

Annexure I

B. Voc. in Software Technologies Overview and Structure

Bachelor of Vocation (B. Voc.) in Software Technologies

About the Programme

The B. Voc. (Software Technologies) programme aims to create trained manpower with the requisite practical skills and capabilities for various well-defined job roles and mould students into competent young and highly productive professionals.

The skill-based courses adopted from Sector Skill Council are suitably complemented by well-thought out courses in general education to provide additional knowledge. In addition to professional development the courses offered also aim at inculcating values and attitudes conducive to good citizenship. The outcome is a unique programme that offers students a flexible, new-age avenue for higher education.

Programme Aim

The aim of the Programme is to develop skilled professionals in the area of Software Technologies

Programme Objectives:

As per revised OA23.1.1

Programme Specific Objectives:

- (i) To provide judicious mix of knowledge and skills in the areas of Software Technologies, Development & Application, and General Education.
- (ii) To provide flexibility to the students by means of pre-defined entry and multiple exit points.
- (iii) To provide industry oriented education in Software Technologies that prepares individual to transition into a broad range of career options at each exit point, including continually upgrading professional, communication, analytic, and technical skills.
- (iv) To ensure that students obtain adequate knowledge, skills and competencies, who after successfully completing courses of
 - a. Certificate in Software Technologies is industry ready to be employed as Junior Software Developer.

- b. Diploma in Software Technologies is industry ready to be employed as Web Developer.
 - c. Advanced - Diploma in Software Technologies are industry ready to be employed as Media Developer and Master Trainer for Junior Software Developer.
 - d. Degree in Software Technologies is ready for transition into a broad range of options: industry (*software/application developer, and Associate-Desktop Publishing*), government, professionals, entrepreneurs and in professional education.
- (v) To provide foundation to graduates to pursue professional careers and take up higher learning courses such as MBA, MCA, MCM, as well as research.

Duration and NSQF level of the Programme: As per revised OA23.1.3

The duration of the Programme shall be as given in the table below

Nomenclature	Duration	NSQF Level
Certificate	One Semester	Level 4
Diploma	Two Semesters	Level 5
Advanced Diploma	Four Semesters	Level 6
B. Voc Degree	Six Semesters	Level 7

Intake Capacity: 40

Eligibility: As per revised OA23.1.4

Admission and Fee Criteria: As per revised OA23.1.5

Programme Structure: As per revised OA23.2 - **Presented in Annexure I below**

Scheme of Examination: As per revised OA 23.3

Verification and Revaluation: As per revised OA-23.4

Award of Certificate, Diploma, Advanced Diploma and B.Voc. Degree:

As per revised OA-23.5

Certificate in Software Technologies (Semester I)

After successfully completing the courses of Semester-I, the students are expected to acquire the skills to be employable as **Junior Software Developer**

NSQF Level / Semester	Course Code	Course Title	Course Credits		Marks
Level 4 / Semester I	General Education				
	STG101	Fundamentals of Computers and Programming	Theory	3	75
	STG102	Web Designing Concepts	Theory	3	75
	STG103	Quantitative Techniques	Theory	2	50
	STG104	Environmental Studies – I	Theory	2	50
	STP101	Software Laboratory – I	Practical	2	50
	Skill Development Qualification Pack & On-Job-Training (OJT)				
	STS101	Junior Software Developer(SSC/Q0508)	Theory	7	600
			Practical & OJT	11	
Semester Total				30	900

Diploma in Software Technologies (Semester II)

After successfully completing the courses of Semester-II, the students are expected to acquire the skills to be employable as **Web Developer**

NSQF Level / Semester	Course Code	Course Title	Course Credits		Marks
Level 5 / Semester II	General Education				
	STG201	OOPS with Java	Theory	3	75
	STG202	Graphics Designing	Theory	3	75
	STG203	E-Commerce	Theory	2	50
	STG204	Environmental Studies - II	Theory	2	50
	STP201	Software Laboratory - II	Practical	2	50
	Skill Development				
	Qualification Pack & On-Job-Training (OJT)				
	STS201	Web Developer (SSC/Q0503)	Theory	7	700
Practical & OJT			11		
Semester Total				30	1000

Advanced Diploma in Software Technologies (Semester III)

After successfully completing the courses of Semester-III, the students are expected to acquire the skills to be employable as **Media Developer**

NSQF Level / Semester	Course Code	Course Title	Course Credits		Marks
Level 6 / Semester-III	General Education				
	STG301	Data Structures	Theory	3	75
	STG302	Audio and Visual Media	Theory	3	75
	STG303	Reasoning Techniques	Theory	4	100
	STP301	Software Laboratory - III	Practical	2	50
	Skill Development Qualification Pack & On-Job-Training (OJT)				
	STS301	Media Developer (SSC/Q0504)	Theory	7	700
			Practical & OJT	11	
Semester Total				30	1000

Advanced Diploma in Software Technologies (Semester IV)

After successfully completing the courses of Semester-IV, the students are expected to acquire the skills to be employable as **Master Trainer for Junior Software Developer**

NSQF Level / Semester	Course Code	Course Title	Course Credits		Marks
Level 6 / Semester-IV	General Education				
	STG401	Python Programming	Theory	3	75
	STG402	Software Engineering & Testing	Theory	3	75
	STG403	Creative Thinking	Theory	4	100
	STP401	Software Laboratory - IV	Practical	2	50
	Skill Development				
	Qualification Pack & On-Job-Training (OJT)				
	STS401	Master Trainer for Junior Software Developer - (SSC/Q0509)	Theory	7	700
Practical & OJT			11		
Total				30	1000

B. Voc. Degree in Software Technologies (Semester V)

After successfully completing the courses of Semester-V, the students are expected to acquire the skills to be employable as **User Interface (UI) Developer**

NSQF Level / Semester	Course Code	Course Title	Course Credits		Marks
Level 7 / Semester- V	General Education				
	STG501	Mobile Application Development	Theory	3	75
	STG502	Human Computer Interaction	Theory	3	75
	STG503	Advanced Quantitative Techniques	Theory	4	100
	STP501	Software Laboratory - V	Practical	2	50
	Skill Development Qualification Pack & On-Job-Training (OJT)				
	STS501	UI Developer - (SSC/Q0502)	Theory	7	800
			Practical & OJT	11	
Total				30	1100

B. Voc. Degree in Software Technologies (Semester VI)

After successfully completing the courses of Semester-VI, the students are expected to acquire the skills to be employable as **Software Developer/ Associate – Desktop Publishing**

NSQF Level / Semester	Course Code	Course Title	Course Credits		Marks
Level 7 / Semester- VI	General Education				
	STG601	E1: Advanced Graphics Designing	Theory - Elective	3	75
		E2: RDBMS			
	STG602	Computer Networks	Theory	3	75
	STG603	Entrepreneurship Development	Theory	4	100
	STP601	Software Laboratory - VI	Practical	2	50
	Skill Development Qualification Pack (Any One) & On-Job-Training (OJT)				
	STS601	Software Developer - (SSC/Q0501)	Theory	7	700
		Associate-Desktop Publishing - (SSC/Q2702)			
		Practical & OJT	11		
Total				30	1000

Annexure II

B. Voc. in Software Technologies

Syllabus (Semester – I)

Course Title: Fundamentals of Computers & Programming

Course Code: STG101

Marks: 75

Type: General Theory

Credits: 03

Hours: 45

Prerequisite Courses: None

Course objectives:

- To enhance the students' understanding of information technology
- To have a thorough understanding of the basic computer structure & Storage
- To have understanding of basic concepts of operating system.
- To introduce the basic concepts of programming

Learning Outcome:

After successful completion of this course, the student will

- Be aware of the components of IT System and Binary Number System
- Explain the basics of Computer architecture and Memory Subsystem.
- Explain basic operating system & storage management concepts
- Explain the basic concepts of programming

Syllabus

1.	Information Technology Basics	05 Lectures
	Information Technology : Definition and components; Data & Information- Definition, Types of data, Data Representation - Coding Schemes(ASCII and UNICODE); Software – Definition, Categories of Software, Applications of Information Technology in various sectors, Internet Applications and Emerging Technologies	
2.	Introduction To Programming Concepts	15 Lectures
	Basic Programming Concepts, Steps Involved in Computer Programming – Problem Definition – Outlining The Solution – Algorithms & Flowchart, Types of Errors in programming, Translators, Interpreter & compilers, Languages Types, Programming approaches, Features of good algorithm and programming	
	Developing algorithm, flow charts and pseudo code of different types of problems.	

	Understanding variables, basic Operators, Understanding blocks, Data Types Declaring and using data types: int, float etc. Basic Input and Output	
3.	Number Systems	08 Lectures
	Number Systems: Decimal to binary conversion and vice versa Binary number representation (signed, 1's Complement and 2's complement), Binary Arithmetic - addition, subtraction, Multiplication, Division, Binary to octal, hexadecimal conversion and vice versa. Floating point representation.	
4.	Computer Technology	07 Lectures
	Basic elements of a computer system: Processor, Main Memory, I/O Modules, System Bus Function and structure of a computer, Interconnection of components, Performance of a computer. Overview of Princeton (Von Neumann) and Harvard architecture. Memory Subsystem: Characteristics of memory system, the memory hierarchy, Types of ROM & RAM, Cache memory unit - Concept of cache memory, Organization of a cache memory unit	
5.	Introduction to Operating System & Storage Management	10 Lectures
	Operating Systems: Definition & Functions, Operating system Structure, operating system operations, Relationship between Kernel, OS, and Hardware, Operating system services, System calls, Types of system calls. Storage Management: File System, Concepts, File Organization and Access Methods, Directory and Disk Structure. Secondary Storage Structure - Overview, disk structure, Disk attachment, Disk scheduling	

Recommended Text Books:

1. Pradeep K. Sinha and Priti Sinha, Computer Fundamentals, BPB Publications
2. A. Silberchatz, Galvin, Gagne, Operating System Concepts, Wiley publication, 8th Edition
3. V. Rajaraman, Computer Fundamentals, PHI Learning, 6th edition

Recommended Reference Books:

1. ITL Education Solutions Limited, Introduction to Information Technology, Pearson Education
2. M. Morris Mano, Computer System Architecture, Pearson Education, 3rd Edition, 2008
3. William Stallings, Operating Systems: Internals and Design Principles, Prentice Hall, 6th Edition
4. R.G. Dromey, How to solve it by Computers, Pearson Education

Course Title: Web Designing Concepts

Course Code: STG102

Marks: 75

Type: General Theory

Credits: 03

Hours: 45

Prerequisite Courses: None

Course objectives:

To acquire skills in developing web applications using latest tools and technology in web designing and good user interfaces covering important design principles such as learn ability, visibility, error prevention, efficiency and graphic design

Learning Outcomes:

On completion of the course the student will be able to

- To setup up and use a web server for testing and deploying web applications
- To be able to design simple static web pages using html tags
- To apply styling using standardized pure CSS
- To write client side scripts to validate and perform client side processing using a scripting language
- To use DOM concepts for client side scripting
- Implementation of user interfaces following design principles and using technologies such as HTML, CSS, JavaScript and JQuery.

Syllabus

1.	Web Design Principles - Overview	06 Lectures
	Basic principles involved in developing a web site, Planning process, rules of web designing, designing a navigation bar, Page design, Home Page Layout, Design Concept – learnability, visibility, error prevention, efficiency, graphic design. Design Patterns for GUI – View tree, Listener, Widget, and Model-View-Controller, Approaches to GUI programming – Procedural, Declarative, and Direct Manipulation. Web UI – HTML, JavaScript, JQuery. Brief History of Internet, what is World Wide Web, Why create a website, Web Standards	
2.	Structure and Style with HTML and CSS	
	A. HTML	5 Lectures
	Introduction. The development process, basic HTML, formatting and fonts, commenting code, color, hyperlink, lists, tables, images, simple HTML forms, web site structure, Meta tags, Character entities, Unicode fonts.	
	B. HTML5	5 Lectures
	Introduction, New Elements, Canvas, SVG, Drag/Drop, Geolocation, Video, Audio, Input types, form elements, form attributes, semantic, web storage, app cache, web workers, SSE	
	C. CSS	5 Lectures

	Introduction – Syntax, inline, internal and external style, Id & Class, Backgrounds, Text, Fonts, Links, Lists, Tables. CSS Box Model – Border, Outline, Margin, Padding. Advanced - Grouping/Nesting, Dimension, Display, Positioning, Floating, Align, Pseudo-class, Pseudo-element, Navigation Bar, Image Gallery, Image Opacity, Image Sprites, Media Types, Attribute Selectors.	
	D. CSS3	5 Lectures
	Introduction, Borders, Backgrounds, Gradients, Text Effects, Fonts, 2D Transforms, 3D Transforms, Transitions, Animations, Multiple Columns.	
3.	Javascript	10 Lectures
	Introduction - What is JavaScript, Understanding Events, JavaScript Example, and External JavaScript. Basic Elements – Comment, Variable, Global Variable, Data Types, Operators, If Statement, Switch, Loop: for and while, Function. JavaScript Objects – objects, Array. Browser Object Model - Browser Objects, Window Object, Document Object – getElementById, getElementsByName, getElementsByTagName, innerHTML property, inner Text property. Validation- form validation, email validation.	
4.	Introducing jQuery	05 Lectures
	JQuery : Introduction - Syntax, Selectors, Events. Effects- Hide/Show, Fade, Slide, Animate, stop(), Callback, Chaining. HTML/CSS- Add, Remove, CSS Classes, css(), Dimensions, slider. Traversing, ancestors, descendants, siblings, filtering.	
5.	Introducing XML	04 Lectures
	Introduction to XML , Need , Features ,XML namespaces , XML DTD ,XML Schemas XML Sheets , Types of XML Packages	

Recommended Text Books:

1. Elisabeth Robson, Eric Freeman, —Head First HTML and CSS, O'Reilly
2. Kogent Learning Solutions Inc., —HTML5 Black Book: Covers CSS3, Javascript, XML, XHTML, Ajax, PHP and JQuery, Pearson Education.

Recommended Reference Books:

1. Steven M. Jacobs, Ben Shneiderman, —Designing the User Interface: Strategies for effective human-computer interaction, 5th Edition, Pearson Education
2. Kogent Learning Solutions Inc. HTML 5 in simple steps Dreamtech Press
3. Ivan Bayross, —HTML 5 and CSS 3 Made Simple, BPB publication

Course Title: Quantitative Techniques

Course Code: STG103

Marks: 50

Type: General Theory

Credits: 02

Hours: 30

Prerequisite Courses: None

Course objectives:

To build quantitative aptitude that are essential requirement in understanding various concepts and to solve problems effectively.

Learning Outcomes:

On successful completion of the course, the students will be able to

- Create, solve and interpret basic data and quantitative models.
- Make sound arguments based on quantitative aptitude and/or careful analysis of data.
- Exhibit critical thinking by developing and expressing sound arguments from given premises to related conclusions
- Effectively communicate the substance and meaning of mathematical problems and their solutions.

Syllabus:

1.	Module I	10 Lectures
	Number System, Progressions, Averages, Allegations, Percentage, ratio & proportion	
2.	Module II	10 Lectures
	Profit & loss, discount, simple and compound interest, Time and Work, Time and distance, population growth and depreciation of value of articles	
3.	Module III	10 Lectures
	Mensuration, Cartesian Coordinate System, functions, inequalities, quadratic equations	

Recommended Text Books:

1. Arun Sharma, How to Prepare for Quantitative Aptitude for the CAT, 8th edition, McGraw Hill Education (India) Private Ltd.
2. R S Aggarwal, Quantitative Aptitude for Competitive Examinations, S Chand Publishing; Revised edition (21 February 2017)

Course Title: Environmental Studies - I

Course Code: STG104

Marks: 50

Type: General Theory

Credits: 02

Hours: 30

Prerequisite Courses: None

Course objectives:

- To provide students with basic knowledge of environment and its aspects.
- To sensitize students about environmental issues.

Learning Outcomes:

On successful completion of the course, the students will be able to

- Explain the various aspects of Environment.
- Develop perspectives around environmental issues.
- Take practical steps to conserve the environment and ecosystem

Syllabus:

1.	Environment and Natural Systems	18 Lectures
	Environment: Meaning, Significance, natural resources and alternatives. Man-Nature relation and interaction with respect to Food, Clothing, Shelter and Occupation Ecosystem: Concept, Structure, Functions, Components (producers, consumers, decomposers), Energy flow in an ecosystem, Ecological succession; Ecological niche (concept); major ecosystems in brief Biodiversity: meaning, hotspots of biodiversity in India, threats to biodiversity; bio geographical classification of India; Conservation (Case studies); genetically modified foods Role of the individual in conservation of natural resources.	
2.	Environmental Degradation and Environmental Pollution	12 Lectures
	Environmental Degradation: Meaning; Degradation of Urban Land, Forest and Agricultural Land due to natural causes and human interference. Environmental pollution: Types of Environmental Pollution: Water, Air, Marine, Land, Noise, Thermal Pollution; for each type of pollution - meaning, Quality standards (where applicable), sources of pollution, pollutants, effects	

Recommended Text Books:

1. Mahua Basu, S. Xavier, Fundamental of Environmental Studies, Edition 2016 Cambridge University
2. N. Balsubramanya, Gurudeep, Chatwal , Environmental Studies , Himalaya Publication.
3. Singh, Anoop Kumar, Environmental Management in mining areas, Ishwar Books

Recommended References:

1. S.C. Santra, Environmental Science, New Book Agency Pvt Ltd. Kolkata.
2. Shinde, Pendse, Donge, Environmental Education, Sheth Publication.

3. Dr. Vijay Kumar, Environmental Studies Text Book, Himalaya Publication.
4. Kumari Veena, Environmental Pollution and Health Hazard, Ishwar Book
5. Singh Chandrama Prakash and Kuter, Sustainable Development and Environment

Course Title: Software Laboratory – I

Course Code: STP101

Marks: 50

Type: General Practical

Credits: 02

Hours: 60

Course objectives:

- To identify the various hardware and software components of a PC
- To trouble-shoot basic.
- To apply the concepts and understanding of algorithm, flowchart and pseudo code design
- To implement web designing concepts and principles such as HTML, CSS, JavaScript and JQuery and develop web pages

Learning Outcome:

After successful completion of this course, the student will be able to

- Identify the components of a PC
- Troubleshoot basic hardware and software problems
- Explain the basic commands of Windows and Linux
- Design efficient algorithm and pseudo-code
- Implement the concepts learnt in Web Designing course

List of suggested Practical:

*A student shall complete practical of minimum 20 Hours and 28 Hours duration from **Section A** and **Section B** respectively.*

Section A - Based on STG101

1. Exploring the Functions and Components of a PC [06 Lectures]
 - a. Recognizing PC components
 - b. Identifying BIOS ROM, Accessing BIOS via the CMOS Setup Program, Configuring and Clearing CMOS Setup Program Passwords, Configuring BIOS Setting
 - c. Identifying Internal Expansion Slots, Installing Expansion Cards, Managing Hardware with Device Manager, Removing and Labeling Components and Cables, Removing a Motherboard, Identifying Motherboard Features.
 - d. Troubleshooting Hard Drive Installations, Configuring Multiple Displays.
 - e. Replacing and Upgrading RAM, Adjusting Power Management to Optimize Battery Life
 - f. PC fault finding techniques
2. Exploring Linux / Windows Operating System [04 Lectures]
 - a. Demo/Review of Installing, Partitioning and formatting disk, Installing applications device drivers, creating, modifying and deleting user accounts
 - b. Study of basic OS commands.
- c. Writing algorithms & designing flowcharts of at least 08 different types of problems. [14 Lectures]

Section B - Based on STG102

1. Create a HTML page with the following : [04 Lectures]
 - a. title heading paragraph emphasis strong and image elements
 - b. complex HTML table
 - c. simple HTML Form covering major form elements
 - d. Embed Video in an HTML page
2. Using CSS do the following : [08 Lectures]
 - a. Create a Navigation bar (with dropdown) with CSS
 - b. Create a CSS Grid
 - c. Create a CSS3 based button
 - d. make an image rounded shape
 - e. Create a CSS based sticky footer
 - f. Create CSS3 Corner Ribbon
 - g. Create CSS3 blurry text effect
 - h. Create image cross fade with CSS3 transition
 - i. Set style for link hover active and visited states of hyperlink
3. Write JavaScript functions to : [10 Lectures]
 - a. accept a string as a parameter and converts the first letter of each word of the string in upper case
 - b. Check whether a given credit card number is valid or not.
 - c. Check whether a given value is a valid url or not.
 - d. Check whether a given email addresses is valid or not.
 - e. print an integer with commas as thousands separators
 - f. Remove items from a dropdown list.
4. Use JQuery to : [08 Lectures]
 - a. Disable buttons
 - b. Make textbox read only
 - c. Uncheck check boxes
 - d. Confirm again
 - e. Sort
 - f. Switch rows and columns
5. Introduction to angular JS/reactjs, Json [06 Lectures]

A mini project combining all the technologies learnt using a front-end development framework such as bootstrap is recommended.

Recommended Text Books: *As mentioned in STG101 and STG102*

Course Title: SSC/Q0508 Junior Software Developer
Course Code: STS101
Marks: 600

Type: Skill (Theory, Practical, & OJT)
Credits: 18 (Theory-7, Practical & OJT-11)
Hours: 440

Prerequisite Courses: None

Course/Package objectives:

To enable students to:

- Demonstrate basic computer and internet, aptitude for analyzing information and making logical conclusions
- Design algorithms to solve problems and convert them into code using the appropriate programming language constructs
- Read and execute a test case and record the outcome in the appropriate format
- Provide complete, accurate and up-to-date data/information to the appropriate people in the required formats on time
- Understand various job roles and work activities of a “Junior Software Developer”.
- Provide an opportunity to work on innovative, thought-provoking industry based projects
- Gain new technical and professional development skills, and give the student an opportunity to establish an interest in industrial/commercial activities.
- Develop constructive work habits
- Utilize and strengthen their communication skills and build healthy work relationships
- Determine what professional and personal competencies they need to develop and address in their transition to professional life
- Maintain a healthy, safe and secure working environment

Learning Outcomes:

Students will have competencies and skills as mentioned in the Model Curriculum of JUNIOR SOFTWARE DEVELOPER designed by NASSCOM (REFERENCE ID: SSC/Q0508, version 1.0)

Package Syllabus (Adopted Model Curriculum of Nasscom)

- *Lecture Hours and Marks Distribution as per Nasscom standard mentioned in Model Curriculum*
- *Unit Contents as mentioned in the courseware designed by Nasscom for Junior Software Developer*

1.	Introduction
	Unit 1.1 – Introduction to IT-ITeS Sector Unit 1.2 – Introduction to the Training Program
2.	Core Skills

	Unit 2.1 – Communication Skills Unit 2.2 – Team Work
3.	Basic of IT (SSC/N0506)
	Unit 3.1 – Basic Computer Skills Unit 3.2 – Internet
4.	Assist in Performing Software Construction and Software Testing Entry-Level Tasks in the It Services Industry (SSC/N0506)
	Unit 4.1 – C Programming Unit 4.2 – Analyze Various Concepts of PHP Unit 4.3 – Introduction to MYSQL Unit 4.4 – SQL using Oracle
5.	Manage Work to Meet Requirements (SSC/N9001)
	Unit 5.1 – Self and Work Management
6.	Work Effectively with Colleagues (SSC/N9002)
	Unit 6.1 – Team Work and Communication 211
7.	Maintain a Healthy, Safe and Secure Working Environment (SSC/N9003)
	Unit 7.1 – Hazards at Workplace Unit 7.2 – Dealing with Emergencies
8.	Data and Information Management (SSC/N9004)
	Unit 8.1 – Provide Data /information in Standard Formats
9.	Develop your Knowledge, Skills and Competence (SSC/N9005)
	Unit 9.1 – Learning and Self Development
10.	Employability & Entrepreneurship Skills
	Unit 10.1 – Personal Strengths & Value Systems Unit 10.2 – Digital Literacy: A Recap Unit 10.3 – Money Matters Unit 10.4 – Preparing for Employment & Self Employment Unit 10.5 – Understanding Entrepreneurship Unit 10.6 – Preparing to be an Entrepreneur

On-Job-Training (OJT) Component (Compulsory)

It is required by a student to carry-out and complete OJT in the relevant industry or working environment and in the relevant skill area (Junior Software Developer).

- **Duration of OJT:** Minimum 45 Hours

- A Student need to submit the detailed report of the OJT along with the 'Certificate of Completion'.

Recommended Text Books:

Nasscom courseware module for Junior Software Developer

Qualification Pack Hyperlink:

https://nsdcindia.org/sites/default/files/QP_SSC-Q0508_Junior-Software-Developer.pdf

Syllabus

B. Voc. in Software Technologies (Semester – II)

Course Title: Java Programming

Course Code: STG201

Marks: 75

Type: General Theory

Credits: 03

Hours: 45

Prerequisite Courses: None

Course Objective:

- To learn and implement the concepts of java and object oriented programming, exception & file handling, threading, applets, swing and jdbc.

Learning Outcomes:

On completion of the course students will

- Become acquainted with the concepts of java & objected orientation.
- Explain the concept of exception handling, multithreading, & interface design.
- Design and develop the Object Oriented working modules using AWT, Swing, multithreading, and Jdbc constructs

Syllabus

1.	Introduction to Java & Object-Oriented Paradigm	07 Lectures
	Introduction to Java: Features of Java, JDK Environment, Java and Internet. Object Oriented Programming Concept: Overview of Programming, Paradigm, Classes, Abstraction, Encapsulation, Inheritance, Polymorphism, Difference between OOP & OOC Java Programming Fundamental: Structure of java program, Data types, Constants, Variables, Operators, Keywords, Naming Convention, Decision Making (if, switch), Looping(for, while),Type Casting	
2.	Classes, Objects, and Packages in Java	12 Lectures
	Classes and Objects: Creating Classes and objects, Memory allocation for objects, Constructor, Use of Inheritance, Use of Polymorphism, Method Overloading, Method Overriding, Nested and Inner classes Arrays and Strings Classes: Arrays, Creating an array, Types of Arrays, String class Methods, String Buffer methods. Abstract Class, Interface and Packages: Modifiers and Access Control, Abstract classes and methods, Interfaces, Interfaces versus Abstract Classes, Packages Concept, Creating user defined packages, Collections	

3.	Exception & File Handling	07 Lectures
	Exception Handling: Exception types, Using try catch and multiple catch, Nested try, throw-throws and finally, Creating User defined Exceptions. File Handling: Byte Stream, Character Stream, File IO Basics, File Operations, Creating file, Reading file, Writing File	
4.	Multithreading & Applet Programming	07 Lectures
	Multithreading: Thread - Synchronization - Messaging - Runnable Interface - Inter thread Communication - Deadlock - Suspending, Resuming and stopping threads Applet Programming: Introduction, Types Applet, Applet Life cycle, Creating Applet, Applet tag	
5.	Introduction to AWT, Swing, and Jdbc	12 Lectures
	AWT & Swing: What is AWT & Swing? Difference between AWT and Swing, The MVC Architecture and Swing, Layout Manager and Layouts. The JComponent Classes: – Components (JButton, JLabel, JText, JTextArea, JCheckBox and JRadioButton, JList, JComboBox, JMenu and JPopupMenu Class, JMenuItem and JCheckBoxMenuItem, JRadioButtonMenuItem, JScrollBar), Dialogs (Message, confirmation, input), JFileChooser, JColorChooser Event Handling: Event sources, Listeners, Mouse and Keyboard Event Handling, Adapters, Anonymous inner class Basic Java Data Base Connectivity: JDBC Fundamentals, Establishing Connectivity and working with connection interface, Working with statements, Creating and Executing SQL Statements, Working with Result Set Objects	

Recommended Text Books:

1. E Balagurusamy, Programming with JAVA, 4th Edition, TMH
2. Cay Horstmann, BIG Java, Wiley Publication, 3rd Edition, 2009

Recommended References:

1. Herbert Schildt, Java 7, The Complete Reference, 8th Edition, 2009.
2. Cay S Horstmann, Gary Cornell, Core Java Volume I- Fundamentals, Prentice Hall, 9th Edition
3. Cay S Horstmann, Gary Cornell, Core Java Volume II- Advanced Features, Prentice Hall, 9th Edition

Course Title: Graphics Designing

Course Code: STG202

Marks: 75

Type: General Theory

Credits: 03

Hours: 45

Prerequisite Courses: None

Course Objective:

- To understand the concepts of graphic designing.
- To make them understand electronic procedures of designing/producing and editing graphics using industry standard software (Adobe Creative Suite) and processes.
- To create, format, illustrate, design, and edit/revise the graphic design elements

Learning Outcomes:

Upon completion of this course, the student should be able to:

- Identify and use the Elements and Principles of Design in a decisive fashion
- Explain and use color to communicate ideas to others
- Explain the principles of graphic placement
- Utilize Software tools to edit images, and create artistic imagery

Syllabus

1.	DESIGN ELEMENTS CONCEPTS	05 Lectures
	• Elements of graphic design: Line, Shape, Forms, Space, Colour, Colour wheel, Colour Harmony, Colour meaning in various context such as culture, religion, gender and emotional factor, texture, Type	
2.	DESIGN PRINCIPLES CONCEPTS	05 Lectures
	• Basic principles of design: balance, proportion, rhythm, emphasis, unity etc. Laws of perception -Gestalt theory: similarity, proximity, continuity, closure etc. Scale and proportion in design • Graphic Visualization: visual composition and layout, Grids in graphics composition	
3.	TYPOGRAPHY CONCEPTS	07 Lectures
	Typography - Typeface, Typeface Family, Font, Anatomy Of Type, Typographic Measurement – Point And Pica, Text Type And Display Type, Classification Of Type - Old Style, Transitional Period, Modern, Slab Serif, Sans Serif, Script, Decorative Etc. Selection Of A Type Face In Design-Clarity: Readability And Legibility, Integration With Visuals, Concept And Theme Etc	

4.	RASTER GRAPHIC CONCEPTS (DESIGNING & EDITING)	15 Lectures
	• Raster Images – Image Resolution – RGB, CMYK, Lab & Other Colour Modes / Channels And Their Applications –Colour Palate And Swatches. Basic Drawing– Using Airbrush, Pencil, Paint Brush Tools. Concept Of Layers – Transparency And Blending Modes – Creative Use Of Layers And Blending Modes, Layer Mask. Selection Tools-Path Options And Selection-Alpha Channel, Type Tool And Its Properties. Concept Of GIF Animation, Image Compression: Lossy And Lossless Compression Formats • Photo Restoration Technique- Clone Tool, Patch Tool, Sponge Tool, Burn Tool, Dodge Tool Etc. Adjusting Hue Saturation And Value, Use Of Levels And Curves, Use Of Colour Histogram, Treatment Of RAW Files, HDR Toning.	
5.	VECTOR GRAPHIC CONCEPTS (DESIGNING & EDITING)	13 Lectures
	• Vector Graphic-What Is Vector, Properties of Vector Graphics - Stroke and Fill Tools – Basic Shapes, Bezier Drawing with the Pen Tool – Creative Use of Shapes – Using the Pathfinder – Boolean Operations Using Shapes. Vector Drawing Techniques – Node Editing – Tracing from Raster Images – Different Styles Of Vector Illustrations. Using Colour in Vector Graphics – Different Colour Palettes – Gradients And Gradient Mesh. • Using Type Tools And Type Controls – Type Along A Path –Concept Of Alignment And Text Flow Options, Filters And Effects.	

Recommended References:

1. Mark A. Thomas, Poppy Evans, Exploring the Elements of Design, Cengage Learning
2. Alexander W. White, The Elements of Graphic Design: Space, Unity, Page Architecture, and Type, Allworth Press
3. Ambrose, Harris, The Fundamentals Of Typography, Bloomsbury Publishing India Private Limited; 2nd Revised edition edition
4. Adobe Creative Team, Adobe Photoshop Classroom in a Book, Adobe Press
5. Adobe Creative Team, Adobe Illustrator Classroom In A Book, Adobe Press
6. Gary David Bouton, CorelDRAW X - Official Guide, TMH
7. Derek Lea, Beyond Photoshop, Focal Press
8. Dan Moughamian, Adobe Digital Imaging How-Tos, Adobe Press
9. Daniel James, Crafting Digital Media, Apress

Course Title: E-Commerce
Course Code: STG203
Marks: 50

Type: General Theory
Credits: 02
Hours: 30

Prerequisite Courses: None

Course Objectives:

This course aims to study the various background processes involved in E-Commerce website. As a part of the course the student will study the various activities associated with ecommerce, and will also be able to understand the various technologies and security mechanisms used in ecommerce websites

Learning Outcomes:

On Completion of this course the student will

- Explain the working of an E-Commerce website
- Be aware of the various E-Commerce Strategies.
- Evaluate and Apply appropriate Payment mechanisms to the e-commerce website

1.	INTRODUCTION TO E-COMMERCE	10 Lectures
	• E-Commerce: In the Beginning: what Is E-Commerce? – Advantages and Limitations of E-Commerce – The Role of Strategy in E-Commerce – Value chains in E-Commerce – Integrating E-Commerce – Managerial Implications - The Internet and the WWW : The Internet today – Unique benefits of the internet - Searching online – BBS and Pay services – The Language of the Internet .	
2.	E-COMMERCE SITE DESIGNING & LAUNCHING	08 Lectures
	• Launching a Business on the Internet : The life cycle approach – The business planning and strategizing Phase – Hardware, software, Security and the setup phase – Design phase – The marketing phase – The fulfillment phase – The maintenance and enhancement phase. • Designing Web sites: The life cycle of site building – Building a web site – Web navigation design - Design criteria – Hiring a web designer - Website evolution and usability testing : Anatomy of a site – Cookies – What makes a website useful – Website content and Traffic management.	
3.	E-MARKETING & E-PAYMENT	12 Lectures
	• Marketing on the internet : the pros and cons of online shopping – internet marketing	

	techniques – The E-cycle of internet marketing – Marketing your Presence – Attracting Customers – Tracking Customer – Customer service - Web based business to business Ecommerce : B2B Ecommerce – B2B Models – B2B tools –EDI. • Payment Systems: From barter to money – Requirements for the internet based payment – Electronic payment media – Issues and implications – E-security: Security in cyberspace – Designing for security – Kinds of threats (risk) – Virus – Security protection and recovery - Securing your system.
--	---

Recommended Text Books:

1. Elias M. Awad, Electronic Commerce, Prentice-Hall of India, 2008
2. Ravi Kalakota and Andrew B. Whinston, Electronic Commerce A Manager's Guide, Pearson Education.

Recommended References:

1. Kamlesh K Bajaj and Debjani Nag, E-Commerce The Cutting Edge of Business, Second Edition, Tata McGraw Hill
2. Gary P. Schneider, E-Commerce Strategy, Technology and Implementation, CENGAGE Learning INDIA Private Limited, Reprint 2008

Course Title: Environmental Studies - II
Course Code: STG204
Marks: 50

Type: General Theory
Credits: 02
Hours: 30

Prerequisite Courses: None

Course objectives:

- To provide students with basic knowledge of environment and its aspects.
- To sensitize students about environmental issues.

Learning Outcomes:

On successful completion of the course, the students will be able to

- Explain the various aspects of Environment.
- Develop perspectives around environmental issues.
- Take practical steps to conserve the environment and ecosystem

Syllabus:

1.	Global Environmental Issues, Human Population and Environment	12 Lectures
	Global Environmental Issues: Climate Change, Global Warming and Green House Effect, Acid Rain, Depletion of Ozone layer; ecological and carbon footprints; Role of the individual in the prevention of environmental degradation and pollution Population Growth, World and Indian scenario, Population explosion, Population and Environmental Degradation Urbanization: Urban population growth and Environmental problems, Food: Sources of food, Global and Indian food demand scenario, Limits of food production, Environmental effects of Agriculture. Environment and Human Health, Climate and Health, Infectious Diseases, Water-Related Diseases, Climate protection protocols	
2.	Environmental Management	12 Lectures
	Environmental Assessment – Environmental Impact Assessment (EIA), Environmental Auditing, Environmental accounting Environmental management: concept and significance; Emerging environment management strategies, Indian initiatives. Carbon Bank, carbon credits and carbon offsets Environmental Protection Movements and reputed NGOs in India. Sustainable development: meaning and significance; sustainable development goals (SDGs). Environmental Ethics: Environmental Ethics and Environmental values, Code of Ethics, Importance and limitations of ethics; Environmental Ethics in India.	

3.	Environmental Legislation and Goa's Initiatives for Environmental Protection	06 Lectures
	Environmental Acts and Regulations: Brief description and major provisions of • Water (Prevention and Control of Pollution) Act 1974, • Air (Prevention and Control of Pollution) Act 1981, • Environment Protection Act, 1986 • Coastal Regulation Zones Rules • E Waste (Management) Amendment Rules, 2018 Functions and Role of Goa State Pollution Control Board	

Recommended Text Books:

1. Mahua Basu, S. Xavier, Fundamental of Environmental Studies, Edition 2016 Cambridge University
2. N. Balsubramanya, Gurudeep, Chatwal , Environmental Studies , Himalaya Publication.
3. Singh, Anoop Kumar, Environmental Management in mining areas, Ishwar Books

Recommended References:

1. Kumar Rajesh Prakash Kutir, Mining and Environmental Sustainability
2. S.C. Santra, Environmental Science, New Book Agency Pvt Ltd. Kolkata.
3. Shinde, Pendse, Donge, Environmental Education, Sheth Publication.
4. Dr. Vijay Kumar, Environmental Studies Text Book, Himalaya Publication.
5. Kumari Veena, Environmental Pollution and Health Hazard, Ishwar Book
6. Singh Chandrama Prakash and Kuter, Sustainable Development and Environment
7. https://www.tutorialspoint.com/environmental_studies/environmental_studies_classification_of_ecosystem.htm

Course Title: Software Laboratory – II
Course Code: STP201
Marks: 50

Type: General Practical
Credits: 02
Hours: 60

Course objectives:

- To understand and implement the concepts of Object Oriented Programming and Java learnt in STG201
- To implement the graphic design concepts learnt in STG202 using appropriate tools

Learning Outcome:

After successful completion of this course, the student will be able to

- Implement the Object Oriented Concepts in solving real world problems
- Efficiently apply the concepts of AWT, Swing, multithreading, and Jdbc.
- Implement the understanding of principles of graphic placement
- Utilize Software tools to edit images, create artistic imagery, and create publications.

List of suggested Practical:

A student shall complete practical of minimum 24 Hours duration from each Section

Section A - Based on STG201

Use of any SDK and IDE like eclipse, Netbeans etc...

1. Simple java programs to understand different constructs [06 Lectures]
2. Java programs to understand Object Oriented Concepts (Objects, Inheritances, Polymorphism, Abstract Class, Interfaces, Overloading, Overriding, Object copying/cloning, static and final) [12 Lectures]
3. Java programs to understand Packages, exception handling, and file handling [06 Lectures]
4. Programs to understand Applet, threading, Swing and jdbc. [06 Lectures]
5. Mini-Project (As an Assignment)

Section B - Based on STG202

Software Recommended: Adobe Suite (Illustrator, Photoshop, CorelDraw, InDesign), GIMP, INKSCAPE

1. Design Basics and Tessellation of geometric shapes [06 Lectures]

- | | |
|--|---------------|
| 2. Raster Graphics (Design, Image Restoration & Correction) | [10 Lectures] |
| 3. Vector Graphics (Design, Logo or corporate identity design) | [08 Lectures] |
| 4. Designs based on typography | [03 Lectures] |
| 5. Symbols or Icons Designs | [03 Lectures] |

Recommended Books:

For Section A: *As mentioned in STG201*

For Section B: *As mentioned in STG202 and*

1. Ambrose Harris, The Production Manual, AVA Publishing
2. Timothy Samara, Design Elements, a Graphic Style Manual, Rockport Publishers

Course Title: SSC/Q0503 Web Developer
Course Code: STS201
Marks: 700

Type: Skill (Theory, Practical, & OJT)
Credits: 18 (Theory-7, Practical & OJT-11)
Hours: 440

Prerequisite Courses: None

Course/Package objectives:

To enable students to:

- Apply their understanding of concepts in Analysis and Design of Web based Applications
- Apply their understanding of concepts in developing media content and graphic designs for software products and Applications
- Understand importance of policies, procedures and guidelines when contributing to the design of Web based applications as well as developing media content and graphic designs for software products and Applications
- Understand and Provide data/information in standard formats
- Gain new technical and professional development skills, and give the student an opportunity to establish an interest in industrial/commercial activities.
- Develop constructive work habits
- Utilize and strengthen their communication skills and build healthy work relationships
- Determine what professional and personal competencies they need to develop and address in their transition to professional life

Learning Outcomes:

Students will have competencies and skills as mentioned in the Model Curriculum of Web Developer designed by NASSCOM (REFERENCE ID: SSC/Q0503, version 1.0)

Package Syllabus (Adopted Model Curriculum of Nasscom)

- *Lecture Hours and Marks Distribution as per Nasscom standard mentioned in Model Curriculum*
- *Unit Contents as mentioned in the courseware designed by Nasscom for Web Developer*

1.	Web Design Basics (SSC/N0501)
	Unit 1.1 – Introduction to the Industry and the Job Role Unit 1.2 – Introduction to the basic of Web Development Unit 1.3 – SDLC Unit 1.4 – Principles of Web Design Unit 1.5 – Phases of Web Development
2.	Planning and Design (SSC/N0501)

	Unit 2.1 – Planning for Web Development Unit 2.1 – The Web Designing Process Unit 2.1 – Introduction to the Web Application Development
3.	Web Development Process (SSC/N0503)
	Unit 3.1 – The Web Development Process Unit 3.2 – Developing Reusable Content Unit 3.3 – Designing Accessible Web Content Unit 3.4 – Evaluating the Websites
4.	Web Information Security (SSC/N0503)
	Unit 4.1 – Web Information and Security
5.	UAT and Deployment (SSC/N0503)
	Unit 5.1 – The Web Deployment Process
6.	Introduction to Java & JavaScript (SSC/N0501,SSC/N0503)
	Unit 6.1 – Introduction to Java & JavaScript
7.	Portfolios (SSC/N0501, SSC/N0503)
	Unit 7.1 – Portfolio Design
8.	Manage your Work to Meet Requirements (SSC/N9001)
	Unit 8.1 – Understanding Scope of Work and Working within limits of Authority Unit 8.2 – Work and Work Environment Unit 8.3 – Maintaining Confidentiality
9.	Working Effectively with Colleagues (SSC/N9002)
	Unit 9.1 – Effective Communication Unit 9.2 – Working Effectively
10.	Maintain a Healthy, Safe and Secure Working Environment (SSC/N9003)
	Unit 10.1 – Need for Health and Safety at Work Unit 10.2 – Analyst Security Operations Centre's Role Unit 10.3 – Emergency Situations Unit 10.4 – Skills for Maintaining Health and Safety at Work
11.	Provide Data/Information in Standards Formats (SSC/N9004)
	Unit 11.1 – Information and Knowledge Management Unit 11.2 – How to Manage Data/Information Effectively Unit 11.3 – Skills Required to Manage Data and Information Effectively
12.	Develop Knowledge, Skills & Competence (SSC/N9005)
	Unit 12.1 – Importance of Self Development Unit 12.2 – Knowledge and Skills Required for the Job Unit 12.3 – Avenues for Self-Development Unit 12.4 – Planning for Self-Development

On-Job-Training (OJT) / Project Component (Compulsory)

It is required by a student to carry-out and complete

- **OJT** in the relevant industry or working environment and in the relevant skill area (Web Developer).

Duration of OJT: Minimum 45 Hours

A Student needs to submit the detailed report of the OJT along with the 'Certificate of Completion'.

OR

- **Project /Mini-Project** on Web Application Development

A Student needs to submit the project report along with the finished product and other artifacts (if any).

Recommended Text Books:

1. Nasscom courseware module for Web Developer

Qualification Pack Hyperlink:

https://nsdcindia.org/sites/default/files/QP_SSC-Q0503_Web-Developer.pdf

Annexure III

B. Voc. in Software Technologies (Syllabus Semester – III)

Course Title: Data Structures

Type: General Theory

Course Code: STG301

Credits: 03

Marks: 75

Hours: 45

Prerequisites Courses: Programming Concepts

Course Objective

In this course, the student will learn and understand the concept of various Data structures (Array, Stack, Queue, Linked List, and Trees), sorting techniques and use them effectively in solving the problems.

Learning Outcomes

On successful completion of the course the students will

- Explain the concepts of Array, Stack, Queue, Linked list, Trees, Graphs and its applications in solving real life problems
- Be able to implement various data structures and sorting techniques.

Syllabus:

1.	Arrays, Searching & Sorting	12 Lectures
	Arrays - Single and Multi Dimensional Arrays, String functions (strlen, strcpy, strcat, strcmp, strcmpi etc) using arrays. Linear Search & binary Search, Sorting Techniques- Bubble, Selection, Insertion, Merge.	
2.	Data Structure and Linked List	08 Lectures
	Data Structure - Meaning, Data type, ADT, Need of Data Structure, Types Linked List – Concept, Representation, Operations, and Applications.	
3.	Stack, and Queue	10 Lectures
	Concept, Representation, Operations, and Applications.	
4.	Trees	10 Lectures
	Concept & Terminologies, Binary Tree, Binary Tree Traversals, Binary Search Tree – construction, Operations on BST – create, Insert, delete, traversals, counting leaf, non-leaf & total nodes, non recursive in order traversal, Expression Tree. Introduction to AVL, B and B+ Trees	

5.	Graphs	05 Lectures
	Concept & terminologies, Graph Representation – Adjacency matrix, adjacency list, Traversals – BFS & DFS, Application of BFS, DFS – Shortest path, Backtracking.	

Recommended Text Books:

1. Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman, Data Structures & Algorithms, Pearson Education India;
2. R. Venkatesan, S. Lovelyn Rose, Data Structures, Wiley; Second edition

Recommended References:

1. Michael T. Goodrich, Roberto Tamassia , Data Structures and algorithms in Java, John Wiley & sons inc.,5th Edition, International Student version.
2. Langsam Yedidiah, Augenstein J. Moshe, Tenenbaum M. Aaron , Data Structures using C and C++, Pearson Education, Second Edition
3. Gilbeg Richard, Forouzan Behrouz, Data Structures: A Pseudocode Approach with C++, Cengage Learning, Second Edition

Course Title: Audio and Visual Media
Course Code: STG302
Marks: 75

Type: General Theory
Credits: 03
Hours: 45

Prerequisites Courses: None

Course objectives:

- To understand the broad range of basics of audio and visual media concepts.
- To develop the skill to produce and distribute digital, audio, video, animations, and presentations.
- To develop Creativity and publish a self-contained multimedia Application using multimedia authoring tools in various application areas

Learning Outcomes:

On successful completion of the course, the students will be able to

- Identify various properties of audio, video and animation data
- Explain compression techniques of various media types
- Familiarize with basics and latest advances in multimedia
- Develop competencies in Audio-Visual Capture and Editing using Software tools
- Create and Manipulate Animation and multimedia presentation.

Syllabus:

1.	AUDIO MEDIA CONCEPTS	12 Lectures
	Digitization of Sound: Fundamental Characteristics of Sound; Analog V/s Digital sound; Basics of digital sound- Sampling, Frequency, Sound Depth Sample rate and Bit-Depth; Elements of Audio System; Nyquist Theorem, SNR, SQNR	
	Working with Audio: Linear & Non Linear Quantization, Audio Filtering , Audio Quality VS Data Rate, Synthetic sound, MIDI to WAV conversion process , Overview of audio File Formats (WAV, VOC, AVI, MP3, MP4, Ogg, Verbose etc.)	
	Basic concepts of Quantization & Transmission of Audio: Coding of Audio, Pulse Code Modulation, Differential Coding of Audio, Lossless Predictive Coding , DPCM,DM,ADPCM	
	Concepts of Audio Editing, Mixing, Effects and Transitions	
2.	VIDEO MEDIA CONCEPTS	12 Lectures
	Analog and Digital Video, graphics accelerator cards, DirectX, AV/DV and IEEE1394 cards, Video Broadcast Standards – Frame rate (NTSC, PAL, SECAM), frame size - HDTV.	
	Basics of video capturing and recording formats like S-VHA Video, Component (YUV),	

	Component & Composite Digital Media & Instrument. Concepts of Frame rates, resolution, image size and color depth; Intraframe, Interframe, and Hybrid compression; Video file formats, codecs and compression options. Process of digitizing, Transferring digital footage, Editing, mixing, scrubbing, and applying transitions and effects Concepts of shooting platforms, Lighting, Chroma Key or Blue Screen, Titling and the Titler, overlays, Adding narration and soundtracks. Distribution options: Internet, CD/DVD, DCP and KDM	
3.	2D DIGITAL ANIMATION	10 Lectures
	Basics of animation, Terms used in Animation, Principle and use of animation in multimedia; Effect of resolutions, pixel depth, Images size on quality and storage. Drawing, Working with Colour, Using Imported Artwork, Adding Sound, Working with Objects, Using Layers, Using Type, Using Symbols and Instances, Creating Animation, Thumbnails, sequential movement drawing, drawing for motion, Creating interactive movies, Creating Printable movies, Publishing and Exporting, features & limitations, creating simple animations for the Web. Animation file formats. Development process of Animation Character - Anatomy & Body Language, 2 D virtual drawing for animation	
4.	ACTION SCRIPT	06 Lectures
	Introduction, Reference Panel, Syntax, Adding an Action to Your Script, Key frame, Object, and Button; Controlling The Timeline & Movie Content, Loops, Modifying objects, Using ActionScript with Text	
5.	3D MODELING BASICS	05 Lectures
	Modeling Basics: Principles of 3D modeling, concepts like polygons, nurbs, and sub surface modeling etc. Primitive and compound Objects, Meshes, Polys & Splines	

Recommended Text Books:

1. Ranjan Parekh, Principles of Multimedia, The McGraw - Hills Company, 12th Reprint 2011.
2. Vaughan Tay, Multimedia: Making it Work, 8th edition, Tata McGraw-Hill
3. Jeffcoate Judith, Multimedia in Practice, Technology and Applications, PHI.
4. Adobe Creative Team, Adobe Flash Professional CS6 Classroom in a Book, Adobe Press, 1st Edition

Recommended References:

1. Peter Wells, A Beginners guide to Digital Video, AVA Publishing –Switzerland, 2004.
2. Richard Williams, The Animator's Survival Kit, Faber; Main - Revised edition

3. Steve Roberts, Character Animation Fundamentals: Developing Skills for 2D and 3D Character Animation, Routledge, 1st Edition
4. Roland Hess, Blender Foundations: The Essential Guide to Learning Blender, Focal Press
5. Using ADOBE® PREMIERE® PRO CS5 & CS5.
6. James E Shuman, Multimedia in Action, Vikas Publishing House.
7. Elson-Cook, Principles of Interactive Multimedia, McGraw Hill Higher Education.
8. Daniel Carter and Michael Courtney, Anatomy for the Artist, Parragon Inc
9. Kelly L. Murdoc, 3Ds Max 6 Bible, Wiley
10. John M. Blain, The Complete Guide to Blender Graphics: A Beginner's Guide, A K Peters/CRC Press, 1st Edition

Course Title: Reasoning Techniques

Course Code: STG303

Marks: 100

Type: General Theory

Credits: 04

Hours: 60

Prerequisites Courses: None

Course Objective

To build logical and reasoning aptitude that is essential requirement in understanding various concepts and to solve problems effectively.

Learning Outcomes

On successful completion of the course the students will

- Create, solve and interpret basic data and logical models.
- Make sound arguments based on reasoning and/or careful analysis of data.
- Exhibit critical thinking by developing and expressing sound arguments from given premises to related conclusions
- Effectively communicate the substance and meaning of logical problems and their solutions.

Syllabus:

1.	Module I	12 Lectures
	Logic, Statements, Arguments, and Assumptions, Statements and Course of Action, Logical Venn Diagrams, Statements and Conclusions, Syllogism	
2.	Module II	12 Lectures
	Seating Arrangement, Ranking & Time Sequence Test, Blood Relations, Direction Sense Test, Conditions & Grouping, Simple & Coded Inequality, Decision Making, Clocks and Calendar, Situation Reaction Test	
3.	Module III	15 Lectures
	Decision-making, Judgment, Problem-solving, Analogies, Analysis, Differences, Discrimination Arithmetic series, Similarities, Verbal & figure classification, Space visualization, Observation Simple Problems on Data interpretation and Data sufficiency	
4.	Module IV	15 Lectures
	Mathematical Logic: Introduction, Statements, Logical Connectives and Compound Statements: Negation, Conjunction, Disjunction, Implication, Converse and Inverse, logical Equivalence, Tautologies: Contradiction, Contingency, Algebra of Propositions, Argument, Predicate and Quantifiers.	

	Mathematical induction, deduction, proof by contradiction, program correctness.	
5.	Module V	06 Lectures
	Boolean algebra - Boolean functions, truth table, DeMorgan's theorem, logic gates, Realization of Boolean Function using logic gates, Simplification using Karnaugh map (Simple Problems).	

Recommended Text Books:

1. Arun Sharma, How to Prepare for Logical Reasoning for the CAT, 8th edition, McGraw Hill Education (India) Private Ltd.
2. A.K. Gupta, Logical and Analytical Reasoning, Ramesh Publishing House; 34th edition
3. Peeyush Bhardwaj, Analytical & Logical Reasoning for CAT & Other Management Entrance Tests, Arihant Publications; 4th edition
4. Rosen H. Kenneth, Discrete Mathematics and its Applications, Tata McGraw Hill, 7th edition

Recommended References:

1. Ananta Ashisha, Data Interpretation & Data Sufficiency, Arihant Publications; Third edition
2. MK Pandey, Magical Book Series - Analytical Reasoning, BSC Publishing Co. Pvt. Ltd., 2017
3. Daniel Kahneman, Thinking Fast and Slow, Farrar, Straus and Giroux; Reprint edition
4. Dr. R.S. Aggarwal, A Modern Approach to Verbal & Non-Verbal Reasoning, 2018, S. Chand Publication

Course Title: Software Laboratory – III
Course Code: STP301
Marks: 50

Type: General Practical
Credits: 02
Hours: 60

Course objectives:

- To understand and implement the concepts of Data Structures learnt in STG301
- To implement the audio-visual design concepts learnt in STG302

Learning Outcomes:

After successful completion of this course, the student will be able to

- Implement various data structures and sorting techniques.
- Implement and apply Array, Stack, Queue, Linked list, Trees, Graphs in solving real life problems
- Develop competencies in Audio-Visual Capture and Editing using Software tools
- Create and Manipulate Animation and multimedia presentation.

List of suggested Practical:

A student shall complete practical of minimum 24 Hours duration from each Section

Section A - Based on STG301

- | | |
|---|---------------|
| 1. Array and String Operations | [04 Lectures] |
| 2. Searching & Sorting Techniques | [06 Lectures] |
| 3. Stacks, Queues, and Linked List (Singly & Doubly) – creation, addition, deletion, and traversals | [12 Lectures] |
| 4. Binary Search Tree: create, add, delete, traversals, and display nodes. | [06 Lectures] |
| 5. Representation of Graphs & Traversal. | [02 Lectures] |

The lab sessions should be using appropriate IDE. The problems for lab session should be supplemented with assignments to be completed and submitted by students based on above mentioned topics.

A real time problem for software development should be evaluated as Assignments so as to enable students to acquire programming competencies.

Section B - Based on STG202

Practical can be done using Proprietary or Open Source Software as per industry standards. For example Adobe premiere pro, After Effects, Audition, Audacity, Sound forge, Movie maker, Openshot, Blender, 3DMax, flash, iMovie, 3D Max, Blender, etc ... tools may be used

1. Audio Editing [10 Lectures]
Recording Sound using Sound Recorder (Capture), Sound capture through sound editing software, Sound editing, Noise correction, Effect enhancement; Voice Recognition; Importing audio and saving audio from Audio CD. Sound Quality types: CD Quality, Radio Quality, Telephone Quality.
2. Video Editing [10 Lectures]
 - a. Record video from video capture devices, webcams, screen capture or even streaming video and Video Editing. Prepare video with rough cut
 - b. Prepare Video content with title and special effects. Record Video content and learn about video compressions.
 - c. Prepare Video content for streaming.
3. 2D Animation using Action script [06 Lectures]
4. 3D Modeling [04 Lectures]

An Assignment combining all the media elements /concepts learnt using various editors is recommended like Capturing and streaming audios and videos, Designing and creating 2D animation clips and games, 3D modeling.

Recommended Books: *As mentioned in STG301 and STG302*

Course Title: SSC/Q0504 Media Developer
Course Code: STS301
Marks: 700

Type: Skill (Theory, Practical, & OJT)
Credits: 18 (Theory-7, Practical & OJT-11)
Hours: 440

Prerequisite Courses: None

Course/Package objectives:

To enable Students to

- Understand concepts of designing and improving the look and feel, functionality and graphic appeal of the developed application
- Understand concepts of developing media content and graphic designs for software products and Applications
- Understand and Provide data/information in standard formats
- Gain new technical and professional development skills required in the field
- Develop constructive work habits
- Utilize and strengthen their communication skills and build healthy work relationships
- Develop their professional knowledge, skills and competence
- Maintain a healthy, safe and secure working environment

Learning Outcomes:

Students will have competencies and skills as mentioned in the Model Curriculum of MEDIA DEVELOPER designed by NASSCOM (REFERENCE ID: SSC/Q0504)

Package Syllabus (Adopted Model Curriculum of Nasscom)

→ *Lecture Hours and Marks Distribution as per Nasscom standard mentioned in Model Curriculum*

1.	Contribute to the design of software products and applications (SSC/N0501)
2.	Develop media content and graphic designs for software products and applications (SSC/N0503)
3.	Manage your work to meet requirements (SSC/N9001)
4.	Work effectively with colleagues (SSC/N9002)

5.	Maintain a healthy, safe and secure working environment (SSC/N9003)
6.	Provide data/information in standard formats (SSC/N9004)
7.	Develop your knowledge, skills and competence (SSC/N9005)

On-Job-Training (OJT) / Project Component (Compulsory)

It is required by a student to carry-out and complete

- **OJT** in the relevant industry or working environment and in the relevant skill area (Media Developer).

Duration of OJT: Minimum 45 Hours

A Student needs to submit the detailed report of the OJT along with the 'Certificate of Completion'.

OR

- **Project /Mini-Project** on Media Development

A Student needs to submit the project report along with the finished product and other artifacts (if any).

Recommended Text Books:

Qualification Pack Hyperlink:

https://nsdcindia.org/sites/default/files/QP_SSC-Q0504_Media-Developer.pdf

Syllabus

B. Voc. in Software Technologies (Semester – IV)

Course Title: Python Programming

Type: General Theory

Course Code: STG401

Credits: 03

Marks: 75

Hours: 45

Prerequisites Courses: None

Course Objective

- To read and write simple Python programs.
- To develop Python programs with conditionals and loops.
- To define Python functions and call them
- To use Python data structures -- lists, tuples, dictionaries.
- To do input/ output with files in Python.
- To understand application areas of Python

Learning Outcomes

Upon completion of the course, students will be able to

- Read, write, and execute simple Python programs.
- Structure simple Python programs for solving problems.
- Decompose a Python program into functions.
- Represent compound data using Python lists, tuples, and dictionaries.
- Read and write data from/to files in Python Programs.

Syllabus

1.	INTRODUCTION TO PYTHON	11 Lectures
	Python interpreter and interactive mode; values and types: int, float, boolean, string, and list; variables, expressions, statements, tuple assignment, precedence of operators, comments; modules and functions, function definition and use, flow of execution, parameters and arguments; Illustrative programs: exchange the values of two variables, circulate the values of n variables, distance between two points.	
2.	PROGRAM FLOW CONTROL	10 Lectures
	Conditionals: Boolean values and operators, conditional (if), alternative (if-else), chained conditional (if-elif-else); Iteration: state, while, for, break, continue, pass; Fruitful functions: return values, parameters, local and global scope, function composition, recursion; Strings: string slices, immutability, string functions and methods, string module; Lists as arrays.	

3.	LIST, TUPLE AND DICTIONARY	08 Lectures
	Lists: list operations, list slices, list methods, list loop, mutability, aliasing, cloning lists, list parameters; Tuples: tuple assignment, tuple as return value; Dictionaries: operations and methods; advanced list processing - list comprehension; Illustrative programs: selection sort, insertion sort, merge sort, histogram.	
4.	FILES, MODULES, PACKAGES	08 Lectures
	Files and exception: text files, reading and writing files, format operator; command line arguments, errors and exceptions, handling exceptions, modules, packages; Illustrative programs: word count, copy file.	
5.	APPLICATION AREAS	08 Lectures
	Google Translate, Sentiment Analysis: Analyse Facebook data, Image processing, Page rank: How Google works	

Recommended Text Books:

1. Allen B. Downey, Think Python: How to Think Like a Computer Scientist, Updated for Python 3, Shroff/O'Reilly Publishers, 2016 (<http://greenteapress.com/wp/thinkpython/>)
2. Guido van Rossum and Fred L. Drake Jr, An Introduction to Python – Revised and updated for Python 3.2, Network Theory Ltd., 2011.

Recommended References:

1. John V Guttag, Introduction to Computation and Programming Using Python“, Revised and expanded Edition, MIT Press , 2013
2. Robert Sedgewick, Kevin Wayne, Robert Dondero, Introduction to Programming in Python: An Inter-disciplinary Approach, Pearson India Education Services Pvt. Ltd., 2016.
3. Timothy A. Budd, Exploring Python, Mc-Graw Hill Education (India) Private Ltd., 2015.
4. Kenneth A. Lambert, Fundamentals of Python: First Programs, CENGAGE Learning, 2012.
5. Charles Dierbach, Introduction to Computer Science using Python: A Computational Problem Solving Focus, Wiley India Edition, 2013.
6. Paul Gries, Jennifer Campbell and Jason Montojo, Practical Programming: An Introduction to Computer Science using Python 3, Second edition, Pragmatic Programmers, LLC, 2013.

Course Title: Software Engineering and Testing
Course Code: STG402
Marks: 75

Type: General Theory
Credits: 03
Hours: 45

Prerequisites Courses: None

Course Objective

- To make students familiar with the software development principles, models and tools
- To make students understand theories, methods, and technologies applied for software development.
- To understand the Project Management Concepts
- To study basic concepts and strategies of software testing, process, strategies and matrix.
- To understand Software Quality Assurance Concepts and Standards

Learning Outcomes

On completion of the course students will be able to

- Acquaint with historical and modern software methodologies
- Apply the software life cycle models & appreciate the development process
- Develop project plan and control it during the software development cycle.
- Comfortably apply the testing strategies and methods.
- Explain Software Quality Assurance Concepts and Standards

Syllabus

1.	INTRODUCTION	06 Lectures
	Characteristics, Software Engineering goals, Software Process Models: Waterfall, Spiral, Prototyping, Agile, Fourth Generation Techniques and Software Development Life Cycle. Requirements Elicitation Techniques, Feasibility Analyses, Software Requirements Document (SRS)	
2.	Software Project Management	05 Lectures
	Software project management, Project planning and control, cost estimation, project scheduling using PERT and GANTT charts.	
3.	Software Design	12 Lectures
	Basic issues in software design, modularity, cohesion, coupling and layering, function-oriented software design: DFD and Structure chart, Object-oriented software development – Use case, Class Diagram, Activity, Sequence and State chart diagram	
4.	Software Testing	12 Lectures
	Fundamentals of testing, White-box and black-box testing, Test coverage analysis and test case design techniques, Mutation testing, Static and dynamic analysis, Software Metrics, Types of Software Metrics.	

5.	Software Quality Assurance Concepts and Standards	10 Lectures
	Quality Concepts, Quality Control, Quality Assurance, SQA Activities, Software Reviews, Formal Technical Reviews, Review Guidelines, Software Reliability, Software Safety, Quality Assurance Standards, ISO 9000, ISO 9001:2000, Quality Factors, CMM, TQM, Six Sigma, SPICE, Software Quality Assurance Metrics.	

Recommended Text Books:

1. Roger S. Pressman, Software Engineering – A Practitioner’s Approach, TMH, 7e, 2009.
2. Richard Fairley, Software Engineering Concepts, TMH, 2008.
3. Jalote Pankaj, An Integrated Approach to Software Engineering, Third Edition, Narosa Publishing House
4. Srinivasan Desikan and Gopalaswamy Ramesh, Software Testing- Principles and Practices, Pearson

Recommended References:

1. Ian Sommerville, Software Engineering, Pearson Education Asia, 2007.
2. Hans Van Vliet, Software Engineering: Principles and Practices, Wiley, 2008.
3. Mahesh Matha, Object Oriented Analysis and Design Using UML ,(1e)
4. Craig Larman, Agile and Iterative Development – A Manager’s Guide, Pearson Education
5. Jorgensen, P. (2014) Software Testing, A Craftsman’s Approach (4e), CRC Press: Boca
5. Copeland, L. (2004) A Practitioner’s Guide to Software Test Design, Artech House
6. Kathy Schwalbe, Introduction to Project Management, First Edition, Course Technology;2006
7. Myers, G. (2011) The Art of Software Testing, (3e), John Wiley & Sons
8. Martin Fowler, UML Distilled, Addison-Wesley Professional, 3rd Edition.

Course Title: Creative Thinking
Course Code: STG403
Marks: 100

Type: General Theory
Credits: 04
Hours: 60

Prerequisites Courses: None

Course Objective

This is a course on study of creative/lateral thinking and problem solving techniques those are essential to solve real world problems. Causal, deductive, and inductive arguments are described as well as the use of persuasion.

Learning Outcome

Upon completion of this course, students should be able to

- Identify the benefits of employing creative/lateral-thinking processes.
- Apply creative/lateral-thinking and problem-solving theories to real-world problems.
- Develop strategies to overcome the barriers that inhibit creative thinking.
- Identify strategies for creating an organizational culture that embraces and sustains creative-thinking practices
- Identify strategies to solve complex problems in a collaborative way.
- Apply creative/lateral-thinking principles to develop persuasive arguments that employ legal, moral, and aesthetic reasoning.

Syllabus

1.	Module I	10 Lectures
	• The way the mind works • Difference between lateral and vertical thinking • Attitudes towards lateral thinking • Basic nature of lateral thinking	
2.	Module II	12 Lectures
	• The use of lateral thinking • Techniques • The generation of alternatives • Challenging assumptions • Innovation	
3.	Module III	12 Lectures
	• Suspended judgement • Design • Dominant ideas and crucial factors • Fractionation • The reversal method	

4.	Module IV	12 Lectures
	• Brainstorming • Analogies • Choice of entry point and attention area • Random stimulation	
5.	Module V	14 Lectures
	• Concepts/divisions/polarization • The new word po • Blocked by openness • Description/problem solving/design	

Recommended Text Books:

1. Edward De Bono, Lateral Thinking: Creativity Step by Step, Harper Perennial; Reissue edition (24 February 2015)
2. Ken Watanabe, Problem Solving 101: A simple book for smart people, Vermilion

Recommended References:

1. R G Chaudhari, Training Techniques of Creative Problem Solving: Trainers Manual, Notion Press, Inc.; 1st edition
2. Mahon N, Basics Advertising 03: Ideation, AVA Publishing (October 26, 2011)
3. Brian Tracy, Creativity & Problem Solving: The Brian Tracy Success Library, Manjul Publishing House
4. Michael Sloan, The Art Of Problem Solving 101: Improve Your Critical Thinking And Decision Making Skills And Learn How To Solve Problems Creatively, Make Profits Easy LLC
5. Ruggiero, V. R., The art of thinking: A guide to critical and creative thought (11th ed.), Longman (2015).
6. Proctor, T., Creative Problem Solving for Managers: Developing Skills for Decision Making and Innovation, Routledge, 4th edition

Course Title: Software Laboratory – IV

Course Code: STP401

Marks: 50

Type: General Practical

Credits: 02

Hours: 60

Course objectives:

- To understand and implement the concepts of Python Programming learnt in STG401
- To implement the software engineering and testing concepts learnt in STG402

Learning Outcomes:

After successful completion of this course, the student will be able to

- Write, test, and debug Python programs.
- Develop Python programs step-wise by defining functions and calling them.
- Use Python lists, tuples, dictionaries for representing compound data.
- Read and write data from/to files in Python
- Design SRS, Use Case, and other software modeling diagram
- Develop project plan and execute, and control it
- Design, and execute test suites effectively

List of suggested Practical:

A student shall complete practical of minimum 24 Hours duration from each Section

Section A - Based on STG401

- 1. PYTHON BASICS** [08 Lectures]
 - Installation & IDE,
 - Working with variables & different DataTypes
 - Basic Input and Output
 - Types of operations
 - Math, Strings, and Variables
- 2. PROGRAM FLOW CONTROL** [08 Lectures]
 - Conditional Statements
 - Control Structures
 - Repetition Structures
 - Working with Text
 - Functions
- 3. LIST, TUPLE AND DICTIONARY** [08 Lectures]
 - Lists
 - Tuple
 - Dictionaries
- 4. FILES, MODULES, PACKAGES** [06 Lectures]
 - File Input and Output
 - Exception Handling

The lab sessions should be using appropriate IDE. The problems for lab session should be supplemented with assignments to be completed and submitted by students based on above mentioned topics.

Section B - Based on STG402

Class may be divided into groups of 5 for the practical session. (Case Study Based)

Use of any UML CASE Tool, Project Management Tool, and Collaboration / Versioning Tool

1. Development of SRS, Requirement Gathering & Feasibility Study [04 Lectures]
2. Creating WBS, Gantt and tracking Gantt chart using any IT PM Tool [04 Lectures]
3. Software Analysis and Designing [08 Lectures]
 - a. Requirement Analyses using Use Case Analysis
 - b. UML Diagrams (Class Diagram, Sequence Diagram, Activity & State Chart Diagram)
4. Software Testing [14 Lectures]
 - a. Using Selenium IDE, Write a test suite containing minimum 4 test cases.
 - b. Install Selenium server and demonstrate it using a script in Java/PHP
 - c. Write and test a program to login a specific web page
 - d. Write and test a program to update 10 student records into table into Excel file
 - e. Write and test a program to select the number of students who have scored more than 60 in any one subject (or all subjects)
 - f. Write and test a program to provide total number of objects available on the page
 - g. Write and test a program to get the number of list items in a list /combo box
 - h. Write and test a program to count number of check boxes on the page checked and unchecked count.

Recommended Books: *As mentioned in STG401 and STG402*

Course Title: SSC/Q0509

Master Trainer for Junior Software Developer

Course Code: STS401

Marks: 700

Type: Skill (Theory, Practical, & OJT)

Credits: 18 (Theory-7, Practical & OJT-11)

Hours: 440

Course/Package objectives:

To enable student to

- Understand concepts to accurately describe software products/applications & their features
- Establish clearly the customization requirements and identify accurately the changes required for each customized version of software products/ applications.
- Acquire skills to implement and document changes using standard templates and tools.
- Create appropriate unit test cases for the changes, execute and document results.
- Design and Access reusable components and tools from the organization's knowledge base.
- Update organization's knowledge base with the code developed.
- Employ Programming Lab Oriented Pedagogical Skills as a Master Trainer in the IT Industry
- Engage Pedagogical Skills as a Master Trainer.
- Develop constructive work habits
- Utilize and strengthen their communication skills and build healthy work relationships
- Develop professional and personal knowledge, skills and competence
- Maintain a healthy, safe and secure working environment.

Learning Outcomes:

Students will have competencies and skills as mentioned in the Model Curriculum of MASTER TRAINER FOR JUNIOR SOFTWARE DEVELOPER designed by NASSCOM (REFERENCE ID: SSC/Q0509, version 1.0)

Package Syllabus (Adopted Model Curriculum of Nasscom)

→ *Lecture Hours and Marks Distribution as per Nasscom standard mentioned in Model Curriculum*

1.	Assist in performing software construction and software testing entry-level tasks in the IT Services industry (SSC/N0506)
2.	Employ Programming Lab Oriented Pedagogical Skills as a Master Trainer in the IT Industry (SSC/N0507)
3.	Engage Pedagogical Skills as a Master Trainer (SSC/N0508)
4.	Manage your work to meet requirements (SSC/N9001)
5.	Work effectively with colleagues (SSC/N9002)
6.	Maintain a healthy, safe and secure working environment (SSC/N9003)

7.	Develop your knowledge, skills and competence (SSC/N9005)

On-Job-Training (OJT) / Project Component (Compulsory)

It is required by a student to carry-out and complete

- **OJT** in the relevant industry or working environment and in the relevant skill area (Master Trainer for Junior Software Developer).

Duration of OJT: Minimum 45 Hours

A Student needs to submit the detailed report of the OJT along with the 'Certificate of Completion'.

OR

- **Project /Mini-Project** : Junior Software Developer Training

A Student needs to submit the project report along with the finished product and other artifacts (if any).

Recommended Text Books:

Qualification Pack Hyperlink:

https://nsdcindia.org/sites/default/files/QP_SSC-Q0509_Master-Trainer-for-Software-Developer.pdf

Syllabus

B. Voc. in Software Technologies (Semester – V)

Course Title: Mobile Application Development

Course Code: STG501

Marks: 75

Type: General Theory

Credits: 03

Hours: 45

Prerequisite Courses: Java Programming, Web Designing

Course Objectives:

- To understand the fundamentals of Android operating systems
- To use Android software development tools
- To develop software with reasonable complexity on mobile platform
- To deploy software to mobile devices
- To debug programs running on mobile devices

Learning Outcomes:

Upon successful completion of this class, the learner will be able to

- Demonstrate their understanding of the fundamentals of Android operating systems
- Efficiently develop software with reasonable complexity on mobile platform
- Use the development tools in the Android development environment
- Use the major components of Android API set to develop their own apps
- Demonstrate their ability to deploy software to mobile devices
- Debug programs running on mobile devices
- Package and prepare their apps for distribution on the Google Play Store

Syllabus

1.	INTRODUCTION TO ANDROID OPERATING SYSTEM	09 Lectures
	Android OS design & Features – Android development framework, SDK features, Installing and running applications on Eclipse platform, Creating AVDs, Types of Android applications, Best practices in Android programming, Android tools. Android application components – Android Manifest file, Externalizing resources like values, themes, layouts, Menus etc, Resources for different devices & languages, Runtime Configuration Changes Android Application Lifecycle – Activities, Activity lifecycle, activity states, monitoring state changes	
2.	ANDROID USER INTERFACE	09 Lectures
	Measurements – Device and pixel density independent measuring units Layouts – Linear, Relative, Grid and Table Layouts User Interface (UI) Components – Editable and non editable Text Views, Buttons, Radio and Toggle Buttons, Checkboxes, Spinners, Dialog and pickers Event Handling – Handling clicks or changes of various UI components Fragments – Creating fragments, Lifecycle of fragments, Fragment states, Adding fragments to Activity,	

	adding, removing and replacing fragments with fragment transactions, interfacing between fragments and Activities, Multi-screen Activities	
3.	INTENTS AND BROADCASTS	09 Lectures
	Intent – Using intents to launch Activities, Explicitly starting new Activity, Implicit Intents, Passing data to Intents, Getting results from Activities, Native Actions, using Intent to dial a number or to send SMS Broadcast Receivers – Using Intent filters to service implicit Intents, Resolving Intent filters, finding and using Intents received within an Activity Notifications – Creating and Displaying notifications, Displaying Toasts	
4.	PERSISTENT STORAGE	09 Lectures
	Files – Using application specific folders and files, creating files, reading data from files, listing contents of a directory Shared Preferences – Creating shared preferences, saving and retrieving data using Shared Preference Database – Introduction to SQLite database, creating and opening a database, creating tables, inserting retrieving and deleting data, Registering Content Providers, Using content Providers (insert, delete, retrieve & update)	
5.	OTHER TOPICS	09 Lectures
	Alarms – Creating and using alarms Understanding Mobile Networking Fundamentals - Accessing the Internet (HTTP), Browsing the web with WebView, Calling PHP From Android: Pass Android application data to PHP. Using Internet Resources – Connecting to internet resource, using download manager Location Based Services – Finding Current Location and showing location on the Map, updating location Publishing Android Applications: Preparing for Publishing, Versioning, Digitally Signing Your Android Applications, Deploying APK Files, Using the adb exe Tool, Using a Web Server, Publishing on the Android Market, Creating a Developer Profile, Submitting Your Apps	

Recommended Text Books:

1. Reto Meier, Professional Android 4 Application Development, , Wiley India, (Wrox) , 2012
2. Jonathan Simon, Head First Android Development, O'Reilly media.
3. Rick rogers, John Lombardo, Android Application Development, O'Reilly
4. James C Sheusi, Android Application Development for Java Programmers, Cengage Learning

Recommended References:

1. Wei-Meng Lee, Beginning Android 4 Application Development, Wiley India (Wrox), 2013
2. Building Android Apps In Easy Steps, McGraw-Hill Education
3. Lauren Darcey and Shane Conder, Android Wireless Application Development, Pearson Education, 2nd ed.

Course Title: Human Computer Interaction

Course Code: STG502

Marks: 75

Type: General Theory

Credits: 03

Hours: 45

Prerequisites Courses: None

Course Objective

- To learn the foundations of Human Computer Interaction.
- To become familiar with the design technologies for individuals
- To be aware of mobile HCI.
- To learn the guidelines for user interface.

Learning Outcome

Upon completion of the course, the students should be able to:

- Design effective dialog for HCI
- Design effective HCI for individuals and persons with different-abilities.
- Assess the importance of user feedback.
- Explain the HCI implications for designing multimedia/ ecommerce/ e-learning Web sites.
- Develop meaningful user interface.

Syllabus

1.	FOUNDATIONS OF HCI	09 Lectures
	The Human: I/O channels – Memory – Reasoning and problem solving; The Computer: Devices – Memory – processing and networks; Interaction: Models – frameworks – Ergonomics – styles – elements – interactivity- Paradigms. - Case Studies	
2.	DESIGN & SOFTWARE PROCESS	09 Lectures
	Interactive Design: Basics – process – scenarios – navigation – screen design – Iteration and prototyping. Persona, storyboard, HCI in software process: Software life cycle – usability engineering – Prototyping in practice – design rationale. Design rules: principles, standards, guidelines, rules. Evaluation Techniques – Universal Design	
3.	MODELS AND THEORIES	09 Lectures
	HCI Models: Cognitive models: Socio-Organizational issues and stakeholder requirements – Communication and collaboration models-Hypertext, Multimedia and WWW.	

4.	MOBILE HCI	09 Lectures
	Mobile Ecosystem: Platforms, Application frameworks- Types of Mobile Applications: Widgets, Applications, Games- Mobile Information Architecture, Mobile 2.0, Mobile Design: Elements of Mobile Design, Tools. - Case Studies	
5.	WEB INTERFACE DESIGN	09 Lectures
	Designing Web Interfaces – Drag & Drop, Direct Selection, Contextual Tools, Overlays, Inlays and Virtual Pages, Process Flow - Case Studies	

Recommended Text Books:

1. Alan Dix, Janet Finlay, Gregory Abowd, Russell Beale, Human Computer Interaction, 3rd Edition, Pearson Education, 2004 (UNIT I, II & III)
2. Brian Fling, Mobile Design and Development, 1st Edition, O'Reilly Media Inc., 2009 (UNIT – IV)
3. Bill Scott and Theresa Neil, Designing Web Interfaces, 1st Edition, O'Reilly, 2009. (UNIT-V)

Course Title: Advanced Quantitative Techniques

Course Code: STG503

Marks: 100

Type: General Theory

Credits: 04

Hours: 60

Prerequisites Courses: None

Course Objective

- To build mathematical foundation that is essential requirement in understanding various concepts.
- To understand appropriate statistical techniques for grouping, displaying, analyzing and interpreting statistical data

Learning Outcomes:

On successful completion of the course, the students will be able to

- Understand the basic principles of sets and operations in sets.
- Apply counting principles to determine probabilities.
- Demonstrate an understanding of matrices and determinants
- Apply basic statistical concepts & techniques for quantification of data.
- Independently calculate basic statistical parameters (measures of central tendency and dispersion, correlation and regression coefficients, indexes)
- Interpret the meaning of the calculated statistical indicators based on the acquired knowledge

Syllabus:

1.	Set, Relation, and Functions	12 Lectures
	SETS: Sets, Subsets, Equal Sets, Universal Sets, Finite and Infinite Sets, Operation of Sets, Union, Intersection and Complement of sets, Cartesian product, Cardinality of Sets, Simple Applications. RELATION: Properties of Relation, Equivalence Relation FUNCTIONS: Domain and Range, Onto, Into and One-to-One- Functions, Composite and Inverse functions, Hashing functions and Recursive Functions, growth of functions	
2.	Counting Principles	12 Lectures
	Permutations; Combinations; Counting; Probability Summation; Basics of recurrence relations	
3.	Matrices and Determinants	12 Lectures
	Definition, Minors, Cofactors, Properties of Determinants MATRICES: Definition, Types of matrices, Multiplication of matrices, Adjoint, Inverse, Cramer's Rule, Rank of matrix, Dependence of vectors, Eigen vectors of a matrix	
4.	Statistical Sampling and Central Tendency	12 Lectures
	Collection, classification, tabulation and presentation of data; the concept and methods of	

	sampling, sample types Measures of central tendency - mean, median, mode, quartiles, deciles and percentiles and their applications in data analyses	
5.	Measures of Dispersion and Relation	12 Lectures
	Measures of Dispersion- Range - Quartile Deviation – Mean Deviation - Standard Deviation. Coefficient of Variation Meaning and use of correlation – Types of correlation-Karl Pearson’s correlation coefficient – Spearman’s Rank correlation. Calculation of Correlation; Regression analysis, comparison between correlation and Regression – Regression Equations, Interpretation of Regression Co-efficients	

Note: It is recommended to use excel/spreadsheets for Unit4 and Unit5

Recommended Text Books:

1. Rosen H. Kenneth, Discrete Mathematics and its Applications, Tata McGraw Hill, 7th edition
2. Gupta. S. C. Fundamental of Statistics, Himalaya Publishing House, Mumbai, 6th edition
3. Kolman, Busby, Ross, Discrete Mathematical structures, Pearson

Recommended References:

1. Sarkar Kumar Swapan, A Textbook of Discrete Mathematics, S Chand & Company, 2005.
2. J.K. Sharma, Discrete Mathematics, Macmillan India Ltd., Second Edition – 2005
3. Spiegel. M. R. and Stephens. J. L., Shaum’s Outlines Statistics, Tata McGraw-Hill, India, 2011
4. Sanchetti D.C and Kapoor V.K .Statistics - Theory, Methods and Application, Sultan Chand & Sons , New Delhi, 7th edition, 2010.

Course Title: Software Laboratory – V
Course Code: STP501
Marks: 50

Type: General Practical
Credits: 02
Hours: 60

Course objectives:

- To understand and implement the concepts of Mobile Application Development learnt in STG501

Learning Outcome:

After successful completion of this course, the student will be able to

- Demonstrate their ability to develop software with reasonable complexity on mobile platform
- Proficiently use the development tools in the Android development environment
- Use the major components of Android API set to develop their own apps
- Debug programs running on mobile devices
- Package and prepare their apps for distribution on the Google Play Store

List of suggested Practical:

A student shall complete practical of minimum 50 Hours duration

Based on STG501

1. Understanding & Configuring Tools [04 Lectures]
 - a) Installing Java, and ADT bundle - Eclipse Integrated Development Environment (IDE), Creating Android Virtual Devices (AVDs)
 - b) Creating a New Android Project, Defining the Project Name and SDK Settings, Project Configuration Settings, Configuring the Launcher Icon, Creating an Activity, Running the Application in the AVD, Stopping a Running Application, Modifying the Example Application, Reviewing the Layout and Resource Files.
 - c) Familiarize with the Directory Structure of an Android Project , Common Default Resources Folders, Leveraging Android XML, Screen Sizes, Launching Your Application: The AndroidManifest.xml File
2. Creating First Android Application [02 Lectures]
3. Develop an application that uses [06 Lectures]
 - a) GUI components, Font and Colours
 - b) Layout Managers and event listeners.
4. Develop a native calculator application. [04 Lectures]

5. Write an application that draws basic graphical primitives on the screen. [04 Lectures]
6. Develop an application that makes use of RSS Feed. [04 Lectures]
7. Implement an application that implements Multi threading [06 Lectures]
8. Develop an application that makes use of Notification Manager [04 Lectures]
9. Develop a native application that uses GPS location information. [06 Lectures]
10. Develop an application that makes use of database. [06 Lectures]
11. Implement an application that writes data to the SD card. [04 Lectures]
12. Write a mobile application that creates alarm clock [06 Lectures]
13. Develop a mobile application to send an email. [04 Lectures]

A mini-project combining all the technologies/concepts learnt using an IDE such as Xamarin and Eclipse is recommended as assignment or internal evaluation.

Recommended Books: *As mentioned in STG501*

Course Title: SSC/Q0502 UI Developer
Course Code: STS501
Marks: 800

Type: Skill (Theory, Practical, & OJT)
Credits: 18 (Theory-7, Practical & OJT-11)
Hours: 440

Course/Package objectives:

To enable students to

- Acquire skills to create complex user interfaces for a variety of applications, such as software's, databases and websites.
- Learn the UI/UX design of software products and applications
- Develop media content and graphic designs for software products and applications
- Develop software code to specification
- Provide data/information in standard formats
- Gain new technical and professional development skills
- Develop constructive work habits
- Utilize and strengthen their communication skills and build healthy work relationships
- Determine what professional and personal competencies they need to develop and address in their transition to professional life
- Maintain a healthy, safe and secure working environment
- Develop knowledge, skills and competence

Learning Outcomes:

Students will have competencies and skills as mentioned in the Model Curriculum of UI DEVELOPER designed by NASSCOM (REFERENCE ID: SSC/Q0502, version 1.0)

Package Syllabus (Adopted Model Curriculum of Nasscom)

→ *Lecture Hours and Marks Distribution as per Nasscom standard mentioned in Model Curriculum*

1.	Contribute to the design of software products and applications(SSC/N0501)
2.	Develop software code to specification(SSC/N0502)
3.	Develop media content and graphic designs for software products and applications (SSC/N0503)
4.	Manage your work to meet requirements(SSC/N9001)
5.	Work effectively with colleagues(SSC/N9002)
6.	Maintain a healthy, safe and secure working environment(SSC/N9003)

7.	Provide data/information in standard formats(SSC/N9004)
8	Develop your knowledge, skills and competence(SSC/N9005)

On-Job-Training (OJT) / Project Component (Compulsory)

It is required by a student to carry-out and complete

- **OJT** in the relevant industry or working environment and in the relevant skill area (User Interface Developer).

Duration of OJT: Minimum 45 Hours

A Student needs to submit the detailed report of the OJT along with the 'Certificate of Completion'.

OR

- **Project /Mini-Project** : User Interface Development

A Student needs to submit the project report along with the finished product and other artifacts (if any).

Recommended Text Books:

Qualification Pack Hyperlink:

https://nsdcindia.org/sites/default/files/QP_SSC-Q0502_UI-Developer.pdf

Syllabus

B. Voc. in Software Technologies (Semester – VI)

Course Title: Advanced Graphics Designing

Course Code: STG601E1

Marks: 75

Type: General Theory

Credits: 03

Hours: 45

Prerequisite Courses: Graphics Designing

Course objectives:

- To understand the concept of color theory and color psychology in graphic design
- To learn about layout design in graphic design and publication
- To learn about the process of creating and publishing books and eBooks

Learning Outcomes:

On successful completion of the course, the students will be able to

- Demonstrate the ability to apply effective color scheme in design
- Create artistic and effective design layout
- Prepare and publish documents and book/eBooks for various Platforms
- Explain the concepts involved in Publishing

Syllabus:

1.	COLOR THEORY	09 Lectures
	Color & color theory – Additive & subtractive -Terms to describe color, - color separation technique Direct & indirect method – GATF color triangles & color circle their use – modern color spaces -color matching – color original - color originals, selection and their characteristics – method of color measurement– color Gamut. Color Profiles .ico files Prepress color proofing: DDCP- inkjet-thermal wax – chromalin proofing- factors in proofing- substrate- color of ink solid ink density- trapping tone reproduction proofing methods- soft proof- digital proof photomechanical proof- press proof- other proofing methods	
2.	COLOR PSYCHOLOGY	09 Lectures
	Color Symbolism, Color Psychology, and Historical & Contemporary use of Color. Local color and subjective use of color. Emotional effects. Personal Color preferences. Color psychology in context of Indian sub-continent.	
3.	LAYOUT DESIGN	09 Lectures
	Layout Design: Directing The Eye, Backwards Movement, Application Of Design Principles In Lay Out, Free Style Lay Out, Grid Design, Formats, Margins, Columns And Gutters, Page Depth, Working With Imagery, Borders And Rules. Consistency In Design: Creating Style	

	Guides And Printing Instructions.	
4.	LAYOUT VISUALIZATION	09 Lectures
	Visualization of various layouts- magazine, newspaper, books, screen media etc. Creating a Suitable Grid, Title and Cover Policies. Selecting and Using Type family, White Space, Colour, Headlines, The Masthead etc.	
5.	E- PUBLISHING	09 Lectures
	Paper types and sizes, GSM, etc. Creating a book –Adding documents to the book – Synchronizing Style Source – Page numbering – Creating Table of Contents – indexing – Preflight – Exporting Documents – Exporting to E-book format – Printing technologies Interactive PDF and Other E-Pub Formats, Interaction Between Movies, Sound Clips URL's And Other E-Books, E-Publication for Various Platforms.	

Recommended Reference Books:

1. Adams Morioka, Color Design Workbook: A Real World Guide to Using Color in Graphic Design, Rockport Publishers
2. John T Draw, Color Management: A Comprehensive Guide for Graphic Designers, Allworth; 1 edition
3. David E. Carter, The Big Book of Layouts, Harper Design
4. Beth Tondreau, Layout Essentials -100 Design Principles for Using Grids, Rockport Publishers
5. Chris Frost, Designing for Newspapers and Magazines, Routledge
6. Jost Hochuli and Robin Kinross, Designing Books: Practice and Theory, Hyphen Press
7. Kindle Direct Publishing, Building Your Book for Kindle, Kindle Direct Publishing

Course Title: Relational Database Management System (RDBMS)

Course Code: STG601E2

Marks: 75

Type: General Theory

Credits: 03

Hours: 45

Prerequisite Courses: Knowledge of Data Structures and Programming skills.

Course objectives:

To provide a strong formal foundation in database concepts, technology and to apply it in the field of software development

Learning Outcomes:

After successful completion of this course, the students will

- Explain the need and working of Data Base and Data Base Management Systems.
- Demonstrate competency in designing database models and design.
- Learn the RDBMS Concepts and data manipulation using SQL.

Syllabus:

1.	INTRODUCTION TO DATABASE MANAGEMENT SYSTEM	05 Lectures
	Basic concepts and definitions – Data Dictionary – Database – DBMS – DBA - Disadvantages of File oriented System – Advantages and Disadvantages of DBMS – Schema, Subschema– Three-Level Architecture of DBMS - Functions and Services of DBMS - Database Languages – Data Models (Hierarchical, Network, and Relational Model)	
2	RELATIONAL DATABASE DESIGN	10 Lectures
	Relational Algebra – Relational Calculus – Codd’s Rules - Structured Query Language (SQL) – Entity – Relationship (ER) Model - Relationship sets, Attributes, Constraints, Mapping Cardinalities, Keys, ER diagrams, Weak entity sets, Strong entity sets. Database Development Life Cycle (DDLC) – Functional Dependency. NORMALIZATION: need for normalization, First Normal Form (1NF) – Second Normal Form (2NF) – Third Normal Form (3NF)	
3.	DATA DEFINITION AND MANIPULATION IN SQL	15 Lectures
	Data Definition: Data types, Creation, Insertion, Viewing, Updating, Deletion of tables, modifying the structure of the tables, Renaming, Dropping of tables. Data Constraints – I/O constraints, Primary key, foreign key, unique key constraints, ALTER TABLE command. Database Manipulation: Computations done on table data Select command, Logical operators, Range searching, Pattern matching, Grouping data from tables in SQL, GROUP BY, HAVING clauses, Joins – Joining multiple tables, Joining a table to itself. Views Creation, Renaming the column of a view, destroys view, Granting & revoking privileges.	
4.	PROGRAM WITH SQL	10 Lectures
	Data types: Using set and select commands, procedural flow, if, if /else, while, goto, global variables, Security Locks, types of locks, levels of locks. Cursors Working with cursors, Error Handling, Developing stored procedures, create, alter and drop, passing and	

	returning data to stored procedures, using stored procedures within queries, building user defined functions, creating and calling a scalar function, implementing triggers, creating triggers, multiple trigger interaction.	
5.	TRANSACTION AND SECURITY MANAGEMENT	05 Lectures
	Transaction Concepts – Concurrency Control – Database Backup – Types of Database Failures – Database Recovery Types – Database Security	

Recommended Text Books:

1. Elmasri & Navathe, Fundamentals of Database Systems, Pearson Education.
2. Abraham Silberschatz, Henry F Korth, S. Sudarshan, Database System Concepts, McGraw-Hill Education

Recommended References:

1. S. K. Singh, Database Systems Concepts, Design and Applications, 2nd Edition, Pearson Education, 2006
2. C.J. Date, Introduction to Database Systems, Addison Wesley

Course Title: Computer Networks
Course Code: STG602
Marks: 75

Type: General Theory
Credits: 03
Hours: 45

Prerequisite Courses: Fundamentals of Computers & Programming

Course objectives:

- To provide a basics of networking concepts.
- To understand the foundations covering the physical layer, data link layer, network layer and the transport layer.

Learning Outcomes:

After successful completion of this course, the students will

- Explain the basics of data communication and exchange
- Explain various techniques and rules for device communication

Syllabus:

1.	Introduction to System Security	09 Lectures
	Introduction to Computer networks, Topology, categories of networks, Internetwork, Internet, Network Models, Layered model, OSI and TCP/IP models, Transmission media, Wired and unwired media.	
2	Physical and Data link layer	09 Lectures
	Physical layer- , Analog and Digital data, Periodic and A periodic signals, Composite signals, Digital data transmission, Transmission Modes , Switching Data link layer - Error detection and correction, Types of errors, Single bit error and Burst error, Cyclic Redundancy Check (CRC), Error correction Single bit error correction, Hamming code Data compression, Flow control, Error control, Wired LANs, Ethernet IEEE standards.	
3.	Network layer	09 Lectures
	Networking and Internetworking devices Repeater, Bridges, Routers, Gateways, Logical addressing IPv4 & IPv6 addresses, Network Address Translation (NAT), Internet protocols, internetworking, Datagram, Transition from IPv4 to IPv6, Address Mapping, Error reporting and multicasting Delivery, Forwarding and Routing algorithms	
4.	Transport and Application Layer	09 Lectures
	Transport layer - Process to process Delivery: UDP, TCP and SCTP, Congestion control and Quality of Service,	

	Application Layer - Domain Name Systems Remote Login – Email FTP, WWW, HTTP Network management SNMP	
5.	Network Security	09 Lectures
	Network Security – Basics of Cryptography Digital Signature, Encryption, Decryption, Firewall, Data Translation Compression – Mails Services – Directory services – File Transfer and Access Management Protocol (FTAM) – Common Management Information Protocol (CMIP).	

Recommended Text Books:

1. Behrouz A. Forouzan, Data Communications and Networking, McGraw Hill Education (India) Pvt. Ltd., 4th Edition, 2006.

Recommended References:

1. Andrew S. Tanenbaum, Computer Networks, Prentice Hall of India, 4th Edition, 2002.
2. James F. Kurose & Keith W. Ross, Computer Networking: A Top-Down Approach, Pearson India, 5th Edition, 2012

Course Title: Entrepreneurship Development
Course Code: STG603
Marks: 100

Type: General Theory
Credits: 04
Hours: 60

Prerequisite Courses: None

Course objectives:

- To familiarize the students with the concept of entrepreneurship
- To identify and develop the entrepreneurial talents of students
- To create awareness of the role of entrepreneurs in the economic development
- To help students understand the role of creativity and innovation in entrepreneurial startups
- To learn about business idea generation and preparation of Business Plan
- To identify the various sources of finance for entrepreneurial ventures
- To equip students with the latest developments in the field of entrepreneurship

Learning Outcomes:

On successful completion of the course, the students will be able to

- Explain the concepts involved in entrepreneurship
- Get insights into his/her own creative and innovative entrepreneurial skills and evaluate opportunities for a new venture
- Demonstrate the ability to prepare a business plan for a venture
- Evaluate various financial sources for launching entrepreneurial ventures
- Introduce business formats in tune with emerging trends

Syllabus:

1.	Introduction: Overview of Entrepreneurship	12 Lectures
	Concept of Entrepreneurship and Entrepreneur, Qualities and skills of an Entrepreneur, Intrapreneurs, Types of Entrepreneurs, Factors influencing Entrepreneurship; Role of entrepreneurs in economic development	
2.	Entrepreneurship, Creativity and Innovation	12 Lectures
	Creativity and entrepreneurship, Steps in Creativity, Organizational actions that enhance/hinder creativity, Innovation and inventions, Using left brain skills to harvest right brain ideas. Sources of Innovation in Business; Managing Organizations for Innovation and Positive Creativity	
3.	Business Idea & Business Plan	12 Lectures
	Business Idea – Meaning & Sources, Evaluation of business idea. Business Plan – meaning & uses, Project Report – meaning, importance & contents	

4.	Financing the Entrepreneurial Business	12 Lectures
	Arrangement of funds; Traditional sources of financing, new innovative sources; Schemes of Govt. of Goa; Make in India Program	
5.	Emerging trends in Entrepreneurship	12 Lectures
	New generation entrepreneurs - Social entrepreneurs, Edupreneurs, Health entrepreneurs, Tourism entrepreneurs, Ecopreneurs, Techpreneurs, etc. New trends in entrepreneurship - Startup accelerators, Student Sandbox and Business Lab, Co-working spaces, Boot camps, etc. Type of Class Assignments 1. Presentations on Innovative Entrepreneurs from Goa. 2. Preparation of Business Plan/ Project Report	

Recommended Text Books:

1. Vasant Desai, Dynamics of entrepreneurial development and management, Himalaya Publishing House, 2010
2. Romeo Mascarenhas, Entrepreneurship Management, Vipul Prakashan

Reference Books:

1. David H. Holt, Entrepreneurship-New Venture Creation, Pearson India Education Services Pvt. Ltd., Noida-India, Fourth Impression 2018
2. Jasmer Singh Saini, Entrepreneurship Development Programmes, Deep & Deep Publications (P) Ltd., 2002.
3. Jose Paul, Ajith Kumar, Paul T. Mampilly, Entrepreneurship Development, Himalaya Publishing House, Mumbai, 2001
4. C B Gupta, N P Srinivasan, Entrepreneurial Development, Sultan Chand & Sons, New Delhi, Revised Edition 2010
5. C B Gupta Entrepreneurship and Small Business Management Seventh Revised Edition, 2017
6. Peter Drucker, Innovation and Entrepreneurship: Practice and Principles, Harper & Row, New York, 6th Edition.

Course Title: Software Laboratory – VI

Course Code: STP601

Marks: 50

Type: General Practical

Credits: 02

Hours: 60

Course objectives:

- To understand and implement the concepts learnt in Elective course STG601 E1/E2

Learning Outcome:

After successful completion of this course, the student will be able to

- Demonstrate the practical ability of the concepts learnt in elective course STG601 E1/E2

Choose the section mapped with the elective course (STG601) chosen by you.

List of suggested Practical:

A student shall complete practical of minimum 50 Hours duration

Section A - Based on STG601E1

Software Recommended: Adobe Suite

1. Colour Scheme understanding and Implementation [10 Lectures]
2. Tessellation of shapes (geometric and organic) with various colour schemes [06 Lectures]
3. Layout Design for publication [14 Lectures]
4. Designing and Publishing Booklet, Portfolio, Books, Newspaper, Magazines[20 Lectures]
5. Designing and Publishing interactive PDF and Other E-Pub Formats, Kindle etc...[10 Lectures]

Recommended Reference Books: *As mentioned in STG601E1 and*

1. Ambrose Harris, The Production Manual, AVA Publishing
2. Timothy Samara, Design Elements, a Graphic Style Manual, Rockport Publishers

Section B - Based on STG601E2

1. Familiarization of the relational database: Data definition commands CREATE, ALTER, DROP, Adding Constraints Primary key, foreign key, unique key, check, not null. [10 Lectures]
2. Basic SQL queries INSERT, SELECT, DELETE, UPDATE, Using multiple tables, ordering of rows using ORDER BY, GROUP BY option, Set operations using UNION, EXCEPT, INTERSECT, Substring Comparison using LIKE operator, IN, BETWEEN operator. [12 Lectures]
3. Complex Queries, Nested Queries, EXISTS and UNIQUE, DISTINCT functions, NULL values, Renaming of attributes and Joining of tables, Aggregate functions and grouping. [12 Lectures]
4. Programs involving views [06 Lectures]
5. Programs involving cursors [06 Lectures]
6. Programs involving triggers [06 Lectures]
7. Stored procedures, stored procedures with parameters. [06 Lectures]
8. Transaction Control Language(TCL) commands [02 Lectures]

Recommended Books: *As mentioned in STG601E2*

Course Title: SSC/Q0501, Software Developer
Course Code: STS601
Marks: 700

Type: Skill (Theory, Practical, & OJT)
Credits: 18 (Theory-7, Practical & OJT-11)
Hours: 440

Course/Package objectives:

To enable Students to:

- Contribute to the design of software products and applications
- Develop software code to specification
- Manage their work to meet requirements
- Work effectively with colleagues
- Maintain a healthy, safe and secure working environment
- Provide data/information in standard formats
- Develop their knowledge, skills and competence

Learning Outcomes:

Students will have competencies and skills as mentioned in the Model Curriculum of SOFTWARE DEVELOPER designed by NASSCOM (REFERENCE ID: SSC/Q0501, version 1.0)

Package Syllabus (Adopted Model Curriculum of Nasscom)

— *Lecture Hours and Marks Distribution as per Nasscom standard mentioned in Model Curriculum*

1.	Programming and Algorithms
2.	Analysis and Design of Software Applications
3.	Application Development
4.	Self and work Management
5.	Team Work and Communication
6.	Managing healthy and safety
7.	Data and information Management
8.	Learning and self Development

On-Job-Training (OJT) / Project Component (Compulsory)

It is required by a student to carry-out and complete

- **OJT** in the relevant industry or working environment and in the relevant skill area (Software Development).

Duration of OJT: Minimum 45 Hours

A Student needs to submit the detailed report of the OJT along with the 'Certificate of Completion'.

OR

- **Project /Mini-Project** : Software Development

A Student needs to submit the project report along with the finished product and other artifacts (if any).

Recommended Text Books:

Qualification Pack Hyperlink:

https://nsdcindia.org/sites/default/files/QP_SSC-Q0501_Software_Developer.pdf

Course Title: SSC/Q2702, Associate -DTP
Course Code: STS601
Marks: 700

Type: Skill (Theory, Practical, & OJT)
Credits: 18 (Theory-7, Practical & OJT-11)
Hours: 440

Course/Package objectives:

To enable students to

- Understand and Provide/control access to publications
- Publish Content
- Manage work to meet requirements
- Work effectively with colleagues
- Develop constructive work habits
- Maintain a healthy, safe and secure working environment
- Provide data/information in standard formats
- Develop your knowledge, skills and competence

Learning Outcomes:

Students will have competencies and skills as mentioned in the Model Curriculum of Associate -DTP designed by NASSCOM (REFERENCE ID: SSC/Q2702, version 1.0)

Package Syllabus (Adopted Model Curriculum of Nasscom)

— *Lecture Hours and Marks Distribution as per Nasscom standard mentioned in Model Curriculum*

1.	Provide/control access to publications
2.	Publish Content
3.	Manage your work to meet requirement
4.	Work effectively with colleagues
5.	Maintain a healthy, safe and secure working environment
6.	Provide data/information in standard formats
7.	Develop your knowledge, skills and competence

On-Job-Training (OJT) / Project Component (Compulsory)

It is required by a student to carry-out and complete

- **OJT** in the relevant industry or working environment and in the relevant skill area (Desktop Publishing).

Duration of OJT: Minimum 45 Hours

A Student needs to submit the detailed report of the OJT along with the 'Certificate of Completion'.

OR

- **Project /Mini-Project** : Desktop Publishing

A Student needs to submit the project report along with the finished product and other artifacts (if any).

Recommended Text Books:

Qualification Pack Hyperlink:

<https://www.sscnasscom.com/qualification-pack/SSC/Q2702/>

Annexure IV

List of Qualification Packs Considered for B.Voc.(Software Technologies)

2

1. SSC/Q0508 Junior Software Developer
https://nsdcindia.org/sites/default/files/QP_SSC-Q0508_Junior-Software-Developer.pdf
2. SSC/Q0503 Web Developer
https://nsdcindia.org/sites/default/files/QP_SSC-Q0503_Web-Developer.pdf
3. SSC/Q0504 Media Developer
https://nsdcindia.org/sites/default/files/QP_SSC-Q0504_Media-Developer.pdf
4. SSC/Q0509 Master Trainer for Junior Software Developer
https://nsdcindia.org/sites/default/files/QP_SSC-Q0509_Master-Trainer-for-Software-Developer.pdf
5. SSC/Q0502 UI Developer
https://nsdcindia.org/sites/default/files/QP_SSC-Q0502_UI-Developer.pdf
6. SSC/Q0501, Software Developer
https://nsdcindia.org/sites/default/files/QP_SSC-Q0501_Software_Developer.pdf
7. SSC/Q2702, Associate –DTP
<https://www.sscnasscom.com/qualification-pack/SSC/Q2702/>

Annexure - V

Subject : B.Sc Computer Science (T.Y.B.Sc. Structure)

Semester	S r l	Course Code	Course Title	Credit s	Marks	Hours
TYBSc (Semester V & VI) (Honours Programme)						
Semester V	Core Courses					
	1	CSC105	Computer Networks	4T+2P	100T+50	60T+60P
	2	CSC106	Object Oriented Programming	4T+2P	100T+50	60T+60P
	3	CSC107	Software Engineering	4T+2P	100T+50	60T+60P
	Discipline Specific Electives (Any 2 DSE's)					
	4 . 5	CSD101	Human Computer Interaction	3T+1P	75T+25P	45T+30P
		CSD102	Data Mining	3T+1P	75T+25P	45T+30P
		CSD103	Natural Language Processing	3T+1P	75T+25P	45T+30P
		CSD104	Embedded Systems	3T+1P	75T+25P	45T+30P
Core Courses						
Semester VI	6	CSC108	Mobile Application Development	4T+2P	100T+50	60T+60P
	7	CSC109	Full Stack Web Development	4T+2P	100T+50	60T+60P
	8	CSC110	Internet of Things	4T+2P	100T+50	60T+60P
	Discipline Specific Elective (Any 1 DSE)					
	9 . .	CSD105	Network Security	3T+1P	75T+25P	45T+30P
		CSD106	Multimedia Techniques	3T+1P	75T+25P	45T+30P
		CSD107	Introduction to Data Analytics	3T+1P	75T+25P	45T+30P
	1	CSP101	Project	4	100	
TYBSc (Semester V & VI) (General Programme)						
Semester V	Skill Enhancement Course					
	1	CSS107	Agile Software Development	3T+1P	75T+25P	45T+30P
	2	CSS108	Network Administration	3T+1P	75T+25P	45T+30P
	Discipline Specific Elective (Any 1 DSE from the following)					
	3 . .	CSD101	Human Computer Interaction	3T+1P	75T+25P	45T+30P
		CSD102	Data Mining	3T+1P	75T+25P	45T+30P
		CSD103	Natural Language Processing	3T+1P	75T+25P	45T+30P
		CSD104	Embedded Systems	3T+1P	75T+25P	45T+30P
Semester VI	Skill Enhancement Course					
	4	CSS109	PHP Programming	3T+1P	75T+25P	45T+30P
	Discipline Specific Elective / Project (any 1)					
	5 . .	CSD105	Network Security	3T+1P	75T+25P	45T+30P
		CSD106	Multimedia Techniques	3T+1P	75T+25P	45T+30P
		CSD107	Introduction to Data Analytics	3T+1P	75T+25P	45T+30P
CSP101		Project	4	100		

Core Courses

Course Code : CSC105		Course Title : Computer Networks		Semester : V	
Theory Marks : 100 Practical Marks : 50 Total Marks : 150		Theory Credits : 04 Practical Credits : 02 Total Credits : 06		Theory Lectures : 60 Practical Lectures : 60	
Course prerequisites : CSC101 Introduction to Programming using C					
Course objectives: • To provide a strong background of Network Concepts. • To be familiar with the components required to build and design different types of networks. • To explain the various protocols associated with the network layers					
Learning Outcomes : Upon completion of the course, students should be able to: • Describe the network models and networks based on type and topology. • Categorize and use transmission media based on their characteristics and applications. • Detect and correct errors using various techniques. • Explain different protocols for data transmission at the DLL. • Be able to setup networks and also implement subnetting. • Be able to apply different transport and application layer protocols.					
Course content:					
Unit		Topic			
#	Title		Content	Lectures	
I	Data Communication		Introduction: Beginnings of Networking and data communication, ARPAnet	10L	
			Networks: Components and Categories, Types of Connections, Topologies, Transmission Modes		
			Switching: Circuit switching, Message switching, Packet switching,		
			Protocols and Standards: Layered Architecture, OSI model, TCP/IP model;		
			Applications of Networks		
II	Physical Layer		Functions of Physical layer	8L	
			Data Encoding: Manchester, Differential Manchester		
			Transmission Media: Twisted pair, Coaxial Cable, Fiber Optics, Wireless Media		
			Physical layer Devices: Hub, Repeater		
III	Data Link Layer		Functions of Data link layer	14L	
			Data Framing techniques: Character Count, Character Stuffing, Bit Stuffing		
			Error detection and correction: Parity, CRC, Hamming code		

			Elementary Data Link Protocols: Stop and wait, Sliding window protocols - Go back-N: ARQ, Selective repeat ARQ		
			MAC Sublayer, Random Access Protocols: ALOHA, CSMA, CSMA/CD, CSMA/CA, Collision free protocols		
			Network Standards: IEEE 802.3 (Ethernet) frame format , Categories of standard ethernet – 10BaseT, 10BaseF, Bridged ethernet, separating collision domains , Switched ethernet , Fast ethernet IEEE 802.11 Architecture, frame structure		
			Data Link layer devices: Bridges, Switches		
IV	Network Layer		Functions of Network layer	12L	
			Network Service types: Virtual Circuits, Datagrams		
			Routing Algorithms: Shortest path routing, Flooding, Distance Vector routing, Link State routing; Hierarchical Routing		
			Congestion Control: Algorithms & Congestion Prevention Policies		
			Internet Protocols: IP Frame Format, IP Addressing, Subnets, Internet Control Protocols: ICMP, ARP, RARP, DHCP		
			Internetworking, Network layer device : Routers		
V	Transport Layer and Application Layer		Functions of Transport layer	10L	
			Transport Services: Connectionless, Connection-oriented, Transport service primitives Berkley sockets, Gateways		
			Transport layer Protocols: User Datagram Protocol, Transmission Control Protocol; Quality of Service parameters		
			Functions of Applications layer	06L	
			Electronic Mail; Domain Name System		
Text Book: Behrouz A. Forouzan; Data Communications and Networking , McGraw Hill Education; Fifth Edition					
References: Andrew S. Tanenbaum; Computer Networks, Pearson Education India;5 th Edition					
Suggested List of Practical : 1. Create scenario and study the performance of network with star topology through NS-2 simulation. 2. Implementation of framing using Bit stuffing and Character stuffing 3. Simulation of Stop and Wait Protocol using NS-2.					

4. Implementation of Dijkstra algorithm for Shortest Path Routing.
5. Configuring TCP/IP on a desktop.
6. Using diagnostic Network Commands: ping, traceroute, netstat, nslookup.
7. Simulate a Mobile Adhoc network (MANET) using NS2
8. Using Network protocol analyzer tool like ethereal (wireshark) or tcpdump to analyze network traffic
9. IP address manipulation (Extract network id and Host id given netmask)
10. Simulation of Congestion Control Algorithms using NS2
11. Implementation of IP fragmentation and reassembly.
12. Simple TCP client and server application (Single server-single client) – String manipulation and arithmetic operations.
13. Simple UDP client and server application (Single server-single client) – String manipulation and arithmetic operations.

Note :

Languages/Tools: C/C++, NS-2, Wireshark/TCPDump should be used for practicals.

		Programming			
Theory Marks : 100		Theory Credits : 04		Theory Lectures : 60	
Practical Marks : 50		Practical Credits : 02		Practical Lectures : 60	
Total Marks : 150		Total Credits : 06			
Course Pre-requisite : CSC101 Introduction to programming using C					
Course objectives: • To present the object oriented method, in viewpoint of software engineering — of the methods, tools and techniques for developing quality software in production environments. • To study how practicing software developers, in industrial as well as academic environments, can use object technology to improve the quality of the software they produce • Introduce Java Programming Environment and Design Patterns					
Learning Outcomes : Upon completion of the course students should be able to: • Use the characteristics of an object-oriented programming language in a program. • Use the basic object-oriented design principles in computer problem solving. • Use the basic principles of software engineering in managing complex software project • Write Java programs using classes and object • Implement Design Patterns in Java Programs					
Course content:					
Unit		Topic			
#	Title	#	Content	Lecture s	
I	CRITERIA OF OBJECT ORIENTATION		• On the criteria • Method and language • Implementation and environment • Libraries	4	
II	TOWARDS OBJECT TECHNOLOGY		• The ingredients of computation • Functional decomposition • Object-based decomposition • Object-oriented software construction • Issues	5	
III	THE STATIC STRUCTURE: CLASSES		• Objects are not the subject • Avoiding the standard confusion • The role of classes • A uniform type system • A simple class • Basic conventions • The object-oriented style of computation • Putting everything together	6	
IV	THE RUN-TIME STRUCTURE: OBJECTS		• Objects • Objects as a modelling tool • Manipulating objects and references • Creation procedures • More on references • Operations on references	6	

			• Attachment: reference and value semantics • Dealing with references: benefits and dangers		
V	MEMORY MANAGEMENT		• What happens to objects? • The casual approach • Reclaiming memory: the issues • Programmer-controlled deallocation • The component-level approach • Automatic memory management • Reference counting • Garbage collection • Practical issues of garbage collection	6	
VI	INTRODUCTION TO INHERITANCE		• What is inheritance? • Overriding and Polymorphism • Typing for inheritance • Dynamic binding • Deferred features and classes • The meaning of inheritance • The role of deferred classes	6	
VII	MULTIPLE INHERITANCE		• Examples of multiple inheritance • Feature renaming • Flattening the structure • Repeated inheritance	3	
VIII	EXCEPTION HANDLING		• Basic concepts of exception handling • Handling exceptions • An exception mechanism • Exception handling in Java	3	
IX	GENERICITY		• Horizontal and vertical type generalization • The need for type parameterization • Generic classes • Arrays • Generics and collection framework in Java	9	
X	DESIGN PATTERNS : INTRODUCTION		• What is a Design Pattern? • Describing Design Patterns. • How Design Patterns solve Design Problems • How to select a Design Pattern • How to Use a Design Pattern	2	
XI	CREATIONAL PATTERNS		• Factory Method • Prototype	2	

			• Singleton		
XII	STRUCTURAL PATTERNS		• Adaptor • Composite • Decorator • Façade • Proxy	4	
XIII	BEHAVIORAL PATTERNS		• Chain of Responsibility • Command • Iterator • Observer • State • Strategy	4	

Text Books:

1. Bertrand Meyer , Object Oriented Software Construction, Prentice Hall; Second edition
2. Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides, Design Patterns : Elements of Reusable Object-Oriented Software, Pearson
3. Khalid A. Mughal and Rolf W. Rasmussen, A Programmer's Guide to Java SCJP Certification, Addison-Wesley

Reference Books

1. Kathy Sierra, Bert Bates , Head First Java - A Brain-Friendly Guide, O'Reilly, Second Edition
2. Kathy Sierra, Elisabeth Freeman, Head First Design Patterns - A Brain-Friendly Guide, O'Reilly

Suggested list of practical :

Programs covering the following topics may be done :

1. Use of command line environment and run-time environment in Java (javac and java)
2. Creating classes
3. Constructors and overloading
4. Object composition using references
5. Use of standard libraries like Math, String, util.*
6. Inheritance
7. Overriding, polymorphism and dynamic binding
8. Abstract class, interfaces and multiple interface inheritance
9. Use of static keyword
10. Exception handling
11. Arrays
12. Collection framework – ArrayList, Maps
13. Minimum one exercise on each design pattern

Tools like Notepad, Eclipse may be used to do the practical.

Course Code : CSC107		Course Title : Software Engineering		Semester : V	
Theory Marks : 100 Practical Marks : 50 Total Marks : 150		Theory Credits : 04 Practical Credits : 02 Total Credits : 06		Theory Lectures : 60 Practical Lectures : 60	
Course prerequisites : -					
Course objectives: To study various methods used for software development with a stress on Agile Software Development					
Learning Outcomes : Upon completion of the course students should be able to:: • Explain Evolution and fundamentals of software engineering methods • Apply Agile software development method – Scrum • Apply refactoring techniques • Perform software testing using various quality assurance methods • Explain Source Control Tools					
Course content:					
Unit		Topic			
#	Title		Content	Lectur es	
I	Introduction to Software Engineering		Introduction to Software Engineering, Software Development phases(Requirements, Analysis, design and implementation, testing and maintenance), SDLC, Waterfall methodology, Prototyping and Iterative, Reverse engineering, reengineering	8	
II	Introduction to Source Control tools		Introduction to Source Control tools - versioning, check-in/checkout, commit, branching, merging, synchronization	4	
III	Agile Approach		Agile Approach: Agile Framework, Agile Manifesto, Agile Principles, Extreme Programming, Scrum	8	
IV	Software Project Management using Scrum		Software Project Management using scrum : User stories, Estimation using story points, sprint, backlog(product and sprint), Scrum team, scrum artifacts, scrum ceremonies	20	
V	Design and Implementation using XP		Design and implementation using XP: TDD, refactoring(code smells and refactoring techniques) , Unit testing, Pair Programming	10	
VI	Quality Assurance		Quality assurance (Verification & Validation): Testing approaches, Types of testing, testing tools- JUnit, Selenium, Build tools, Iteration and Release planning, Introduction to Continuous Integration	10	

Text Books:

1. Pankaj Jalote , Integrated Approach to Software Engineering, Narosa Publishing House
2. Chris Sims and Hillary Louise Johnson , Elements of Scrum, Dymaxicon, LLC
3. Martin Fowler, Refactoring, Addison Wesley; 2nd edition

Reference Books:

1. Ken Schwaber, Mike Beedle , Agile Software Development with Scrum, Pearson Education
2. S. Kenneth Rubin, Essential Scrum: A Practical Guide to the Most Popular Agile Process, Pearson Education
3. Kent Beck, Extreme Programming Explained: Embrace Change, Addison Wesley, 2nd Edition

Suggested List of Practical :

1. Git
2. User stories, Estimation
3. Burndown charts, Scrum board, Trello,
4. JUnit, Selenium
5. Refactoring exercises(pair programming)
6. Debugging and defect tracking using Bugzilla
7. Maven Build
8. Javadoc

The above practical should be done using a mini project using scrum and by performing refactoring exercises using pair programming. The tools Eclipse, Git, Selenium, Bugzilla, Trello are suggested.

Theory Marks : 100 Practical Marks : 50 Total Marks : 150		Theory Credits : 04 Practical Credits : 02 Total Credits : 06		Theory Lectures : 60 Practical Lectures : 60	
Course prerequisites : 1). CSC106 Object Oriented Programming 2). CSC103 Database Management Systems					
Course objective : Introduce mobile application development for the Android platform using XML and Java/Kotlin. Include developing simple applications that could run on Android phones and tablets. Cover Android application terminologies, components and coding.					
Learning Outcomes : Upon completion of the course students should be able to : • Describe the anatomy of a mobile app. • Use Android components in designing simple mobile applications. • Identify the significance of each of the Android basic building blocks and determine when to use which component. • Discuss the data storage options available on android platform and perform basic CRUD operations on persistent data. • Design complete Android app by integrating the android building blocks and using firebase as backend tool.					
Course content:					
Unit		Topic			
#	Title	#	Content	Lectures	
I	Introduction to Mobile Apps and Android	A	Introduction: Need for Mobile Apps. Different types of Mobile Apps. Android vs. Other mobile platforms Open Handset Alliance (OHA) Features of Android Android Limitations	03	
		B	Mobile Navigation: Basic patterns, Pros and Cons Screen independent design - Resolution and density independence (px, dip, dp, sip, sp)	02	
		C	Android Pre-requisites: Java/Kotlin Programming fundamentals Introduction to XML Introduction to Build System (Gradle)	05	
		D	Android Architecture: Overview of Android Architecture Internals Linux Kernel, Libraries, Android Runtime, Application Framework, Dalvik VM and .apk files Emulator – Android Virtual Device: Definition, Features, Examples	03	
		E	Android Basic Building blocks: Activities, Services, Broadcast Receivers and Content	02	

			Providers UI Components: Views and notifications		
II	Activities and UI	A	Introduction: Activities, Views, layouts (LinearLayout and RelativeLayout and WebView) Responsive UI with ConstraintLayout Significance of application manifest file	04	
		B	Creating User Interface (UI): Common UI components (TextView, EditText, Button, Checkbox, RadioButton , ToggleButton, Spinner, Pickers) Activity lifecycle Understanding the exception handler	04	
		C	Event Handling: onClick(), onLongClick(), onFocusChange(), OnKey(), onTouch(), onCreateContextMenu(), onCreateOptionsMenu()	04	
		D	Intents: Intents uses, intent types (Implicit & Explicit), Passing data(Direct, Bundle & Parcelable) Implicit Intents – Intent Filters and Intent Resolution Process, Pending intents	04	
		E	Advanced UI: Building Layouts with an Adapter (GridView & ListView), Custom Adapters, Menus (Options menu & Context menu), Toast, Custom Toast, Dialogs, Status bar Notifications.	04	
III	Broadcast Receivers and Services	A	Broadcast Receivers (BR): Broadcast receiver registration (Static & Dynamic), Broadcast Receiver Classes, Sticky & non-sticky BR, BR Security ,Understanding Broadcast action, category and data, Sending & Receiving Broadcast	03	
		B	Services: Overview of services in Android, Implementing a Service, Service lifecycle, Inter Process Communication (AIDL Services).	03	
		C	Web Services and WebView : Consuming web services, Receiving HTTP Response (XML, JSON), Parsing JSON and XML, Using WebView	03	
IV	Multithreading	A	Multithreading: Background processing in android	02	
		B	Threads running on UI thread: Handlers & Runnable, AsyncTask	02	
V	Data Storage	A	Introduction to data storage: Shared Preferences: Introduction, Preferences types, operating modes Android File System:	04	

			Internal storage, External storage.		
		B	SQLite: Basics of SQLite Database, Data Types, SQLite Connections, SQLiteOpenHelper class	03	
		C	SQLite Queries: Working with cursors, Inserting, updating, and deleting contents of SQLite	03	
		D	Firestore: Introduction, Firestore services (Authentication, Cloud messaging, Database query)	02	

Reference Books :

1. Dawn Griffiths and David Griffiths, Head First Android Development: A Brain-Friendly Guide, Shroff/O'Reilly; Second edition, 2017.
2. John Horton, Android Programming for Beginners, Packt Publishing Limited , 2015.
3. Erik Hellman, Android Programming: Pushing the Limits (MISL-WILEY) , Wiley,2013
4. Iyanu Adelekan, Kotlin Programming by Example, Packt Publishing Limited,2018.
5. Valentino Lee, Heather Schneider, and Robbie Schell, Mobile Applications: Architecture, Design, and Development, Prentice Hall, 2004.
6. Rajiv Ramnath, Roger Crawfis, and Paolo Sivilotti, Android SDK 3 for Dummies, Wiley, 2011.

Web References

1. <https://developer.android.com/>
2. <http://www.tutorialspoint.com/android/>
3. <http://www.appinventor.org/>
4. <http://www.smashingmagazine.com/guidelines-for-mobile-web-development/>
5. <https://www.smashingmagazine.com/2017/05/basic-patterns-mobile-navigation/>

Suggested List of Practical :

1). Setup Android Studio Environment

- Install or update Android Studio to its latest version.
- Ensure that your Android app:
 - Targets API level 16 (Jelly Bean) or later
 - Uses Gradle 4.1 or later
- Create Android Studio Environment,
- Explore Android Studio IDE
- Set up a device or emulator for running your app (Android studio's Emulator, Genymotion, Koplayer, Memu)
- Sign into Firebase using your Google account.
- Connect your Android app to Firebase

Testing and Debugging Android Application (Use of Dalvik Debug Monitor Server (DDMS), Use of Step Filters, Breakpoints, Suspend and Resume, use of LogCat (Verbose, Debug, Info, Warn, Error, Assert), Use of Perspectives

2). Create Simple Calculator Application

Layout design with constraint layout, Implement app, Debug and find errors, Installation of .apk into

your android mobile

3). Create Tic tac toe Game

Layout design with TableLayout , Add colors, Buttons call event, Find winner, Play with device.

4). Create Zoo app

- ListView layout design, Load listView with Data, Load different views in ListView, ListView events, Add or Remove item to listView

5). Create Restaurant App

Design GridView Layout, Load gridView with list of foods, Show Food details

6). Web services - Find City sunrise time app

HTTP calls, JSON and XML, UI layout design, HTTP calls and JSON read

7). Sqlite database - My notes

Design Add notes layout, Use Menu bar and Intent, Styles, Add notes to Sqlite database, List notes from Sqlite database, Delete notes from Sqlite database, Update notes in Sqlite database

8). Complete Alarm App

- Layout design, Broadcast Receiver and Service concept, Broadcast Receiver and Set alarm time SharedPreferences and start app with OS

9). Use Firebase Realtime Database, to build a mobile system

- User login with phone number, Menus for contact and main activity, Dummy contact list data, Pick contacts from phone. Save my trackers in Shared preferences, Firebase signIn anonymously, Save user info into Firebase, Save my trackers in Realtime- database, People who I find using dummy data, People who I find using Firebase data, Load user's contacts, Send phone location to the server, Find missing phone location
- Services Vs Broadcast receiver
- Service, Run App in background

Note :

Languages/Tools: Java/Kotlin, XML, Android Studio, AVD (Android studio's Emulator, Genymotion, Koplayer, Memu), Firebase may be used for practical.

Theory Marks : 100 Practical Marks : 50 Total Marks : 150		Theory Credits : 04 Practical Credits : 02 Total Credits : 06		Theory Lectures : 60 Practical Lectures : 60	
Course prerequisites : - Basic programming knowledge					
Course objective : • To Introduce concepts for internet of things and the different devices involved in IOT. • To introduce cloud concepts and its use in IOT • Gain hands on experience of working with different sensors/actuators and their use in IOT projects. • To gain knowledge of Arduino, NodeMcu, Raspberry pi Boards and to develop IOT projects by integrating these boards with a cloud platform.					
Learning Outcomes : Upon completion of the course students should be able to : • Explain the requirements and components of an IOT system. • Develop different IOT projects using cloud technology • Develop IOT Projects using the Arduino, NodeMcu, Raspberry pi Boards and a cloud platform such as Nodered or similar.					
Course content:					
Unit		Topic			
#	Title	#	Content	Lectures	
I	IOT Concepts	A	Introduction Definition, modern day IoT applications, Baseline technologies-M2M,WoT, IOT categories- industrial and consumer, IOT components	03	
		B	Sensors and Actuators sensors, transducers, sensor features, resolution, analog sensors, digital sensors, scalar sensors, vector sensors, sensor types. Actuators-types-hydraulic, pneumatic, electrical, thermal/mechanical ,motors-DC, Servo, Stepper, relays, motor drivers for interfacing	06	
		C	IOT Networks IoTWF Standardized Architecture, Connectivity Protocols- MQTT,SMQTT; communication protocols-IEEE 802.15.4,802.11,LORA wireless protocol ,ZigBee	06	
II	IOT Boards	A	Arduino Introduction to Arduino Programming-features of arduino, Arduino IDE, sketch, sketch structure, supported data types, Arduino function libraries, operators, control statements, arrays, String functions, Interrupts, sensor interface with Arduino, DHT sensor library, types of motor actuators, Arduino servo library	10	
		B	Raspberry Pi and comparative study Introduction to Raspberry Pi – specifications, GPIOs, Features of EsP8266 ,comparative studies of Arduino	06	

			uno, raspberry pi, nodemcu boards and their applications		
III	Cloud Technology	A	Introduction to cloud computing definition, characteristics, components, service models-IaaS, Pass, SaaS, Deployment models-public, private, hybrid, open source and commercial clouds-examples, facilities offered	10	
		B	Cloud computing case studies Microsoft Azure-features, Azure as PaaS, Azure as IaaS; OpenStack-components and features, Firebase cloud service features	06	
		C	Visual tool for wiring IOT NodeRed, its features, installing on Raspberry pi	04	
		D	Wireless sensor networks definition, limitations; Sensor cloud-definition, difference with WSN, Actors in sensor cloud, architecture	05	
		E	Fog computing Introduction, why use fog computing, when to use fog computing, architecture of fog, fog nodes, working of fog, applications of fog	04	

Reference Books :

1. Arshdeep Bagha, Vijay Madiseti , Internet of Things: A Hands-On Approach, Orient Blackswan Private Limited
2. Adrian McEwen , Designing The Internet of Things, Wiley
3. Hanes David, Salguero Gonzalo , IoT Fundamentals: Networking Technologies, Protocols and Use Cases for the Internet of Things , Pearson Education

Suggested List of Practical :

1. Blink an LED . Traffic lights using Arduino board.
2. Night security light using PIR motion sensor and photo resistor.
3. Arduino weather station with temperature, humidity, pressure date and time.
4. Controlling a server motor using arduino.
5. Setting up Raspberry pi and blinking LED
6. Capturing an image using Raspberry Pi
7. DHT22 interfaced with Raspberry Pi to record the temperature.
8. Setting up server on Raspberry Pi and sending the recorded temperature to the server.
9. Installing Nodered, creating and testing a simple flow in Nodered.
10. Controlling an LED with Nodered.
11. Use of Digital Smoke and gas sensor to detect gas/smoke with ESP8266, MQTT and Nodered.
12. Controlling lamps and outlets using Arduino and MQTT

Note :

Tools like Arduino IDE, python editor may be used

Course Code : CSC109	Course Title : Full Stack Web Development	Semester : VI
Theory Marks : 100	Theory Credits : 04	Theory Lectures : 60

Practical Marks : 50 Total Marks : 150		Practical Credits : 02 Total Credits : 06		Practical Lectures : 60	
Course prerequisites : - Working knowledge of HTML,CSS, JavaScript					
Course objective: - To know the core concepts of Node js and React for server side and client side coding . - To gain insight and understand the working of MVC architecture with MERN - To introduce the no-sql database - MongoDB					
Learning Outcomes : Upon completion of the course students should be able to:: - Explain the significance of each of the MERN components. - Develop a CRUD application using MongoDB. - Develop applications using NODEjs, React to understand the different aspects of these technologies. - Design and implement a full-fledged application using all the components of the MERN Stack					
Course content:					
Unit		Topic			
#	Title	#	Content	Lectures	
I	JavaScript	A	JavaScript basics: Introduction, Syntax and Statements, Comments, Operators, Variables, Z `Assignment, Loops, If and Switch conditions, break & Continue, Data Types: Number & Number Methods, Strings and String Methods, Functions, Callbacks, Arrays, Array Methods, Looping through an array (Array Iteration)	03	
		B	JavaScript advanced features Let and Const, Let inside loops, String Templates For of Loops, map, reduce, filter, Arrow Functions Class, Class properties and methods, object, this operator, spread function, Class Constructor, Class Inheritance, Modules import and export, Form validation using validation API	06	
		C	AJAX Ajax - request object creation, forwarding the request, accepting response object and display on webpage, JSON syntax, XmlHttpRequest Object	02	
II	Node Js	A	Introduction Advantages of Node JS , Node.js Process Model, Working in REPL, Node JS Console.	02	
		B	Node JS Modules Functions, Buffer, Module, Module Types : Core Modules, Local Modules, Module.Exports	03	

		C	Node Package Manager NPM, Installing packages Locally, Adding dependency in package.json, Installing packages globally, Updating packages.	02	
		D	Web Server Creating Web Server, Handling HTTP Requests, Sending Requests.	03	
		E	File System Fs.readFile, Writing a File, Writing a file asynchronously, Opening a file, Deleting, Other IO Operations.	03	
		F	Events EventEmitter Class, Returning event emitter, Inheriting events	03	
III	React	A	React elements and JSX What is react? advantages and disadvantages Overview of JSX, creating elements with JSX, JSX components, properties in JSX, condition statements in JSX, Rendering an Element into the DOM, Naming Conventions	05	
		B	Components & Component Life Cycle Overview of Components, Props, State, component composability, Life Cycle Methods, Reusable Components	06	
		C	Forms Submitting the form data to server using react component and updating state, Validating Props, Call back events.	06	
IV	Express Js	A	Introduction Introduction to Express, MVC pattern, initial node server setup, adding data to the server(ready data from mockaroo)	02	
		B	Routing	04	

			create a basic route, add a static route for file server, routing with express : routing parameters, routing handler, routing common methods, routing chaining,		
		C	HTTP Interaction: Handling Form Data, Handling Query Parameters, Cookies and Sessions	03	
		D	RESTful Services Creating and Consuming RESTful Services, Using Templates	03	
V	MongoDb		Introduction to MongoDB, Document-oriented database, key features, databases and collections , CRUD operations, Text search operations	04	

Reference Books :

1. Basarat Ali Syed , Beginning Nodejs, Appress
2. Vasan Subramanian , Pro MERN Stack **Full Stack Web App Development with Mongo, Express, React, and Node**, Appress, 1st Edition
3. Eddy Wison,Iriarte Koroliova ,MERN Quick Start Guide: Build web applications with MongoDB, Express.js, React, and Node , Packt

Web References

1. <https://docs.mongodb.com>
2. <https://medium.com/poka-techblog/>
3. <https://javascript.info/>
4. <https://reactjs.org/>

Suggested List of Practical :

1. JavaScript
 - a) Demonstration of const, let, string templates, callbacks, arrow functions, class, class-properties, methods.
 - b) Handling asynchronous request using AJAX and JSON with a simple server script that just returns JSON data.
2. Nodejs - Installation and Nodejs Core
 - a) Installing Nodejs
 - b) Use of global object
 - c) Argument variables with process.argv
 - d) Standard input and standard output
3. Nodejs Modules

- a) Core Modules
- b) Collecting information with Readline
- c) Handling events with EventEmitter
- d) Exporting custom modules
- e) Creating child process with exec/spawn

4. The File system

- a) Listing directory files
- b) Reading files
- c) writing and appending files
- d) Directory creation
- e) Readable file streams
- f) Writable file streams

5. The Http Module

- a) Making a request
- b) Building a web server
- c) serving files
- d) serving JSON data
- e) Collecting POST data

6. The Node Package manager

- a) Installing node NPM
- b) Initializing a package-json file
- c) Adding node packages
- d) Managing global directory
- e) Updating a package
- f) Removing a package
- g) File servers with httpster

7. Introduction to React Elements

- a) Installing create-react-app
- b) Generate a project
- c) Create react elements
- d) Refactor elements using JSX

8. React components

- a) Create a react component
- b) Add component properties
- c) Creating a component(ex Book) with data
- d) Adding custom methods
- e) creating function components.

9. Props and states

- a) Compose components
- b) Display child components
- c) Introducing state
- d) Using setstate
- e) pass state as props

f)Conditional rendering

10. Additional react features

a) Component life cycle

b)Fetching and rendering JSON data

c)Using forms with react.

11.Mongodb

a) installation

b)collection and basic operations(find, create, update ,delete, drop)

c)Reading and writing to MongoDB database using APIs

12. Developing a simple CRUD application using the MERN stack.

Note :

The Tools visual Studio code/sublime may be used for practical

Discipline Specific Electives

Course Code : CSD101	Course Title : Human Computer Interaction	Semester : V
Theory Marks : 75	Theory Credits : 03	Theory Lectures : 45

Practical Marks : 25 Total Marks : 100		Practical Credits : 01 Total Credits : 04		Practical Lectures : 30	
Course objectives: • To introduce the foundations of Human Computer Interaction, design technologies and user interface design and development. • Learn the foundations of Human Computer Interaction • Be familiar with the design technologies for individuals and persons with disabilities • Learn the guidelines for user interface design and development • Be aware of mobile HCI					
Course Outcomes : Upon completion of the course students should be able to:: • Develop meaningful user interface • Assess the importance of user feedback • Design effective HCI for individuals and persons with disabilities • Develop persona, conduct interview • Develop storyboard and design prototype • Design GUI, Web UI and Reports. • Perform Heuristic Evaluation of the design					
Course content:					
Unit		Topic			
#	Title		Content	Lectures	
I	FOUNDATIONS OF HCI		The Human: I/O channels, Memory, Reasoning and problem solving; The computer: Devices, Memory, processing and networks; Interaction: Models, frameworks, Ergonomics, styles, elements, interactivity, Paradigms	8	
II	DESIGN – RULES AND TECHNIQUES		Interactive Design basics: process, scenarios, navigation, screen design, Iteration and prototyping. Usability engineering, Prototyping in practice, design rationale. Design rules: principles, standards, guidelines, rules. Evaluation Techniques, Universal Design.	8	
III	MODELS AND THEORIES		Cognitive models, Socio-Organizational issues and stake holder requirements; Communication and collaboration models-Hypertext, Multimedia and WWW	8	
IV	MOBILE HCI		Mobile Ecosystem: Platforms, Application frameworks, Types of Mobile Applications: Widgets, Applications, Games; Mobile Information Architecture, Mobile 2.0, Mobile Design: Elements of Mobile Design, Tools.	8	
V	WEB INTERFACE DESIGN		Designing Web Interfaces: Drag & Drop, Direct Selection, Contextual Tools, Overlays, Inlays and Virtual Pages, Process Flow	8	
VI	CONTEMPORARY		Future Domains, IHCI and Case Studies	5	

	INTERFACE DESIGN TECHNOLOGY			
Reference Books : 1. Alan Dix, Janet Finlay, Gregory Abowd, Russell Beale; Human Computer Interaction; Pearson Education, 2004 (UNIT I,II and III), 3rd Edition. 2. Brian Fling; Mobile Design and Development , OReilly Media Inc., 2009 (UNIT –IV) 3. Bill Scott and Theresa Neil ; Designing Web Interfaces; OReilly, 2009 (UNIT V), First Edition				
Suggested list of practical : 1. Paper Prototyping using templates 2. Story boarding 3. Conducting survey interview and summarizing the result 4. Persona- conducting contextual interview and developing persona 5. GUI design- form design, menu design, help, error messages 6. Web UI design- pages, navigation, controls, (Ajax) 7. Report designs 8. Heuristic evaluation				

Course Code : CSD102	Course Title : Data Mining	Semester : V
Theory Marks : 75 Practical Marks : 25 Total Marks : 100	Theory Credits : 03 Practical Credits : 01 Total Credits : 04	Theory Lectures : 45 Practical Lectures : 30

Course objectives:

- To get an understanding of the general properties of data in large databases
- Understand a variety of real-world applications that require mining
- To introduce the basic concepts of Data Warehouse and Data Mining techniques.
- Examine the types of the data to be mined and apply pre-processing methods on raw data using data mining software.
- Become familiarized with association analysis, classification and cluster analysis of data objects.
- To discover interesting patterns, analyze and estimate the accuracy of popular data mining algorithms using different data sets.
- Get introduced to the challenges in mining complex data types.
- To develop skills of using data mining software for solving practical data mining problems.

Learning Outcomes : Upon completion of the course students should be able to:

- Design a DatawarehouseSchema
- Use Classification and prediction methods to solve problems
- Identify suitable clustering methods for different applications

Course content:

Unit		Topic		Lecture s
#	Title		Content	
I	Data Mining Overview		Evolution of Database Technology, What is Data Mining, Scope of Data Mining, Task of Data mining, Which Kind of Applications are Targeted-Business Intelligence, Web Search Engines, Common Data Mining Application Domains, Benefits of Data Mining, Data Mining and Society	3L
II	Data Pre-Processing		Data Objects and Attribute Types, Data Pre-processing – Data Quality: a Reason to Pre-process the Data, Major Task in Data Pre-processing-Data Cleaning-Missing Data; Noisy Data; Inconsistent Data, Need of Data Integration-Issues in Data Integration, Data Transformation, Need of Data Reduction, Data Visualization	4L
III	Data warehousing and OLAP		Introduction to Data Warehouse, Understanding a Data Warehouse, Data Warehouse Schema- Star, Snowflake, Fact Constellations, Data Warehouse Modeling: OLAP and Data Cube-What is OLAP, what is Data Cube, Data Cube as a Multidimensional Data Model, Dimensions: The Role of Concept Hierarchies, OLAP in Data Warehouse, OLAP Vs OLTP, Types of OLAP, Data Usage in Data Warehouse, Data Warehousing, A Three Tier Data Warehouse Architecture, Data Warehouse Design Process, Data Warehousing to Data Mining, Data Warehouse Applications	7L
IV	Basics of Data Mining:		Data Mining and Knowledge Discovery, What kind of Data can be Mined, Technologies used in Data Mining- Statistics; Database and Data	9L

			Warehouse Systems; Information Retrieval; Machine Learning; Pattern Recognition, Data Mining System Architecture, Data Mining Techniques, Issues in Data Mining- Mining Methodology and User Interaction Issues; Performance Issues; Diverse Data Type Issues		
V	Association Analysis		Introduction to Association Analysis, Frequent Patterns, Market Basket Analysis, Association Rule Mining-Problem Definition, Important Concepts; The Apriori Algorithm: Finding Frequent Itemsets Using Candidate Generation; Pseudocode for Apriori; Example of Apriori for Generating Frequent Itemsets ; Example of Apriori for Generating Association Rules, Mining Multilevel Association Rules, Mining Multidimensional Association Rules, Other Applications of Association Rule Mining	9L	
VI	Classification and Prediction		Introduction, Classification and Prediction Techniques, How Does Classification work, Building the Classifier; Using Classifier for Classification, General Approach to Classification, Classification and Prediction Issues. Classifier Accuracy-Confusion Matrix: Accuracy; Recall; Precision; F-Measure, Type-I, Type-II errors.	5L	
VII	Cluster Analysis		Introduction to Cluster Analysis, What is Clustering, Clustering Applications, Requirements for Clustering Algorithms, Major Clustering Methods- Partitioning Methods, Hierarchical Methods, Density Based Methods	5L	
VIII	Trends in Data Mining		Introduction, Mining Complex Data Types-Temporal Data Mining, Streaming Data Mining, Spatial Data Mining, Text Mining and Multimedia Data Mining, Web Mining-Categories of Web Mining	3L	

Text Book :

1. Jiawei Han and Micheline Kamber, "Data Mining Concepts and Techniques," 1st Edition Indian Reprint 2001, Harcourt India Private Limited.
2. Margaret Dunham, "Data Mining: Introductory and Advanced Topics," 1st Edition, 2003, Prentice Hall (Pearson Publication).
3. Arun K Pujari: Data Mining Techniques, 2nd Edition, Universities Press, 2009.

Reference Books :

1. Pang-Ning Tan, Michael Steinbach, Vipin Kumar: Introduction to Data Mining, Pearson Education, 2005.
2. Gupta G. K., Introduction to Data Mining with Case Studies, Prentice Hall India Learning Private Limited; Third edition (2014)

Suggested list of practical:

1. Create employee.arff dataset with attributes gender, age, salary, performance in the following format. Also create employee.csv dataset. View both the files.

```
@relation employee
@attribute gender{male,female}
@attribute age numeric @attribute
confirmed {yes,no}
@attribute salary{10000,15000,20000,25000,30000,35000} @attribute
performance{poor,average,good}
@data male,25,yes,10000,poor
male,25,no,15000,poor
female,25,yes,10000,poor
female,25,yes,15000,poor
male,28,no,15000,poor
female,28,no,10000,poor
male,30,yes,20000,average
female,30,no,20000,average
female,30,no,25000,average
male,30,no,25000,average
male,35, yes,30000,good
female,40,yes,35000,good
male,40,yes,30000,good
male,40,yes,35000,good
female,40,yes,35000,good
```

2. Apply preprocess on employee.arff and employee.csv dataset.
3. Carry out filter operation on employee dataset to do the following-1)Add attributes qualification of nominal type and experience of numeric type 2) Remove attribute confirmed 3) Normalize the dataset. Perform preprocess on the modified dataset.
4. Create studentbuyspc.arff dataset in the following format. Also create studentbuyspc.csv dataset. Perform preprocess on the datasets. (Add 20 more record instances of your choice and get the results).

```
@relation studentbuyspc @attribute
age {<30,30-40,>40}
@attribute income {low, medium, high} @attribute
student {yes, no}
```

```

@attribute credit-rating {fair, excellent}
@attribute buyspc {yes, no}
@data
<30, high, no, fair, no
<30, high, no, excellent, no
30-40, high, no, fair, yes
>40, medium, no, fair, yes
>40, low, yes, fair, yes
>40, low, yes, excellent, no
30-40, low, yes, excellent, yes
<30, medium, no, fair, no
<30, low, yes, fair, no
>40, medium, yes, fair, yes
<30, medium, yes, excellent, yes
30-40, medium, no, excellent, yes
30-40, high, yes, fair, yes
>40, medium, no, excellent, no

```

5. Create PhoneBuyDecision.arff dataset in the following format. Perform preprocess on the dataset. Identify noisy and missing data. (Add 10 more record instances of your choice and get results). Visualize the results.

```

@relation SmartPhoneBuyingDecision
@attribute age {Youth,middle,senior}
@attribute income {high,medium,low}
@attribute student {no,yes}
@attribute savingaccount {no,yes}
@attribute buyssmartphone {iphone6s,SamsungS7}
@data
Youth,high,no,no,iphone6s
Youth,high,no,yes,iphone6s
middle,high,no,SamsungS7
senior,medium,no,no,SamsungS7
senior,low,yes,no,SamsungS7
senior,low,yes,yes,iphone6s
middle,low,yes,yes,SamsungS
Youth,medium,no,no,iphone6s
Yout,low,yes,no,SamsungS7
senior,medium,yes,no,SamsungS7
Youth,medium,yes,yes,SamsungS7
midd,medium,no,yes,SamsungS7
middle,high,yes,no,SamsungS7
senior,medium,no,yes,SamsungS7
middle,high,yes,no,SamsungS7
senior,medium,no,yes,iphone6s
middle,low,yes,yes,SamsungS7
Youth,med,no,no,iphone6
Youth,high,no,no,iphone6s
senior,low,yes,no,SamsungS7
senior,low,yes,iphone6s
Youth,low,yes,no,SamsungS7
middle,high,no,no,SamsungS7
Youth,medium,yes,yes,SamsungS
senior,medium,no,SamsungS7
seni,medium,no,no,SamsungS7
middle,medium,no,yes,SamsungS7

```

6. Create the following five different datasets with 25 or more record instances in .arff and .csv file formats. Perform preprocess on these datasets. Analyze the results.

a) Dataset1: Patient

patient_id	A unique patient identification string
Sex	male/female
patient_type	child/adult
Sickness	allergy/acidity/migraine
treatment_type	allopathy/ayurveda/homeopathy
treated_for_years	<1/1-2/>2
treatment_effective	yes/no

b) Dataset2: Recruitment

Registraion_id	A unique registration identification string
current_employment	nil/temporary/contract
years_of_experience	Numeric
academic_background	graduate/masters
communication_skill	good/average/poor
technical_skill	good/average/poor
Recruited	yes/no

c) Dataset3: Applicant

applicant_id	A unique student identification string
Sex	male/female
Address	panjim/ mapusa/ margao/vasco
Community	general/sc/st/obc
course_applied	bcom/ba/bsc
twelfth_percentage	Numeric
transport_needed	true/false
Admitted	yes/no

d) Dataset4: Project

project_id	A unique project identification string
Title	title of project (nominal)
Department	computer/electronics/chemistry/physics/math
Expenditure	Numeric
Duration	number of years (numeric)
Authority	UGC/DST/AICTE/INSA
coordinator	Project Coordinator appointed or not (yes/no)
start_year	year of start of project (numeric)
Status	ongoing/withdrawn/completed

e) Dataset5: Bank

Id	a unique customer identification number
Age	age of customer in years (numeric)
Sex	MALE / FEMALE
Region	city/rural/urban/town
Income	income of customer (numeric)
Married	is the customer married (YES/NO)
Children	number of children (numeric)
Car	does the customer own a car (YES/NO)

land property	customer has a land property (YES/NO)
flat_or_house	customer own a flat or house (YES/NO)
save_acct	customer has a saving account (YES/NO)
current_acct	customer has current account (YES/NO)
loan_sanctioned	customer got sanctioned with personal loan (YES/NO)

7. Consider a real life data mining system of Stock Market/Airline/Retail Industry/Financial Organisation/Telecommunication/Social Media or any other of your choice. Collect the data, Create the dataset and perform preprocess on the dataset. Visualize the results.

8. Use studentbuyspc.arff dataset created with the attributes age ,income, student, credit-rating, buyspc. Perform classify (using J48 classifier (*C4.5 algorithm*)) on the dataset and carry out performance evaluation of the classifier.

9. Create twodcluster.arff dataset with the following data. Perform Cluster (using SimpleKmeans) on the dataset. Visualize cluster assignments.

ID	X	Y
1	1.0	1.0
2	1.5	2.0
3	3.0	4.0
4	5.0	7.0
5	3.5	5.0
6	4.5	5.0
7	3.5	4.5

10. Create customer.arff dataset in the following format. Perform Preprocess, Classify and Cluster on the dataset.

```
@relation customer
@attribute gender {male,female}
@attribute age {youth,middle,senior}
@attribute income numeric
@attribute class {A,B}
@data
male,youth,50000,A
female,youth,40000,A
male,youth,10000,B
male,middle,40000,A
female,middle,12000,B
male,senior,45000,A
female,middle,45000,A
female,senior,13000,B
male,senior,13000,B
female,youth,15000,B
male,youth,15000,B
female,middle,45000,A
male,middle,10000,B
male,senior,12000,B
female,senior,50000,A
```

11. create shopping.arff dataset in the following format. Perform Association Rule Mining with Apriori (Associate) and visualize the results.

```
@relation shopping
@attribute milk {0,1}
@attribute bread {0,1}
@attribute butter {0,1}
@attribute beer {0,1}
@attribute honey {0,1}
```

```
@data
```

```
1,1,0,0,1
0,1,0,1,0
0,1,1,0,0
1,1,0,1,0
1,0,1,0,0
0,1,1,0,0
1,0,1,0,0
1,1,1,0,1
1,1,1,0,0
```

12. Convert the following shopping data to the necessary format and carry out Association Rule Mining with Apriori on the dataset. (Add 10 more record instances of your choice and get the results)

```
1 bread,milk,biscuit,cornflakes
2 bread,tea,bournvita
3 jam,maggi,bread,milk
4 maggi,tea,biscuit
5 bread,tea,bournvita
6 maggi,tea,cornflakes
7 maggi,bread,tea,biscuit
8 jam,maggi,bread,tea
9 bread,milk
10 coffee,cock,biscuit,cornflakes
11 coffee,cock,biscuit,cornflakes
12 coffee,suger,bournvita
13 bread,coffee,cock
14 bread,suger,biscuit
15 coffee,suger,cornflakes
16 bread,suger,bournvita
17 bread,coffee,suger
18 bread,coffee,suger
19 tea,milk,coffee,cornflakes
```

13. Create weather.arff dataset using the following relation. Perform preprocess, classify, cluster and associate on the dataset. Convert temperature and humidity attributes to numeric type, make necessary changes to the respective data. Visualize the results

Relation: weather.symbolic					
No.	outlook Nominal	temperature Nominal	humidity Nominal	windy Nominal	play Nominal
1	sunny	hot	high	FALSE	no
2	sunny	hot	high	TRUE	no
3	overcast	hot	high	FALSE	yes
4	rainy	mild	high	FALSE	yes
5	rainy	cool	normal	FALSE	yes
6	rainy	cool	normal	TRUE	no
7	overcast	cool	normal	TRUE	yes
8	sunny	mild	high	FALSE	no
9	sunny	cool	normal	FALSE	yes
10	rainy	mild	normal	FALSE	yes
11	sunny	mild	normal	TRUE	yes
12	overcast	mild	high	TRUE	yes
13	overcast	hot	normal	FALSE	yes
14	rainy	mild	high	TRUE	no

14. Use five created datasets: 1.Patient 2.Recruitment 3.Applicant 4.Project 5.Bank . Perform preprocess, classify, cluster and associate on the datasets. Analyze the results.

15. Consider the following details of a Movie. Make your assumptions and create a dataset movie.arff. Apply preprocess, classify, cluster and associate on the datasets. Analyze the results
Year;Length;Title;Subject;Actor;Actress;Director;Popularity;Awards

16. Identify two data sets from UCI machine learning repository or <https://sites.google.com/site/labitis462/lab-tutorials> or the default datasets available with the software installation or create datasets of your choice. Perform preprocess, classify, cluster and associate on these dataset.

Note :

1. Practical are to be done using open source software tool like WEKA/python or any other machine learning software.
2. Questions are framed as per the format of WEKA. Make Necessary changes to use the datasets with other tools

Course Code : CSD103		Course Title : Natural Language Processing		Semester : V	
Theory Marks : 75 Practical Marks : 25 Total Marks : 100		Theory Credits : 03 Practical Credits : 01 Total Credits : 04		Theory Lectures : 45 Practical Lectures : 30	
Course prerequisites : Programming in Python					
Course objectives: - To learn the fundamental concepts of Natural Language Processing(NLP) and its applications in day to day life. - To learn Application of NLP using NLTK.					
Learning Outcomes : - Students will be able to recognize importance of NLP tasks in their day-to-day work and contribute to the development of resources and tools for NLP. - Students will be able to contribute to the development of resources and tools for NLP using NLTK.					
Course content:					
Unit		Topic			
#	Title	#	Content	Lectures	
I	Introduction to NLP	A	Language, Thought and Understanding, History of NLP, stages of NLP, Ambiguity, Models and Algorithms	2	
		B	Noisy Channel: Argmax Based Computation, Noisy Channel Application to NLP.	4	
II	Regular Expressions and Ngram	A	Regular Expressions, Basic Regular Expression Patterns, Disjunction, Grouping, and Precedence	2	
		B	Advanced Operators, Regular Expression substitution and counting Words, Simple N-gram.	2	
III	Finite-State Automata	A	Using an FSA for Recognition, Formal Languages	2	
		B	Non-Deterministic FSAs, Using an NFSA to Accept Strings, Recognition as Search.	2	
IV	Morphology	A	English Morphology, Inflectional Morphology, Derivational Morphology,	2	
		B	Finite-State Morphological, Parsing, The Lexicon and Morphotactics, Morphological Parsing with Finite-State Transduces, N-grams and word counting in corpora.	4	
V	Probabilistic Parsing	A	Parsing Algorithms, Evidence for Deeper Structure; Top Down Parsing Algorithms, Noun Structure; Top Down Parsing Algorithms, Non-noun Structure and Parsing Algorithms	6	
		B	Probabilistic parsing; sequence labelling, Part of Speech Tagging for Indian languages. Accuracy Measure.	4	
VI	HMM and Maximum Entropy Models	A	Markov chain, HMM Forward Algorithm, Viterbi Algorithm	5	
		B	HMM training: Forward- Backward Algorithm. And Maximum Entropy Models.	5	
VII	Wordnet and Word Sense		Wordnet, Metonymy and Word Sense	5	

	Disambiguation		Disambiguation, Overlap Based Method and Supervised Based Method.		
Text Book: 1. Jurafsky, Dan and Martin, James, Speech and Language Processing, Second Edition, Prentice Hall, 2008 References A. NPTEL Course on Natural Language Processing https://nptel.ac.in/courses/106101007/ B. Cousera course on Natural Language Processing https://www.coursera.org/learn/languageprocessing C. Books a. Allen, James, Natural Language Understanding, Second Edition, Benjamin/Cumming, 1995. 2. Charniack, Eugene, Statistical Language Learning, MIT Press, 1993 b. Manning, Christopher and Heinrich, Schutze, Foundations of Statistical Natural Language Processing, MIT Press, 1999. c. Natural Language Processing with Python--- Analyzing Text with the Natural Language Toolkit http://www.nltk.org/book_1ed/ Suggested list of practical : 1. Downloading and installation of NLTK tool. 2. Case study on WordNet (Download corpus of various languages). 3. To perform Tokenization of words and sentences using NLTK tool. 4. To study POS tagging and shallow parsing (chunking). 5. To study and implement Stemming with python NLTK. 6. To study and implement Lemmatization with python NLTK. 7. To create a Bag of Words based on minimum four documents in a single folder (feature extraction). 8. To find Synonyms for words in WordNet (IndoWordNet). 9. To construct n-gram (unigram, bigram, trigram) model for a given input (collocation extraction). 10. To study regular expression tagger. Mini Projects – a. Addition of New Synsets to the IndoWordnet b. Creation of Domain Specific POS tagged Text Corpus c. Creation of annotated Speech Corpus of minimum 20 minutes duration					

Course Code : CSD104		Course Title : Embedded Systems		Semester : V	
Theory Marks : 75 Practical Marks : 25 Total Marks : 100		Theory Credits : 03 Practical Credits : 01 Total Credits : 04		Theory Lectures : 45 Practical Lectures : 30	
Course prerequisites :					
Course objectives: • Conceptualize the basics of Embedded systems • Understand fundamentals of Real Time Operating Systems					
Learning Outcomes : • Explain the function and use of embedded system hardware and Interfacing I/O devices. • Identify various sensors, actuators and their use					
Course content:					
Unit		Topic			
#	Title		Content	Lectures	
I	Introduction		Introduction to embedded systems, Microprocessors and Microcontrollers, Components of Embedded system & its classification, characteristics of embedded system.	5L	
II	Interrupts		Microprocessor Architecture, Interrupt Basics, shared Data problem, Interrupt latency	8L	
III	Survey of Software Architecture		Round Robin, Round Robin with interrupt, Function-Queue-Scheduling Architecture, Real time OS Architecture	8L	
IV	Introduction to RTOS System		Task and Task states, Task and Data, Semaphores and shared Data	6L	
V	More OS services		Message Queues, Mailboxes and pipes, Timer functions, Events, Memory Management, Interrupt routines in RTOS Environment	8L	
VI	Embedded software Development Tools		Host and Target machines, Linkers/Locators for Embedded Systems, Getting Embedded software into the Target system	10L	
Text Book: 1). David E Simon, “An Embedded Software Primer”, Pearson India, 1 st Edition					

References

1. Tony Givargis Frank Vahid ; Embedded System Design: A Unified Hardware / Software Introduction, Wiley; Student edition
2. D. Patranabis, “ Sensors and Transducers”, PHI learning Private Limited

Suggested list of practical :

1. Interfacing sensors
2. Interfacing output devices
3. Interfacing input devices
4. Interfacing actuators
5. Programming with Raspberry Pi
6. Blink an LED, Traffic lights using Arduino Board
7. Monitoring Data over Cloud
8. Building Web app to control devices
9. A mini Project

Note :

Programs to be executed on some of the Embedded boards like Arduino, Intel Edison, Raspberry Pi, Bolt, etc that covers the above tasks.

Theory Marks : 75 Practical Marks : 25 Total Marks : 100		Theory Credits : 03 Practical Credits : 01 Total Credits : 04		Theory Lectures : 45 Practical Lectures : 30	
Course Code : CSD105		Course Title : Network Security		Semester : V	
Course objectives: - To learn the principles and practices of Cryptography and Network Security. - To enable the students understand the various methods of encryption, decryption and authentication.					
Learning Outcomes : - Explain the need and concepts of security. - Apply encryption techniques to secure data in transit across data networks.					
Course content:					
Unit		Topic			
#	Title	#	Content	Lectures	
I	Concepts of Security and Classical Encryption Techniques	A	Introduction: The need for security, Security approaches, Security attacks, Security Services, Security Mechanisms	10L	
		B	A Model for Network Security: Symmetric and asymmetric models		
		C	Encryption techniques: Substitution techniques – Caesar, Mono alphabetic, Polyalphabetic , Playfair , Vernam cipher Transposition techniques – Rail fence		
		D	Steganography: Difference between steganography and cryptography Techniques – Text steganography, Image steganography Applications, Limitations		
II	Symmetric and Asymmetric Key Algorithms	A	Algorithm types and Modes: Block Cipher Operation, Electronic Code Book, Cipher Block Chaining, Block Cipher Principles,	10L	
		B	Symmetric Key Cryptography		
		C	Diffie Hellman Key Exchange Algorithm, The Data Encryption Standard.		
		D	Asymmetric Key Cryptography: Overview, RSA algorithm Comparison between symmetric and asymmetric key cryptography, Digital Signatures		
III	Hash Functions and Message	A	Message Digest, MD5, SHA-1 and SHA-512	10L	

	Authentication Codes	B	Message Authentication Requirements		
		C	Message Authentication Functions, MAC, HMAC		
		D	Applications of Cryptographic Hash Functions		
IV	Digital Certificates	A	Public Key Infrastructure	9L	
		B	Digital Certificates: Technical details of Digital Certificates, Certification Authority, Digital Certificate Creation and verification		
		C	Certificate Hierarchies and Self-signed Digital Certificates.		
V	Firewall and Virtual Private Network	A	Introduction to network security techniques: IP Security	6L	
		B	Firewalls		
		C	Virtual Private Networks.		

References:

1. Kahate Atul, "Cryptography and Network Security", Tata McGraw-Hill.
2. Charlie Kauffman, Radia Perlman, Mike Spciner, "Network Security", Pearson Education
3. Behrouz A Forouzan, "Cryptography and Network security", McGraw Hill
4. Stallings William, "Cryptography and Network Security: Principles and Practices", 5th edition, Prentice Hall

Supplementary Reading

- 1). Alfred J. Menezes, Paul C. van Oorschot, "Handbook of Applied Cryptography", Jaypee Medical

Web Reference

1. <https://www.edureka.co/blog/steganography-tutorial>

Suggested list of practical :

1. Implementation of Caesar Cipher
2. Implementation of Vigenere Cipher
3. Implementation of Playfair Cipher
4. Implementation of Rail Fence Technique.
5. Implementation of Vernam Cipher
6. Perform Steganography using simple DoS commands and tools such as OpenStego
7. Use openSSI/JCrypt tool (or any other equivalent) and demonstrate asymmetric, symmetric cryptography, hashing and digital/PKI signatures / certificates.
8. Use of password cracking tools (ophcrack, John the Ripper) .Verify the strength of passwords using these tools.

9. Firewall and VPN setup

Note :

Languages/Tools: C/C++, openSSL/Jcrypt, OpenStego, ophcrack/ John the Ripper to be used for practical

Course Code : CSD106		Course Title : Multimedia Techniques		Semester : VI	
Theory Marks : 75 Practical Marks : 25 Total Marks : 100		Theory Credits : 03 Practical Credits : 01 Total Credits : 04		Theory Lectures : 45 Practical Lectures : 30	
Course prerequisites :					
Course objectives: To make the students aware of • Color Models and color harmony, • Raster and Vector Graphics formats & basic Graphic editing, • Font types, selection of fonts • Audio formats, codecs, basic audio editing, filters • Video formats, codecs, basic video editing, filters and transitions • Data compression.					
Learning Outcomes : • Explain Multimedia Concepts • Develop their Creativity and publish a self-contained multimedia Application using multimedia authoring tool in various application areas.					
Course content:					
Unit		Topic			
#	Title		Content	Lectures	
I	Introduction		Introduction to Multimedia, Social & Ethical considerations, Digital Representations, Standards	3	
II	Color Theory		Color Basics, Color Systems, additive and subtractive colors, HSL model, Color Wheel, Complementary Colors, After Images, Color Combinations, Color & Contrast, Color Psychology, Itten's Contrasts, Proportion & Intensity, Contrast & Dominance, Shades & Tints, Color Studies; Color Gamut, ICC profiles, Gamma Correction.	10	
III	Computer Graphics		Difference between Raster and Vector Graphics, Raster graphics : resolution, image compression, file formats, manipulation, Geometrical transformations; Vector graphics – fundamentals, file formats, shapes, transforms and filters	8	
IV	Text and Layout		character set, fonts, layout & Text in graphics	6	

V	Sound		Sampling, quantization, Sound Design, Audio Codec & file formats, processing sound, sound editing and effects, compression	7	
VI	Animation		Types of Animation, Keyframe, Sprite , Perception of vision, Human Color Perception, animated gifs, interpolating motion	5	
VII	Video		Aspect Ratio, Frame Size, Frame Rate, Regions, Video Codec & Formats, Processing, Delivery	6	

Reference Books:

1. Nigel Chapman, Jenny Chapman; "Digital Multimedia"; Wiley India Edition, 2nd Edition
2. Roger Parker; "One-Minute Designer"; Hungry Minds Inc,U.S.; 2nd edition
3. Ranjan Parekh, "Principles of Multimedia", McGraw Hill Education; 2 edition
4. Tay Vaughan, "Multimedia Making It Work"; Mc Graw Hill, Eighth Edition

Suggested list of practical :

1. Image compositing : Remove background and combine images to create a work of art
2. Learn to create images for Print, Web and Video
3. Design a Logo for a company
4. Design a Brochure for given Product and details. Learn about different file formats
5. Design a poster with given information and learn about image compression
6. Edit the sound file and Learn about Effects and Filters of sound.
7. Record your voice and learn about Audio Compression
8. Learn Audio mixing and streaming of audio content
9. Learn about Video editing – Prepare video with rough cut.
10. Prepare video content with title and special effects.
11. Record video content and learn about video compressions.
12. Prepare Video content for vimeo / youtube.

Note :

Practical can be done using GIMP, Inkscape, Scribus, Blender, Audacity, Lightworks / Kdenlive

Course Code : CSD107		Course Title : Introduction to Data Analytics		Semester : VI	
Theory Marks : 75 Practical Marks : 25 Total Marks : 100		Theory Credits : 03 Practical Credits : 01 Total Credits : 04		Theory Lectures : 45 Practical Lectures : 30	
Course prerequisites: Students are expected to have basic knowledge of algorithms and reasonable programming experience and some familiarity with basic linear algebra and basic probability and statistics.					
Course objectives: Become familiar with methods of Data Science and their practical usefulness					
Learning Outcomes : Upon completion of the course students should be able to : • Describe what Data Science is and the skill needed to be a data analyst. • Explain in basic terms what statistical inference means. Identify probability distributions commonly used as foundations for statistical modeling. Fit a model to data. • Use Python to carry out basic statistical modeling and analysis.					
Course content:					
Unit		Topic			
#	Title		Content	Lectures	
I	Introduction to Data Science		Data Science, Big Data, significance of data science, Datafication, Current landscape of perspectives, Skill sets needed , The Data Science Process	4	
II	Statistical Inference		Populations and samples, statistical modelling, probability distributions, fitting a model	6	
III	Exploratory Data Analysis and the Data Science Process		Basic tools(plots, graphs and summary statistics) of Exploratory Data Analysis, Philosophy of EDA	6	
IV	Feature Generation and Feature Selection (Extracting Meaning From Data)		Motivating application: user (customer) retention, Feature Generation (brainstorming, role of domain expertise, and place for imagination) , Feature Selection algorithms	5	
V	Basic Machine Learning Algorithms		Classification and Clustering algorithms, Linear Regression, Logistic regression, k-Nearest Neighbors (k-NN), k-means , Decision Trees, Random Forests	10	
VI	Mining Social-Network Graphs		Social networks as graphs, Clustering of graphs, Direct discovery of communities in graphs, Partitioning of graphs, Neighborhood properties in graphs	6	

VII	Data Visualization		Basic principles, ideas and tools for data visualization, Examples of inspiring (industry) projects,	4	
VIII	Data Science and Ethical Issues		Discussions on privacy, security, ethics, A look back at Data Science, Next-generation data scientists	4	

Text Books:

1. Cathy O’Neil and Rachel Schutt.; “Doing Data Science, Straight Talk From The Frontline”, O’Reilly, 2014.

Reference Books :

1. Jure Leskovek, Anand Rajaraman and Jeffrey Ullman, Mining of Massive Datasets v2.1, Cambridge University Press, 2014 (free online)
2. Kevin P. Murphy, Machine Learning: A Probabilistic Perspective, 2013.
3. Foster Provost and Tom Fawcett, Data Science for Business: What You Need to Know about Data Mining and Data-analytic Thinking, 2013.
4. Trevor Hastie, Robert Tibshirani and Jerome Friedman. Elements of Statistical Learning, Second Edition, 2009 (free online)
5. Avrim Blum, John Hopcroft and Ravindran Kannan. Foundations of Data Science.
6. Mohammed J. Zaki and Wagner Miera Jr, Data Mining and Analysis: Fundamental Concepts and Algorithms, Cambridge University Press, 2014.

Suggested list of practical :

1. Predict the class of a flower based on the available attributes for Iris data set.
2. Dream Housing Finance company deals in all home loans. They have presence across all urban, semi urban and rural areas. Customer first apply for home loan after that company validates the customer eligibility for loan. Company wants to automate the loan eligibility process (real time) based on customer detail provided while filling online application form. These details are Gender, Marital Status, Education, Number of Dependents, Income, Loan Amount, Credit History and others. To automate this process, they have given a problem to identify the customers segments, those are eligible for loan amount so that they can specifically target these customers.
3. Refer to the Wine Quality data set on the following link:
<https://archive.ics.uci.edu/ml/machine-learning-databases/wine-quality/>
 Predict the quality of wine.

Note : All programs to be implemented using Python

Skill Enhancement Courses

Course Code : CSS107		Course Title : Agile Software Development		Semester : V	
Theory Marks : 75 Practical Marks : 25 Total Marks : 100		Theory Credits : 03 Practical Credits : 01 Total Credits : 04		Theory Lectures : 45 Practical Lectures : 30	
Course objectives: To understand what is the software development and it's various stages like analysis, designing, coding, testing, documenting with a stress on Agile Software Development.					
Learning Outcomes : Upon completion of the course students should be able to:: • Explain Evolution and fundamentals of software engineering methods • Apply Agile software development method – Scrum, XP, TDD • Explain and draw UML diagrams • Perform software testing using various quality assurance methods • Perform various software engineering steps using Agile methodologies • Apply refactoring techniques • Explain Source Control Tools					
Course content:					
Unit		Topic			
#	Title		Content	Lectures	
I	Introduction to Agile Software Development		Introduction to Agile Software Development: Understanding how traditional software development works and it's problems; Role of Agile practices in the world of software development and Tools used	5	
II	Agile Project Planning and Management using Scrum		Agile Project Planning And Management using Scrum: Requirement Analysis, Estimation techniques, Iteration planning, Introduction to development practices, user stories, sprint, backlog, velocity, burndown chart, story points	10	
III	Introduction to UML		introduction to UML- class diagram, sequence diagram, use case diagram	5	
IV	Extreme Programming and Test Driven Development		XP & TDD : Test Driven Development & Pair Programming, Introduction to QA Practices: Fail Fast & Automated functional testing, Introduction to Continuous Integration	10	

V	Coding and Testing Practices		Coding and testing practices: Practicing TDD and pair programming as alternative to traditional documentation; Unit testing using Junit, test case, test suite, bad smells & Refactoring.	10	
VI	Introduction to Source Control Tools		Introduction to Source Control tools - versioning, check-in/checkout, commit, branching, merging	5	

Text Books:

1. Chris Sims and Hillary Louise Johnson , “Elements of Scrum”, Dymaxicon, LLC
2. Martin Fowler, “Refactoring”, Addison Wesley; 2nd edition
3. Martin Fowler , “UML Distilled”, Addison Wesley
4. . Kent Beck, Extreme Programming Explained: Embrace Change, Addison Wesley, 2nd Edition

Reference Book:

1. Ken Schwaber, Mike Beedle , “Agile Software Development with Scrum”, Pearson Education

Suggested Practical list :

1. Git
2. User stories, Estimation
3. Burndown charts, Scrum board, Trello, Bugzilla
4. JUnit, Selenium
5. Refactoring
6. Debugging
7. Maven Build
8. Javadoc

Note :

Practical may be done using the tools: Eclipse, Git, Selenium, Bugzilla, Trello

Course Code : CSS108		Course Title : Network Administration		Semester : V	
Theory Marks : 75		Theory Credits : 03		Theory Lectures : 45	
Practical Marks : 25		Practical Credits : 01		Practical Lectures : 30	
Total Marks : 100		Total Credits : 04			
Course objectives: Provide knowledge of networking as well as related equipment and terminologies.					
Learning Outcomes : Upon completion of the course, students should be able to: • Install, configure and manage computer networks. • Administer switching and routing devices. • Install and configure TCP/IP for network and Internet connectivity.					
Course content:					
Unit		Topic			
#	Title	#	Content	Lectures	
I	Introduction to Computer Networks	A	Introduction: Advantages of Networking, Peer-to-Peer and Client/Server Network.	8L	
		B	Network Topologies: Bus, ring, star, mesh, hybrid		
		C	Type of Networks: PAN, LAN, MAN, WAN, Internet and Internet,		
		D	Data Communication modes: Simplex, Half-Duplex and Full-Duplex transmission		
		E	Wired and wireless networking: Ethernet, Wi-Fi, Bluetooth, Mobile Networking,		
II	Wired Media	A	Cabling: Unshielded twisted-pair (UTP), shielded twisted-pair (STP), Fiber Optic and coaxial cable	10L	
		B	Connectors: RJ-45, RJ-11, BNC, Straight Tip (ST);, Subscriber Connector (SC), Local Connector (LC). Understanding color codes of twisted pair cable. 568A and 568B convention. Crimping and punching practices		
		C	Wired testing equipment’s: Multimeter, cable testers, loopback plug		
III	Wireless Media	A	Bluetooth: Features, radio communication, typical purpose	10L	
		B	Near-field communication (NFC): Features, radio communication, typical purpose		
		C	WiFi: Features, radio communication, typical purpose Key types of WiFi networking security – WEP, WPA		
		D	Wireless Standards: 802.11a, 802.11b/g/n		

			Network adapters, Wireless access points, wireless extension points		
		E	Securing a wireless network: Common wireless security threats - Rogue Access points, Denial of Service, Misconfigured Access points, Wireless Phishing, Eavesdroppers Security methods – passwords, SSID, WEP/WPA, MAC address filtering, IP filtering		
IV	Network Components	A	Networking devices: Repeaters, Transceivers, Bridges, Hubs, Switches	08	
		B	Internetworking devices: Routers, gateways, Firewall		
		C	IP addressing: IP versions, IP V4 classes, static and dynamic IP addresses, subnetting and supernetting		
		D	Types of Routing protocols: RIP, IGRP		
V	Network Application and Security	A	Protocols: Simple Mail Transfer Protocol (SMTP), Hyper Text Transfer Protocol (HTTP/HTTPS).	09	
		B	Concept of Dynamic Host Control Protocol (DHCP).		
		C	Domain Name System (DNS):		
		D	Introduction to network security: Digital signature, Public Key Infrastructure, Digital certificates		

Reference Books :

1. Networking, All-in-one desk reference –by Doug Lowe –Wiley India Publication 2009.
2. Wireless & Mobile Networks, by Sunil Kumar S Manvi, Mahaballeshwara S. Kakkasageri, Wiley India Publication 2011.

Suggested Practical list :

1. Identify various Network device like : (a) Switch (Normal and Managed), (b) Router(Normal and wireless), (c) Rack, Patch Panel, i/o box, (d) Access Point etc.
2. Crimping - straight and cross CAT 6 cables. Punching practice in IO Box and patch panel. Fabricate a long cable, Test the fabricated cable using a cable tester
3. Connecting systems to the Switches, Checking Status LED on Switches, Checking Patch Panel
4. Use Ping, Path Ping and Tracert
5. Connect computers with Network with Drop cable and using WiFi configuration.
6. Configure WEP & WAP, configure tethering,
7. Check MAC address, IP Addressing technique (IPv4/IPv6) Subnet Calculation, Static IP Addressing, Dynamic IP Addressing
8. Configure IP & MAC Filters
9. Setup Bluetooth Network, Enable Discovery of Bluetooth Devices, Pairing of Bluetooth devices, Security settings for Bluetooth Devices
10. Use openSSL/JCrypt tool to demonstrate hashing and digital/PKI signatures & certificates

11. Configuring DHCP, Sharing Resource & Internet connection - Sharing Resource and advance sharing setting. Installing Proxy Server. Exposure and using Internet.

Note :

Languages/Tools/equipment's: Cat 6 (or equivalent) cable, Crimping/punching tool, I/O box, Rack, Patch panels, 8/16 port Switch, WIFI enabled/supported devices, JCrypt/openSSL

Course Code : CSS109		Course Title : PHP Programming		Semester : VI	
Theory Marks : 75 Practical Marks : 25 Total Marks : 100		Theory Credits : 03 Practical Credits : 01 Total Credits : 04		Theory Lectures : 45 Practical Lectures : 30	
Course objectives: To give an understanding of web software development using PHP					
Learning Outcomes : Upon completion of the course, students should be able to: • <u>Create websites using PHP/MYSQL.</u>					
Course content:					
Unit		Topic			
#	Title		Content	Lectures	
I	Web Technologies		Introduction to Web technology, Web pages and Browsing, Dynamically generated Web Pages, HTTP-Basic request response paradigm, format and important headers , Session Management using Cookies and hidden fields.	03L	
II	PHP basics		<u>Basic syntax, comments, data types,</u> Type juggling. Variables: Basics, Predefined variables and scope, Constants: define, const, Magic constants, Expressions, Operators, Control structures, Function, Need of Function , declaration and calling of a function , PHP Function with arguments, Default Arguments in Function, Function argument with call by value, call by reference, Scope of Function Global and Local, callbacks.	10L	
III	String Manipulation and Regular Expression:		Creating and accessing String , Searching & Replacing String; Formatting, joining and splitting String , String Related Library functions ; Use and advantage of regular expression over inbuilt function; Use of preg_match(), preg_replace(), preg_split() functions in regular expression.	04L	
IV	PHP Arrays		Indexed, Associative, Multidimensional, Creating index based and Associative array, Accessing array , Looping with Index based array, with associative array using each() and foreach(), Library functions for arrays.	04L	
V	Form Processing		PHP Form Processing, GET and POST form methods, Dealing with multi value fields, Redirecting a form after submission, PHP Superglobals,: scope, \$GLOBALS, \$_SERVER, \$_REQUEST, \$_POST, \$_GET, \$_FILES, \$_COOKIE, \$_SESSION , Form validations, \$_SERVER[“PHP_SELF”],htmlspecialchars() function, PHP Form security, validating form data, sanitising form data .	08L	
VI	PHP		Date and Time functions, include, require,	06L	

	Advanced		require_once, File Upload : Steps involved, \$_FILES; Cookie variables : creation, retrieval, modification, deletion; Session variables: starting sessions, creating, deleting ; Cookies v/s Sessions, usage of superglobals \$_COOKIE and \$_SESSION.	
VII	Database Programming		PHP data objects, Connecting to MySQL, Running Simple Select Statements, Fetch Modes, Getting the Last Insert Id, Running Simple INSERT, UPDATE, or DELETE statements, Running Statements With Parameters, Named Placeholders, Prepared Queries.	06L
VIII	Security		Introduction, General considerations, Filesystem Security, Database Security, SQL Injections, Error Reporting, Hiding PHP.	04L

Reference Books :

1. Steven Holzner, "PHP: The Complete Reference Paperback", McGraw Hill Education (India), 2007.
2. Timothy Boronczyk, Martin E. Psinas, "PHP and MYSQL (Create-Modify-Reuse)", Wiley India Private Limited, 2008.
3. Robin Nixon, "Learning PHP, MySQL, JavaScript, CSS & HTML5", 3rd Edition Paperback, O'reilly, 2014.
4. Luke Welling, Laura Thompson, "PHP and MySQL Web Development", 4th Edition, Addition Paperback, Addison-Wesley Professional, 2008.
5. David Sklar, Adam Trachtenberg, "PHP Cookbook: Solutions & Examples for PHP Programmers", 2014.

Suggested Practical list :

- a) Install and configure Apache, PHP and MySQL
 - b) Session management using cookies, sessions and hidden fields
 - c) Database CRUD operations (using PDO's)
 - d) File Uploading (with file type restrictions).
 - e) Any 15 from the following :
1. Create a PHP page using functions for comparing three integers and print the largest number.
 2. Write a function to calculate the factorial of a number (non-negative integer). The function accept the number as an argument.
 3. WAP to check whether the given number is prime or not.
 4. Create a PHP page which accepts string from user. After submission that page displays the reverse of provided string.
 5. Write a PHP function that checks if a string is all lower case.
 6. Write a PHP script that checks whether a passed string is palindrome or not? (A palindrome is word, phrase, or sequence that reads the same backward as forward, e.g., madam or nurses run)
 7. WAP to sort an array.
 8. Write a PHP script that removes the whitespaces from a string.
Sample string : 'The quick " " brown fox'
Expected Output : Thequick""brownfox
 9. Write a PHP script that finds out the sum of first n odd numbers.
 10. Create a login page having user name and password. On clicking submit, a welcome message should be displayed if the user is already registered (i.e.name is present in the database) otherwise

error message should be displayed.

11. Write a PHP script that checks if a string contains another string.
12. Create a simple 'birthday countdown' script, the script will count the number of days between current day and birth day.
13. Create a script to construct the following pattern, using nested for loop.

```
*  
* *  
* * *  
* * * *  
* * * * *
```

14. Write a simple PHP program to check that emails are valid.
15. WAP to print first n even numbers.
16. \$color = array('white', 'green', 'red')
Write a PHP script which will display the colors in the following way :
Output :
white, green, red,
 - green
 - red
 - white
17. Using switch case and dropdown list display a “Hello” message depending on the language selected in drop down list.
18. Write a PHP program to print Fibonacci series using recursion.
19. Write a PHP script to replace the first 'the' of the following string with 'That'.
Sample : 'the quick brown fox jumps over the lazy dog.'
Expected Result : That quick brown fox jumps over the lazy dog.

FYBCA CBCS Course Codes

SEMESTER- I		
Course Code	Subjects	Course Credits
CAC-101	Problem Solving and Programming Concepts	4
CAC-102	Computer Organization and Architecture	4
CAC-103	Basic Mathematics	4
CAC-104	Problem Solving and Programming Laboratory	2
	GE To be selected by College from approved list	4
ESA-101	Environmental Studies	2
	SEC To be selected by College from approved list	2
	Total	22
SEMESTER – II		
Course Code	Subjects	Credits
CAC-105	Data Structures	4
CAC-106	Operating Systems Concepts	4
CAC-107	Applied Mathematics	4
CAC-108	Data Structures Laboratory	2
	GE To be selected by College from approved list	4
ESA-102	Environmental Studies	2
	SEC To be selected by College from approved list	2
	Total	22

List of Generic Electives (GE)

Course Code	Subjects	Course
-------------	----------	--------

		Credits
CAG-101	Business Accounting	4
CAG-102	Cost Accounting	4
CAG-103	Advertising	4
CAG-104	Human Resource Management	4
CAG-105	Entrepreneurship Development	4
CAG-106	Marketing Fundamentals	4

<u>List of Skill Enhancement Courses (SEC)</u>		
Semester I		
Course Code	Subjects	Course Credits
CAS-101	IT Tools Laboratory	2
CAS-102	Programming in Scratch	2
CAS-103	Digital Photography	2
CAS-104	Open Source Software	2
Semester II		
CAS-105	Operating Systems Laboratory	2
CAS-106	Programming in Python	2
CAS-107	HTML & CSS	2
CAS-108	PHP Programming	2