th Edition, 2017
Reference Books				
1	Software Engineering: A Precise Approach	Pankaj Jalote	Wiley	3rd Edition, 2010
2	Object Oriented Modelling and design with UML	Michael Blaha, James Rumbaugh	Pearson Education	2nd Edition, 2007
3	Software Engineering- A Practitioners approach	Roger S. Pressman	Tata McGraw Hill	9th Edition, 2023

Web links and Video Lectures (e-Resources):

- **Software Engineering Essentials:** **https://www.edx.org/course/software-engineering-essentials**
- **Basics to Advanced fundamentals of SE:** **https://www.udemy.com/the-complete-software-engineering-from-basics-to-advanced/**
- **Agile Software Development:** **https://www.udemy.com/fundamentals-of-agile-software-development/**
- **Scrum Master Tales:** **https://www.udemy.com/scrumtalesstoriesfromascrummastersdiary/**
- **Test your agile and scrum knowledge:** https://www.udemy.com/test-your-agile-and-scrum-knowledge/?couponCode=DISCUDEMY.COM

Course Articulation Matrix

Course Outcomes (COs)	Program Outcomes (POs)													
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
22CSE51.1	3		1				1							1
22CSE51.2	3			1			1							
22CSE51.3		2	1	2										1
22CSE51.4		2	1	2	1				1	1				1
22CSE51.5			1	2	1									1
22CSE51.6			1					2	2	3	1			

1: Low 2: Medium 3: High

Computer Networks			
Course Code	**22CSE52**	CIE Marks	50
Course Type (Theory/Practical/Integrated)	Integrated	SEE Marks	50
		Total Marks	100
Teaching Hours/Week (L:T:P)	3:0:2	SEE	3 Hours
Total Hours	40 hours Theory + 10 Lab slots	Credits	04

Course Learning Objectives: The objective of the course is to

- Develop knowledge of network elements and associated network layer protocols and uses of computer networks.
- Analyze various network layers' design issues and its quality-of-service requirements.
- Apply the knowledge of computer networking to develop network applications.
- Develop the knowledge of computer networking modern tools and technological advancements.

Module-1 Introduction (08 hours)

Uses of computer networks: Business Applications, Home Application, Mobile Users, Social Issues; Network hardware: Local Area Networks, Metropolitan Area Networks, Wide Area Networks, Wireless Networks, Home Networks, Internet works; Network software: Protocol Hierarchies, Design Issues for the Layers, Connection-Oriented and Connectionless Services, Service Primitives. Reference Models: OSI Reference Model and TCP/IP Reference Model.
Chapter 1: 1.1-1.3, 1.4.1, 1.4.2

Module-2 Network Layer - Routing and Congestion Control (8 hours)

Routing algorithms: The Optimality Principle, Shortest Path Routing, Flooding, Distance Vector Routing, Link state Routing, Hierarchical Routing.
Congestion Control Algorithms: General Principles of Congestion Control, Congestion Prevention Policies, Congestion Control in Virtual-Circuit Subnets, Congestion Control in Datagram Subnets, Load Shedding, Jitter Control.
Chapter 5: 5.2, 5.3

Module-3 Quality of Service in Networks and Internetworking (8 hours)

Quality of Service: Application Requirements, Traffic Shaping, Packet Scheduling, Admission Control.
Internetworking: How networks differ, How Networks Can Be Connected, Tunneling, Internetwork Routing, Fragmentation**;** The Network Layer in the Internet: The IPv4 Protocol, IP Addresses, IPv6, Internet Control Protocols.
Chapter 5: 5.4, 5.5, 5.6: 5.6.1-5.6.4, 5.6.6-5.6.8

Module-4 The Transport Layer (8 hours)

The Transport Service: Services Provided to the Upper Layers, Transport Service Primitives, An Example of Socket Programming. **Elements of Transport Protocols:** Addressing, Connection Establishment, Connection Release.
The Internet Transport Protocols (UDP and TCP): Introduction to UDP, Introduction to TCP, The TCP Service Model, The TCP Protocol, The TCP Segment Header, TCP Connection Establishment, TCP Connection Release, and Sliding Window protocols.
Chapter 6: 6.1,6.2,6.4,6.5.

Module-5 The Application Layer (8 hours)

The Domain Name System DNS: The DNS Name Space, Domain Resource Records, Name Servers,
Electronic Mail: Architecture and Services, The User Agent, Message Formats
WWW: Architectural overview, Static Web Pages, Dynamic Web Pages and . HTTP (Hypertext Transfer Protocol), Web Search.
Chapter 7: 7.1, 7.2, 7.3

A–Demonstration (Compulsorily to be conducted):

A1. Implement three nodes point–to–point networks with duplex links between them using NS2. Set the queue size, vary the bandwidth, and find the number of packets dropped.

A2. Implement transmission of ping messages/traceroute over a network topology consisting of 6 nodes using NS2 and find the number of packets dropped due to congestion.

B–Exercise (compulsorily to be conducted):

B1. Write a program to find the shortest path between vertices using the bellman-ford algorithm

B2. Write a program for congestion control using a leaky bucket algorithm.

B3. Using TCP/IP sockets, write a client-server program to make client sending the file name and the server to send back the contents of the requested file if present.

C–Structured Enquiry (compulsorily any 4 to be conducted):

C1. Analyze network packets using Wireshark (IPv6, Ipv4, ICMP, and ICMP6).

C2. Implement an Ethernet LAN using n nodes and set multiple traffic nodes using NS2 and plot conges the window for different source/destination.

C3. Develop a program such that client will send a simple ping message to a server and receive a corresponding pong message back from the server using any programming message.

C4. Implement IPv4 address classifier (A, B, C, D, and E) using any programming language.

C5. Implement simple FTP protocol using any programming language.

C6. Create IPv4 or IPv6 packets using any programming language.

D–Open Ended Experiments (any one):

D1. Implement IPv4 router using any programming language.

D2. Implement packet sniffer using any programming language.

Course Outcomes: At the end of the course the student will be able to:	
22CSE52.1	Identify various uses of networks, Design issue of network layers, and summarize its quality-of-service requirements.
22CSE52.2	Illustrate various routing and congestion control algorithms.
22CSE52.3	Outline the internet protocol's structure and demonstrate the working of internet protocols.
22CSE52.4	Identify transport layer services and show the working of transport layer protocols.
22CSE52.5	Appraise application layer protocols.
22CSE52.6	Apply modern networking tools.

Sl. No.	**Title of the Book**	**Name of the Author/s**	**Name of the Publisher**	**Edition and Year**
Textbooks				
1	Computer Networks	Andrew S. Tanenbaum David J. Wetherall	Pearson	4th Edition, 2014
2	A First Course in the Finite Element Method	Daryl L. Logan	Cengage Learning India	5th Edition, 2020
Reference Books				
1	Computer Networking. A Top-Down Approach	Kurose and Ross	Tata McGraw-Hill	5th Edition, 2010
2	Computer Networks: A Systems Approach	Bruce S. Davie and Larry L. Peterso	The Morgan Kaufmann Series in Networking	5th Edition, 2011

Web links and Video Lectures (e-Resources):

- **Computer Networks and Internet Protocol, IIT Kharagpur: https://www.youtube.com/playlist?list=PLbRMhDVUMngf-peFloB7kyiA40EptH1up**
- **TCP/IP Tutorial and Technical Overview: https://www.redbooks.ibm.com/redbooks/pdfs/gg243376.pdf**
- **RFCs: http://www.ietf.org/rfc.html**
- **Computer Networks:** https://www.cse.iitk.ac.in/users/dheeraj/cs425/
- **Web Resources for Computer Networks, 5/e:** https://www.cs.vu.nl/~ast/CN5/

Course Articulation Matrix

Course Outcomes (COs)	Program Outcomes (POs)													
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
22CSE52.1				3		3			3					
22CSE52.2	2				2									2
22CSE52.3				2					2					
22CSE52.4			2										2	
22CSE52.5			2			2								2
22CSE52.6					3				1					2

1: Low 2: Medium 3: High

Full Stack Development			
Course Code	**22CSE53**	CIE Marks	50
Course Type (Theory/Practical/Integrated)	Integrated	SEE Marks	50
		Total Marks	100
Teaching Hours/Week (L:T:P)	3:0:2	SEE	3 Hours
Total Hours	40 hours Theory + 10 Lab slots	Credits	04

Course Learning Objectives: The objective of the course is to

- Illustrate the Semantic Structure of HTML and CSS
- Design Client-Side programs using JavaScript
- Understand the basics of React and create components and lifecycle
- Build applications using React JSX and Node.js
- Design databases using MongoDB

Module-1 HTML and CSS (08 hours)

Introduction to HTML: HTML Syntax, Semantic Markup, Structure of HTML Documents, HTML5 Semantic Structure Elements

Introduction to CSS: What is CSS, CSS Syntax, Location of Styles, Selectors, The Cascade: How Styles Interact, The Box Model, CSS Text Styling.

Textbook 1: Chapter 2, 3

Module-2 JavaScript and React Basics (08 hours)

JavaScript: Client-Side Scripting: What is JavaScript and What can it do? JavaScript Design Principles, Where does JavaScript Go? Syntax, JavaScript Objects, The Document Object Model (DOM), JavaScript Events, Forms.

The Life of a Component: A Custom Function Component, A Custom Class Component, Properties, State

Textbook 1: Chapter 6 and Textbook 2: Chapter 1, 2

Module-3 Advanced React (08 hours)

The Life of a Component: A Note on DOM Events, Props Versus State, Props in Initial State, Lifecycle Methods.

JSX: Whitespace in JSX, Comments in JSX, HTML Entities, Spread Attributes, Returning Multiple Nodes in JSX.

Building the App's component: Setup, start coding, css, The components, Logo and body, Button Component, Forms

Textbook 2: Chapter 2, 5,7

Module-4 Node.js (08 hours)

Welcome to Node.js: Built on JavaScript, Asynchronous and evented, DIRTy applications, DIRTy by default. **Node programming fundamentals:** Organizing and reusing Node functionality, Asynchronous programming techniques, Sequencing asynchronous logic. **Web application development with Node:** HTTP server fundamentals, Building a RESTful web service, Serving static files, Accepting user input from forms, Securing application with HTTPS

Textbook 3: Chapter 1, 3, 4

Module-5 MongoDB (08 hours)

Getting Started: Documents, Collections, Databases, Starting MongoDB, Introduction to the MongoDB Shell, Data Types,

Editing Documents: Inserting, Removing, Updating Documents.

Querying: Introduction to find, Query Criteria, Type-Specific Queries, $where Queries, Cursors.

Textbook 4: Chapter 2, 3, 4

PRACTICALMODULE

A–Demonstration (Compulsorily to be conducted):

A1. Create a webpage with HTML describing your department. Use paragraph and list tags. A2.

A2. Apply various colors to suitably distinguish key words. Also apply font styling like italics, underline and two other fonts to words you find appropriate. Also use header tags.
A3. Create links on the words e.g. “Wi-Fi” and “LAN” to link them to Wikipedia pages.
A4. Insert an image and create a link such that clicking on image takes user to other page.
A5. Change the background color of the page. At the bottom create a link to take user to the topof the page.

PART – B (Compulsorily to be conducted):

B1. i Create a table to show your class time-table. (CSS and HTML)
ii Include course name, instructors, days of week and time slots. Include at least 5 coursesand make sure the table is properly formatted with appropriate table headers, rows and cells.

B2. i Write an HTML page that contains a selection box with a list of 5 countries,
ii when theu user Select a country its capital should be printed next to the list;
iii Add CSS to Customize theproperties of font of the capital (color, bold, and font size)

B3. Write a JavaScript to design a simple calculator to perform the following operations: sum, product, difference and quotient (Java Script)

B4. Build your own Button component using React JS and render it three times. On user click, it shouldalert which button was clicked.

B5. Build a form that accepts a first name and a last name. And instead of a boring "Submit" button, make a button that says "Greet Me" that when clicked, will alert "Hello [first name] [last name]!".

B6. Create a custom component for rendering each joke present in an array. Using the map function, map through each object in the array. Use the custom component to render each object.

PART – C

C1. Create an HTTP server listening on port 1337, which sends Hello, World! to the browser and using Express

C2. Develop a simple react app that displays an unordered list of fruits and ordered list of selected students for an event

C3. Implement node.js and mongo dB application for inserting and displaying the data.

PART D

(Open Ended Problems: any one)

1. Create a Tic Tac Toe game using React JS
2. Develop React app that performs student registration to a course. It should also display list of students registered for any selected course.

Course Outcomes: At the end of the course the student will be able to:	
22CSE53.1	Apply HTML and CSS syntax and semantics to build web pages
22CSE53.2	Design Client-Side Scripts using JavaScript
22CSE53.3	Apply the concepts of React to create Components and design applications using the Lifecycle method
22CSE53.4	Build applications using React JSX
22CSE53.5	Develop applications using Node.js
22CSE53.6	Construct Databases using MongoDB

Sl. No.	Title of the Book	Name of the Author/s	Name of the Publisher	Edition and Year
Textbooks				
1	Fundamentals of Web Development	Randy Connolly, Ricardo Hoar	Pearson Education India	1st Edition, 2015
2	React:Up & Running: Building Web Applications	Stoyan Stefanov	O'Reilly Media,Inc.	2nd Edition, 2020
3	Node.js in Action	Mike Cantelon, Marc Harter, T.J. Holowaychuk, and Nathan Rajlich	Manning Publications	1st Edition, 2014
4	MongoDB: The Definitive Guide	Shannon Bradshaw, Eoin Brazil, Kristina Chodorow	O'Reilly Media,Inc.	3rd Edition, 2019
Reference Books				
1	Learning PHP, MySQL & JavaScript with jQuery, CSS and HTML5	Robin Nixon	O'Reilly Publications	4th Edition, 2015
2	Professional JavaScript for Web Developers	Nicholas C Zakas	Wrox/Wiley India	3rd Edition, 2012
3	Node.js Web Development	David Herron	Packt Publishing	4th Edition, 2018
4	Fullstack React: The Complete Guide toReactJS and Friends	Anthony Accomazzo, Ari Lerner, Nate Murray, Clay Allsopp, David Gutman, Tyler McGinnis	Fullstack.io	1st Edition, 2017

Web links and Video Lectures (e-Resources):

- NPTEL Videos on Compiler Design: https://archive.nptel.ac.in/courses/106/104/106104123/
- **NPTEL Video on System Programming**: https://www.youtube.com/watch?v=zj2VsRnji6c&list=PLM-jfaoaU5iy0Zq-NlWycsfzyMKrAV7jw
- **NPTEL Video on System Software:**https://www.youtube.com/watch?v=VG9VopzV_T0

Course Articulation Matrix

Course Outcomes (COs)	Program Outcomes (POs)													
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
22CSE53.1			3										1	
22CSE53.2					2								1	
22CSE53.3					2				2					
22CSE53.4					2									
22CSE53.5									2					
22CSE53.6			2						1					

1: Low 2: Medium 3: High

Automata Theory and Computability			
Course Code	**22CSE54**	CIE Marks	50
Course Type (Theory/Practical/Integrated)	Theory	SEE Marks	50
		Total Marks	100
Teaching Hours/Week (L:T:P)	2:2:0	SEE	3 Hours
Total Hours	40	Credits	03

Course Learning Objectives: The objective of the course is to

- Introduce core concepts in Automata and Theory of Computation
- Identify different Formal Language Classes and their Relationships
- Design Grammars and Recognizers for different formal languages
- Prove or disprove theorems in automata theory using their properties
- Determine the decidability and intractability of Computational problems

Module-1 Basic Concepts of Automata Theory (8 hours)

Languages and Strings: Strings, Languages.
Finite State Machines (FSM): Deterministic FSM, Regular languages, Designing DFSM, Nondeterministic FSMs, From FSMs to Operational Systems, Minimizing FSM
Textbook 1: Chapter 2, Chapter 5: 5.1-5.5, 5.7.2

Module-2 Regular Expressions and Languages (8 hours)

Regular Expressions (RE): What is a RE? Kleene's theorem,
Regular Grammars: Definition, Regular Grammars and Regular Languages.
Regular and Nonregular Languages: How many RLs? Showing that a language is regular, Closure properties of RLs, Showing That a Language is Not Regular.
Textbook 1: Chapter 6: 6.1-6.2(6.2.1-6.2.3), Chapter 7: 7.1-7.2, Chapter 8: 8.1-8.4

Module-3 Context-Free Grammars and PDA (8 hours)

Context-Free Grammars (CFG): Introduction to Rewrite Systems and Grammars, CFGs and languages, Simplifying CFGs, Derivations and Parse trees, Ambiguity, Normal Forms. Pushdown **Automata (PDA):** Definition of Non-deterministic PDA, Deterministic and Non deterministic PDAs.
Textbook 1: Chapter 11: 11.1, 11.2, 11.4, 11.6-11.8, Chapter 12: 12.1-12.2

Module-4 Decision Procedures for CFLs and Turing Machines (8 hours)

Algorithms and Decision Procedures for CFLs: Decidable questions, Undecidable questions.
Turing Machine: Turing machine model, Representation, Language acceptability by TM, Design of TM, Techniques for TM construction. Variants of Turing Machines (TM), The model of Linear Bounded automata.
Textbook 1, Chapter 14: 14.1-14.2, Textbook 2: Chapter 9: 9.1-9.4, 9.6, 9.7, 9.8

Module-5 Decidability and Complexity (8 hours)

Decidability: Definition of an algorithm, decidability, decidable languages, Undecidable languages, Halting problem of TM, Post correspondence problem, Supplementary Examples. **Complexity:** Growth rate of functions, Classes of P and NP, Quantum Computation: Quantum computers, Church-Turing thesis.
Textbook 2: Chapter 10: 10.1-10.7, Chapter 12: 12.1-12.2, 12.8(12.8.1, 12.8.2)

Course Outcomes: At the end of the course the student will be able to:	
22CSE54.1	Demonstrate an understanding of the core concepts in automata theory and construct regular expressions using the concepts learnt.
22CSE54.2	Construct different models of computation using finite state machines and deduce optimum solutions for the problems

22CSE54.3	Identify grammars for different language classes and perform computations by applying it to various applications
22CSE54.4	Classify problems with respect to the push down automata model of computation
22CSE54.5	Inspect the decision procedures for context free grammars and solve the problems using formal models like Turing Machines
22CSE54.6	Examine the decidable and undecidable problems, the complexity of the problem and Quantum Computation.

Sl. No.	**Title of the Book**	**Name of the Author/s**	**Name of the Publisher**	**Edition and Year**
Textbooks				
1	Automata, Computability and Complexity	Elaine Rich	Pearson Education Asia	1st Edition, 2019
2	Theory of Computer Science	K L P Mishra, N Chandrasekaran	Prentice Hall India	3rd Edition, 2012
Reference Books				
1	Introduction to languages and the theory of computation	J Martin	Tata McGraw-Hill	3rd Edition, 2007
2	Elements and Theory of Computation	C Papadimitrou and C. L. Lewis	Prentice Hall India	3rd Edition, 2012
3	Mathematical Foundation of Computer Science	Y.N.Singh	NewAge International	1st Edition, 2005
4	A Formal Languages and Automata Theory	C K Nagpal	Oxford University Press	1st Edition, 2012

Web links and Video Lectures (e-Resources):

- **Theory of Computation & Automata Theory:** https://www.youtube.com/watch?v=58N2N7zJGrQ&list=PLBlnK6fEyqRgp46KUv4ZY69y XmpwKOIev
- **Automata Theory : https://www.youtube.com/watch?v=wXMkeJyI7YE&list=PLm_MSC lsnwm gKA0hAZkIrPO-BlYAtiEV**
- **Computer - Theory of Automata, Formal Languages and Computation**: https://www.youtube.com/watch?v=-aIRqNnUvEg&list=PL85CF9F4A047C7BF7
- **Introduction to Automata, Languages and Computation :** https://www.youtube.com/watch?v=_2w9UX17m_k&list=PLbRMhDVUMngcwWkzVTm_ kFH6JW4JCtAUM
- **Computer Science - Theory of Computation** : https://www.youtube.com/watch?v=al4AK6ruRek&list=PLbMVogVj5nJSd25WnSU144

Course Articulation Matrix

Course Outcomes (COs)	Program Outcomes (POs)													
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
22CSE54.1	3													1
22CSE54.2	3													1
22CSE54.3		2												1
22CSE54.4		2												1
22CSE54.5				2										1
22CSE54.6				2										1

1: Low 2: Medium 3: High

Mobile Application Development Laboratory			
Course Code	**22CSE55L**	CIE Marks	50
Course Type (Theory/Practical/Integrated)	Practical	SEE Marks	50
		Total Marks	100
Teaching Hours/Week (L:T:P)	0:0:2	SEE	3 Hours
Total Hours	20 Hours	Credits	01

Course Learning Objectives: The objective of the course is to

- Utilize Android Platform and Android Studio Environment to develop android application.
- Design responsive user interfaces that work across a wide range of devices.
- Demonstrate methods in storing, sharing and retrieving data in Android applications

Experiments

A–Demonstration:

A1. App user interface designing – mobile UI Layout (Layout, View) UI Control (TextView), EditText, Button, ImageButton, ToggleButton, RadioGroup, RadioButton, CheckBox, ProgressBar, Spinner, DayPicker, TimePicker), Drawable, Menu (Option, Context, popup)

A2. Android Application Component: Activity –states and life cycle, interaction amongst activities. Services – state and lifecycle. Notifications, Broadcast Receivers, Content Provider, Fragments. Intents: Implicit and Explicit Intent

A3. App Functionality Beyond User Interface: Threads, Async task, Notification, Location Based Service, Telephony and SMS APIs, Text to Speech, Camera

A4. Data Handling: Shared preferences, mobile databases such as SQLite, and enterprise data access, Android multimedia: Multimedia-audio/video playback and record. Sensors: Location awareness and native hardware access (sensors such as accelerometer and gyroscope). Android Web Service, Android Google Maps, Android Bluetooth, Navigation.

B–Exercise (compulsorily to be conducted):

B1. Design four checkboxes namely any four food items and one button. Find total amount of food items selected in Toast message after clicking the button.

B2. Create an application which generates a random color on each click.

B3. Implement option menu concept in application to choose between two activities.

B4. Implement context menu concept in application to change the background color.

C–Structured Enquiry (compulsorily to be conducted):

C1. Develop an application to insert the data entered by a user into a database and display all the values in database.

C2. Develop an application to send SMS using Intent class.

C3. Implement phone call concept in application by passing number from the user.

C4. Develop an app to capture the image using camera and set it as background for your app.

D–Open Ended Experiments (any one):

D1. Develop an application to search for a given USN from a student database and call to that student.

D2. Develop an application to toast your joining date and course selected for engineering using date picker and list view.

Course Outcomes: At the end of the course the student will be able to:	
22CSE55L.1	Utilize the IT tool like Android Platform and Android Studio Environment to develop android application.
22CSE55L.2	Develop adaptive, responsive user interfaces that work across a wide range of devices.
22CSE55L.3	Infer long running tasks and background work in Android applications
22CSE55L.4	Demonstrate methods in storing, sharing and retrieving data in Android applications.

22CSE55L.5	Inspect different methods of sharing data using services
22CSE55L.6	Develop Application to store and retrieve data from database.

Sl. No.	Title of the Book	Name of the Author/s	Name of the Publisher	Edition and Year
Textbooks				
1	Anubhav Paradhan, Anil V Deshpande,	Mobile apps Development	Wiley India Pvt Ltd	1st Edition, 2014
Reference Books				
1	Android Programming – Pushing the Limits	Erik Hellman,	Wiley India Pvt Ltd	1st Edition, 2014
2	Dawn Griffiths and David Griffiths	Head First Android Development	O‟Reilly SPD Publishers	1st Edition, 2015
3	Android Programming: The Big Nerd Ranch Guide	Bill Phillips, Chris Stewart and Kristin Marsicano	Big Nerd Ranch Guides	3rd Edition 2017

Web links and Video Lectures (e-Resources):

- **Android Developer Fundamentals Course:** https://www.gitbook.com/book/google-developer-training/android-developer-fundamentalscourse-concepts/details
- **TechA Android Application Development with Kotlin Certification:** TOC - TechA Android Application Development with Kotlin Certification | Infosys Springboard (onwingspan.com)
- **TechA Flutter Application Developer Certification:** TOC - TechA Flutter Application Developer Certification | Infosys Springboard (onwingspan.com)

Course Articulation Matrix

Course Outcomes (COs)	Program Outcomes (POs)													
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
22CSE55L.1	3				3							2		3
22CSE55L.2		3			3							2		3
22CSE55L.3	3				3							2		3
22CSE55L.4		3			3							2	3	
22CSE55L.5				2	3							2	3	
22CSE55L.6				2	3							2	3	

1: Low 2: Medium 3: High

Big Data and Analytics			
Course Code	**22CSE561**	CIE Marks	50
Course Type (Theory/Practical/Integrated)	Theory	SEE Marks	50
		Total Marks	100
Teaching Hours/Week (L:T:P)	3:0:0	SEE	3 Hours
Total Hours	40 Hours	Credits	03

Course Learning Objectives: The objective of the course is to

- Explain the Hadoop framework and Hadoop Distributed File system
- Demonstrate MapReduce programming model to process the big data
- Illustrate the concepts of NoSQL using MongoDB for Big Data
- Analyze text mining, web mining and Social network system

Module-1 Introduction to Big Data Analytics (8 hours)

Introduction to Big Data Analytics: Big Data, Scalability and Parallel Processing, Designing Data Architecture, Data Sources, Quality, Pre-Processing, Data Storage and Analysis, Big Data Analytics Applications and Case Studies.

Textbook 1: Chapter 1.2-1.5.3, 1.6, 1.7

Module-2 Introduction to Hadoop and HDFS (8 hours)

Introduction to Hadoop (T2): Introduction, Hadoop and its Ecosystem, Hadoop Distributed File System, MapReduce Framework and Programming Model, Hadoop Yarn.

Hadoop Distributed File System Basics (T1): HDFS Design Features, Components, HDFS User Commands.

Essential Hadoop Tools (T1): Using Apache Sqoop, Flume, Oozie, and HBase.

Textbook 1: Chapter 2.1-2.5, Textbook 2: Chapter 3, Textbook 2: Chapter 7.3-7.6

Module-3 MapReduce, Hive and Apache Spark (8 hours)

MapReduce, HBase and Spark: Introduction, MapReduce Map Tasks, Reduce Tasks and MapReduce Execution, Composing MapReduce for Calculations and Algorithms

Apache Hbase: Using Apache Hbase, HBase Data Model Overview, HBase Example Walk-Through

Apache Spark: A gentle Introduction to Spark: Basic architecture, Spark Language API, Starting Spark, Spark Session, DataFrames, Partitions, Transformations, Lazy Evaluation, Actions, Spark UI

Textbook 1: Chapter 4.1-4.6, Textbook 2: Chapter 7, Textbook 3: Chapter 2

Hands On:

1) Demonstrate MapReduce program in python to find occurrences of each word in a text file.
2) Write MapReduce program to demonstrate the use of Aggregate functions on suitable products.
3) Demonstrate MapReduce program on Addition and Subtraction of two Matrices.
4) Demonstrate Problems on Apache Spark.

Module-4 NoSQL and MongoDB database (8 hours)

NoSQL data store and MongoDB database: Introduction, NoSQL Data store, NoSQL Data Architecture Patterns, NoSQL to Manage Big Data, Shared-Nothing Architecture for Big Data Tasks, MongoDB Database.

Textbook 1: Chapter 3.1-3.6

Hands On: 1) Create the Database and implement Insert, Delete and Search functions using MongoDB query language.

Module-5 Social Network Analytics (8 hours)

Text, Web Content, Link, and Social Network Analytics: Introduction, Text mining, Web Mining, Web Content and Web Usage Analytics. Page Rank, Structure of Web and analyzing Web Graph. Social Network as Graphs and Social Network Analytics.

Textbook 1: Chapter 9.1-9.5

Course Outcomes: At the end of the course the student will be able to:	
22CSE561.1	Interpret fundamentals of Big Data analytical and application systems.
22CSE561.2	Illustrate the concept of Hadoop framework, HDFS and tool services.
22CSE561.3	Demonstrate MapReduce programming model to process Big Data.
22CSE561.4	Interpret Essential Hadoop Hive and Pig tools.
22CSE561.5	Illustrate the concepts of NoSQL using MongoDB for Big Data storage
22CSE561.6	Analyze the text content, web content and Social Network Analytics for effective analytics.

Sl. No.	**Title of the Book**	**Name of the Author/s**	**Name of the Publisher**	**Edition and Year**
Textbooks				
1	Big Data Analytics Introduction to Hadoop, Spark, and Machine-Learning	Raj Kamal and Preeti Saxena	McGraw Hill Education	1st Edition, 2018
2	Hadoop 2 Quick-Start Guide: Learn the Essentials of Big Data Computing in the Apache Hadoop 2 Ecosystem.	Douglas Eadline	Pearson Education	1st Edition, 2016.
Reference Books				
1	Big Data and Hadoop	V.K. Jain	Khanna Book Publishing	1st Edition, 2017
2	Data Analytics	Maheshwari	McGraw	1st Edition, 2017
3	Big Data Analytics: A Hands-On Approach	Arshdeep Bahga, Vijay Madisetti	VPT Publications	1st Edition, 2018
4	Big Data Analytics	Seema Acharya	Wiley	2nd Edition, 2019

Web links and Video Lectures (e-Resources):

1) **Fundamentals of Big Data :** https://www.simplilearn.com/what-is-big-data-analytics-article
2) **Introduction to Hadoop:** https://www.geeksforgeeks.org/hadoop-an-introduction/
3) **View of MapReduce concepts :** https://www.guru99.com/introduction-to-mapreduce.html
4) **Essential Hadoop Tools:** https://www.educba.com/hadoop-tools/

Course Articulation Matrix

Course Outcomes (COs)	**Program Outcomes (POs)**													
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
22CSE561.1	1	1	1	1				1					1	1
22CSE561.2			1	3	2								1	2
22CSE561.3		1		2						1	1	1	2	1
22CSE561.4		1		2	2					1		1	2	1
22CSE561.5		1		1	1								1	1
22CSE561.6			1		1									2

1: Low 2: Medium 3: High

Blockchain Technology			
Course Code	**22CSE562**	CIE Marks	50
Course Type (Theory/Practical/Integrated)	Theory	SEE Marks	50
		Total Marks	100
Teaching Hours/Week (L: T:P)	3:0:0	SEE	3 Hours
Total Hours	40 hours	Credits	03

Course Learning Objectives:

- Outline the basics of Blockchain Technology
- Explain the foundations of bitcoins.
- Appraise the idea of Ethereum Blockchain and Smart Contract
- Explore Solidity Programming language to develop smart contract.
- Illustrate Hyperledger fabric and its framework

Module-1 Introduction to Blockchain Technology (8 hours)

Introduction: Blockchain 101: Distributed systems, History of blockchain, Introduction to blockchain, Types of blockchain, CAP theorem and blockchain, Benefits and limitations of blockchain.
Decentralization: Decentralization using Blockchain, Methods of decentralization, Routes to decentralization, Decentralized organizations. **Textbook 1**: **Ch 1, Ch 2**

Module-2 Introduction to Bitcoins (8 hours)

Introducing Bitcoins: Bitcoin definition, Digital keys and addresses: Private keys in Bitcoin, Public keys in Bitcoin, Addresses in Bitcoin, Transactions: Transaction life cycle, Transaction data structure, Types of transaction, transaction verification. Blockchain: structure of a block, structure of block header, genesis block. Mining: Task of miners, mining rewards, Proof of work, mining algorithm, hash rate, mining systems, mining pools.
Bitcoin Network and Payments: The bitcoin network, wallets: Non deterministic wallets, deterministic wallet, Hierarchical deterministic wallet, brain wallet, paper wallet, hardware wallet, online wallet, mobile wallet. Bitcoin payments, Innovation in Bitcoin: Bitcoin Improvement Proposals, advanced protocols, segregated witness, bitcoin cash, bitcoin unlimited, bitcoin gold, bitcoin investment- buying and selling bitcoins.
Textbook 1: Ch 5, Ch 6

Module-3 Smart Contracts and Ethereum (8 hours)

Smart Contracts and Ethereum 101: Smart Contracts: Definition, Ricardian Contracts. Ethereum 101: Introduction, Ethereum blockchain.
Further Ethereum: Blocks and Blockchain, The genesis block, block validation mechanisms, Block difficulty, Gas Fee schedule, Forks in blockchain, Nodes and miners, The consensus mechanism, CPU mining, GPU mining, Benchmarking, mining rigs, mining pools.
Textbook 1: Ch 10, Ch 11

Module-4 Introduction to Solidity (8 hours)

Introducing solidity: Types, Value types: Boolean, Integers, Address, Array value types (fixed size and dynamically sized byte arrays), Literals, Integer literals, String literals, Hexadecimal literals, Enums, Function types, Internal functions, External functions, Reference types, Arrays, Structs, Data location, Mappings, Global variables, Control structures, Events, Inheritance, Libraries, Functions, Layout of a solidity source code file. Truffle Basics and Unit Testing, Debugging Contracts Remix IDE: Programs execution. **Textbook 1: Ch 13**

Module-5 Hyperledger fabric (8 hours)

Exploring Hyperledger Fabric: Building on the foundations of open computing, Fundamentals of the Hyperledger project, The Linux Foundation, Hyperledger, Open source and open standards, Hyperledger frameworks, tools, and building blocks, Hyperledger Fabric component design, Principles of Hyperledger design, Hyperledger Fabric reference architecture, Hyperledger Fabric runtime architecture, Strengths and advantages of componentized design.
Textbook 2: Ch 2

Course Outcomes: At the end of the course the student will be able to:	
22CSE562.1	Explain the fundamental building blocks of Blockchain technology.
22CSE562.2	Outline the basics of Bitcoin, bitcoin network and payments.
22CSE562.3	Appraise the concepts of smart contract and basics of Ethereum.
22CSE562.4	Develop block chain-based solutions and write smart contract using Solidity.
22CSE562.5	Illustrate Hyperledger fabric and its framework, design principles and architecture.
22CSE562.6	Analyse the principles of Hyperledger design.

Sl. No.	**Title of the Book**	**Name of the Author/s**	**Name of the Publisher**	**Edition and Year**
Textbooks				
1	Mastering Block Chain: Distributed Ledger Technology, Decentralization and Smart Contracts Explained	Imran Bashir	Packt Publishing	1st Edition, 2017
2	Block Chain with Hyperledger Fabric: Building Decentralized Applications using Hyperledger Fabric 2	Salman Baset, Luc Desrosiers, Nitin Gaur, Petr Novotny, Anthony O'Dowd, Venkatraman Ramakrishna	Packt Publishing	2nd Edition, 2020
Reference Books				
1	Blockchain: Blueprint for a New Economy	Melanic Swan	O'Reilly	1st Edition 2015
2	Blockchain Basics	Daniel Drescher	Apress	1st Edition, 2017

Additional Resources: Web links/NPTEL Courses

- https://www.simplilearn.com/learn-blockchain-basics-skillup
- https://www.youtube.com/watch?v=0UvVOMZqpEA
- https://www.slideshare.net/Mithileysh/blockchain-technology-181440314
- https://intellipaat.com/blog/what-is-blockchain-technology/

Course Articulation Matrix

Course Outcomes (COs)	Program Outcomes (POs)													
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
22CSE562.1	3		2			3							3	
22CSE562.2	3		2										3	
22CSE562.3	2		2			3	3						3	
22CSE562.4	2		3		3	2						3	3	
22CSE562.5	2		2		2								3	
22CSE562.6	2		2		2								3	

1: Low 2: Medium 3: High

NoSQL Databases			
Course Code	**22CSE563**	CIE Marks	50
Course Type (Theory/Practical/Integrated)	Theory	SEE Marks	50
		Total Marks	100
Teaching Hours/Week (L:T:P)	3:0:0	SEE	3 Hours
Total Hours	40 hours	Credits	03

Course Learning Objectives: The objective of the course is to

- Learn the difference between relational database and non-relational database
- Define, compare, and use the different types of NoSQL Databases
- Perform aggregate functions and learn the usage of Document-oriented, Key Value Pairs, Column-oriented and Graph databases
- Gain knowledge of replication, sharding, distribution, and resilience in NoSQL databases

Module-1 Introduction (08 Hours)

Early Database Management Systems, The Relational Database Revolution, Motivations for Not Just/No SQL (NoSQL) Databases.
Data Management with Distributed Databases, ACID and BASE, Four Types of NoSQL Databases.
Textbook 1: Ch 1, 2

Module-2 Key-Value Databases (08 Hours)

Essential Features of Key-Value Databases, Keys: More Than Meaningless Identifiers, Values: Storing Just About Any Data You Want.
Key-Value Database Terminology: Key-Value Database Data Modeling Terms, Key-Value Architecture Terms.
Designing for Key-Value Databases: Key Design and Partitioning, Limitations of Key-Value Databases.
Textbook 1: Ch 3, 4, 5

Module-3 Document Databases (08 Hours)

What Is a Document? Avoid Explicit Schema Definitions, Basic Operations on Document Databases.
Document and Collection Terms, Types of Partitions, Data Modeling and Query Processing.
Normalization, Denormalization, and the Search for Proper Balance: One-to-Many Relations, Many-to-Many Relations, The Need for Joins.
Textbook 1: Ch 6,7,8

Module-4 Column Family Databases (08 Hours)

Differences and Similarities to Key-Value and Document Databases, Architectures Used in Column Family Databases, When to Use Column Family Databases.
Basic Components of Column Family Databases, Structures and Processes: Implementing Column Family Databases.
Guidelines for Designing Tables. **Textbook 1: Ch 9,10,11**

Module-5 Graph Databases (08 Hours)

What Is a Graph? Graphs and Network Modeling, Advantages of Graph Databases
Elements of Graphs, Operations on Graphs, Properties of Graphs and Nodes, Types of Graphs. Getting Started with Graph Design, Querying a Graph, Tips and Traps of Graph Database Design. Guidelines for Choosing a NoSQL Database. **Textbook 1: Ch 12,13,14,15**

Course Outcomes: At the end of the course the student will be able to:	
22CSE563.1	Define, compare, and use the four types of NoSQL Databases (Document-oriented, Key-Value Pairs, Column-oriented and Graph).
22CSE563.2	Demonstrate the detailed architecture, define objects, load data, query data and performance tune Key-Value NoSQL databases.

22CSE563.3	Evaluate the suitability of document databases for managing semi-structured and unstructured data in modern application architectures.
22CSE563.4	Illustrate the principles of schema design and normalization in document databases.
22CSE563.5	Explain the concepts of Column-Family Data Store for efficient storage and retrieval of wide-column data models.
22CSE563.6	Explore the use of graph databases for managing and querying highly interconnected data.

Sl. No.	Title of the Book	Name of the Author/s	Name of the Publisher	Edition and Year
Textbooks				
1	NoSQL For Mere Mortals	Dan Sullivan	Pearson Education India	1st Edition, 2015
Reference Books				
1	NoSQL distilled: a brief guide to the emerging world of polyglot persistence.	Sadalage, Pramod J, Martin Fowler	Pearson Education	1st Edition, 2013
2	NoSQL for Dummies	Adam Fowler	John Wiley & Sons	1st Edition, 2015
3	Mongodb: The Definitive Guide- Powerful and Scalable Data Storage	Kristina Chodorow	O'Reilly Publications	2nd Edition, 2013

Web links and Video Lectures (e-Resources):

- **https://www.geeksforgeeks.org/introduction-to-nosql/ (Introduction to NoSQL)**
- **https://www.mongodb.com/nosql-explained (What is NoSQL?)**
- **https://www.w3schools.in/mongodb/introduction-to-nosql (MongoDB - Introduction to NoSQL)**
- **https://www.ibm.com/topics/nosql-databases (What is a NoSQL database?)**
- **https://www.guru99.com/nosql-tutorial.html (NoSQL Tutorial)**
- **https://www.youtube.com/watch?v=zG6CHYCx6ag (NoSQL database)**
- **https://www.youtube.com/watch?v=vb8xZ-bvxbg&list=PLLAZ4kZ9dFpOFJ9JcVW9u4PlSWO-VFoao (MongoDB)**

Course Articulation Matrix

Course Outcomes (COs)	Program Outcomes (POs)													
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
22CSE563.1	2													
22CSE563.2		2												1
22CSE563.3		2												1
22CSE563.4	2													
22CSE563.5			2						1					
22CSE563.6			2		1						3			

1: Low 2: Medium 3: High

Salesforce Developer			
Course Code	**22CSE564**	CIE Marks	50
Course Type (Theory/Practical/Integrated)	Theory	SEE Marks	50
		Total Marks	100
Teaching Hours/Week (L:T:P)	3:0:0	SEE	3 Hours
Total Hours	40 hours	Credits	03

Course Learning Objectives:

- To be intended for an individual who has experience developing and deploying basic business logic and user interfaces.
- Train the individuals to the next level, who have the knowledge, skills, and experience in building custom applications on the Lightning Platform.
- To learn the fundamental programmatic capabilities of the Lightning Platform to develop custom business logic and interfaces to extend Salesforce using Apex, Visualforce, and basic Lightning Components.
- To use the programmatic capabilities in practice with the Lightning Platform, including practical application of the skills and concepts.

Module-1 Platform Developer (8 hours)

Apex & .NET Basics: Map .NET Concepts to the Lightning Platform, Understand Execution Context, Use Asynchronous Apex, Debug and Run Diagnostics

Formulas and Validations: Use Formula Fields,Implement Roll-Up Summary Fields, Create Validation Rules

Data Modeling : Understand Custom & Standard Objects, Create Object Relationships, Work with Schema Builder

Data Management: Import Data, Export Data

Approve Records with Approval Processes: Customize How Records Get Approved, Build an Approval Process

Record-Triggered Flows: Triggered Flows, Build a Record-Triggered Flow, Add a Scheduled Task to Your Flow, Meet Flow Trigger Explorer

Search Solution Basics: Choose the Right Search Solution, Build Search for Common Use Cases, Optimize Search Results

Apex Basics & Database: Get Started with Apex, Use sObjects, Manipulate Records with DML, Write SOQL Queries, Write SOSL Queries

Apex Triggers : Get Started with Apex Triggers, Bulk Apex Triggers.

Module-2 Triggers (8 hours)

Triggers and Order of Execution: Performing a sequence of events in a order when a record is saved with an insert, update, or upsert statement.

Asynchronous Apex: Asynchronous Processing Basics, Use Future Methods, Use Batch Apex, Control Processes with Queueable Apex, Schedule Jobs Using the Apex Scheduler, Monitor Asynchronous Apex.

Module-3 Introduction to Visualforce (8 hours)

Visualforce & Lightning Experience: Use Visualforce in Lightning Experience, Develop Visualforce Pages for Lightning Experience, Explore the Visualforce App Container, Share Visualforce Pages Between Classic and Lightning Experience, Manage Navigation, Understand Important Visual Design Considerations, Know Which Features to Avoid in Lightning Experience.

Visualforce Basics: Get Started with Visualforce, Create & Edit Visualforce Pages, Use Simple Variables and Formulas, Use Standard Controllers, Display Records, Fields, and Tables, Input Data Using Forms, Use Standard List Controllers, Use Static Resources, Create & Use Custom Controllers.

Module-4 Web Components (8 hours)

Lightning Web Components Basics: Discover Lightning Web Components, Create Lightning Web Components, Deploy Lightning Web Component Files, Handle Events in Lightning Web

Components, Add Styles and Data to a Lightning Web Component Secure Server-Side **Development:** Write Secure Apex Controllers, Mitigate SOQL Injection, Mitigate Cross-Site Request Forgery.
Module-5 Testing & Debugging (8 hours)
Developer Console Basics: Get Started with the Developer Console, Navigate and Edit Source Code, Generate and Analyze Logs, Inspect Objects at Checkpoints, Execute SOQL and SOSL Queries **Command-Line Interface:** Learn About the Command-Line Interface, Explore Command Structure and Navigation, Set Up Command-Line Tools. **Org Development Model:** Plan for Changes to Your Org, Develop and Test Changes Locally, Test and Deploy Changes **Apex Testing**: Get Started with Apex Unit Tests, Test Apex Triggers, Create Test Data for Apex Tests **Find and Fix Bugs with Apex Replay Debugger**: Launch Your Trailhead Playground, Set Up Visual Studio Code, Set Up Apex Replay Debugger, Debug Your Code **Debug Logs**: Debug Log Details, Debug Log Order of Precedence, Debug Log Levels, Searching a Debug Log, Delete Debug Logs, Debug Log Filtering for Apex.

Course Outcomes: At the end of the course the student will be able :	
22CSE564.1	Acquire a fundamental understanding of the CRM
22CSE564.2	To earn experience on Salesforce tools necessary to effectively generate useful applications on the Salesforce platform to support customer requirements.
22CSE564.3	Understand the tools and techniques of CRM
22CSE564.4	Gain experience in using the Salesforce tools to complete projects focused on obtaining actionable insights from complex data.
22CSE564.5	Dive deeply into a Salesforce Developer practice to fully prepare to use knowledge gained in the course to add significant value in a professional setting.
22CSE564.6	Be able to utilize knowledge and skills to continue learning and adapting to other advanced Salesforce technologies.

Sl. No.	**Title of the Book**	**Name of the Author/s**	**Name of the Publisher**	**Edition and Year**
Textbooks				
1	Beginning Salesforce Developer	Michael Wicherski	Wiley Apress Publisher	2018
2	Learning Salesforce Development with Apex	Paul Battisson	BPB Publishers	2020
3	Advanced Apex Programming in Salesforce	Advanced Apex Programming in Salesforce	PACKT Publisher	2018
4	Mastering Apex Programming	Paul Battisson	Paul Battisson	2020
Reference Books				
1	Learning Salesforce Lightning Application Development: Build and test Lightning Components for Salesforce Lightning Experience using Salesforce DX	Mohith Shrivastava	PACKT Publisher	2018

2	Lightning Web Components (LWC) Development on the Salesforce Platform: A Salesforce developer's guide to building, testing, and deploying Lightning Web Components	Brian Cline	PACKT Publisher	2023

Additional Resources: Web links/NPTEL Courses

- **Use the Trailhead Platform:** https://www.salesforce.com/blog/what-is-trailhead/
- **The Salesforce Developer Trailmix : https://trailhead.salesforce.com/users/trjha3/trailmixes/salesforce-developer-catalyst-v-3-0**

Course Articulation Matrix

Course Outcomes (COs)	Program Outcomes (POs)													
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
22CSE564.1		2			1							2		
22CSE564.2					1			3		2				
22CSE564.3						2				2				
22CSE564.4						2				2				
22CSE564.5						2				2				
22CSE564.6						2				2				

1: Low 2: Medium 3: High

Research Methodology and Intellectual Property Rights			
Course Code	**22RMI57**	CIE Marks	50
Course Type (Theory/Practical/Integrated)	Theory	SEE Marks	50
		Total Marks	100
Teaching Hours/Week (L:T:P)	(2:0:0)	SEE	3 Hours
Total Hours	25 hours	Credits	02

Course Learning Objectives:

1. To understand the basic concepts related to research
2. To learn the concept of literature survey, review and technical writing
3. To discuss the basics of intellectual property
4. To explain the patents, copyrights, trademarks, industrial designs and geographical indications

Module-1 Research Methodology and Literature Survey (5 hours)

Research Methodology: Meaning, Objectives, Types of research, Method versus methodology, Research process, Criteria of good research.

Literature Survey, Literature Review: Introduction, process, databases and management tools. Identifying gap areas from literature review. Plagiarism: Introduction, tools for detection, avoiding plagiarism. Illustrations.

Textbook 1: Chapter 1 , Textbook 2: Ch 7-9, 14-17.

Module-2 Technical Writing and Presentations (5 hours)

Research Paper Writing: Importance, steps of writing research papers, Contents of a research article, Illustrations.

Thesis Writing: Synopsis, Introduction, Literature review, Aim and Objectives, Methodology, Time frame, Results and discussions, Conclusions. Illustrations.

Research Proposal Writing: Preliminary requirements for proposal writing, Standard heads in research proposal. Illustrations.

Textbook 2: Chapter 20-22, 26-28, 35.

Module-3 Introduction to IPR and Patents (5 hours)

Introduction to Intellectual Property: Types of IP, Role of IP in the economic and cultural development of the society, IP governance, IP as a global indicator of innovation, National IPR Policy in India. Textbook 3: Chapter 1,

Patents: Conditions for patent, Non-patentable matters, Inventions Eligible for Patenting**,** Salient features of the Indian Patent 1970**,** Process of patenting**,** Types of patent applications**,** Patent infringements**.** Case examples. Textbook 3: Chapter 2: 2.1.

Module-4 Copyright and Trademarks (5 hours)

Copyright: Classes of copyrights, Salient features of the Indian Copyright Act 1957, Criteria for copyright, Copyrights of the author, Copyright Infringements, Non-Copyright Work, Process of copyright registration. Copyright cases.

Trademark: Eligibility Criteria, Classification, Trade Mark Rules 2017, Advantages of registration, Types of trademark registered in India, Process for Trademarks Registration, Case examples.

Textbook 3: Chapter 2: 2.2 and 2.3.

Module-5 Industrial Designs and Geographical Indications (5 hours)

Industrial Designs: Introduction, Eligibility criteria, Famous industrial designs, Features of Design Act 2000, Non-Protectable industrial designs in India, Procedure for Registration of Industrial Designs, Case examples.

Geographical Indications (GIs): Introduction, Rights granted to holders, Popular GIs registered in India, salient features of Geographical Indications of Goods (Registration & Protection) Act, 1999, Non-Registerable GI, Procedure for GI Registration**,** Case examples.

Textbook 3: Chapter 2: 2.4 and 2.5.

Course Outcomes: At the end of the course the student will be able :	
22RMI57.1	To conduct literature survey, review and define a research problem.
22RMI57.2	To follow research ethics and develop the art of writing technical papers and reports.
22RMI57.3	To discuss the role of Intellectual Property and Patents in India.
22RMI57.4	To explain the various aspects of Copyright and Trademark in Indian context.
22RMI57.5	To explain legal aspects of Industrial Designs and Geographical Indications in India.
22RMI57.6	To discuss the case studies related to the different Intellectual Property.

Sl. No.	**Title of the Book**	**Name of the Author/s**	**Name of the Publisher**	**Edition and Year**
Textbooks				
1	Research Methodology: Methods and Techniques	C R Kothari and Gaurav Garg	New Age International Publishers	4th Edition 2019
2	Academic Writing	Ajay Semalty	B S Publications	2021
3	Intellectual Property: A Primer for Academia	Prof. Rupinder Tewari and Ms. Mamta Bhardwaj	Publication Bureau, Panjab University, India	2021
Reference Books				
1	Research Methodology: A Step-by-Step Guide for Beginners	Ranjit Kumar	Sage Publications India Pvt Ld New Delhi	4th Edition 2014
2	Intellectual Property Rights – Laws and Practice	The Institute of Company Secretaries of India, New Delhi	Delhi Computer Services, New Delhi	2018

Additional Resources: Web links/NPTEL Courses

https://ipindia.gov.in/ (Official website of Intellectual Property India)
https://dpiit.gov.in/policies-rules-and-acts/policies/national-ipr-policy
https://www.icsi.edu/media/webmodules/FINAL_IPR&LP_BOOK_10022020.pdf
https://corpbiz.io/learning/design-infringement-in-india/
https://nptel.ac.in/courses/121106007 (Introduction to Research (Research Methodology))
https://nptel.ac.in/courses/109105112 (Introduction on Intellectual Property to Engineers)

Course Articulation Matrix

Course Outcomes (COs)	Program Outcomes (POs)													
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
22RMI57.1	-	**2**	-	-	**1**	-	-	-	-	-	-	**2**	-	-
22RMI57.2	-	-	-	-	**1**	-	-	**3**	-	**2**	-	-	-	-
22RMI57.3	-	-	-	-	-	**2**	-	-	-	**2**	-	-	-	-
22RMI57.4	-	-	-	-	-	**2**	-	-	-	**2**	-	-	-	-
22RMI57.5	-	-	-	-	-	**2**	-	-	-	**2**	-	-	-	-
22RMI57.6	-	-	-	-	-	**2**	-	-	-	**2**	-	-	-	-

1: Low 2: Medium 3: High

Emerging Technologies: A Primer			
Course Code	22ETP58	CIE Marks	100
Teaching Hours/Week (L:T:P)	(0:0:2)	SEE Marks	-
Credits	1	Exam Hours	03

Course Learning Objectives:

1. To develop a strong awareness of the ethical and societal implications associated with emerging technologies.
2. To instil practical skills related to AI (Artificial Intelligence), Blockchain, Digital Twins, RPA (Robotic Process Automation), and Cybersecurity.
3. To enable experiences of working on a team project, allowing students to apply their knowledge and skills to a real-world problem and present their findings effectively.

Module-1: AI and Web 3.0 (06 Hours)

Introduction to Emerging Technologies: Overview of the course, Importance of staying updated with emerging technologies, Ethical and societal considerations.

Artificial Intelligence (AI): Definition and history of AI, Machine learning and deep learning, Applications of AI in various industries, In-Class Assignment: AI in Everyday Life, Homework Assignment: Building a Simple Chatbot.

Web 3.0: Blockchain and Metaverse - Introduction to Blockchain technology, Metaverse and its potential, In-Class Assignment: Creating a Simple Smart Contract, Homework Assignment: Exploring a Metaverse Platform.

Module-2: Smart Manufacturing and Robotic Process Automation (06 Hours)

Smart Manufacturing and Digital Twins: The concept of Smart Manufacturing, Role of IoT and sensors, Digital Twins and their applications, In-Class Assignment: Explore the designs of Digital Twins, Homework Assignment: Analysing a Smart Manufacturing Case Study.

Robotic Process Automation: Understanding Robotic Process Automation (RPA), Types of robots and their applications, Human-robot collaboration, In-Class Assignment: Automating a Task with RPA, Homework Assignment: Researching Advances in Robotics.

Module-3: Cybersecurity and Quantum Computing (06 Hours)

Cybersecurity: Importance of cybersecurity in the digital age, Threats and vulnerabilities, Security best practices, In-Class Assignment: Ethical Hacking Simulation, Homework Assignment: Creating a Cybersecurity Plan.

Quantum Computing: Introduction to Quantum Mechanics, Quantum bits (qubits) and quantum gates, Quantum supremacy and real-world applications. Homework Assignment: Exploring Quantum Computing Research.

Module-4: Project Work (06 Hours)

Team Formation, Synopsis submission, Mid-Term Progress Review, Final Project Presentation.

Course Outcomes: At the end of the course the student will be able to:	
22ETP58.1	Assess the ethical and societal impacts of emerging technologies, demonstrating critical thinking skills.
22ETP58.2	Apply AI and Web 3.0 concepts to develop practical solutions and explore real-world applications.
22ETP58.3	Apply RPA principles and tools to automate common tasks to boost productivity.
22ETP58.4	Explain common cybersecurity threats and recommend best practices to safeguard digital assets.
22ETP58.5	Explain the fundamentals of quantum computing and its real-world applications.
22ETP58.6	Develop a solution using emerging technologies for a real-world problem in teams.

Sl. No.	Title of the Book	Name of the Author/s	Name of the Publisher	Edition and Year
Textbooks				
1	Artificial Intelligence: A Modern Approach	Stuart Russell, Peter Norvig	Pearson	Fourth Edition, 2020
2	Blockchain Technology	Chandramouli Subramanian, Asha A George, Abhilash K A and Meena Karthikeyan	Universities Press (India) Pvt. Ltd.	First Edition 2020
3	Metaverse and Web 3: A Beginner's Guide: A Beginner's Guide: A Digital Space Powered with Decentralized Technology	Utpal Chakraborty	BPB Publications	First Edition, 2022
4	Learning Robotic Process Automation: Create Software robots and automate business processes with the leading RPA tool - UiPath	Alok Mani Tripathi	Packt Publishing	First Edition 2018
5	Cybersecurity: The Beginner's Guide: A comprehensive guide to getting started in cybersecurity	Dr. Erdal Ozkaya	Packt Publishing Limited	First Edition 2019
6	Quantum Computing: A Gentle Introduction	Eleanor G. Rieffel, Wolfgang H. Polak.	MIT Press	First Edition 2014
Reference Books				
1	Smart Manufacturing Technologies for Industry 4.0: Integration, Benefits, and Operational Activities	Edited By: Jayakrishna Kandasamy, Kamalakanta Muduli, V. P. Kommula, Purushottam L. Meena	CRC Press	First Edition 2022
2	The Robotic Process Automation Handbook: A Guide to Implementing RPA Systems	Tom Taulli	Apress Berkeley, CA	2020
3	The Cyber Security Handbook: Prepare for, respond to and recover from cyber-attacks with the IT Governance Cyber Resilience Framework (CRF)	Alan Calder	IT Governance Publishing	First Edition 2020

Web links/Video Lectures:

Introduction to Emerging Technologies:

1. https://aiethics.princeton.edu/case-studies/case-study-pdfs/
2. https://research.aimultiple.com/ai-ethics/
3. https://news.harvard.edu/gazette/story/2020/10/ethical-concerns-mount-as-ai-takes-bigger-decision-making-role/
4. https://www.sciencedirect.com/science/article/pii/S0268401223000816
5. https://www.youtube.com/watch?v=G2fqAlgmoPo
6. https://www.youtube.com/watch?v=zizonToFXDs

Web 3.0: Blockchain and Metaverse

1. What is Ethereum? | ethereum.org
2. Navigating Remix — Remix - Ethereum IDE 1 documentation (remix-ide.readthedocs.io)

3. Solidity — Solidity 0.6.8 documentation (soliditylang.org)
4. https://www.youtube.com/watch?v=nalMdCI_pv8&t=765s
5. The Decentralized Autonomous Organization and Governance Issues by Usman W. Chohan :: SSRN
6. Ethereum Smart Contract Best Practices (consensys.github.io)
7. https://hackernoon.com/hack-solidity-reentrancy-attack

Smart Manufacturing and Digital Twins:

1. https://www.youtube.com/watch?v=nwFed03fS_s
2. https://www.youtube.com/watch?v=ScmK-bKJ4MI

RPA and Robotics:

1. https://www.youtube.com/watch?v=9URSbTOE4YI
2. https://www.youtube.com/watch?v=UEbw7dIOg0g
3. https://www.uipath.com/resources/automation-case-studies
4. https://www.ibm.com/products/robotic-process-automation/case-studies

Cybersecurity:

1. https://www.getastra.com/blog/security-audit/what-is-vapt/
2. https://owasp.org/www-project-top-ten/
3. https://owasp.org/www-project-mutillidae-ii/
4. https://www.youtube.com/watch?v=JAtwZoW76-I
5. Threat modelling (STRIDE framework): https://learn.microsoft.com/en-us/azure/security/develop/threat-modeling-tool-threats
6. Cyber Kill Chain:https://www.lockheedmartin.com/en-us/capabilities/cyber/cyber-kill-chain.html

Quantum Computing:

1. https://www.youtube.com/watch?v=e3fz3dqhN44
2. https://quantumai.google/

Course Articulation Matrix

Course Outcomes (COs)	Program Outcomes (POs)											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
22ETP58.1	-	-	-	-		**3**	-	**2**	-	--	-	-
22ETP58.2	-	**2**	-	-	**3**	-	-	-		-	-	**1**
22ETP58.3	-	-	-	**3**	**2**	-	-	-	--	-	-	-
22ETP58.4	-	-	-	-	**3**	-		-	-	-	-	**1**
22ETP58.5	**2**	-	-	-	**3**	-	-	-	-	-	-	-
22ETP58.6	-	-	**2**	-	**3**	-	--	-	**2**	-	-	**1**

1: Low 2: Medium 3: High

VI Semester

System Software & Compiler Design			
Course Code	**22CSE61**	CIE Marks	50
Course Type (Theory/Practical/Integrated)	Integrated	SEE Marks	50
		Total Marks	100
Teaching Hours/Week (L:T:P)	2:2:2	SEE	3 Hours
Total Hours	40 hours Theory + 10 Lab slots	Credits	04

Course Learning Objectives: The objective of the course is to

- Analyze the working of system software.
- Identify source file, object file and executable file structures and libraries.
- Distinguish the front-end and back-end phases of the compiler and their importance.
- Develop the knowledge of implementing lexical analyzer and syntax analyzer.

Module-1 System Software, Assemblers (8 hours)

Background: Introduction to System Software, Machine Architecture of SIC and SIC/XE. **Assemblers**: Basic assembler functions, machine dependent assembler features, machine independent assembler features.

Text book 1: Chapter 1: 1.1, 1.2, 1.3.1, 1.3.2 Chapter 2: 2.1-2.3

Module-2 Compiler Design, Lexical analysis (8 hours)

Introduction: Language Processors, The structure of a compiler, The evaluation of programming languages, The science of building a compiler, Applications of compiler technology. **Lexical Analysis**: The role of lexical analyzer, Input buffering, Specifications of token, recognition of tokens.

Text book 2: Chapter 1: 1.1-1.5 Chapter 3: 3.1-3.4

Module-3 Syntax Analysis (8 hours)

Syntax Analysis: Introduction, Context Free Grammars, Writing a grammar, Top-Down Parsers, Bottom-Up Parsers.

Text book 2: Chapter 4: 4.1-4.5

Module-4 Lex and Yacc (8 hours)

Lex and Yacc: The Simplest Lex Program, Grammars, Parser-Lexer Communication, A YACC Parser, The Rules Section, Running LEX and YACC, LEX and Hand-Written Lexers, Using LEX - Regular Expression, Examples of Regular Expressions, A Word Counting Program, Using YACC - Grammars, Recursive Rules, Shift/Reduce Parsing, What YACC Cannot Parse, A YACC Parser - The Definition Section, The Rules Section, The LEXER, Compiling and Running a Simple Parser, Arithmetic Expressions and Ambiguity.

Text book 3: Chapter 1, 2 and 3

Module-5 Syntax Directed Translation, Intermediate Code Generation, Code Generation (8 hours)

Syntax Directed Translation: Syntax directed definition, evaluation orders for SDD's, Applications of syntax directed Translation scheme.

Intermediate code generation: Variants of syntax trees, Three address code.

Code generation: Issues in the design of a code generator, The target language.

Text book 2: Chapter 5.1-5.3, 6.1-6.2, 8.1-8.2

PRACTICAL MODULE

A–Demonstration

A1. Demonstration of LEX program

A2. Demonstration of YACC program

B–Exercise (compulsorily to be conducted):

B1. Write a LEX program to recognize valid arithmetic expressions. Identifiers in the expression could be only integers and operators could be + and *. Count the identifiers & operators present and print them separately.

B2. Write a Lex program to count the number of characters, words, lines, tabs and spaces used in the input file.
B3. Write a LEX program to eliminate comment lines in a C program and copy the resulting program into a separate file.
B4. Write a Lex program to search for a word in a file.

C–Exercise (compulsorily - any 4 to be conducted):

C1. Write a YACC program to recognize valid identifiers, operators, and keywords in the given text (C program) file.
C2. Develop, Implement, and Execute a program using YACC tool to recognize all strings ending with b preceded by n a's using the grammar an b (note: input n value)
C3. Write a YACC program to evaluate arithmetic expressions involving operators: +, -, *, and /.
C4. Design, develop and implement YACC/C program to construct Predictive / LL(1) Parsing Table for the grammar rules: A →aBa , B →bB | ε. Use this table to parse the sentence: abba$
C5. Design, develop and implement YACC/C program to demonstrate Shift Reduce Parsing technique for the grammar rules: E →E+T | T, T →T*F | F, F →(E) | id and parse the sentence: id + id * id.

D–Open Ended Experiments (any two):

D1. Using the appropriate tool (Lex/Yacc), write a program to recognize for statements of C.
D2. Using the appropriate tool (Lex/Yacc), write a program to recognize if-else statements of C.
D3. Using the appropriate tool (Lex/Yacc), write a program to recognize switch statements of C.
D4. Using the appropriate tool (Lex/Yacc), write a program to recognize while statements of C.

Course Outcomes: At the end of the course the student will be able to:	
22CSE61.1	Choose the functions and features in assembling SIC/XE assembly language programs.
22CSE61.2	Identify and implement various techniques in designing and developing lexical analyzer.
22CSE61.3	Utilize various techniques and algorithms in designing and developing the parsers.
22CSE61.4	Make use of lex and yacc tools for implementing different concepts of system software.
22CSE61.5	Analyze the implementation aspects of a compiler using Syntax-Directed Translation and Code generation.
22CSE61.6	Examine the techniques of Lex and Yacc to recognize different statements in programming languages.

Sl. No.	**Title of the Book**	**Name of the Author/s**	**Name of the Publisher**	**Edition and Year**
Textbooks				
1	System Software	Leland. L. Beck, D Manjula	Pearson	3rd Edition, 2012
2	Compilers-Principles, Techniques and Tools	Alfred V Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman	Pearson	2nd Edition, 2013
3	Lex & Yacc	Doug Brown, John Levine, Tony Mason	O'Reilly Media	1st Edition, 2012
Reference Books				
1	Systems Programming	Srimanta Pal	Oxford university press	1st Edition, 2016
2	Compiler Design	K Muneeswaran	Oxford University Press	1st Edition, 2013

3	Systems Programming and operating system	D. M. Dhamdhere	Tata MC Graw-Hill	2nd Edition, 1993

Web links and Video Lectures (e-Resources):

- **NPTEL Videos on Compiler Design:** **https://archive.nptel.ac.in/courses/106/104/106104123/**
- **NPTEL Video on System Programming:** https://www.youtube.com/watch?v=zj2VsRnji6c&list=PLM-jfaoaU5iy0Zq-NlWycsfzyMKrAV7jw
- **NPTEL Video on System Software:** **https://www.youtube.com/watch?v=VG9VopzV_T0**

Course Articulation Matrix

Course Outcomes (COs)	Program Outcomes (POs)													
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
22CSE61.1	2	2												1
22CSE61.2		2		1										2
22CSE61.3		2		1										2
22CSE61.4				1	3			1						
22CSE61.5				2										1
22CSE61.6				1	3			1	1					

1: Low 2: Medium 3: High

Machine Learning			
Course Code	**22CSE62**	CIE Marks	50
Course Type (Theory/Practical/Integrated)	Integrated	SEE Marks	50
		Total Marks	100
Teaching Hours/Week (L:T:P)	3:0:2	SEE	3 Hours
Total Hours	40 hours Theory + 10 Lab slots	Credits	04

Course Learning Objectives: The objective of the course is to

- Understand the importance of Machine Learning in problem-solving.
- Analyze various learning algorithms using Machine learning.
- Apply the knowledge of Bayesian learning and Neural networks
- Analyze various unsupervised learning techniques.
- Develop the knowledge of Reinforcement learning

Module-1: Introduction to Machine Learning (8 Hours)

Introduction to ML: Introduction, Well posed problems, Examples, Data Representation, Domain Knowledge, Forms of Learning.

Supervised Learning Rationale and Basics: Learning from Observations, Bias & Variance, Occam's Razor Principle, Heuristic search in Inductive Learning, Ensemble Learning, Estimating Generalization Errors, Metrics for Assessing Regression & Classification, Classification on Imbalanced Dataset, Classification Metrics.

T1: 1.1-1.6, 2.1-2.10

Module-2 Statistical Learning (8 hours)

ML and Inferential Statistical Analysis, Descriptive Statistics in Learning Techniques, Bayesian Reasoning: A Probabilistic Approach to Inference, K-NN Classifier, Discriminant and Regression Functions, Linear Regression (LR) with Least Square Error Criterion, Least Square Regression with Regularization, Logistic Regression for Binary Classification Tasks.

T1: 3.1-3.8

Module-3: Decision Tree Learning (8 hours)

Introduction, Basics, Measures for Evaluating Splits, ID3, C4.5 and CART decision trees, Pruning the Tree, Regression Trees, Learning Tree-Based Ensembles, Bagging, Random Forests, Boosting. **T1: 6.1-6.10**

Module-4: Learning with SVM, ANN(8 hours)

Learning with Support Vector Machines: Introduction, Linear Discriminant Functions for Binary Classification, Linear Maximal Margin Classifier, Linear Soft-Margin Classifier

Artificial Neural Network: Introduction, NN Representation, Appropriate Problems, Perceptron, Back Propagation Algorithm, Remarks on Backpropagation Algorithm

T1: 4.1-4.4, T2: 4.1-4.6

Module-5: Data Clustering, Reinforcement Learning (8 hours)

Data Clustering: Unsupervised learning, Exploratory Data Analysis, Cluster Analysis, Standard Clustering Techniques, K-Means Clustering, Expectation-Maximization (EM) Algorithm and Gaussian Mixtures Clustering

Reinforcement Learning: Introduction, The learning task, Q Learning

T1: 7.1-7.6, T2: 13.1-13.3

Laboratory Experiments
Demonstration (All)

1. Visualize decision surfaces of different classifiers using Matlab.
2. Demonstrate Classification using the Classification Lerner App in Matlab using the appropriate dataset. Analyze the results.
3. Demonstrate Regression using the Regression Lerner App in Matlab using the appropriate dataset. Analyze the results.

Structured Inquiry (All)

1. Build an ANN and train with Backpropagation algorithm using appropriate data set. Do not use library/package APIs.
2. Write a program to implement linear regression. Do not use library/package APIs.
3. Implement the non-parametric locally weighted regression algorithm in order to fit data points. Select the appropriate data set for your experiment and draw graphs. Do not use library/package APIs.
4. Write a program to implement the Naïve Bayesian classifier for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering few test data sets. Use Python ML library API's in the program
5. Write a program to implement k-Nearest Neighbour algorithm to classify the Iris data set. Print both the correct and wrong predictions. Analyze the results with various values of k. Use Python ML library API's in the program.
6. Apply EM algorithm to cluster a set of data. Use the same data set for clustering using k-Means algorithm. Compare the results of these two algorithms. Use Python ML library API's in the program.
7. Implement Random Forest Classifier using appropriate dataset and perform hyper parameter tuning. Use Python ML library API's in the program.

Open-ended experiments (Any One)

1. Write a program to build a Decision tree based ID3 algorithm using an appropriate data set and classify new samples. Do not use python library/package APIs
2. Design and implement two ensemble techniques for classifying the data. Use Python ML library API's in the program.

Course Outcomes: At the end of the course the student will be able to:	
22CSE62.1	Understand the rationale and basics of machine learning
22CSE62.2	Apply statistical learning techniques for classification and regression
22CSE62.3	Perform supervised learning using decision trees.
22CSE62.4	Apply the concepts of ANNs and SVMs for supervised learning.
22CSE62.5	Analyze unsupervised machine learning algorithms and apply them for clustering the data.
22CSE62.6	Understand and apply reinforcement learning techniques.

Sl. No.	**Title of the Book**	**Name of the Author/s**	**Name of the Publisher**	**Edition and Year**
Textbooks				
1	Applied Machine Learning	M. Gopal	McGraw Hill	2nd Edition, 2022
2	Machine Learning	Tom Mitchell	McGraw Hill,	1st Edition, 2017
Reference Books				
1	Machine Learning	S. Sridhar, M Vijayalakshmi	Oxford University Press	1st Edition, 2021
2	The Elements of Statistical Learning	Trevor Hastie, Robert Tibshirani, Jerome Friedman	Springer	2nd Edition 2008

Web links and Video Lectures (e-Resources)

- **NPTEL course on Machine Learning,**
  https://onlinecourses.nptel.ac.in/noc22_cs24/preview
- **NPTEL course on Introduction to Machine Learning:**
  https://onlinecourses.nptel.ac.in/noc22_cs29/ preview
- **NPTEL course on Fundamentals Of Artificial Intelligence,**
  https://nptel.ac.in/courses/112103280
- **Resources on Artificial Intelligence**, https://cse.iitk.ac.in/users/cs365/2015/resources.html

Course Articulation Matrix

Course Outcomes (COs)	Program Outcomes (POs)													
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
22CSE62.1	3	2	2	3	2		1	1	1	2	1	2	2	2
22CSE62.2	2	2	3	3	3	2	1	1	2	2	1	2	1	
22CSE62.3	1	2	2	3	3			1	1	2	1	3	1	1
22CSE62.4	2	2	3	3	3	2	1	1	1	2	1	2	1	
22CSE62.5	2	1	2	2	2		1		1	2		2		1
22CSE62.6	2	1	2	3	3		1	1	2	2	1	2		

1: Low 2: Medium 3: High

UI/UX DESIGN			
Course Code	**22CSE63**	CIE Marks	50
Course Type (Theory/Practical/Integrated)	Theory	SEE Marks	50
		Total Marks	100
Teaching Hours/Week (L: T:P)	3:0:0	SEE	3 Hours
Total Hours	40 hours	Credits	03

Course Learning Objectives:

- Explore the Fundamentals of UX Design.
- Examine the Concepts of Menus, User Interfaces, and Design Processes
- Develop and Organize User-Friendly Information Architectures
- Utilize Tools and Techniques for Prototyping
- Apply Methods for Testing and Evaluating UX Designs

Module- Foundational Elements of UI/UX: (8 hours)

User Experience and Why It Matters, Everyday Miseries, Introducing User Experience, From Product Design to User Experience Design, designing (for) Experience: Use Matters, User Experience, and the Web, Good User Experience Is Good Business, Minding Your Users
Meet the Elements, The Five Planes, The Surface Plane, The Skeleton Plane, The Structure Plane, The Scope Plane, The Strategy Plane, Building from Bottom to Top, A Basic Duality, The Elements of User Experience the Strategy Plane, The Scope Plane, The Structure Plane, The Skeleton Plane, The Surface Plane, Using the Elements
Textbook 1: Ch 1,2

Module-2 User Interface Design Process and System Menus (8 hours)

The User Interface Design process- Obstacles, Usability, Human characteristics in Design, Human Interaction speeds, Business Functions-Business definition and requirement analysis, Basic business functions, Design standards. System menus and navigation schemes- Structures of menus, Functions of menus, Contents of menus, Formatting of menus, Phrasing the menu, selecting menu choices, Navigating menus, Kinds of graphical menus.
Textbook 3: Part 2

Module-3 Design Process and Information Architecture: (8 hours)

Defining Project Scope and Requirements, Functional Specification, Prioritizing requirements Content requirements, Information Architecture Basics, Interaction Design Principles, Wireframing and Layout Design.
Textbook 1: Ch 4,5

Module-4 Prototyping and Visual Design (8 hours)

The Skeleton Plane: Interface Design, Navigation Design, and Information Design.
The Surface Plane: Sensory Design: -Defining the surface, Making Sense of the senses, Contract and uniformity, Internal and External Consistency, Color Palettes and Typography
Textbook 1: Ch 6,7

Module-5 Testing and Evaluating UX Designs (8 hours)

Usability Testing on 10 Cents a Day, Usability as common courtesy, Accessibility, Cascading style sheet and you
Textbook 2: Ch 9,10,11

Course Outcomes: At the end of the course the student will be able	
22CSE63.1	Summarize the foundational understanding of UX design.
22CSE63.2	Define the UI design process, its obstacles, and pitfalls in the development process of UI.
22CSE63.3	Identify the need for defining a good structure of menus along with their functions and contents.

22CSE63.4	Develop knowledge and skills to effectively plan, structure, and organize digital products, ensuring user-friendly interfaces.
22CSE63.5	Experiment with various visual design aspects, visual look and feel of the user experiences.
22CSE63.6	Analyze the need of usability testing and identify various testing methods to apply on the designed User Interface.

Sl. No.	Title of the Book	Name of the Author/s	Name of the Publisher	Edition and Year
Textbooks				
1	The Elements of User Experience: User-centered Design for the Web	Jesse James Garrett	New Riders	2nd Edition, 2010
2	Don't Make Me Think: A Common Sense Approach to Web Usability	Steve Krug	New Riders	2nd Edition, 2006
3	The Essential Guide to User Interface Design	Wilbert O. Galitz,	John Wiley & Sons	2nd Edition, 2002
Reference Books				
1	Design the User Interface	Ben Sheiderman	Pearson Education	1998
2	The Essential of User Interface Design	Alan Cooper	Wiley- Dream Tech Ltd	2002

Additional Resources: Web links/NPTEL Courses

- User Interface Design By Prof. Saptarshi Kolay IIT Roorkee: https://onlinecourses.nptel.ac.in
- https://www.andacademy.com/lp/ui-ux-design-diploma
- https://www.interaction-design.org/courses
- https://www.coursera.org/specializations/ui-ux-design

Course Articulation Matrix

Course Outcomes (COs)	Program Outcomes (POs)													
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
22CSE63.1	2												2	
22CSE63.2						2							2	
22CSE63.3			2			2								
22CSE63.4			2			2								
22CSE63.5			2							3				
22CSE63.6			2							3				

1: Low 2: Medium 3: High

Robotic Process Automation			
Course Code	**22CSE641**	CIE Marks	50
Course Type (Theory/Practical/Integrated)	Theory	SEE Marks	50
		Total Marks	100
Teaching Hours/Week (L:T:P)	3:0:0	Exam Hours	3 Hours
Total Hours	40 Hours	Credits	03

Course Learning Objectives:

- Understand the basic concept of RPA
- Acquire knowledge on Bots and its development
- Learn the different types of variables, Control Flow and data manipulation techniques.
- Understand Image, Text and Data Tables Automation
- Know Various types of RPA vendors

Module-1 Introduction to Robotic Process Automation (8 Hours)

RPA Foundations – What is RPA, Flavours of RPA, History of RPA, the benefits of RPA, The Downsides of RPA, A Compared to BPO, BPM and BPA, Consumer Willingness for Automation, The Workforce of the Future.

RPA Skills: On-Premise Vs. the Cloud, Web Technology, Programming Languages and Low Code, OCR, Databases, APIs, AI, Cognitive Automation, Kanban and Waterfall, DevOps, Flowcharts

Textbook 1: Ch 1,2

Module-2 Bot Development (8 Hours)

Introduction, Installation of UiPath, Flowcharts and sequences, Log message, Variables,Loops and conditionals, for each loop, IF/THEN/ELSE conditionals, Switch,Debug, Common UiPath Functions, The UiPath Orchestrator, Best Practices for Bot development.

Textbook 1: Ch 7

Module-3 Sequence, Flowchart, and Control Flow and Data Manipulation (8 Hours)

Sequence, Flowchart, and Control Flow: Sequencing the workflow, Activities, Control flow, various types of loops, and decision making, Step-by-step example using Sequence and Flowchart, Step-by-step example using Sequence and Control flow. Data Manipulation: Variables and Scope, Collections, Arguments - Purpose and use, Data table usage with examples, Clipboard management, File operation with step-by-step example, CSV/Excel to data table and vice versa (with example).

TextBook 2 : Ch 3, Ch 4

Module-4 Taking Control of the Controls (8 Hours)

Taking Control of the Controls: Finding and attaching windows, Finding the control, Techniques for waiting for a control, Act on controls - mouse and keyboard activities, working with UiExplorer, Handling events, Revisit recorder, Screen Scraping, when to use OCR, Types of OCR available, How to use OCR, Avoiding typical failure points.

TextBook 2 : Ch 5

Module-5 Open source RPA and RPA Vendors (8 Hours)

RPA Vendors: UiPath, Automation Anywhere, Blue Prism, EdgeVerve, PEGA, WorkFusion, Kofax,SAP Contextor,NICE, Future of RPA, Open Source RPA- Open RPA, UI.Vision, Robot Framework, Robocorp, Orchestra, TagUI, Conclusion.

Textbook 1: Ch 10,11

Course Outcomes: At the end of the course the student will be able :	
22CSE641.1	Describe RPA Fundamentals and Applications
22CSE641.2	Demonstrate the development and deployment of Bots using UiPath
22CSE641.3	Implement control flow structures (sequences, flowcharts, and loops) and data

	manipulation techniques
22CSE641.4	Apply techniques to control and interact with UI elements in RPA
22CSE641.5	Compare various RPA vendors and explore open-source RPA tools
22CSE641.6	Integrate RPA with advanced technologies like OCR, AI, and Cognitive Automation, leveraging web technologies

Sl. No.	Title of the Book	Name of the Author/s	Name of the Publisher	Edition and Year
Textbooks				
1	The Robotic Process Automation	Tom Taulli	Apress	1st Edition , 2020
2	Learning Robotic Process Automation	Alok Mani Tripathi	Packt Publishing	1st Edition, 2018
Reference Books				
1	"Introduction to Robotic Process Automation :A Primer"	Frank Casale , Rebecca Dilla, Heidi Jaynes, Lauren Livingston	The Institute for Robotic Process Automation (IRPA)	2018
2	Robotic Process Automation: Guide To Building Software Robots, Automate Repetitive Tasks & Become An RPA Consultant	Richard Murdoch	Richard Murdoch & RPA Ultra	2020

Additional Resources: Web links/NPTEL Courses

- https://www.uipath.com/rpa/robotic-process-automation
- https://irpanetwork.com/
- https://www.udemy.com/course/robotic-process-automation-fundamentals-and-build-a-robot/
- https://www.hfsresearch.com/research/the-rpa-bible-your-practical-technical-guide-to-rpa/

Course Articulation Matrix

Course Outcomes (COs)	Program Outcomes (POs)													
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
22CSE641.1	2					1				3				
22CSE641.2		1						3						
22CSE641.3		3							2		2			
22CSE641.4		1	3				2							
22CSE641.5				2			2				2	2		
22CSE641.6	1				3					1				

1: Low 2: Medium 3: High

Agile and Scrum Technology			
Course Code	**22CSE642**	CIE Marks	50
Course Type (Theory/Practical/Integrated)	Theory	SEE Marks	50
		Total Marks	100
Teaching Hours/Week (L:T:P)	3:0:0	Exam Hours	3 Hours
Total Hours	40 Hours	Credits	03

Course Learning Objectives:

- To explain fundamentals of Agile methodology and Principles.
- To explain Scrum principles.
- To explain and apply sprint practices.
- To apply practices of XP and Incremental design.
- To develop methods to eliminate waste.
- To apply Kanban to visualize workflows, manage flow, limit WIP, and drive continuous improvement.

Module-1 Introduction to Agile Methodologies and Agile Principles (8 Hours)

Learning Agile: What is Agile? Understanding Agile Values, No Silver Bullet, Agile to the Rescue, Adding Agile Makes a Difference. A Fractured Perspective, How a Fractured Perspective Causes Project Problems. The Agile Manifesto, Purpose Behind Each Practice. Individuals and Interactions Over Processes and Tools, Working Software Over Comprehensive Documentation, Customer Collaboration Over Contract Negotiation, Responding to Change Over Following a Plan, Principles Over Practices. Understanding the Elephant, Methodologies Help You Get It All in Place at Once, Where to Start with a New Methodology

The Agile Principles: The 12 Principles of Agile Software, The Customer Is Always Right..Right? ,"Do As I Say, Not As I Said". Delivering the Project, Better Project Delivery for the Ebook Reader Project. Communicating an Working Together, Better Communication for the Ebook Reader Project.

Textbook 1: Ch 1 -3

Module-2 Scrum Framework (8 Hours)

Scrum Framework: Overview, Scrum Roles, Scrum Activities and Artifacts, Sprints , Sprint Planning , Sprint Execution , Daily Scrum ,Done ,Sprint Review , Sprint Retrospective , Closing

Scrum And Self-Organizing Teams: The Rules of Scrum, Act I: I Can Haz Scrum?, Everyone on a Scrum Team Owns the Project, The Scrum Master Guides the Team's Decisions, The Product Owner Helps the Team the Value of the Software, Everyone Owns the Project, Scrum Has Its Own Set of Values ,Status Updates Are for Networks!,

Daily Scrum: The Whole Team Uses the Daily Scrum, Feedback and the Visibility-Inspection-Adaptation Cycle, The Responsible Moment, How to Hold an Effective Daily Scrum. The Product Owner Makes or Breaks the Sprint, Visibility and Value, How to Plan and Run an Effective Scrum Sprint.

Textbook 1: Ch 4, Textbook 2 :Ch 2.

Module-3 Sprint and Scrum Planning (8 Hours)

Sprint: Overview, Time boxed, Short Duration, Consistent Duration, No Goal-Altering Definition of Done

Requirements and User Stories: What are user stories, Invest in good stories, Story Mapping.

Scrum Planning and Collective Commitment: User Stories, Velocity, and Generally Accepted Scrum Practices, Make Your Software Useful, User Stories Help Build Features Your Users Will Use, Conditions of Satisfaction, Story Points and Velocity, Burndown Charts, Planning and Running a Sprint Using Stories, Points, Tasks, and a Task Board.

Textbook 1: Ch 4 ,Textbook 2 :Ch 4

Module-4 XP (8 Hours)

XP and Embracing Change: Going into Overtime, The Primary Practices of XP, Programming Practices, Integration Practices, Planning Practices, Team Practices, Why Teams Resist Changes, and How the Practices Help. The Game Plan Changed, but We're Still Losing, The XP Values Help the Team Change Their Mindset, XP Helps Developers Learn to Work with Users, Practices Only "Stick" When

the Team Truly Believes in Them.

XP, Simplicity, AND Incremental Design: Code and Design, Code Smells and XP Teams Look for Code Smells and Fix Them, Hooks, Edge Cases, and Code That Does Too Much. Make Code and Design Decisions at the Last Responsible Moment, Refactoring.

Effective Design Through Incremental Development: Use Continuous Integration to Find Design Problems, Avoid Monolithic Design, Incremental Design and the Holistic XP Practices. Team Members Trust Each Other and Make Decisions Together. The XP Design, Planning, Team, and Holistic Practices Form an Ecosystem.

Textbook 1: Ch 6, 7

Module-5 Lean, Eliminating Waste, And Seeing the Whole (8 Hours)

Lean, Eliminating Waste, And Seeing the Whole: Lean Thinking, Commitment, Options Thinking, and Set-Based Development, Creating Heroes and Magical Thinking. Eliminate Waste, Use a Value Stream Map to Help See Waste Clearly, Gain a Deeper Understanding of the Product, See the Whole, Find the Root Cause of Problems That You Discover. Deliver As Fast As Possible, Use an Area Chart to Visualize Work in Progress, Control Bottlenecks by Limiting Work in Progress.

KANBAN, flow, And Constantly Improving: The Principles of Kanban, Find a Starting Point and Evolve Experimentally from There. Stories Go into the System; Code Comes Out, Improving Your Process with Kanban, Visualize the Workflow, Limit Work in Progress. Measure and Manage Flow, Managing Flow with WIP Limits Naturally Creates Slack. Make Process Policies Explicit So Everyone Is on the Same Page. Emergent Behavior with Kanban.

The Agile Coach: The Principles of Coaching

Textbook 1:Ch 8, 9 , 10

Course Outcomes: At the end of the course the student will be able :	
22CSE642.1	To explain the foundational principles, values, and practices of Agile methodology.
22CSE642.2	To apply the Scrum Framework, manage Sprints, foster self-organizing teams, and deliver high-value outcomes through collaboration and continuous improvement.
22CSE642.3	To explain Scrum and team organization, describe Sprints and their purpose, and apply the concept of User Stories in development.
22CSE642.4	To explain the primary practices of XP, identify and find Design problems, apply incremental design practice.
22CSE642.5	To Implement Lean principles to eliminate waste, optimize processes, and deliver value efficiently using tools like value stream maps.
22CSE642.6	To utilize Kanban to visualize workflows, limit work in progress, manage flow, and foster continuous improvement in Agile teams.

Sl. No.	**Title of the Book**	**Name of the Author/s**	**Name of the Publisher**	**Edition and Year**
Textbooks				
1	Learning Agile	Andrew Stellman, Jill Alison Hart	O'Reilly	1st Edition,2015
2	Essential Scrum : A practical guide to the most popular Agile process	Rubin K	Addison-Wesley,	1st Edition, 2013
Reference Books				
1	Head first Agile,	Andrew stellman, Jennifer Green	O'Reilly	1st Edition, 2017
2	"Scrum Guide"	Ken Schwaber and Jeff Sutherland	Scrum.org, Scrum Alliance	1st Edition, 2020

Additional Resources: Web links/NPTEL Courses

- https://www.agilealliance.org
- https://www.scrum.org
- https://www.scrumtrainingseries.com
- https://www.agilemanifesto.org
- https://www.scrumguides.org
- https://www.theserverside.com/video/Scrum-methodology-explained

Course Articulation Matrix

Course Outcomes (COs)	Program Outcomes (POs)													
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
22CSE642.1	3	2			2				3	2		2		
22CSE642.2	3		2					2	3					
22CSE642.3	3	2	2			2			3	3				
22CSE642.4	3	3	3			2			2					
22CSE642.5	3			2	3				2	3				
22CSE642.6	3	2	3						3					

1: Low 2: Medium 3: High

Go Programming			
Course Code	**22CSE643**	CIE Marks	50
Course Type (Theory/Practical/Integrated)	Theory	SEE Marks	50
		Total Marks	100
Teaching Hours/Week (L: T:P)	3:0:0	SEE	3 Hours
Total Hours	40 hours	Credits	03

- **Course Learning Objectives:**
- To apply the basic program structures and data types of Golang
- To build functions and methods of Golang.
- To utilize linear and non-linear data structures for sorting and searching algorithms in Golang
- To discover the applications of Golang in Network programming and web services.

Module-1 Introduction to Go (8 hours)

Program Structure: Names, declarations, variables, assignments, Type declarations, Packages and Files, Scope
Basic Data types: Integers, floating point numbers, complex numbers, Booleans, strings, constants.
Composite Data types: Arrays, Slices, Maps, Structs, JSON
Textbook 1: Chapter 2, Chapter 3, Chapter 4

Module-2 Functions and Methods (8 hours)

Functions: Function declarations, recursion, multiple return values, errors, function values, anonymous functions, variadic functions, deferred function calls, panic, recover.
Methods: Method declarations, methods with a pointer receiver, composing types by struct embedding, method values and expressions, encapsulation.
Textbook 1: Chapter 5, Chapter 6

Module-3 Foundations of Network Programming with GO (8 hours)

http.Roundtripper: http.Response, http.Request, http.Transport, TCP/IP, IPv4 and IPv6, nc(1) command line utility, Reading configuration of network interfaces, performing DNS lookups: Getting the NS records of a domain, Getting the MS records of a domain
Creating a web server in GO: using the atomic package, profiling an HTTP server, creating a website in GO, HTTP tracing, creating a web client in GO, timing out HTTP connections, wireshark and tshark tools, gRPC and Go
Textbook 2: Chapter 12

Module-4 Go and Microservices (8 hours)

Introduction to microservices: Building a simple web server, reading and writing JASON, Unmarshalling JSON to GO constructs, Routing in net/http, context, RPC in Go standard library.
Designing a great API: Restful API, URIs, URI format, URI path design for REST services, HTTP verbs, URI query design, Response codes, HTTP headers, RPC API, versioning API, object type standardization, Documenting APIs
Textbook 4: Chapter 1, Chapter 2

Module-5 Database and Golang (8 hours)

In-memory storage, File storage: reading and writing CSV files, gob package
Go and SQL: Setting up database, connecting to the database, creating a post, retrieving a post, updating a post, deleting a post, getting all posts, Go and SQL relationships: setting up databases, one-to-many relationships, Go relational mappers: sqlx, Gorm
Textbook 3: Chapter 6

Course Outcomes: At the end of the course the student will be able:	
22CSE643.1	Identify and apply the programming constructs of Golang
22CSE643.2	Develop functions and methods in Golang
22CSE643.3	Utilize linear, non-linear data structures for searching and sorting algorithms

22CSE643.4	Apply Golang for computer network applications
22CSE643.5	Build Golang programs for storing data in CSV files
22CSE643.6	Design Golang programs for managing SQL database

Sl. No.	Title of the Book	Name of the Author/s	Name of the Publisher	Edition and Year
Textbooks				
1	The Go Programming language	Alan A. A. Donovan & Brian W. Kernighan	Addison-Wesley	1st Edition, 2016
2	Mastering Go	Mihalis Tsoukalos	Packt	2nd Edition, 2019
3	Go Web Programming	Sau Sheong Chang	Manning Publications	1st Edition, 2016
4	Building Microservices with Go	Nic Jackson	Packt	1st Edition, 2017
Reference Books				
1	Go in Action	William Kennedy	Manning Publications	1st Edition, 2016

Additional Resources: Web links/NPTEL Courses

- **Go Programming certificate course**: https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013846693824962560169 6_shared/overview

Course Articulation Matrix

Course Outcomes (COs)	Program Outcomes (POs)												
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
22CSE643.1		3	3										3	
22CSE643.2		3	3		3								3	
22CSE643.3		3	3		3								3	
22CSE643.4			3	2	2							2		3
22CSE643.5		3		2	2							2		3
22CSE643.6		3	3	2	2							2		3

1: Low 2: Medium 3: High

MuleSoft Anypoint Platform Fundamentals			
Course Code	**22CSE644**	CIE Marks	50
Course Type (Theory/Practical/Integrated)	Theory	SEE Marks	50
		Total Marks	100
Teaching Hours/Week (L:T:P)	3:0:0	SEE	3 Hours
Total Hours	40 hours	Credits	03

Objectives of the Course: This course content is aiming at the students:

- Understand basics of Integration and need for integration in organizations.
- Build an application network using API-led connectivity and Anypoint Platform.
- Understand how to use Anypoint Platform to discover, consume, design, build, deploy, manage, and govern APIs in organizations.
- Learn how to connect to databases, files, web services, SaaS applications, JMS queues, and more.
- Get knowledge on how to add application logic, handle errors, and transform data using Data Weave.
- Structure applications to facilitate development and deployment on Anypoint Platform
- Learn the methods on how to handle batch data processing

Module-1 Introduction to Anypoint Platform (8 hours)

Integration Basics: Concept of Integration, Protocols (HTTP, HTTPS, JMS, FTP, FTPS, SOAP, JDBC), Data sharing, Data Replication (Batch, ETL), Data Formats (XML, JSON, CSV, etc), Point-to-point, Hub-sopke, Integration Challenges, SOA

Introducing application networks and API-led connectivity: Explain what an application network is and its benefits, describe how to build an application network using API-led connectivity, explain what web services and APIs are, make calls to secure and unsecured APIs

Introducing Anypoint Platform: Describe the benefits of Anypoint Platform and MuleSoft's approach to be successful with it, Describe the role of each component in building application networks, Navigate Anypoint Platform, Locate APIs and other assets needed to build integrations and APIs in Anypoint Exchange, Build basic integrations to connect systems using Flow Designer

Module-2 API (8 hours)

Designing APIs: Define APIs with RAML, the Restful API Modeling Language, Mock APIs to test their design before they are built, Make APIs discoverable by adding them to the private Anypoint Exchange, Create public API portals for external developers

Building APIs: Use Anypoint Studio to build, run, and test Mule applications, Use a connector to connect to databases, Use the graphical DataWeave editor to transform data, Create RESTful interfaces for applications from RAML files, Connect API interfaces to API implementations, Synchronize changes to API specifications between Anypoint Studio and Anypoint Platform

Deploying and managing APIs: Describe the options for deploying Mule applications, Deploy Mule applications to CloudHub, Use API Manager to create and deploy API proxies, Use API Manager to restrict access to API proxies

Module-3 Application Development (8 hours)

Accessing and modifying Mule events: Log event data, Debug Mule applications, Read and write event properties, Write expressions with the DataWeave expression language, Create variables

Structuring Mule applications: Create applications composed of multiple flows and subflows, Pass events between flows using asynchronous queues, Encapsulate global elements in separate configuration files, Specify application properties in a separate properties file and use them in the application, Describe the purpose of each file and folder in a Mule project, Define and manage application metadata

Consuming web services : Consume web services that have an API (and connector) in Anypoint Exchange, Consume RESTful web services, Consume SOAP web services, Pass parameters to SOAP web services using the Transform Message component, Transform data from multiple services to a canonical format.

Module-4 Error Handling (8 hours)
Controlling event fiow: Multicast events, Route events based on conditions, Validate events **Handling errors:** Handle messaging errors at the application, flow, and processor level, Handle different types of errors, including custom errors, Use different error scopes to either handle an error and continue execution of the parent flow or propagate an error to the parent flow, Set the success and error response settings for an HTTP Listener, Set reconnection strategies for system errors **Writing DataWeave transformations**: Write DataWeave expressions for basic XML, JSON, and Java transformations, Write DataWeave transformations for complex data structures with repeated elements, Define and use global and local variables and functions, Use DataWeave functions, Coerce and format strings, numbers, and dates, Define and use custom data types, Call Mule flows from DataWeave expressions, Store DataWeave scripts in external files.
Module-5 Trigger Flows (8 hours)
Triggering flows: Read and write files, Trigger flows when files are added, created, or updated, Trigger flows when new records are added to a database table, Schedule flows to run at a certain time or frequency, Persist and share data in flows using the Object Store, Publish and consume JMS messages **Processing records:** Process items in a collection using the For Each scope, Process records using the Batch Job scope, Use filtering and aggregation in a batch step.

Course Outcomes: Through the MuleSoft Anypoint Platform Fundamentals course, students will:	
22CSE644.1	Acquire a fundamental understanding of routing of messages through the use of the Mule enterprise service bus.
22CSE644.2	Gain knowledge of connectors that allow for integration with a broad number of data sources, such as JDBC, JMS, and SFTP.
22CSE644.3	Able to utilize knowledge and skills to continue learning and adapting to Salesforce technologies.
22CSE644.4	Dive deeply into Application Programming Interfaces, or APIs, as well as RESTful services, which are both provided by MuleSoft.
22CSE644.5	Learn about encrypting sensitive data to protect it and forcing users to verify themselves with credentials to access the system.
22CSE644.6	Utilize and take Advantage, Within the Projects, of the Scripting Components Offered by MuleSoft.

Sl. No.	Title of the Book	Name of the Author/s	Name of the Publisher	Edition and Year
Textbooks				
1	A practitioner's guide to deploying "MuleSoft APIs and integrations for Salesforce enterprise solutions"	Arul Christhuraj Alphonse, Alexandra Martinez, Akshata Sawant	PACKT Publisher	2022
2	"Hands-on MuleSoft Anypoint Platform – Volume 1"	Nanda Nachimuthu	BPB Publishers	2020
3	"Hands-on MuleSoft Anypoint Platform – Volume 2"	Nanda Nachimuthu	BPB Publishers	2021
4	"MuleSoft A Complete Guide"	Gerardus Blokdyk	5 star books Publishers	2021

Reference Books				
1	"Hands-on MuleSoft Anypoint Platform – Volume 3" Implement various Connectors including database, File, SOAP, Email, VM, JMS, AMQP, Scripting, SFTP, LDAP,Java and ObjectStore	Nanda Nachimuthu	BPB Publishers	2022
2	REST API Design Rulebook: Designing Consistent RESTful Web Service Interfaces	Mark Masse	O'Reilly Publishers	2021
3	Mastering API Architecture: Design, Operate and Evolve API-Based Systems	James Gough, Daniel Bryant, Matthew Auburn	O'Reilly Publishers	2022
4	Learn API Testing: Norms, Practices, and Guidelines for Building Effective Test Automation	Jagdeep Jain	Wiley A press	2022

Additional Resources: Web links/NPTEL Courses

- **Use the Trailhead Platform:** https://www.salesforce.com/blog/what-is-trailhead/
- **The MuleSoft Anypoint Developer Fundamentals course:** https://trailhead.salesforce.com/users/strailhead/trailmixes/getting-started-with-anypoint- platform-d ex-401

Course Articulation Matrix

Course Outcomes (COs)	Program Outcomes (POs)													
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
22CSE644.1			2		2						2	2		
22CSE644.2			2		2						2			
22CSE644.3			2		2						2			
22CSE644.4			2		2						2			
22CSE644.5			2		2						2			
22CSE644.6					2						2	2		2

1: Low 2: Medium 3: High

Introduction to Cyber Security			
Course Code	**22CSE651**	CIE Marks	50
Course Type (Theory/Practical/Integrated)	Theory	SEE Marks	50
		Total Marks	100
Teaching Hours/Week (L:T:P)	3:0:0	SEE	3 Hours
Total Hours	40 hours	Credits	03

Course Learning Objectives:

- To familiarize cybercrime terminologies and perspectives.
- To understand Cyber Offenses and Botnets
- To gain knowledge on tools and methods used in cybercrimes
- To understand phishing and identity theft
- To understand the Cyber forensics
- To recognize the various cybercrime and the measures to overcome those cybercrimes

Module-1 Introduction to Cybercrime (8 hours)

Introduction to Cybercrime: Cybercrime: Definition and Origins of the Word, Cybercrime and Information Security, who are Cybercriminals? Classifications of Cybercrimes, Cybercrime: The Legal Perspective, Cybercrime: An Indian Perspective, Cybercrime and the Indian ITA 2000, A Global Perspectives on Cybercrimes

Textbook:1 Chapter 1 (1.1 to 1.9)

Module-2 Cyber Offenses and Botnets(8 hours)

Cyber Offenses: How Criminals Plan Them: Introduction, How criminals plan the attacks, Social Engineering, Cyber Stalking, Cybercafe & cybercrimes. Botnets: The fuel for cybercrime, Attack Vector.

Textbook:1 Chapter 2 (2.1 to 2.7)

Module-3 Tools and Methods used in Cybercrime (8 hours)

Tools and Methods used in Cybercrime: Introduction, Proxy Servers and Anonymizers, Phishing, Password Cracking, Key Loggers and Spyware, Virus and Worms, Trojan Horses and Backdoors, Steganography, DoS and DDOS Attacks, SQL Injection, Attacks on Wireless networks.

Textbook:1 Chapter 4 (4.1 to 4.10, 4.12)

Module-4 Phishing and Identity Theft (8 hours)

Phishing and Identity Theft: Introduction, Phishing: Methods of Phishing, Phising Techniques, Spear Phishing, Types of Phishing Scams, Phishing Toolkits and Spy Phishing, Phishing Countermeasures, Identity Theft: Personally Identifiable Information(PII), Types of Identity Theft, Techniques of ID Theft, Identity Theft: Countermeasures.

Textbook:1 Chapter 5 (5.1. to 5.3)

Module-5 Computer Forensics (8 hours)

Computer Forensics: Introduction, Historical Background of Cyber Forensics, Digital Forensics Science, Need for Computer Forensics, Cyber Forensics and Digital Evidence, Forensic Analysis of E-Mail, Digital Forensic Life cycle, Chain of Custody Concepts, Network Forensics.

Textbook:1 Chapter 7 (7.1. to 7.9)

Course Outcomes: At the end of the course the student will be able :	
22CSE651.1	Explain the cybercrime terminologies and the various cyber laws.
22CSE651.2	Describe Cyber offenses and Botnets
22CSE651.3	Illustrate Tools and Methods used on Cybercrime
22CSE651.4	Explain phishing and identity thefts
22CSE651.5	Illustrate the need of cyber forensics
22CSE651.6	Justify the measures to overcome the cybercrimes

Sl. No.	Title of the Book	Name of the Author/s	Name of the Publisher	Edition and Year
Textbooks				
1	"Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives"	Sunit Belapure and Nina Godbole	Wiley India Pvt Ltd	1st Edition, 2018
Reference Books				
1	Introduction to Security and Network Forensics	Buchanan, William J. (2011).	CRC Press	1st Edition, 2011
2	Principles of Information Security	Michael E. Whitman, Herbert J. Mattord,	Cengage Learning	2nd Edition 2012

Additional Resources: Web links/NPTEL Courses

- https://youtu.be/hXSFdwIOfnE
- https://onlinecourses.nptel.ac.in/noc23_cs127/preview
- https://onlinecourses.swayam2.ac.in/nou19_cs08/preview

Course Articulation Matrix

Course Outcomes (COs)	Program Outcomes (POs)													
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
22CSE651.1			1			1								
22CSE651.2			2			2								
22CSE651.3			1			1								
22CSE651.4			1			1								
22CSE651.5			1			1								
22CSE651.6			1			1								

1: Low 2: Medium 3: High

Object Oriented Programming with Java			
Course Code	**22CSE652**	CIE Marks	50
Course Type (Theory/Practical/Integrated)	Theory	SEE Marks	50
		Total Marks	100
Teaching Hours/Week (L:T:P)	3:0:0	SEE	3 Hours
Total Hours	40 hours	Credits	03

Course Learning Objectives:

- Learn fundamental features of JAVA
- Set up Java JDK environment to create, debug and run simple Java programs using class.
- Learn Inheritance and packages to solve the problem
- Create multi-threaded programs and Exception handling mechanisms.
- Introduce event driven Graphical User Interface (GUI) programming using applets and swings.
- Develop a code to solve the real time problems using JDBC

Module-1 Introduction to Java(8 hours)

The History and Evolution of Java: Java's magic: The Byte code, The Java Buzzwords,
An Overview of Java: Object-oriented programming, A first Simple Java Programs,
Data types, Variables and Arrays, Operators, Control Statements.
Text Book 1: Chapter 1, Chapter 2, Chapter 3, Chapter 4, Chapter 5

Module-2 Classes and Methods (8 hours)

Introducing Classes: Class Fundamentals, Declaring objects, Asssigning Object Reference Variables, Introducing Methods, Constructors, this keyword, garbage collection, The finilize() method
A Closer Look at Methods and Classes: Overloading Methods , Overloading Constructor, ,A Closer Look at Argument Passing ,Returning Objects ,Recursion ,Introducing Access Control Understanding static ,Introducing final ,Exploring the String Class ,Using Command-Line Arguments, Varargs: Variable-Length Arguments.
Text Book 1: Chapter 6, Chapter 7

Module-3 Inheritance, Packages and Interfaces (8 hours)

Inheritance: Inheritance basics, using super, creating multi level hierarchy, method overriding,Using Abstract Class,Using Final with Inheritance
Packages and Interfaces: Packages, Access Protection , Importing Packages ,Interfaces.
Text Book 1: Chapter 8, Chapter 9

Module-4 Exception Handling and Multithreaded Programming(8 hours)

Exception Handling: Exception-Handling Fundamentals ,Exception Types ,Uncaught Exceptions ,Using try and catch , Multiple catch Clauses ,Nested try Statements ,throw, throws ,finally ,Java's Built-in Exceptions ,Creating Your Own Exception Subclasses ,Chained Exceptions ,Using Exceptions
Multi-Threaded Programming: The Java Thread Model , The Main Thread ,Creating a Thread , Creating Multiple Threads ,Using isAlive( ) and join( ) ,Thread Priorities ,Synchronization, Suspending, Resuming, and Stopping Threads, Using Multithreading.
Text Book 1: Chapter 10, Chapter 11

Module-5 JDBC(8 hours)

JDBC Objects: The concept of JDBC, JDBC driver types, JDBC Packages, A brief overview of JDBC process, Database connection, Associating the JDBC/ODBC bridge with the database, Statement objects, ResultSet, Transaction Processing, Metadata, Datatypes, Exceptions. **TextBook 2:Chapter 6**

Course Outcomes: At the end of the course the student will be able to :	
22CSE652.1	Understand the Object-Oriented concepts in JAVA programming.
22CSE652.2	Design and Develop Object Oriented programs using JAVA to solve multidisciplinary problems.

22CSE652.3	Apply generalization and specialization in programs using inheritance features of object-oriented programming and exception handling.
22CSE652.4	Develop user defined packages, interfaces to solve real world problems.
22CSE652.5	Develop programs using exception handling and multithreaded applications with synchronization.
22CSE652.6	Design the Program using JDBC.

Sl. No.	Title of the Book	Name of the Author/s	Name of the Publisher	Edition and Year
Textbooks				
1	Java The Complete Reference	Herbert Schildt	Tata McGraw Hill	7th Edition, 2007
2	J2EE The Complete Reference	Jim Keogh	Tata McGraw Hill	1st Edition, 2008
Reference Books				
1	Programming with Java	Mahesh Bhave and Sunil Patekar	Pearson Education	1st Edition, 2008
2	Java: A Beginner's Guide	Herbert Schildt,	Tata McGraw Hill Education Private Limited,	5th Edition, 2011

Additional Resources: Web links/NPTEL Courses

- https://onlinecourses.nptel.ac.in/noc22_cs47/preview

Course Articulation Matrix

Course Outcomes (COs)	Program Outcomes (POs)													
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
22CSE652.1														2
22CSE652.2														2
22CSE652.3														2
22CSE652.4														2
22CSE652.5					2								2	
22CSE652.6					2						2		2	

1: Low 2: Medium 3: High

Python Programming			
Course Code	**22CSE653**	CIE Marks	50
Course Type (Theory/Practical/Integrated)	Theory	SEE Marks	50
		Total Marks	100
Teaching Hours/Week (L:T:P)	3:0:0	SEE	3 Hours
Total Hours	40 Hours	Credits	03

Course Learning Objectives:

- Apply the syntax and semantics of the Python programming language.
- Illustrate the process of structuring the data using lists, tuples, Dictionaries.
- Implement looping constructs and functions to manipulate strings.
- Demonstrate the use of built-in functions to work with different types of files.
- Implement the program for working with JSON data

Module-1 Python Basics and Control Flow (8 hours)

Python Basics: Entering Expressions into the Interactive Shell, The Integer, Floating-Point, and String Data Types, String Concatenation and Replication, Storing Values in Variables, Your First Program, Dissecting Your Program.

Flow control: Boolean Values, Comparison Operators, Boolean Operators, Mixing Boolean and Comparison Operators, Elements of Flow Control, Program Execution, Flow Control Statements, Importing Modules, Ending a Program Early with sys.exit()

Textbook 1: Chapters 1 – 2

Module-2 Python Functions and Lists (8 hours)

Functions: def Statements with Parameters, Return Values and return Statements, The None Value, Keyword Arguments and print(), Local and Global Scope, The global Statement, Exception Handling.

Lists: The List Data Type, Working with Lists, Augmented Assignment Operators, Methods, Example Program: Magic 8 Ball with a List, List-like Types: Strings and Tuples, References.

Textbook 1: Chapters 3– 4

Module-3 Pattern Matching with Regular Expressions in Python (8 hours)

Pattern Matching with Regular Expressions: Finding Patterns of Text without Regular expression, Finding Patterns of Text with Regular expression, More Pattern Matching with Regular Expression, Greedy and Non-greedy matching, Findall() method, Character classes, Making your own character classes, the Caret and Dollar sign Characters, Wildcard Characters, Case-insensitive matching, Substituting strings with sub() method.

Textbook 1: Chapters 7

Module-4 Dictionaries and NumPy

Dictionaries and Structuring Data: The Dictionary Data Type, Pretty Printing, Nested Dictionaries.

NumPy Arrays and Vectorized Computation: The NumPy ndarray: A Multidimensional Array Object, Universal Functions: Fast Element-Wise Array Functions, Array-Oriented Programming with Arrays, File Input and Output with Arrays, Linear Algebra, Pseudorandom Number Generation.

Textbook 1: Chapters 5

Textbook 2: Chapters 4

Module-5 Pandas and Matplotlib

Getting Started with pandas: Introduction to pandas Data Structures, Essential Functionality, Summarizing and Computing Descriptive Statistics.

Plotting and Visualization: A Brief matplotlib API Primer, Plotting with pandas and seaborn, Other Python Visualization Tools

Textbook 2: Chapters 5, Chapter 9

Course Outcomes: At the end of the course the student will be able :	
22CSE653.1	Implement python programs to solve problems using flow control and decision-making constructs.

22CSE653.2	Implement looping constructs and functions in python programs and Design, create and execute python programs to solve problems using lists.
22CSE653.3	Design, create and execute python programs to solve problems using core data structures like dictionaries and Implement Python Programs using Strings.
22CSE653.4	Implement regular expressions in python program.
22CSE653.5	Develop a python program to work with Numpy.
22CSE653.6	Develop programs for working with Pandas.

Sl. No.	**Title of the Book**	**Name of the Author/s**	**Name of the Publisher**	**Edition and Year**
Textbooks				
1	Automate the Boring Stuff with Python	Al Sweigart	No Starch Press	1st Edition, 2015
2	Python for Data Analysis	Wes McKinney	O'Reilly Media	2nd Edition, 2017
Reference Books				
1	Python for Everybody: Exploring Data Using Python 3	Charles R. Severance	Shroff Publishers	1st Edition, 2017
2	Introduction to Computer Science Using Python	Charles Dierbach,	Wiley	1st Edition, 2015
3	Introduction to Python Programming	Gowrishankar S, Veena A	CRC Press	1st Edition, 2018

Web links and Video Lectures (e-Resources):

- https://www.learnbyexample.org/python/
- https://www.learnpython.org/

Course Articulation Matrix

Course Outcomes (COs)	**Program Outcomes (POs)**													
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
22CSE653.1	2	1	1	1	1				1			1	1	1
22CSE653.2	2	2	1	1	1				1			1	2	2
22CSE653.3	2	2	1	2	2				2			2	2	
22CSE653.4	1	1	1	1	1				1			2	1	
22CSE653.5	1	1	2	1	2				2			1	2	1
22CSE653.6	1	1	2	1	2				2			1	2	1

1: Low 2: Medium 3: High

Major Project Phase I			
Course Code	**22CSE66**	CIE Marks	100
Course Type (Theory/Practical/Integrated)	Practical	SEE Marks	-
		Total Marks	100
Teaching Hours/Week (L:T:P)	(0:0:4)	SEE	-
Total Hours	48 hours	Credits	02

Course Learning Objectives:

5. Utilize fundamental principles of engineering and interdisciplinary knowledge to identify, analyse, and solve complex problems in the project domain.
6. Develop and execute a comprehensive project plan that includes designing, prototyping, testing, and evaluating a system, component, or process to meet specific needs and constraints.
7. Conduct in-depth research, critically review literature, and integrate innovative solutions or techniques within the project framework.
8. Demonstrate effective teamwork, communication, and collaboration skills in a multidisciplinary environment to achieve project objectives.
9. Incorporate ethical considerations, societal impact, and sustainable practices in the project development, while adhering to professional engineering standards.
10. Prepare and present a well-structured project report, supported by technical documentation and visual aids, and confidently defend the work during project viva-voce or presentations.

1. Project Selection

- **Relevance**: Projects should align with the students' specialization and current industry trends.
- **Innovation**: Projects that offer innovative solutions to existing problems or explore new ideas are encouraged.
- **Feasibility**: The project should be achievable within the given timeframe and resources.
- **Team Composition**: Students can work in teams, typically comprising maximum 4 members.

2. Project Proposal

- **Submission**: Students must submit a detailed project proposal (project synopsis) outlining the problem statement, objectives, methodology, expected outcomes, and a work plan.
- **Approval**: The proposal should be reviewed and approved by the Department Project Evaluation Committee (DPEC).

3. Project Execution

- **Regular Meetings**: Students should meet regularly with their project-guide to discuss progress, challenges, and next steps.
- **Documentation**: Maintain detailed documentation throughout the project in a project work-dairy, including design decisions, experiments, and testing results.
- **Milestones**: Set clear milestones and deadlines to ensure steady progress. These could include design completion, initial prototype, testing, etc.

4. Mid-term Review

- **Progress Presentation**: DPEC shall conduct a mid-term review where students present their progress to a panel of faculty members.
- **Feedback**: Provide constructive feedback and guidance to help students refine their projects.

5. Report Submission

- **Report**: The project report should include an abstract, introduction, literature review, methodology, completed portion of the project work with the available results, discussion, conclusion, and references.
- **Code and Data**: If applicable, students should submit their code, datasets, and any other relevant materials.

6. Project Presentations

- **Oral Presentation**: Students should present their projects to a panel, explaining their work, findings, and contributions.

- **Demonstration**: If possible, include a live demonstration of the project or show relevant simulations and results.
- **Q&A**: Be prepared to answer questions from the panel and justify the project's methodology and conclusions.

7. Evaluation Criteria

- **Originality and Innovation**: Assess the novelty and creativity of the project.
- **Technical Competence**: Evaluate the depth of technical knowledge and problem-solving ability demonstrated.
- **Project Execution**: Consider the effectiveness of project planning, adherence to timelines, and quality of implementation.
- **Presentation and Communication**: Judge the clarity and coherence of the project report, presentation, and the ability to answer questions.

8. Plagiarism Check

- **Academic Integrity**: Ensure that the work submitted is original and properly cites all references and sources.
- **Plagiarism Check**: Run all reports through plagiarism detection software and ensure that similarity index is less than the threshold value (25%).

9. Mentorship and Feedback

- **Feedback:** Students are required to consult with their project guide regularly throughout the project work to seek guidance and feedback.
- **Weekly Meetings:** At least one mentorship meeting every week shall be held and recorded in the project work-dairy.

Continuous Internal Evaluation (CIE)		
Description	**Proposed Dates**	**CIE Weightage (Max 100 marks)**
1. Project Synopsis Evaluation	Beginning of the 6th Semester	20 marks
2. Project Progress Evaluation	Middle of the 6th Semester	30 marks
3. Project Report Evaluation (Phase I)	End of the 6th Semester	50 marks
Marks given for the Project Report shall be the same for all project team members, However, marks may differ for presentations and viva-voce depending upon the individual student performance.		
Semester End Examinations (SEE)		
4. There is No SEE component for Major Project Phase I.		

Course Outcomes: At the end of the course the student will be able to :	
22CSE66.1	Demonstrate the ability to identify, define, and solve complex engineering problems using appropriate methodologies and modern tools.
22CSE66.2	Successfully design, develop, and test an engineering solution that meets specified requirements, addressing technical, economic, environmental, and social constraints.
22CSE66.3	Apply research skills to review existing literature, gather and analyze data, and incorporate innovative or state-of-the-art technologies in the project
22CSE66.4	Collaborate effectively within a team, taking on leadership or supportive roles as needed, while ensuring clear communication and efficient project management.
22CSE66.5	Demonstrate awareness of professional ethics, societal impact, and sustainability in the design and implementation of engineering solutions.

22CSE66.6	Exhibit strong written and oral communication skills by preparing technical reports, project documentation, and delivering persuasive project presentations.

Course Articulation Matrix

Course Outcomes (COs)	**Program Outcomes (POs)**													
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
22CSE66.1	**2**	**3**	**-**	**-**	**1**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**
22CSE66.2	**-**	**-**	**3**	**-**	**-**	**2**	**1**	**-**	**-**	**-**	**-**	**-**	**-**	**-**
22CSE66.3	**1**	**2**	**-**	**3**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**
22CSE66.4	**-**	**-**	**-**	**-**	**-**	**1**	**-**	**-**	**3**	**2**	**2**	**-**	**-**	**-**
22CSE66.5	**-**	**-**	**1**	**-**	**-**	**-**	**2**	**3**	**-**	**-**	**-**	**-**	**-**	**-**
22CSE66.6	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**3**	**2**	**1**	**-**	**-**

1: Low 2: Medium 3: High

Environmental Studies			
Course Code	**22CIV67**	CIE Marks	50
Course Type (Theory/Practical/Integrated)	Theory	SEE Marks	50
		Total Marks	100
Teaching Hours/Week (L:T:P)	1:0:0	SEE	2 Hours
Total Hours	15 hours	Credits	01

Course Learning Objectives: This course will enable

- To create environmental awareness among the students.
- To gain knowledge on different types of pollution in the environment.

Module-1 Introduction to Ecology (3 hours)

Ecosystems (Structure and Function): Forest, Desert, Wetlands, River, Oceanic, and Lake. Biodiversity: Types, Value; Hot spots; Threats and Conservation of Biodiversity, Forest Wealth, and Deforestation.

Module-2 Energy Systems and Natural Resources (3 hours)

Advances in Energy Systems (Merits, Demerits, Global Status, and Applications): Hydrogen, Solar, OTEC, Tidal, and Wind.

Natural Resource Management (Concept and case studies): Disaster Management, Sustainable Mining, case studies, and Carbon Trading.

Module-3 Environmental Pollution and Public Health (3 hours)

Environmental Pollution (Sources, Impacts, Corrective and Preventive measures, Relevant Environmental Acts, Case-studies): Surface and Ground Water Pollution; Noise pollution; Soil Pollution, and Air Pollution.

Waste Management & Public Health Aspects: Bio-medical Wastes; Solid waste; Hazardous wastes; E-wastes; Industrial and Municipal Sludge.

Module-4 Environmental Concerns (3 hours)

Global Environmental Concerns (Concept, policies, and case studies): Groundwater depletion/recharging, Climate Change; Acid Rain; Ozone Depletion; Radon and Fluoride problems in drinking water; Resettlement and rehabilitation of people, Environmental Toxicology.

Module-5 Environmental Management (3 hours)

Latest Developments in Environmental Pollution Mitigation Tools (Concept and Applications): G.I.S. & Remote Sensing, Environment Impact Assessment, Environmental Management Systems, ISO14001; Environmental Stewardship- NGOs. Fieldwork: A visit to an Environmental Engineering Laboratory or Green Building or Water Treatment Plant or Waste water treatment Plant; thought to be Followed by an understanding of the process and its brief documentation (Optional).

Course Outcomes: At the end of the course the student will be able to:	
22CIV67.1	Identify the principles of ecology and environmental issues that apply to air, land, and water issues on a global scale,
22CIV67.2	Develop critical thinking and/or observation skills and apply them to the analysis of a problem or question related to the environment.
22CIV67.3	Demonstrate ecology knowledge of a complex relationship between a biotic and abiotic component.
22CIV67.4	Apply their ecological knowledge to illustrate and graph a problem and describe the realities that managers face when dealing with complex issues.
22CIV67.5	Address problems related to waste management and public health aspects
22CIV67.6	List the Standards and latest tools to mitigate pollution.

Sl. No.	Title of the Book	Name of the Author/s	Name of the Publisher	Edition and Year
Textbooks				
1	Environmental studies	Benny Joseph	Tata McGraw-Hill	Edition 3, 2018
2	Environmental Studies – From Crisis to Cure	R Rajagopalan	Oxford Uni-Press	Edition 3, 2020
Reference Books				
1	A Basic Course in Environmental Studies	Surinder Deswal, Anupama Deswal	Dhanpat Rai Publishing Co. (P) Ltd	2017
2	Text book of Environmental Studies for Undergraduate Courses	Bharucha Erach	Universities Press	Edition 2, 2017
3	Environmental Studies	Ranjit R. J Daniels, Jagdish Krishnaswamy	John Wiley & Sons Inc.	2010
4	Perspective in Environmental Studies	Anubha Kaushik, C P Kaushik	New Age International Pvt. Ltd	Edition 3, 2009

Web links and Video Lectures (e-Resources):

1. **Coursera Course: Introduction to Environmental Science Specialization -** https://coursera.org/share/e6c3c98f7215fd49f688e7ede71a0dfc
2. **NPTEL: Environmental Studies -** https://onlinecourses.swayam2.ac.in/cec22_ge22/preview
3. **Directory of Open Access Books (DOAB) - Environmental Leadership Capacity Building in Higher Education: Experience and Lessons from Asian Program for Incubation of Environmental Leaders : http://link.springer.com/openurl?genre=book&isbn=978-4-431- 54339-8**
4. **Lec 31: Environmental Management Systems (EMS) - https://youtu.be/BYqLRGawoH0**
5. **ISO 14001:2015 Training - Environmental Management - https://youtu.be/2f4pBIvXkBs**

Course Articulation Matrix

Course Outcomes (COs)	Program Outcomes (POs)													
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
22CIV67.1	-	-	-	-	-	1	3	-	-	-	-	-	-	-
22CIV67.2	-	-	-	-	-	-	3	-	-	-	-	-	-	-
22CIV67.3	-	-	-	-	-	-	3	-	-	-	-	-	-	-
22CIV67.4	-	-	-	-	-	-	3	-	-	-	-	-	-	-
22CIV67.5	-	-	-	-	-	2	3	1	-	-	-	-	-	-
22CIV67.6	-	-	-	-	1	-	-	2	-	-	-	-	-	-

1: Low 2: Medium 3: High

Innovation and Intellectual Property			
Course Code	**22IIP68**	CIE Marks	100
Course Type (Theory/Practical/Integrated)	Practical	SEE Marks	-
		Total Marks	100
Teaching Hours/Week (L:T:P)	0:0:2	Exam Hours	3 Hours
Total Hours	20 Hrs	Credits	01

Course Learning Objectives:

1. Learn how to use online databases and search tools for conducting patent searches.
2. Develop skills in analyzing patent documents and identifying relevant prior art.
3. Gain proficiency in evaluating the patentability criteria for engineering inventions.
4. Understand the principles of technology gap analysis and patentability search.
5. Understand the patent drafting and patent prosecution.

Module-1 Basics of Intellectual Property Rights (4 Hours)

Creativity, Invention, and Innovation – Introduction to Intellectual Property Rights-types and Importance – Overview of Patent Law – Intellectual Property Management and Commercialization – Emerging Issues in Intellectual Property – Case Studies and Practical Examples – Ethical and Social Considerations.

Activity: Trademark Design Challenge – IP Case Study Analysis

Module-2 Patent Landscape Analysis – Technology Gap Analysis (4 Hours)

Overview of Patent Databases and Search Tools – Keyword Searching, Classification Searching, and Citation Searching – Methods for Analyzing Patent Data: Patent Counts, Citation Analysis, and Patent Mapping – Technology Gap Analysis – Patent Portfolios – Portfolio Strength Assessment – Identification of Key Players – Competitive Intelligence and Market Analysis.

Activity: Conduct Patent Landscape Analysis for the Proposed Capstone Project.

Module-3 Patentability Assessment (6 Hours)

Significance of Patentability Assessment – Patentability Criteria: Novelty, Non-obviousness (Inventive Step), and Industrial Applicability/Utility – Prior Art Searching and Analysis (Keyword Searching, Classification Searching, and Citation Searching) – Non-Patent Literature Search and Other sources of Prior Art – Patentability Reports and Assessments – Case Studies and Practical Examples.

Activity: Conduct a Patentability Search for the Proposed Capstone Project.

Module-4 Patent Drafting and Prosecution (6 Hours)

Significance of Patent Drafting and Prosecution – Structure and Components of a Patent Application – Drafting of Patent Specifications, Claims, and Drawings – Overview of Patent Prosecution Process

Activity: Prepare a Patent Draft for the Proposed Capstone Project.

Course Outcomes: At the end of the course, the student will be able to:	
22IIP68.1	Demonstrate proficiency in utilizing various online databases and search tools for conducting patent searches.
22IIP68.2	Develop advanced skills in analyzing patent documents to identify relevant prior art, including patents, patent applications, and non-patent literature.
22IIP68.3	Demonstrate a comprehensive understanding of the patentability criteria, including novelty, non-obviousness, and utility.
22IIP68.4	Explain the principles and methodologies of technology gap analysis and its relevance to patentability searches.
22IIP68.5	Gain insight into the patent drafting process, including the structure and components of patent applications, and patent prosecution.
22IIP68.6	Apply the acquired knowledge and skills in conducting practical activities, such as conducting patent landscape analysis, patentability searches, and drafting patent

	applications, to solve real-world problems and challenges in the field of intellectual property rights.

Sl. No.	Title of the Book	Name of the Author/s	Name of the Publisher	Edition and Year
Reference Books/Sources				
1	Intellectual Property-A Primer for Academia (For Module 1)	Rupinder Tewari Mamtha Bhardway	Publication Bureau, Panjab University Chandigarh India	2021
2	Patent Landscape Reports (For Module 2)	WIPO - World Intellectual Property Organization https://www.wipo.int/patentscope/en/programs/patent_landscapes		
3	Guidelines for Preparing Patent Landscape Reports (For Module 2)	Anthony Trippe, Patinformatics, LLC	World Intellectual Property Organization (WIPO)	2015
4	Patent Searching - Tools and Techniques (For Module 3)	David Hunt	John Wiley & Sons Inc	First edition 2007
5	The Complete Patent Book_ Everything You Need to Obtain Your Patent (For Module 4)	James L. Rogers	Sphinx Publishing	First Edition 2003

Additional Resources:

1. WIPO Patent Drafting Manual - Second Edition 2023, https://www.wipo.int/edocs/pubdocs/en/wipo-pub-867-23-en-wipo-patent-drafting-manual.pdf
2. Patent Drafting for Beginners - https://elearn.nptel.ac.in/shop/nptel/patent-drafting-for-beginners/?v=c86ee0d9d7ed
3. The Office of the Controller General of Patents, Designs and Trade Marks, Government of India - https://www.ipindia.gov.in/
4. Copyright Office, Government of India - https://copyright.gov.in/
5. United States Patent and Trademark Office - https://www.uspto.gov/
6. Espacenet – patent search - https://worldwide.espacenet.com/
7. The Lens - Free & Open Patent and Scholarly Search - https://www.lens.org/
8. WIPO PATENTSCOPE - https://patentscope.wipo.int/search/en/search.jsf

Course Articulation Matrix

Course Outcomes (COs)	Program Outcomes (POs)													
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
22IIP68.1	2	-	-	-	3	-	-	-	-	-	-	1	-	-
22IIP68.2	2	-	-	3	-	-	-	-	-	-	-	1	-	-
22IIP68.3	3	-	-	-	-	-	-	-	-	-	1	-	-	-
22IIP68.4	2	-	3	-	-	-	-	-	-	-	-	-	-	-
22IIP68.5	1	3	-	-	-	-	-	-	-	-	-	2	-	-
22IIP68.6	-	-	-	-	2	-	-	-	-	-	-	3	-	-

1: Low 2: Medium 3: High

***

Core Values of the Institution

SERVICE

A Josephite will keep service as the prime goal in everything that is undertaken. Meeting the needs of the stakeholders will be the prime focus of all our endeavors.

EXCELLENCE

A Josephite will not only endeavor to serve, but serve with excellence. Preparing rigorously to excel in whatever we do will be our hallmark.

ACCOUNTABILITY

Every member of the SJEC Family will be guided to deliver on assurances given within the constraints set. A Josephite will always keep budgets and deadlines in mind when delivering a service.

CONTINUOUS ADAPTATION

Every member of the SJEC Family will strive to provide reliable and continuous service by adapting to the changing environment.

COLLABORATION

A Josephite will always seek to collaborate with others and be a team-player in the service of the stakeholders.

Objectives

- Provide Quality Technical Education facilities to every student admitted to the College and facilitate the development of all round personality of the students.
- Provide most competent staff and excellent support facilities like laboratory, library and internet required for good education on a continuous basis.
- Encourage organizing and participation of staff and students in in-house and outside Training programmes, seminars, conferences and workshops on continuous basis.
- Provide incentives and encouragement to motivate staff and students to actively involve in research-innovative projects in collaboration with industry and R &D centres on continuous basis
- Invite more and more number of persons from industry from India and abroad for collaboration and promote Industry-Institute Partnership.
- Encourage consultancy and testing and respond to the needs of the immediate neighbourhood.

St Joseph Engineering College

AN AUTONOMOUS INSTITUTION

Affiliated to VTU, Belagavi | Recognised by AICTE, New Delhi

Accredited by NAAC with A+ Grade

B.E. (CSE, ECE, EEE, ME, CIV), MBA & MCA Accredited by NBA, New Delhi

Vamanjoor, Mangaluru - 575 028, Karnataka, India

Ph: 91-824-2868100 / 2263753 / 54 / 55

E-mail: sjec@sjec.ac.in| Website: www.sjec.ac.in