Semester IV

Course Code	Course Title	Credits	Lectures /Week
USCS401	**Theory of Computation**	**2**	**3**

About the Course:
The course provides a comprehensive insight into theory of computation by understanding grammar, languages and other elements of modern language design. It also helps to develop capabilities to design and develop formulations for computing models and identify its applications in diverse areas.

Course Objectives:
- To give an overview of the theoretical foundations of computer science from the perspective
- of formal languages
- To illustrate finite state machines to solve problems in computing
- To explain the hierarchy of problems arising in the computer sciences.
- To familiarize Regular grammars, context frees grammar.

Learning Outcomes:
After successful completion of this course, students would be able to
- Understand Grammar and Languages
- Learn about Automata theory and its application in Language Design
- Learn about Turing Machines and Pushdown Automata
- Understand Linear Bound Automata and its applications

Unit	Topics	No of Lectures
I	**Automata Theory:** Defining Automaton, Finite Automaton, Transitions and Its properties, Acceptability by Finite Automaton, Nondeterministic Finite State Machines, DFA and NDFA equivalence, Mealy and Moore Machines, Minimizing Automata. **Formal Languages:** Defining Grammar, Derivations, Languages generated by Grammar, Chomsky Classification of Grammar and Languages, Recursive Enumerable Sets, Operations on Languages, Languages and Automata	15
II	**Regular Sets and Regular Grammar:** Regular Grammar, Regular Expressions, Finite automata and Regular Expressions, Pumping Lemma and its Applications, Closure Properties, Regular Sets and Regular Grammar **Context Free Languages:** Context-free Languages, Derivation Tree, Ambiguity of Grammar, CFG simplification, Normal Forms, Pumping Lemma for CFG **Pushdown Automata:** Definitions, Acceptance by PDA, PDA and CFG	15

III	**Linear Bound Automata:** The Linear Bound Automata Model, Linear Bound Automata and Languages. **Turing Machines:** Turing Machine Definition, Representations, Acceptability by Turing Machines, Designing and Description of Turing Machines, Turing Machine Construction, Variants of Turing Machine, **Undecidability:** The Church-Turing thesis, Universal Turing Machine, Halting Problem, Introduction to Unsolvable Problems	**15**	

Textbooks:
1. Theory of Computer Science, K. L. P Mishra, Chandrasekharan, PHI,3rd Edition 2019
2. Introduction to Computer Theory, Daniel Cohen, Wiley, 2nd Edition, 2007
3. Introductory Theory of Computer Science, E.V. Krishnamurthy, Affiliated East-West Press, 2009

Additional References:
1. Theory of Computation, Kavi Mahesh, Wiley India, 2018
2. Elements of The Theory of Computation, Lewis, Papadimitriou, PHI, 2015
3. Introduction to Languages and the Theory of Computation, John E Martin, McGraw-Hill Education, 2010
4. Introduction to Theory of Computation, Michel Sipser, Thomson
5. Introduction to Automata Theory, Languages and Computation, John E. Hopcroft, Pearson Education, 2014

Course Code	Course Title	Credits	Lectures /Week
USCSP401	**Theory of Computation – Practical**	**1**	**3**
1	Write a program for tokenization of given input		
2	Write a program for generating regular expressions for regular grammar		
3	Write a program for generating derivation sequence / language for the given sequence of productions		
4	Design a Program for creating machine that accepts three consecutive one.		
5	Design a Program for creating machine that accepts the string always ending with 101.		
6	Design a program for accepting decimal number divisible by 2.		
7	Design a program for creating a machine which accepts string having equal no. of 1's and 0's.		
8	Design a program for creating a machine which count number of 1's and 0's in a given string.		
9	Design a PDA to accept WCWR where w is any string and WR is reverse of that string and C is a Special symbol.		
10	Design a Turing machine that's accepts the following language an b n c n where n>0		

Course Code	Course Title	Credits	Lectures /Week
USCS402	**Computer Networks**	**2**	**3**

About the Course:
This course introduces computer networks, with a special focus on the Internet architecture and protocols. The course includes topics such as network architectures, addressing, naming, forwarding, routing, communication reliability, the client-server model, web, email and other application layer protocols.

Course Objectives:
- To Understand Basic Concepts of Networking.
- To Understand Working of Network Layer Architecture.
- To Learn Practical Implementation of Basic Routing Algorithms.
- To Learn Different Networking Protocols.

Learning Outcomes:
After successful completion of this course, students would be able to
- Learn basic networking concepts and layered architecture.
- Understand the concepts of networking, which are important for them to be known as a 'networking professionals'.

Unit	Topics	No of Lectures
I	**Introduction:** Networking standards and Administrations, networks, network types – LAN, MAN, WAN. **Network Models:** The OSI model, TCP/IP protocol suite, **Introduction to Physical layer:** Data and signals, periodic analog signals, digital signals, transmission impairment, data rate limits, performance. **Digital transmissions:** Digital-to-digital conversion, analog-to-digital conversion, transmission modes **Analog transmissions:** digital-to-analog conversion, analog-to-analog conversion. **Bandwidth Utilization – Multiplexing and Spectrum spreading:** Multiplexing, Spread Spectrum **Transmission media:** Guided Media, Unguided Media **Switching:** Introduction, Circuit Switched Network, Packet Switching.	**15**
II	**Introduction to Data Link Layer:** Link layer addressing, Data Link Layer Design Issues. **Error detection and correction**: -Block coding, cyclic codes, checksum, forward error correction, error correcting codes, error detecting codes.	**15**

	Data Link Control: DLC services, data link layer protocols, HDLC, Point-to-point protocol.	
	Media Access Control: Random access, controlled access, channelization,	
	Wired LANs – Ethernet: Ethernet Protocol, standard Ethernet, fast Ethernet, gigabit Ethernet, 10 gigabit Ethernet	
	Wired Network: Telephone Network, Cable Network, SONET, ATM	
	Wireless LANs: Introduction, IEEE 802.11 project, Bluetooth, WiMAX, Cellular telephony, Satellite networks.	
	Introduction to Network Layer: Network layer services, packet switching, network layer performance, IPv4 addressing, forwarding of IP packets,	
	Network Layer Protocols : Internet Protocol, ICMPv4, Mobile IP	
III	**Unicast Routing:** Introduction, routing algorithms, unicast routing protocols. **Next generation IP:** IPv6 addressing, IPv6 protocol, ICMPv6 protocol, transition from IPv4 to IPv6. **Introduction to the Transport Layer:** Transport Layer Protocol, User Datagram Protocol, Transmission Control Protocol, SCTP. **Introduction to Application Layer:** Client Server Programming, Iterative Programming. **Standard Client-Server Protocols:** WWW, HTTP, FTP, Electronic Mail, TELNET, Secure Cell, DNS, SNMP **Quality of Service:** Data Flow to improve QoS, Flow control to improve QoS, Integrated service (Intserv), Differentiated Service(Diffserv).	15

Textbooks:
1. Data Communications and Networking, Behrouz A. Forouzan, Fifth Edition, TMH, 2018.
2. Computer Network, Andrew S. Tanenbaum, David J. Wetherall, Fifth Edition, Pearson Education, 2018.

Additional References:
1. Computer Network, Bhushan Trivedi, Oxford University Press, 2016
2. Data and Computer Communication, William Stallings, PHI, 2017

Course Code	Course Title	Credits	Lectures /Week
USCSP402	**Computer Networks – Practical**	1	3
1	Using, linux-terminal or Windows-cmd, execute following networking commands and note the output: ping, traceroute, netstat, arp, ipconfig, Getmac, hostname, NSLookUp, pathping, SystemInfo		
2	Using Packet Tracer, create a basic network of two computers using appropriate network wire. Use Static IP address allocation and show connectivity		
3	Using Packet Tracer, create a basic network of One server and two computers using appropriate network wire. Use Dynamic IP address allocation and show connectivity		
4	Using Packet Tracer, create a basic network of One server and two computers and two mobile / movable devices using appropriate network wire. Show connectivity		
5	Using Packet Tracer, create a network with three routers with RIPv1 and each router associated network will have minimum three PC. Show Connectivity		
6	Using Packet Tracer, create a network with three routers with RIPv2 and each router associated network will have minimum three PC. Show Connectivity		
7	Using Packet Tracer, create a network with three routers with OSPF and each router associated network will have minimum three PC. Show Connectivity		
8	Using Packet Tracer, create a network with three routers with BGP and each router associated network will have minimum three PC. Show Connectivity		
9	Using Packet Tracer, create a wireless network of multiple PCs using appropriate access point.		
10	Using Wireshark, network analyzer, set the filter for ICMP, TCP, HTTP, UDP, FTP and perform respective protocol transactions to show/prove that the network analyzer is working		

Course Code	Course Title	Credits	Lectures /Week
USCS403	**Software Engineering**	**2**	**3**

About the Course:
This course covers a collection of methods which embody an "engineering" approach to the development of software. It discusses the nature of software and software projects, software development models, software process maturity, project planning, management, and estimations. It also underlines the topics on software testing and quality assurance.

Course Objectives:
- To learn and understand the Concepts of Software Engineering
- To learn and understand Software Development Life Cycle
- To apply the project management and analysis principles to software project development.
- To apply the design & testing principles to software project development.

Learning Outcomes:
After successful completion of this course, students would be able to
- Plan a software engineering process life cycle, including the specification, design, implementation, and testing of software systems that meet specification, performance, maintenance and quality requirements
- Analyze and translate a specification into a design, and then realize that design practically, using an appropriate software engineering methodology.
- Know how to develop the code from the design and effectively apply relevant standards and perform testing, and quality management and practice
- Able to use modern engineering tools necessary for software project management, time management and software reuse.

Unit	Topics	No of Lectures
I	**Introduction:** The Nature of Software, Software Engineering, Professional Software Development, Layered Technology, Process framework, CMM, Process Patterns and Assessment Prescriptive Models: Waterfall Model, Incremental, RAD Models Evolutionary Process Models: Prototyping, Spiral and Concurrent Development Model Specialized Models: Component based, Aspect Oriented development, The Unified Process Phases, Agile Development- Agility, Agile Process, Extreme Programming **Requirement Analysis and System Modeling:** Requirements Engineering, Eliciting Requirements, SRS Validation, Components of SRS, Characteristics of SRS, Object-oriented design using the UML - Class diagram, Object diagram, Use case diagram, Sequence diagram, Collaboration diagram, State chart diagram, Activity diagram, Component diagram, Deployment diagram	**15**

II	**System Design:** System/Software Design, Architectural Design, Low-Level Design Coupling and Cohesion, Functional-Oriented Versus Object-Oriented Approach, Design Specifications, Verification for Design, Monitoring and Control for Design **Software Measurement and Metrics:** Process Metrics and Project Metrics, Software Measurement, Object Oriented Metrics, Software Project Estimation, Decomposition Techniques, LOC based, FP based and Use case based estimations, Empirical estimation Models **Software Project Management: Estimation** in Project Planning Process –Software Scope and Feasibility, Resource Estimation, Empirical Estimation Models – COCOMO II, Estimation for Agile Development, The Make/Buy Decision **Project Scheduling** - Basic Principles, Relationship Between People and Effort, Effort Distribution, Time-Line Charts	15
III	**Risk Management** - Risk strategies, Software risks, Risk Identification, projection, RMMM Quality Concepts **Software Quality Assurance** SQA activities, Software reviews, FTR, Software reliability and measures, SQA plan Software Configuration Management, elements of SCM, SCM Process, Change Control Capability Maturity Model **Software Testing :** Verification and Validation, Introduction to Testing, Testing Principles, Testing Objectives, Test Oracles, Levels of Testing, White-Box Testing/Structural Testing, Functional/Black-Box Testing, Test Plan, Test-Case Design	15

Textbooks:
1. Software Engineering, A Practitioner's Approach, Roger S, Pressman, 2019
2. Software Engineering: principles and Practices, Deepak Jain, OXFORD University Press, 2008

Additional References:
1. Software Engineering, Ian Sommerville, Pearson Education, 2017
2. Fundamentals of Software Engineering, Fourth Edition, Rajib Mall, PHI, 2018
3. Software Engineering: Principles and Practices, Hans Van Vliet, John Wiley & Sons, 2010
4. A Concise Introduction to Software Engineering, Pankaj Jalote, Springer

Course Code	Course Title	Credits	Lectures /Week
USCSP403	**Software Engineering – Practical**	1	3

Perform the following exercises for any two projects given in the list of sample projects or any other projects:

1	Write down the problem statement for a suggested system of relevance
2	Perform requirement analysis and develop Software Requirement Specification Sheet (SRS) for suggested system.
3	Draw the function oriented diagram: Data Flow Diagram (DFD) and Structured chart.
4	Draw the user's view analysis for the suggested system: Use case diagram.
5	Draw the structural view diagram for the system: Class diagram, object diagram.
6	Draw the behavioral view diagram : State-chart diagram, Activity diagram
7	Draw the behavioral view diagram for the suggested system: Sequence diagram, Collaboration diagram
8	Draw the implementation and environmental view diagram: Component diagram, Deployment diagram
9	Perform Estimation of effort using FP Estimation
10	Prepare time line chart/Gantt Chart/PERT Chart
11	Develop test cases for unit testing and integration testing
12	Develop test cases for various white box and black box testing

List of sample projects
  a. Student Result Management System
  b. Library management system
  c. Inventory control system
  d. Accounting system
  e. Fast food billing system
  f. Bank loan system
  g. Blood bank system
  h. Railway reservation system
  i. Automatic teller machine
  j. Video library management system
  k. Hotel management system
  l. Hostel management system
  m. Share online trading
  n. Hostel management system
  o. Resource management system
  p. Court case management system

Course Code	Course Title	Credits	Lectures /Week
USCS404	**IoT Technologies**	2	3

About the Course:

The course aims to provide basic understanding of SoC architectures; IoT, different types of IoT platforms and different types of applications that can be built.

Course Objectives:
- Introduce concepts of SoC and IoT
- Introduce various types of IoT platforms
- Interfacing various types of devices using different protocols with IoT
- Understand practical applications of IoT in real life world

Learning Outcomes:

After successful completion of this course, students would be able to
- understand SoC and IoT
- use different types of IoT Platforms and interfaces
- understand and implement an idea of various types of applications built using IoT

Unit	Topics	No of Lectures
I	**Fundamentals of IoT:** Introduction, Definitions & Characteristics of IoT, IoT Architectures, Physical & Logical Design of IoT, Enabling Technologies in IoT, History of IoT, About Things in IoT, The Identifiers in IoT, About the Internet in IoT, IoT frameworks, IoT and M2M. **System on Chip:** What is System on chip? Structure of System on Chip. SoC Elements: FPGA, GPU, APU, Compute Units. **Different types of IoT/SoC Platforms:** Introduction to Raspberry Pi, Arduino & NodeMCU, Introduction to SoC-ARM Architecture, atmega328 architecture	15
II	**Interfacing with IoT Platforms:** Basic hardware components like LED, Button, Camera, 8X8 LED Grid, Motor etc and interfacing them for input/output with IoT devices using PWM, UART, GPIO, I2C, SPI **Using Sensor & Actuators**: Overview of Sensors working, Analog and Digital Sensors, Interfacing of Temperature, Humidity, Motion, Light and Gas Sensor, Level Sensors, Ultrasonic sensors, Interfacing of Actuators, Interfacing of Relay Switch and Servo Motor **IoT and Protocols IoT Security:** HTTP, UPnP, CoAP, MQTT, XMPP, Privacy and Security Issues in IoT.	15

III	**IoT & Web:** Web server for IoT, Sending/Receiving data between web server & IoT device, Cloud for IoT, Node RED, M2M vs IoT Communication Protocols, Basics of WSNs, WSN architecture and types, **IoT Applications:** Modern IoT case studies / applications used in the areas of transportation, agriculture, health care etc **Edge Computing:** Edge computing purpose and definition, Edge computing use cases, Edge computing hardware architectures, Edge platforms, Edge vs Fog Computing, Communication Models - Edge, Fog and M2M.	15

Textbooks:
1. Introduction to IoT Paperback by Sudip Misra , Anandarup Mukherjee , Arijit Roy , Cambridge Press, 2022
2. Jain, Prof. Satish, Singh, Shashi, "Internet of Things and its Applications", 1st Edition, BPB, 2020.
3. Shriram K Vasudevan, Abhishek S Nagarajan, RMD Sundaram, Internet of Things, Wiley, India, 2019
4. IoT and Edge Computing for Architects - Second Edition, by Perry Lea, Publisher: Packt Publishing, 2020

Additional References:
1. Internet of Things by Vinayak Shinde, SYBGEN Learning India Pvt. Ltd, 2020
2. Internet of things, Dr. Kamlesh Lakhwani, Dr. Hemant kumar Gianey, Josef Kofi Wireko, Kamalkant Hiran, BPB Publication, 2020
3. Arduino, Raspberry Pi, NodeMCU Simple projects in easy way by Anbazhagan k and Ambika Parameswari k, 2019.
4. IoT based Projects: Realization with Raspberry Pi, NodeMCU Paperback – February 2020, by Rajesh Singh Anita Gehlot, 2020
5. Mastering the Raspberry Pi, Warren Gay, Apress, 2014

Course Code	Course Title	Credits	Lectures /Week
USCSP404	**IoT Technologies – Practical**	**1**	**3**
1	Preparing Raspberry Pi: Hardware preparation and Installation		
2	Demonstrate Arduino Uno and its pins interfacing with IDE.		
3	GPIO: Light the LED with Python with/without a button using either Uno/Raspberry Pi.		
4	SPI: Camera Connection and capturing Images/Videos using SPI		
5	GPIO: LED Grid Module: Program the 8X8 Grid with Different Formulas		
6	Stepper Motor Control: PWM to manage stepper motor speed using Uno/Raspberry Pi.		
7	Node RED: Connect LED to Internet of Things		
8	Use different types of sensors (LDR, Temperature) with Raspberry Pi/Uno.		
9	Trigger a set of led GPIO on any IoT platform via any related web server		
10	Interface with any sensor and send its value over the internet to the server using any suitable protocol		

Course Code	Course Title	Credits	Lectures /Week
USCS405	**Android Application Development**	**2**	**3**

About the Course:
This course is aimed at creating a skilled IT workforce that is focused on developing Apps for mobile and smart Android-based computing platforms. It familiarises the development of android applications using Kotlin for problems that address real-life needs ranging from intuitive UI to rich multimedia experience.

Course Objectives:
- Kotlin Programming Language for application development
- Creating robust mobile applications on simulators and physical devices
- Creating intuitive, reliable mobile apps using the android services and components
- Handling data local and remote data storage
- Create a seamless user interface that works with different mobile screens

Learning Outcomes:
After successful completion of this course, students would be able to
- Build useful mobile applications using Kotlin language on Android
- Install and configure Android Studio for application development
- Master basic to intermediate concepts of Kotlin required for mobile application development
- Use built-in widgets and components, work with the database to store data
- Master key Android programming concepts and deploy the application on Google Play

Unit	Topics	No of Lectures
I	**Introduction to Kotlin**: Basics of Kotlin, type conversions, comments, Kotlin operators, variables in Kotlin, packages, visibility modifiers, control flow statements, Concept of OOPS in Kotlin, classes in Kotlin, delegation and extension functions, the companion object, **Advanced Concepts in Kotlin:** declaring and calling functions, parameters, and arguments in Kotlin, default argument, variable number of arguments, unit-returning function, explicit return type, lambda expression, coroutines, Collections in Kotlin, Mutable and Immutable Collections, Ranges, type Checks, casting concept, this expression, Null safety, exception handling, annotations **App Development with Android Studio**: Android Architecture, Android Application Framework, Android Virtual Device, Creating and running First Android Application, working with Physical Android Device, Adding Kotlin Files in Android Studio **Basics Of Android**- Application Components: Activities, Intent, and Broadcast Receiver, Services, Fragment, Activity Life Cycle, Content Provider, Widgets, and Notifications	**15**

II	**Designing Android UI**: User Interface (UI), Layout and Its Types, Layout Attribute, working with Views, Android UI Controls, Styles and Themes, Event Handler, setting up themes in Manifest and from the application, dialog in activity, using intents, fragments **Handle Images, Listview And Menu**: ImageView, ImageSwitcher, ListView, Menu, and its types**,** Designing menu in XML, Option menu, Context menu, popup menu, **Screen** Navigation, RecyclerView, Interaction of Views **Data binding in Android-**AdapterView, Spinner, Gallery view, AutotextCompleteView, screen orientation, Design the view dynamically **Implementing Data Persistence**: Data Storage-Shared Preference, Internal And External Storage Storing Data Using SQLite Databases, Content Provider, Firebase Real-Time Data	15
III	**Graphics, Animations, and Integrating Media in Android:** Drawable Class, Animation in Android, MediaPlayer API and in Android, Mediaplayer and AudioManger Class, **Interacting With Camera** and input gestures: Android Camera, Input gestures-multiple touch, swipe, drag, scroll, zoom, Recording **Gathering Location Data:** **Managing Background Tasks:** Broadcaset Receivers, Services, Threads and Process, AsyncTask, JobScheduler, Manage device Awake State **Deploying Android applications on Google Play**-Publishing/Deploy the application, Versioning, signing Application	15

Textbooks:
1. How to Build Android Apps with Kotlin: A hands-on guide to developing, testing, and publishing your first apps with Android, Alex Forrester, Packt Publishing, 2021
2. Android Programming: Crafting UI/UX using Kotlin, SYBGEN Learning, 2020

Additional References:
1. Head First Android Development: A Learner's Guide to Building Android Apps with Kotlin Dawn Griffiths, 3rd Edition, O'Reilly Media, 2021
2. Android Studio 4.2 Development Essentials - Kotlin Edition: Developing Android Apps Using Android Studio 4.2, Kotlin and Android Jetpack, Neil Smyth, Payload Media, 2021
3. Android Programming with Kotlin for Beginners, John Horton, Packt Publishing, 2019
4. Android Development with Kotlin: Enhance your skills for Android development using Kotlin, Marcin Moskala, Packt Publishing

Course Code	Course Title	Credits	Lectures /Week
USCSP405	**Android Application Development – Practical**	1	3
1	i. Write a program using Kotlin to implement control structures and loops. ii. Write a program to implement object-oriented concepts in Kotlin.		
2	i. Create an Android application to design screens using different layouts and UI including Button, Edittext, Textview, Radio Button etc. ii. Write an android application demonstrating response to event/user interaction for 　a. Checkbox 　b. Radio button 　c. Button 　d. Spinner		
3	i. Create an application to create Image Flipper and Image Gallery. On click on the image display the information about the image. ii. Create an application to use Gridview for shopping cart application.		
4	i. Create an Android application to demonstrate implicit and explicit intents ii. Create an application to demonstrate shared preferences		
5	i. Create an Android application to demonstrate the use of Broadcast listeners. ii. Create an Android application to create and use services.		
6	i. Create an Android application to demonstrate XML based animation ii. Create an Android application to display canvas and allow the user to draw on it.		
7	i. Create a media player application in android that plays audio. Implement play, pause, and loop features. ii. Create an Android application to use a camera and capture image/video and display them on the screen.		
8	i. Create an android application to implement Asynctask and threading concepts. ii. Create an Android application to demonstrate the different types of menus. 　a. Pop-up Menu 　b. Context Menu 　c. Option Menu		
9	Create an Android application to record the current location. Based on the current location allow the user to use some useful services/applications		
10	Create a suitable Android application to store and retrieve data in the SQLite database.		
11	Create a suitable Android application to work with Firebase for storing and manipulating data.		

Course Code	Course Title	Credits	Lectures /Week
USCS406	**Advanced Application Development**	**2**	**3**

About the Course:
The course aims at developing scalable, robust, and maintainable web applications using MEAN stack and developing advanced mobile applications using Flutter

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

Learning Outcomes:
After successful completion of this course, students would be able to
- Store the data in NoSQL, document-oriented MongoDB database that brings performance and scalability.
- Use Node.js and Express Framework for building fast, scalable network applications
- Use AngularJS framework that offers declarative, two-way data binding for web applications.
- Integrate the front-end and back-end components of the MEAN stack.
- Develop robust mobile applications using Flutter.

Unit	Topics	No of Lectures
I	**Node.js (N):** Introduction to Node.js. Installing Node.js. The package.json File. The Node.js Event Loop. The I/O Cycle. The Anatomy of a Node.js Module. Creating Node Modules. Exploring the Node.js HTTP Module. Creating an HTTP Webserver with Node.js. Responding to HTTP Requests. Routing in Node.js. Creating a Sample Node.js Application. **MongoDB(M):** Introduction to MongoDB. Installing MongoDB. Using MongoDB Compass. Using Mongo Shell Interface. Connecting to MongoDB. Creating Schemas and Models. Querying Documents Using find(). Inserting Documents Using create(). Updating Documents Using findOneAndUpdate(). Deleting Documents Using findOneAndDelete() & deleteMany()	**15**
II	**Server-Side Development with Express (E):** Introduction to the Express Framework. Installing and Testing Express. Creating a Node.js Express App. Restructuring an Express App. Creating Templates. Using Express Middleware Functions. Creating the List Page. Creating the Details Page. Creating the Edit Page. Creating the Add Page. Deleting Data. REST API Basics. Testing REST APIs. Refactoring APIs. **Understanding Angular.JS(A):** Getting Started with Angular. Creating an Angular Application. Angular Project File Structure. Anatomy of an Angular	**15**

	Component. One-way Data Binding. Two-way Data Binding. Using Nglf Directive. Using NgForOf Directive. Angular Modules. Creating NgModules Using Angular Router. Configuring Templates. Creating Navigations. Working with Template-driven Forms. Working with Reactive Forms. Validating Form Data. Services Dependency Injection (DI). Reading Data from Database. Inserting Data into Database. Updating Data in the Database. Delete Data from Database.		
III	**Understanding Flutter:** Importance of Flutter, Flutter Framework, Android Studio, Flutter SDK, Installing and Configuring Flutter SDK. **Dart Programming**: main() function, Dart Variables, Dart Data Types, Dart Conditional Operators, Control Flow & Loops. Dart Functions - Functions, Function Structure, creating a Function, Function Returning Expression. Object-Oriented Programming (OOP) - Creating a Class, Adding Methods to Classes, Class — Getters and Setters, Class Inheritance, Abstract Class. **Flutter Widgets Fundamentals**: Scaffold Widget, Image Widget, Container Widget, Column and Row Widgets, Icon Widget, Layouts in Flutter, Card Widget, Hot Reload and Hot Restart, Stateful and Stateless Widgets **Navigation and Routing:** Button Widget, App Structure and Navigation, Navigate to a New Screen and Back, Navigate with Named Routes, Send and Return Data among Screens, Animate a Widget across Screens, WebView Widget in Flutter	15	

Textbooks:
1. Node.js, MongoDB and Angular Web Development: The definitive guide to using the MEAN stack to build web applications by Brad Dayley, Brendan Dayley, Caleb Dayley, Pearson, 2018.
2. Beginning Flutter: A Hands On Guide to App Development by Marco L. Napoli, Wrox, 2019

Additional References:
1. Full Stack Javascript Development with Mean - MongoDB, Express, AngularJS, and Node.JS by Adam Bretz, Colin J Ihrig, Shroff/SitePoint, 2015
2. Practical Flutter by Zammetti Frank, Apress, 2019

Course Code	Course Title	Credits	Lectures /Week
USCSP406	**Advanced Application Development – Practical**	**1**	**3**
1	Write a program to implement MongoDB data models		
2	Write a program to implement CRUD operations on MongoDB		
3	Write a program to perform validation of a form using AngularJS		
4	Write a program to create and implement modules and controllers in Angular JS		
5	Write a program to implement Error Handling in Angular JS		
6	Create an application for Customer / Students records using AngularJS		
7	Write a program to create a simple web application using Express, Node JS and Angular JS		
8	Create a simple HTML "Hello World" Project using AngularJS Framework and apply ng-controller, ng-model and expressions		
9	Create an app using Flutter for User Authentication		
10	Create an app using Flutter to implement an Image Gallery		
11	Create an app using Flutter to demonstrate the use of different layouts		
12	Create an app using Flutter to demonstrate navigation in an App		

Course Code	Course Title	Credits	Lectures /Week
USCS4071	**Research Methodology**	2	3

About the Course:
The course aims to understand the basics research, how research problems are defined, research methods are adopted and/or developed, research is undertaken, and how research results are communicated to the peers.

Course Objectives:
- The research methodology course is proposed to assist students in planning and carrying out research projects.
- The students are exposed to the principles, procedures and techniques of implementing research project.
- The course starts with an introduction to research and carries through the various methodologies involved.
- It continues with finding out the literature using technology, basic statistics required for research and finally report writing.

Learning Outcomes:
After successful completion of this course, students would be able to
- Define research, formulate problem and describe the research process and research methods.
- Understand and apply basic research methods including research design, data analysis and interpretation.
- Understand ethical issues in research, write research report, research paper and publish the paper.

Unit	Topics	No of Lectures
I	**Introduction to Research Methodology:** Meaning of Research, Objectives of Research, Motivations in Research, types of Research, Research Approaches, Significance of Research, Research Methods v/s Methodology, Research and Scientific Methods, Research Process, Criteria of Good Research. **Defining the Research Problem:** Concept and need, Identification of Research problem, defining and delimiting Research problem. **Formulating a Research Problem:** Reviewing Literature, formulating a Research Problem, Research Question, Identifying Variables, Constructing Hypothesis **The Research Design**: Meaning, Need for Research Design, Important Concepts, Different Research Designs, Basic Principles of Experimental Designs.	15
II	**Tools for Data Collection**: Collections of Primary Data, Collection of Data through questionnaire and Schedules, other Observation Interview Methods,	15

	Collection of Secondary Data, Selection of appropriate method for data collection, Case Study, Focus Group Discussion, Techniques of developing research tools, viz. Questionnaire and rating scales etc. Reliability and validity of Research tools.

Sampling Design: Steps in Sampling Design, Criteria of Selecting a Sampling Procedure, Characteristics of a Good Sample Design, Different Types of Sample Designs, how to Select a Random Sample. Probability and Non-Probability sampling types and criteria for selection, Developing sampling Frames.

Overview of Hypothesis Testing: What is a Hypothesis? Characteristics of good Hypothesis. Basic Concepts, Procedure for Hypothesis Testing, Flow Diagram for Hypothesis Testing, Tests of Hypotheses, and One sided and two-sided hypothesis, Type – I and Type – II errors, Null Hypothesis-Alternative Hypothesis. | |

| **III** | **Technical Writing:** Writing a Research Proposal, what is a Scientific Paper? Ethics in Scientific Publishing.

Preparing the Text: How to Prepare the Title, how to List the Authors and Addresses, how to Prepare the Abstract, how to Write the Introduction, how to Write the Materials and Methods Section, how to Write the Results, how to Write the Discussion, how to State the Acknowledgments, how to Cite the References.

Preparing the Tables and Figures: How to Design Effective Tables, how to Prepare Effective Graphs, how to Prepare Effective Photographs.

Publishing the Paper: Rights and Permissions, How to Submit the Manuscript, How and When to Use Abbreviations, How to Write a thesis, Outcome of Research, Ethical issues in research | **15** |

Textbooks:
1. Kothari C.R., Research Methodology, New Age International Publication, 2019
2. Research Methodology-A Step-by-Step Guide for Beginners, (4th ed.), Ranjit Kumar, Singapore, Pearson Education, 2018
3. Research Methodology, Vaishali Khairnar, Staredu Solutions India Pvt Ltd, 2020

Additional References:
1. Research Methodology: Methods and Techniques, Dr. R. K. Jain, , Fifth Edition, VEI, 2021
2. Research Methodology, R. Panneerselvam, Second Edition, PHI, 2014
3. Dr. Rachna Jain, Research Methodology, Maximax Publishing House
4. How to Write and Publish a Scientific Paper, Cambridge University Press, Barbara Gastel and Robert A. Day, 2017

Course Code	Course Title	Credits	Lectures /Week
USCS4072	**Management & Entrepreneurship**	2	3

About the Course:
The aim of the course is to develop conceptual understanding of management and administration, and comprehend the environment of making of an entrepreneur. The course focuses on giving students the business management and innovation skills required to succeed in a startup

Course Objectives:
- To understand the idea of management, process and its levels.
- To understand the perception of entrepreneurship, process and its types.
- To understand the concept SSI and steps to start SSI.
- To understand the selection of project, project report, project appraisal, and its feasibility.

Learning Outcomes:
After successful completion of this course, students would be able to
- Understand the meaning of management, functions, administration and its process.
- Understand the foundation of entrepreneurship and its theory, types and its process.
- Identify the steps involved in an entrepreneurial venture (SSI).
- Understand an entrepreneur is converting his business ideas into running concern by selecting the project.

Unit	Topics	No of Lectures
I	**Introduction: Meaning,** Meaning, Characteristics of Management, Nature of Management, Management Functions, Functional Areas of Management, Management and Administration, Role of Management, Levels of Management, Evolution of Management **Planning:** Meaning, Nature, importance, types of planning, types of plans, planning process, decision-making. **Organizing and staffing:** Meaning and Definitions of Organizing, Steps in Organizing, Nature of Organization, Organization Structure, Purpose of Organization, Principles of Organization, Departmentation, Types of Organization, Span of Control, Authority, Power and Responsibility, Delegation of Authority, Centralization and Decentralization, Delegation vs Decentralization, Management by Objectives [MBO], Meaning of Staffing, Nature and Importance of Staffing, Recruitment, Selection.	15
II	**Directing and Controlling:** Meaning and Nature of Direction, Principles of Directing, Leadership and Leadership Styles, Motivation, Communication, Noise and Feedback in Communication, Importance of Communication, Channels of Communication, Types of Communication, Forms of Communication, Coordination, Coordination and Cooperation, Importance	15

	of Coordination, Techniques of Coordination, Managerial Control, Steps in a Control Process, Essentials of a Sound Control System, Control Methods.

Entrepreneurship: Evolution of Concept of Entrepreneur, Concept of Entrepreneur, Characteristics of Entrepreneur, Distinction between Entrepreneur and Manager, Technical Entrepreneur, Charms of Being an Entrepreneur, Functions of an Entrepreneur, Types of Entrepreneurs, Intrapreneurs, Ultrapreneurs, Concept of Entrepreneurship, Evolution of Entrepreneurship, Role of Entrepreneurship in Economic Development, Stages in the Entrepreneurial Process, Barriers to Entrepreneurship | |
| III | **Small Scale Industry:** Meaning and Definition of Small-Scale Industry, Characteristics of SSI, Objectives, Scope, Role of SSI in Economic Development, Advantages of Small-Scale Industries, steps to Start an SSI, Government Policy towards SSI

Preparation of Project: Meaning, Project Classification, Project Identification, Project Report and its significance, Contents of a Project Report, Formulation of Project Report, Planning Commission Guidelines, Network Analysis, Common Mistakes by Entrepreneurs in Project Formulation, Project Appraisal, Identification of Opportunity, Project Feasibility study. | 15 |

Textbooks:
1. Havinal Veerabhadrappa, Management and Entrepreneurship, New Age International Publishers.
2. Kanishka Bedi, Management and Entrepreneurship, Oxford University Press
3. Dr. R. K. Singal, Entrepreneurship Development and Management

Additional References:
1. P. N. Singh, J. C. Saboo, Entrepreneurship Management, 6th Edition, Dr. P. N. Singh Centre for Hrd Publications.
2. Donald L. Sexton & Raymond W. Smilor, The Art and Science of Entrepreneurship, Ballinger, 2022
3. Clifford M.Baumback & Joseph R.Mancuso, Entrepreneurship And Venture Management, Prentice Hall

Evaluation Scheme

I. Internal Evaluation for Theory Courses – 25 Marks

(i) Mid-Term Class Test – 15 Marks

- It should be conducted using any **learning management system** such as **Moodle** (Modular object-oriented dynamic learning environment)
- The test should have **15 MCQ's** which should be solved in a time duration of **30 minutes.**

(ii) Assignment/ Case study/ Presentations – 10 Marks

- Assignment / Case Study Report / Presentation can be uploaded on any **learning management system**.

II. External Examination for Theory Courses – 75 Marks
- Duration: **2.5 Hours**
- Theory question paper pattern:

Question	Based on	Options	Marks
All questions are compulsory.			
Q.1	Unit I	*Any 4 out of 6*	20
Q.2	Unit II	*Any 4 out of 6*	20
Q.3	Unit III	*Any 4 out of 6*	20
Q.4	Unit I, II and III	*Any 5 out of 6*	15

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

III. Practical Examination
- Each core subject carries 50 Marks
  40 marks + 05 marks (journal) + 05 marks (viva)
- Duration: **2 Hours** for each practical course.
- Minimum **80% practical** from each core subjects are required to be completed.
- **Certified Journal is compulsory for appearing at the time of Practical Exam**
- The final submission and evaluation of **journal in electronic form** using a Learning Management System / Platform can be promoted by college.