

Deccan Education Society's

Kirti M. Doongursee College of Arts, Science and Commerce (AUTONOMOUS)



Affiliated to

UNIVERSITY OF MUMBAI

Syllabus for
Program: Bachelor of Science
Course: S.Y.B.Sc.
Subject: Computer Science

Choice Based Credit System (CBCS)
with effect from
Academic Year 2024-2025

PROGRAM OUTCOMES

PO	Description
A student completing Bachelor's Degree in Science Program will be able to	
PO1	To think analytically, creatively and critically in developing robust, extensible and highly maintainable technological solutions to simple and complex problems.
PO2	To apply their knowledge and skills to be employed and excel in CS professional careers and/or to continue their education in CS and/or related postgraduate programmes.
PO3	To be capable of managing complex CS projects with consideration of the human, financial and environmental factors.
PO4	To work effectively as a part of a team to achieve a common stated goal in a professional manner.
PO5	To adhere to the highest standards of ethics, including relevant industry and organizational codes of conduct.
PO6	To communicate effectively with a range of audiences both technical and non-technical.
PO7	To develop an aptitude to engage in continuing professional development
PO8	To understand corporate approach towards the projects in a research oriented qualitative manner

Year of implementation- 2024-2025

Name of the Department-Computer Science

Semester	Course Code	Course Title	Vertical	Credit
III	24CSMJ311	Data Structure	Major	2
	24CSMJ31	Data Structure Practical	Major	2
	24CSMJ312	Advance DBMS	Major	2
	24CSMJ32	Advance DBMS Practical	Major	2
	24CSMR321	Java Application Development	Minor	2
	24CSMRP31	Java Application Development Practical	Minor	2
	24CSOE331	Descriptive Statistics	OE	2
	24CSVC341	Advance Application Development	VSC	2
IV	24CSMJ411	Theory of Computation	Major	2
	24CSMJ41	Theory of Computation Practical	Major	2
	24CSMJ412	Software Engineering	Major	2
	24CSMJ42	Software Engineering Practical	Major	2
	24CSMR421	Internet of Things	Minor	2
	24CSMRP41	Internet of Things Practical	Minor	2
	24CSOE431	Probability Theory	OE	2
	24CSSE441	Android Application Development	SEC	2

SEMESTER-III

Course Code	MAJOR SEM – III	Credits	Lectures /Week
24CSMJ311	Paper I - Data Structure	2	2
Course Outcomes: After successful completion of this course, students would be able to • CO1: Describe and understand elementary data structures such as stack, queue, linked list, tree, graphs • CO2: Understand the various concepts related to different data structures • CO3: Design, use and write programs for elementary data structures such as stack, queue, linked list • CO4: Write algorithms / programs and step by step approach in solving problems with the help of fundamental data structures trees, graphs • CO5: Illustrate the use of different data structures on different types of problems.			
Unit	Topics	No of Lectures	
I	Introduction: Data and Information, Data Structure, Classification of Data Structures, Primitive Data Types, Abstract Data Types Simple Data Structures: Array: Introduction, One Dimensional Array, Memory Representation of One Dimensional Array, Multi-dimensional Arrays, Memory Representation of Two Dimensional Arrays, Advantages and Limitations of Arrays. Stack: Introduction, Operations on the Stack, Memory Representation of Stack, Array Representation of Stack, Recursion. Queue: Introduction, Operations on the Queue, Memory Representation of Queue, Array representation of Queue, Circular Queue, Priority Queue Linked List: Linked List, singly Linked List, Traversal of Linked List, Searching, Insertion in Linked List, Deletion from Linked List, Circular Linked List, Doubly Linked List, Traversal, Search, Insert and Delete in a doubly Linked List, Implementing other Data Structures.	15	

II	Trees: Tree terminologies, Binary Tree, Properties of Binary Tree, Memory Representations of Binary Tree - static and dynamic , Binary Tree Traversals, Binary Search Tree, Operations on Binary Search Tree- search, insert, traversal Graphs: Terminologies, graph representations, shortest path	15
Textbooks: - Schaum's Outlines Data structure Seymour Lipschutz Tata McGraw Hill 2 nd 2005 - Data structure – A Pseudocode Approach with C AM Tanenbaum, Y Langsam and MJ Augustein Prentice Hall India 2 nd 2006 Additional References: - A Simplified Approach to Data Structures Lalit Goyal, Vishal Goyal, Pawan Kumar SPD 1 st 2014 - An Introduction to Data Structure with Applications Jean – Paul Tremblay and Paul Sorenson Tata MacGraw Hill 2 nd 2007 - Data Structure and Algorithm Maria Rukadikar SPD 1 st 2017		

Course Code	MAJOR SEM-III Practical	Credits	Lectures/Week
24CSMJ31	CS Practical 311 Data Structure	2	4
Course Outcomes: After successful completion of this course, students would be able to - CO1: List the errors and warnings for the given input. - CO2: Explain and demonstrate the execution process of the programs. - CO3: Solve the problems based on each algorithm. - CO4: Write modularized program code for implementing various operating system algorithms and integrate them.			
1	Writing simple programs using one dimensional and two dimensional array		
2	Write a program to implement a stack using an array.		
3	Write a program: factorial and Fibonacci series using recursion		
4	Write a program to implement a linear queue		
5	Write a program to implement a circular queue		
6	Write a program to implement a singly linked list		
7	Write a program to implement a doubly linked list		
8	Write a program to implement a stack using a linked list.		
9	Write a program to create and traverse a binary search tree		
10	Write a program to construct and print adjacency matrix representation of a graph.		

Course Code	MAJOR SEM – III	Credits	Lectures /Week
24CSMJ312	Paper II - Advance DBMS	2	2
Course Outcomes: CO 1: Remember the concepts of Database Programming Paradigms. CO 2 Understanding the Concepts of PL/SQL CO 3 Analyze different techniques of PL/SQL (stored procedure, functions, cursors , triggers) CO 4 Develop and Implement different techniques of PL/SQL			
Unit	Topics	No of Lectures	
I	Introduction: Defining variables and constants,PL/SQL expressions and comparisons: Logical Operators, Boolean Expressions, CASE Expressions Handling, Null Values in Comparisons and Conditional Statements, PL/SQL Datatypes: Number Types, Character Types, Boolean Type, Date time and Interval Types. The %TYPE Attribute ,The %ROWTYPE Attribute Overview of PL/SQL Control Structures: Conditional Control: IF and CASE Statements, IF-THEN Statement, IF-THEN-ELSE Statement, IF THEN-ELSEIF Statement, CASE Statement, Iterative Control: LOOP and EXIT Statements, WHILE-LOOP, FOR-LOOP, Sequential Control: GOTO and NULL Statements, Continue Sequences: creating sequences, referencing, altering and dropping a sequence	15	
II	Stored Procedures & Functions: Types and benefits of stored procedures, creating stored procedures, executing stored procedures, altering stored procedures, viewing stored procedures. Create a Simple Function, Execute a Simple Function, recursive function. Triggers: Concept of triggers, Implementing triggers – creating triggers, Insert, delete, and update triggers, viewing, deleting and modifying triggers, and enforcing data integrity through triggers. Ex Cursors: Concept of a cursor, types of cursors: implicit cursors;Explicit cursor, Cursor for loops, Cursor variables, parameterized cursors,nested cursors, FOR UPDATE Clause and WHERE CURRENT Clause, Indexing: create, modify and drop index	15	

Textbooks:

- Ivan Bayross, "SQL,PL/SQL -The Programming language of Oracle", B.P.B. Publications
- Michael Abbey, Michael J. Corey, Ian Abramson, Oracle 8i – A Beginner's Guide, TataMcGraw-Hill,
- 3rd edition
- PL/SQL Language Reference 11g, , Sheila Moore, E. Belden, 2nd edition.
- Ramakrishnam, Gehrke, "Database Management Systems", McGraw- Hill, 3rd edition.

Additional References:

- Robert Sheldon, Geoff Moes, Begning MySQL, Wrox Press
- Joel Murach, Murach's MySQL, Murach

Course Code	MAJOR SEM-III Practical	Credits	Lectures/Week
24CSMJ32	CS Practical 312 Advance DBMS	2	4
Course Outcomes: After successful completion of this course, students would be able to •			
1	Writing PL/SQL Blocks with basic programming constructs by including following: a. Sequential Statements b. unconstrained loop		
2	Writing PL/SQL Blocks with basic programming constructs by including following: a. If...then...Else, IF...ELSEIF...ELSE... END IF b. Case statement		
3	Writing PL/SQL Blocks with basic programming constructs by including following: a. While-loop Statements b. For-loop Statements c. Unconstrained loops		
4	Writing PL/SQL Blocks with basic programming constructs by including Sequences: a. Creating simple Sequences with clauses like START WITH, INCREMENT BY, MAXVALUE, MINVALUE, CYCLE NOCYCLE, CACHE NOCACHE, ORDER NOORDER. b. Creating and using Sequences for tables.		
5	Writing Procedures in PL/SQL Block (IN, OUT, INOUT, DEFAULT keywords). a. Create an empty procedure, replace a procedure and call procedure		

	b. Create a stored procedure and call it c. Define procedure to insert data d. A forward declaration of procedure
6	Writing Functions in PL/SQL Block. a. Define and call a function b. Define and use function in select clause, c. Call function in dbms_output.put_line d. Recursive function e. Count Employee from a function and return value back f. Call function and store the return value to a variable
7	Creating and working with Insert/Update/Delete Trigger using Before/After clause.
8	Write an Implicit and explicit cursor to complete the task.
9	Create packages and use it in SQL block to complete the task.
10	Create Index, modify index and drop index

Course Code	MINOR SEM – III	Credits	Lectures /Week
24CSMR321	Java Based Application	2	2
Course Outcomes: After successful completion of this course, students would be able to • CO1 Design java applications using basic concepts. • CO2 Understand the various Object oriented programming concepts using Java • CO3 Develop and implement client-side and server-side programs. • CO4 Develop and implement exception handling, multithreading. • CO5 Knowledge and implementation of the swing package			
Unit	Topics	No of Lectures	
I	OOPS: Introduction to java, Class, Object, Static Keywords, Constructors, this keyword, Inheritance, Inner class, Anonymous Inner class, super keyword, Polymorphism (overloading and overriding), Abstraction, Encapsulation, Abstract Classes, Interfaces. Packages: Introduction to predefined packages, User Defined Packages, Access specifiers Exception Handling: Introduction, Pre-Defined Exceptions, try-catch-finally, throws, throw, User Defined Exceptions Multithreading: Thread Creations, Thread Life Cycle, Life Cycle Methods, Synchronization, wait() notify() notify all() methods	15	
II	Introduction to JFC and Swing: Features of the Java Foundation Classes, Swing API Components, JComponent Class, Windows, Dialog Boxes, and Panels, Labels, Buttons, Check Boxes, Menus, Toolbars, Implementing Action interface, Pane, JScrollPane, Desktop pane, Scrollbars, Lists and Combo Boxes, Text-Entry Components. Event Handling: Delegation Event Model, Events, Event classes, Event listener interfaces, Using delegation event model, adapter classes. JDBC: Introduction, JDBC Architecture, JDBC Drivers, JDBC Connectivity Model, java.sql package, Using Statement,	15	

	PreparedStatement, CallableStatement, ResultSet, Scrollable and Updatable ResultSet, Navigating and manipulating data, ResultSetMetaData, Managing Transactions in JDBC, JDBC Exception classes, BLOB & CLOB	
Textbooks: 1. Herbert Schildt, Java The Complete Reference, Eleventh Edition, McGraw-Hill Education, 2020 2. Bryan Basham, Kathy Sierra, Bert Bates, Head First Servlets and JSP, O'reilly (SPD), 2018 3. Cay S. Horstmann, Gary Cornell, Core Java™ 2: Volume II–Advanced Features Prentice Hall PTR, 2004		

Course Code	MAJOR SEM-III Practical	Credits	Lectures/Week
24CSMJ42	CS Practical Java Based Application	2	4
Course Outcomes: After successful completion of this course, students would be able to - List the errors and warnings for the given input. - Explain and demonstrate the execution process of the programs. - Solve the problems based on each OO concept. - Write modularized program code for implementing OO concepts.			
1	a. Write a program to create a class and implement the concepts of Constructor Overloading, Method Overloading, Static methods b. Write a program to implement the concept of Inheritance and Method Overriding		
2	a. Write a program to implement the concepts of Abstract classes and methods b. Write a program to implement the concept of interfaces		
3	Write a program to define user defined exceptions and raise them as per the requirements.		
4	Write a program using various swing components to design Java applications to accept a student's resume. (Design form)		
5	a. Write a JDBC program that displays the data of a given table b. Write a JDBC program to return the data of a specified record from a given table c. Write a JDBC program to insert / update / delete records into a given table		
6	a. Construct a simple calculator using the JAVA Swings with minimum functionality. b. Construct a GUI using JAVA Swings to accept details of a record of a given table and submit it to the database using JDBC technology on the click of a button.		
7	a. Write a Servlet that accepts a User Name from a HTML form and stores it as a cookie. Write another Servlet that returns the value of this cookie and displays it.		

	b. Write a Servlet that displays the names and values of the cookie stored on the client. c. Write a Servlet that accepts a User Name from a HTML form and stores it as a session variable. Write another Servlet that returns the value of this session variable and displays it
8	a. Write a registration Servlet that accepts the data for a given table and stores it in the database. b. Write a Servlet that displays all the records of a table.
9	a. Write a JSP that accepts a User Name from a HTML form and stores it as a cookie. Write another JSP that returns the value of this cookie and displays it. b. Write a JSP that displays the names and values of the cookie stored on the client. c. Write a JSP that accepts a User Name from a HTML form and stores it as a session variable. Write another JSP that returns the value of this session variable and displays it.
10	a. Write a JSP code that accepts username and password from HTML file and validates the user from the database b. Write a registration JSP that accepts the data for a given table and stores it in the database. c. Write a JSP that displays all the records of a table

Course Code	OPEN ELECTIVE SEM – III	Credits	Lectures /Week
24CSOE331	Paper II Descriptive Statistics	2	2
Course Outcomes: After successful completion of this course, students would be able to • Enable learners to know descriptive statistical concepts • Enable study of probability concept required for Computer learners			
Unit	Topics	No of Lectures	
I	Data types: attribute, variable, discrete and continuous variable Data presentation : frequency distribution, histogram ogive, curves, stem and leaf display Data Aggregation Measures of Central tendency: Mean, Median, mode for raw data, discrete, grouped frequency distribution. Measures dispersion: Variance, standard deviation, coefficient of variation for raw data, discrete and grouped frequency distribution, quartiles, quantiles Real life examples Moments : raw moments, central moments, relation between raw and central moments Measures of Skewness and Kurtosis: based on moments, quartiles, relation between mean, median, mode for symmetric, asymmetric frequency curve.	15	
II	Correlation and Regression: bivariate data, scatter plot, correlation, nonsense correlation, Karl pearson's coefficients of correlation, independence. Linear regression: fitting of linear regression using least square regression, coefficient of determination, properties of regression coefficients (only statement) Probability: Random experiment, sample space, events types and operations of events Probability definition: classical, axiomatic, Elementary Theorems of probability (without proof) $0 \leq P(A) \leq 1$, $P(A \cup B) = P(A) + P(B) - P(A \cap B)$ $P(A^c) = 1 - P(A)$ $P(A) \leq P(B)$ if $A \subset B$ Conditional probability, 'Bayes' theorem, independence, Examples on Probability	15	

Textbooks:

1. Trivedi, K.S.(2001) : Probability, Statistics, Design of Experiments and Queuing theory, with applications of Computer Science, Prentice Hall of India, New Delhi

Additional References:

1. Ross, S.M. (2006): A First course in probability. 6 th Edⁿ Pearson
2. Kulkarni, M.B., Ghatpande, S.B. and Gore, S.D. (1999): common statistical tests. Satyajeet Prakashan, Pune
3. Gupta, S.C. and Kapoor, V.K. (1987): Fundamentals of Mathematical Statistics, S. Chand and Sons, New Delhi
4. Gupta, S.C. and Kapoor, V.K. (1999): Applied Statistics, S. Chand and Son's, New Delhi
5. Montgomery, D.C. (2001): Planning and Analysis of Experiments, wiley.

Course Code	VOCATIONAL SKILL COURSE SEM – III	Credits	Lectures/ Week
24CSVC141	Paper I - Advance Application Development (Practical Based)	2	2
Course Outcomes: After successful completion of this course, students would be able to • learn MEANstack structure and its components • Understand Meanstack technologies such as MongoDB, Express.js, AngularJS, and Node.js. • Analyze and integrate the front-end and back-end components of the MEAN stack. • Develop Meanstack Framework for building fast and scalable applications			
Unit	Topics	No of Sessions	
I	Node.js(N): Introduction to Node.js. Installing Node.js. The package.json File. The Node.js Event Loop. The I/O Cycle. The Anatomy of a Node.js Module. Creating Node Modules. Exploring the Node.js HTTP Module. Creating an HTTP Web Server with Node.js. Responding to HTTP Requests. Routing in Node.js. Creating a Sample Node.js Application. MongoDB(M): Introduction to MongoDB. Installing MongoDB. Using MongoDB Compass. Using Mongo Shell Interface. Connecting to MongoDB. Creating Schemas and Models. Querying Documents Using find(). Inserting Documents Using create(). Updating Documents Using findOneAndUpdate(). Deleting Documents Using findOneAndDelete() & deleteMany()	15	
II	Server-Side Development with Express (E): Introduction to the Express Framework. Installing and Testing Express. Creating a Node.js Express App. Restructuring an Express App. Creating Templates. Using Express Middleware Functions. Creating the List Page. Creating the Details Page. Creating the Edit Page. Creating the Add Page. Deleting Data. REST API Basics. Testing REST APIs. Refactoring APIs. Understanding Angular.JS(A): Getting Started with Angular. Creating an Angular Application. Angular Project File Structure. Anatomy of an Angular Component. One-way Data Binding. Two-way Data Binding. Using NgIf Directive. Using NgForOf	15	

	Directive. Angular Modules. Creating NgModules Using Angular Router. Configuring Templates. Creating Navigations. Working with Template-driven Forms. Working with Reactive Forms. Validating Form Data. Services Dependency Injection (DI). Reading Data from Database. Inserting Data into Database. Updating Data in the Database. Delete Data from Database.	
Textbooks: - Node.js, MongoDB and Angular Web Development: The definitive guide to using the MEAN stack to build web applications by Brad Dayley, Brendan Dayley, Caleb Dayley, Pearson, 2018. - Beginning Flutter: A Hands On Guide to App Development by Marco L. Napoli, Wrox, 2019 Additional References: - Full Stack Javascript Development with Mean - MongoDB, Express, AngularJS, and Node.JS by Adam Bretz, Colin J Ihrig, Shroff/SitePoint, 2015 - Practical Flutter by Zammetti Frank, Apress, 2019		

Course Code	VSC SEM-III Practical	Credits	Lectures/ Week
24CSSC451	Advanced Application Development Practical	2	2

Course Outcomes:

After successful completion of this course, students would be able to

- To understand all the necessary and important technologies such as MongoDB, Express.js, AngularJS, and Node.js.
- To understand modern app development using Flutter

Unit	Topics
1	Write a program to implement following concepts in Node.js - Event loop - modules - Creating an HTTP Web Server with Node.js. - Responding to HTTP Requests. - Routing in Node.js. - Creating a Sample Node.js Application.
2	Write a program to create a Node.js Express App.
3	Write a program to create custom middleware in Express.
4	Write a program in express to - Create Templates - List Page

	- Details Page - Edit Page - Add Page - Delete page
5	Write a program to implement CRUD operations on MongoDB.
6	Write a program to perform validation of a form using AngularJS
7	Write a program to create and implement modules and controllers in Angular JS
8	Create an application for Customer / Students records using AngularJS
9	Write a program to create a simple web application using Express, Node JS and Angular JS
10	Create a simple HTML “Hello World” Project using AngularJS Framework and apply ng-controller, ng-model and expressions

SEMESTER-IV

Course Code	MAJOR SEM – IV	Credits	Lectures /Week
24CSMJ411	Paper I- Theory of Computation	2	2
Course Outcomes: After successful completion of this course, students would be able to - CO1: Remember and describe different concepts of finite automata,regular expressions, grammars - CO2: Understand and discuss the concepts, properties, types of automata, regular expressions, grammars, PDA and Turing machines.. - CO3: Construct different automata and regular expressions and minimize a DFA - CO4: Construct CFG, simplify CFG,Represent CFG and construct a PDA and a Turing machine - CO5: Compare and analyze different languages and their representations, properties, acceptor machines			
Unit	Topics	No of Lectures	
I	Automata Theory: Introduction, Defining Automaton, Finite Automaton, Transitions and Its properties, Acceptability by Finite Automaton, Nondeterministic Finite State Machines, Mealy and Moore Machines, Minimizing Automata. Regular Sets: Regular Expressions, Finite automata and Regular Expressions, Pumping Lemma and its Applications, Closure Properties.	15	
II	Formal Languages: Defining Grammar, Derivations, Languages generated by Grammar, Chomsky Classification of Grammar and Languages Context-free Languages: Derivation Tree,Ambiguous Grammar, CFG simplification, Normal Forms, PDA Turing Machines: Concept and problems	15	
Textbooks: - Theory of Computer Science, K. L. P Mishra, Chandrasekharan, PHI,3rd Edition 2019 Additional References: - Introduction to Automata Theory, Languages and Computation, John E. Hopcroft, Pearson Education, 2014 - Introduction to Computer Theory, Daniel Cohen, Wiley, 2nd Edition, 2007 - Introductory Theory of Computer Science, E.V. Krishnamurthy, Affiliated East-West Press, 2009			

Course Code	MAJOR SEM-IV Practical	Credits	Lectures/Week
24CSMJ41	CS Practical 411 Theory of Computation	2	4
Course Outcomes: After successful completion of this course, students would be able to • CO1: List the errors and warnings for the given input. • CO2: Explain and demonstrate the execution process of the programs. • CO3: Solve the problems based on each algorithm. • CO4: Write modularized program code for implementing various operating system algorithms and integrate them.			
1	Write a program for tokenization of given input		
2	Design a Program for creating a machine that accepts three consecutive 1.		
3	Design a Program for creating a machine that accepts the string always ending with 101.		
4	Design a program for accepting decimal numbers divisible by 2.		
5	Write a program for generating language strings for given regular expression $1(0+1)^*0$		
6	Write a program for generating language strings for given regular expression $(0+1)^*10(0+1)^*$		
7	Write a program for generating derivation sequence / language for the given sequence of productions		
8	Design a program for creating a machine which accepts string having equal no. of a's and b's starting with all a's.		
9	Design a PDA that accepts the language $a^n b^n$ where $n > 0$		
10	Design a PDA to accept WCWR where W is any string and WR is the reverse of the string W and C is a Special symbol.		

Course Code	MAJOR SEM – II	Credits	Lectures /Week
24CSMJ412	Paper II - Software Engineering	2	2
Course Outcomes: After successful completion of this course, students would be able to - Plan a software engineering process life cycle, including the specification, design, implementation, and testing of software systems that meet specification, performance, maintenance and quality requirements. - Analyze and translate a specification into a design, and then realize that design practically, using an appropriate software engineering methodology. - Know how to develop the code from the design and effectively apply relevant standards and perform testing, and quality management and practice. - Able to use modern engineering tools necessary for software project management, time management and software reuse.			
Unit	Topics	No of Lectures	
I	Introduction: The Nature of Software, Software Engineering, Professional Software Development, Layered Technology, Process framework, CMM, Process Patterns and Assessment Prescriptive Models: Waterfall Model, Incremental, RAD Models Evolutionary Process Models: Prototyping, Spiral and Concurrent Development Model Specialized Models: Component based, Aspect Oriented development, The Unified Process Phases, Agile Development-Agility, Agile Process, Extreme Programming Requirement Analysis and System Modeling: Requirements Engineering, Eliciting Requirements, SRS Validation, Components of SRS, Characteristics of SRS, Object-oriented design using the UML - Class diagram, Object diagram, Use case diagram, Sequence diagram, Collaboration diagram, State chart diagram, Activity diagram, Component diagram, Deployment diagram.	15	
II	System Design: System/Software Design, Architectural Design, Low-Level Design Coupling and Cohesion, Functional-Oriented Versus Object-Oriented Approach, Design Specifications, Verification for Design, Monitoring and Control for Design Software Measurement and Metrics: Process Metrics and Project Metrics, Software Measurement, Object Oriented Metrics, Software Project Estimation, Decomposition Techniques, LOC based, FP	15	

	based and Use case based estimations, Empirical estimation Models Software Project Management: Estimation in Project Planning Process, Software Scope and Feasibility, Resource Estimation, Empirical Estimation Models – COCOMO II, Estimation for Agile Development, The Make/Buy Decision Project Scheduling: Basic Principles, Relationship Between People and Effort, Effort Distribution, Time-Line Charts	
Textbooks: • Software Engineering, A Practitioner's Approach, Roger S, Pressman, 2019 • Software Engineering: principles and Practices, Deepak Jain, OXFORD University Press, 2008 Additional References: • Software Engineering, Ian Sommerville, Pearson Education, 2017 • Fundamentals of Software Engineering, Fourth Edition, Rajib Mall, PHI, 2018 • Software Engineering: Principles and Practices, Hans Van Vliet, John Wiley & Sons, 2010 • A Concise Introduction to Software Engineering, Pankaj Jalote, Springer		

Course Code	MAJOR SEM-IV Practical	Credits	Lectures/Week
24CSMJP42	CS Practical 412 Software Engineering	2	4
Course Outcomes: After successful completion of this course, students would be able to • List the errors and warnings for the given input. • Explain and demonstrate the execution process of the programs. • Solve the problems based on each OO concept. • Write modularized program code for implementing OO concepts.			
1	Write down the problem statement for a suggested system of relevance.		
2	Perform requirement analysis and develop Software Requirement Specification Sheet (SRS) for suggested systems.		
3	Draw the function oriented diagram: Data Flow Diagram (DFD) and Structured chart.		
4	Draw the user's view analysis for the suggested system: Use case diagram.		
5	Draw the structural view diagram for the system: Class diagram, object diagram.		
6	Draw the behavioral view diagram : State-chart diagram, Activity diagram.		
7	Draw the behavioral view diagram for the suggested system: Sequence diagram, Collaboration diagram.		

8	Draw the implementation and environmental view diagram: Component diagram, Deployment diagram.
9	Perform Estimation of effort using FP Estimation.
10	Prepare timeline chart/Gantt Chart/PERT Chart.

Course Code	MINOR SEM – IV	Credits	Lectures /Week
24CSMR421	Paper I - Internet of Things	2	2
Course Outcomes: After successful completion of this course, students would be able to • CO1- Learn different protocols used for IoT design • CO2- Understand roles of sensors in IoT • CO3- Use sensors and actuators for the design of IoT. • CO4- Analyze the different roles of big data, cloud computing and data analytics in a typical IoT system.			
Unit	Topics	No of Lectures	
I	Introduction to IOT: Understanding IoT fundamentals, IOT Architecture and protocols, Various Platforms for IoT, Real time Examples of IoT, Overview of IoT components and IoT Communication Technologies, Challenges in IOT. Arduino Simulation Environment: Arduino Uno Architecture ,Setup the IDE, Writing Arduino Software, Arduino Libraries, Basics of Embedded C programming for Arduino, Interfacing LED, push button and buzzer with Arduino, Interfacing Arduino with LCD. Sensor & Actuators with Arduino: Overview of Sensors working, Analog and Digital Sensors, Interfacing of Temperature, Humidity, Motion, Light and Gas Sensor with Arduino, Interfacing of Actuators with Arduino. Interfacing of Relay Switch and Servo Motor with Arduino.		
II	Basic Networking with ESP8266 WiFi module: Basics of Wireless Networking, Introduction to ESP8266 Wi-Fi Module, Various Wi-Fi library, Web server- introduction, installation, configuration Posting sensor(s) data to web server. IoT Protocols: M2M vs. IOT, Communication Protocols. Cloud Platforms for IOT: Virtualization concepts and Cloud Architecture, Cloud computing, benefits, Cloud services -- SaaS, PaaS, IaaS, Cloud providers & offerings, Study of IOT Cloud platforms, ThingSpeak API and MQTT, Interfacing ESP8266 with Web services.		
Textbooks: 1. Hakima Chaouchi, – “The Internet of Things Connecting Objects to the Web” ISBN : 978-1-84821-140-7, Wiley Publications 2. Olivier Hersent, David Boswarthick, and Omar Elloumi, – “The Internet of Things: Key			

Applications and Protocols”, Wiley Publications

3. Vijay Madiseti and Arshdeep Bahga, — “Internet of Things (A Hands-on-Approach)”, 1st Edition, VPT, 2014.

4. J. Biron and J. Follett, "Foundational Elements of an IoT Solution", O'Reilly Media, 2016.

5. Keysight Technologies, “The Internet of Things: Enabling Technologies and Solutions for Design and Test”, Application Note, 2016.

References:

1. Daniel Minoli, — “Building the Internet of Things with IPv6 and MIPv6: The Evolving World of M2M Communications”, ISBN: 978-1-118-47347-4, Wiley Publications

2. Pethuru Raj and Anupama C. Raman, "The Internet of Things: Enabling Technologies, Platforms, and Use Cases", CRC Press

Course Code	MAJOR SEM-IV Practical	Credits	Lectures/ Week
24CSMJ43	CS Practical 413 Internet Of Things	2	4
Course Outcomes:			
After successful completion of this course, students would be able to			
• CO1- Learn the fundamentals of IoT and Arduino Uno architecture. • CO2- Choose the right microcontroller and C Programming for various Applications. • CO3- Apply different types of sensors for appropriate circuits. • CO4- Design and develop embedded systems using Arduino.			
Unit	Topics		
	Introduction to Arduino		
1	Introduction to Arduino circuits and breadboarding Blinking of LEDs		
2	Program using Light Sensitive Sensors		
3	Program using temperature sensors		
4	Programs using humidity sensors		
5	Programs using Line tracking sensors		
6	Programs using Ultrasonic Sensors		
7	Programs using digital infrared motion sensors		
8	Programs using gas sensors		
9	Programs using servo motors		
10	Programs making Joystick with Arduino		

Course Code	OPEN ELECTIVE SEM – IV	Credits	Lectures /Week
24CSOE431	Paper I - Probability Distribution	2	2
Course Outcomes: After successful completion of this course, students would be able to • To know how the business is carried out through electronic media • To understand the types and ways of doing business over internet • To apply the knowledge after becoming a professional or an entrepreneur • To analyze the security concerns while transacting using electronic media			
Unit	Topics	No of Lectures	
I	Standard distributions : random variable; discrete, continuous, expectation and variance of a random variable, pmf, pdf, cdf, reliability, Introduction and properties without proof for following distributions; binomial, normal, chi-square, t, F. Examples Hypothesis testing: one sided, two sided hypothesis, critical region, p-value, tests based on t, Normal and F, confidence intervals.	15	
II	Analysis of variance: one-way, two-way analysis of variance Non-parametric tests: need of non-parametric tests, sign test, Wilcoxon's signed rank test, run test, Kruskal-Walis tests. Post-hoc analysis of one-way analysis of variance : Duncan's test Chi-square test of association	15	
Textbooks: • Trivedi, K.S.(2009) : Probability, Statistics, Design of Experiments and Queuing theory, with applications of Computer Science, Prentice Hall of India, New Delhi Additional References: • Ross, S.M. (2006): A First course in probability. 6 th Edⁿ Pearson • Kulkarni, M.B., Ghatpande, S.B. and Gore, S.D. (1999): Common statistical tests. • Satyajeet Prakashan, Pune • Gupta, S.C. and Kapoor, V.K. (2002) : Fundamentals of Mathematical Statistics, S. Chand and Sons, New Delhi • Gupta, S.C. and Kapoor, V.K. (4 th Edition) : Applied Statistics, S. Chand and Son's, New Delhi • Montgomery, D.C. (2001): Planning and Analysis of Experiments, Wiley.			

Course Code	SKILL ENHANCEMENT COURSE SEM – IV	Credits	Lectures /Week
24CSSC451	Paper I - Android Application Development	2	2
Course Outcomes: After successful completion of this course, students would be able to - Describe different concepts of Android. - Understand Android OS, grade, Android Studio. - Apply Kotlin programming concepts to Android application development - Analyze Data Storage-Shared Preference,SQLite Databases and Firebase Real-Time Data			
Unit	Topics	No of Lectures	
I	App Development with Android Studio: Android Architecture, Android Application Framework, Android Virtual Device, Creating and running First Android Application, working with Physical Android Device, Adding Kotlin Files in Android Studio Basics Of Android- Application Components: Activities, Intent, and Broadcast Receiver, Services, Fragment, Activity Life Cycle, Content Provider, Widgets, and Notifications	15	
II	Designing Android UI: User Interface (UI), Layout and Its Types, Layout Attribute, working with Views, Android UI Controls, Styles and Themes, Event Handler, setting up themes in Manifest and from the application, dialog in activity, using intents, fragments Handle Images, Listview And Menu: ImageView, ImageSwitcher, ListView, Menu, and its types, Designing menu in XML, Option menu, Context menu, popup menu, Screen Navigation, RecyclerView, Interaction of Views Implementing Data Persistence: Data Storage-Shared Preference, Internal And External Storage Storing Data Using SQLite Databases, Content Provider, Firebase Real-Time Data	15	
Textbooks: 1. How to Build Android Apps with Kotlin: A hands-on guide to developing, testing, and publishing your first apps with Android, Alex Forrester, Packt Publishing, 2021 2. Android Programming: Crafting UI/UX using Kotlin, SYBGENLearning, Additional References: 1. Head First Android Development: A Learner's Guide to Building Android Apps with			

Kotlin Dawn Griffiths, 3rd Edition, O'Reilly Media, 2021

- 2. Android Studio 4.2 Development Essentials - Kotlin Edition: Developing Android Apps Using Android Studio 4.2, Kotlin and Android Jetpack, Neil Smyth, Payload Media, 2021
- 3. Android Programming with Kotlin for Beginners, John Horton, Packt Publishing, 2019
- 4. Android Development with Kotlin: Enhance your skills for Android development using Kotlin, Marcin Moskala, Packt Publishing

Course Code	SEC SEM-IV Practical	Credits	Lectures/ Week
24CSSC451	Android Application Development Practical	2	4

Course Outcomes:

After successful completion of this course, students would be able to

- Understand both the basic and advanced concepts of Kotlin
- The Analyze of android project is to create a successful real-world product that improves the mobile experience for end users
- Design and develop user interfaces for mobile apps using basic building blocks, UI components and application structure using Emulator

Unit	Topics
1	i. Write a program using Kotlin to implement control structures and loops. ii. Write a program to implement object-oriented concepts in Kotlin.
2	Create an Android application to design screens using different layouts and UI including Button, Edittext, Textview, Radio Button etc.
3	Create an Android application to demonstrate implicit and explicit intents
4	Create an application to create Image Flipper and Image Gallery. On click on the image display the information about the image.
5	i. Create an Android application to demonstrate the use of Broadcast listeners. ii. Create an Android application to create and use services.
6	Create an Android application to demonstrate XML based animation
7	Create a media player application in android that plays audio. Implement play, pause, and loop features.
8	Create an Android application to demonstrate the different types of menus. a. Pop-up Menu b. Context Menu c. Option Menu
9	Create a suitable Android application to store and retrieve data in the SQLite database.
10	Create a suitable Android application to work with Firebase for storing and manipulating data.

Evaluation Scheme for Second Year (UG) under NEP (2 credits)

I. Internal Evaluation for Theory Courses – 20 Marks

- 1) Continuous Internal Assessment(CIA) Assignment - Tutorial/ Case Study/ Project / Presentations/ Group Discussion / Ind. Visit. – 10 marks
- 2) Continuous Internal Assessment(CIA) ONLINE Unit Test – 10 marks

II. External Examination for Theory Courses – 30 Marks

Duration: 1 Hours

Theory question paper pattern: All questions are compulsory.

Question	Based on	Marks
Q.1	Unit I	15
Q.2	Unit II	15

- All questions shall be compulsory with internal choice within the questions.
- Each Question may be subdivided into sub questions as a, b, c, d, etc. & the allocation of Marks depends on the weightage of the topic.

III. Practical Examination

- Each core subject carries 50 Marks.
- Duration: 2 Hours for each practical course.
- Minimum 80% practical from each core subjects are required to be completed.
- Certified Journal is compulsory for appearing at the time of Practical Exam

NOTE: To pass the examination, attendance is compulsory in both Internal & External (Theory + Practical) Examinations.