

# **MCA III and IV Semester Syllabus**

## MCA III Semester

<b>23PCA301C</b>	<b>Software Engineering and Agile Practices</b>	<b>Credits: 03</b>
<b>Hrs/Week: L:T:P:S</b> <b>3:0:0:0</b>		CIEMarks:50
<b>Total Hours/Week:</b> 40Hrs		SEEMarks:50

<b>UNIT-I</b>	<b>10 Hrs.</b>
<p>Software Process &amp; Software Development Methods: Professional Software Development, software engineering ethics, Software process models, Process activities, Coping with change. Agile Software Development: Agile methods, Plan-driven and agile development, Extreme programming, Requirements Engineering: Functional and non-functional requirements, The software requirements document, Requirements specification, Requirements engineering processes, Requirements elicitation and analysis, Requirements validation. Requirements change.</p>	
<b>UNIT-II</b>	<b>10 Hrs.</b>
<p>System Modeling: Context models, Interaction models, Structural models, Behavioural models, Architectural Design: Architectural design decisions, Architectural patterns.</p>	
<b>UNIT-III</b>	<b>10 Hrs.</b>
<p>Software testing: Development testing, Test-driven development, Release testing, User testing. Project management: Risk management, Managing people, Teamwork, Project planning, Software pricing, Plan-driven development, Project scheduling, Agile planning, Estimation techniques. Quality management: Software quality, Software standards, Reviews and inspections, Software measurement and metrics.</p>	
<b>UNIT-IV</b>	<b>10 Hrs.</b>
<p>Scrum Framework Foundation of Scrum, pillars of empiricism, Scrum Values, Identifying a Scrum Team; Scrum Events: Spring planning, Implementation and review, Scrum Artifacts: Creating, Managing and refining product backlog. Scrum in Action: Planning and Estimating with Scrum: Estimation Scale, Bucket method, Envisioning the product journey with a product roadmap; Sprint Journey: Refining the Product Backlog, Tracking progress with a Scrum Board, Defects in Sprint; Facets of Scrum: software development practices for Scrum, Source control model for continuous integration, Continuous delivery and continuous deployment, Leveraging testing methods for Scrum, Applying Scrum to remote teams.</p> <p><b>Reference Books:</b></p> <ol style="list-style-type: none"> <li>1. Ian Sommerville, Software Engineering, 10th Edition, Pearson India Education Services Pvt. Ltd., 2020.</li> <li>2. Roger S. Pressman, Software Engineering-A Practitioners approach, 7th Edition, Tata</li> </ol>	

McGraw Hill.

3. Pankaj Jalote, An Integrated Approach to Software Engineering, Wiley India.
4. Fred heath, The Professional Scrum Master Guide, Packt Publishing, 2021.
5. Stacia Viscardi, The Professional Scrum Master's Handbook, Packt Publishing, 2013
6. Andrew T. Pham and David K. Pham, Business-Driven IT-Wide Agile (Scrum) and Kanban (Lean) Implementation, CRC Press.

**Course Outcomes:**

1. Comprehend software process and process models and decide which process model has to be adopted for the given scenarios.
2. For given case study List functional and nonfunctional requirements.
3. Ability to develop system model and architectural design.
4. Analyze the importance of various software testing methods and the role of project planning and quality management in software development.
5. Understand Scrum frame works and its utility in software development.

<b>23PCA302C</b>	<b>Internet of Things</b>	<b>Credits: 04</b>
<b>Hrs/Week:</b> L:T:P:S <b>3:0:2:0</b>		CIEMarks:50
Total Hours/Week: 40 Hrs+12Hrs		SEEMarks:50

<b>UNIT-I</b>	<b>13 Hrs.</b>
Introduction to Internet of Things: Introduction, characteristics of IoT, IoT Ecosystem, IoT reference model, Transducers, Sensors and Actuators, IoT protocols, Domain Specific IoT. Internet of Things Platform Design Methodology, Introduction to python, logical design Using Python	
<b>UNIT-II</b>	<b>13 Hrs.</b>
Programming with Arduino: Introduction to Arduino, Exploring Arduino Uno Learning Board, Arduino IDE, Understanding the eco system of Arduino, Pinout configuration, Digital input and output, Analog input and output, working with sensors and actuators Fundamentals of Arduino Programming, Arduino serial communication. Communication interfaces (SPI and I2C) wired and wireless communication with Arduino using Bluetooth modules, Examples Modules on Arduino.	
<b>UNIT-III</b>	<b>13 Hrs.</b>
Programming with RaspberryPI: Introduction and Exploring RaspberryPI learning Board, Understanding the eco system of Raspberry Pi3/Pi4, Pinout configuration, RaspberryPI Operating System setup and commands, Programming RaspberryPI with Python: python modules like Rpi.GPIO and gpiozero, Raspberry Pi serial communication, Communication interfaces (SPI and I2C), wired and wireless communication with raspberry Pi, Serial communication from raspberry Pi3 to Arduino and vice versa, Monitoring and controlling between raspberry pi.	
<b>UNIT-IV</b>	<b>13 Hrs.</b>
IoT Application Development: Integrating sensors with IoT Dashboards and micro services. Introduction to Flow based IoT Dashboard: Fundamentals of NodeRED, creating basic dashboard. Introduction to MQTT based IoT Dashboard: setup and configuration of dashboard like Things board. Introduction to hosted IoT dashboard services like Adafruit io or thing board hosted service. IoT alert integration: alert integration in the form of email, tweets or any social media post.	
<b>Reference Books</b>	

1.Srinivasa K.G, Siddesh G.M, Hanumantha Raju R, Internet of Things, Cengage,2017,ISBN:9789386858955

2.ArshdeepBahga, Vijay Madiseti, Internet of Things: A Hands-on Approach, Orient Blackswan Private Ltd, July 1st, 2015, ISBN: 8173719543.

3.Raj Kamal, "Internet of Things: Architecture and Design Principles", 1st Edition, McGraw Hill Education, 2017.

4.Maneesh Rao, Internet of Things with Rasberry Pi3,Packt Publishing April 2018

5.The Official Raspberry Pi Handbook by The Magpi Magazine, 2023

### Course Outcomes

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

1. Understand the fundamentals of electronics and hardware devices required for IoT including deployment levels, network protocols and standards.
2. Comprehend various development boards, sensors, actuators, architecture of Arduino, Raspberry Pi, using IDE
3. Interact with Arduino, Raspberry Pi devices.
4. Design, Setup, Configure and Develop IoT Applications (Dashboards) and integrate several essential micro services

**LABORATORY ASSIGNMENTS  
(Tentative)**

Practice Lab: Fundamentals of Electronics and Introduction to variety of devices and development boards used to develop IoT Applications

Identifying the IoT Kit elements: sensors, actuators and development boards and other accessories Study about the principle of operations, operating conditions, cost, tolerance and durability study.

1.	Assembling and Disassembling IoT Kit using Raspberry Pi/Arduino.
2.	Display college name, student name and good bye message on LCD.
3.	Demonstrate functioning of relay with different time interval.
4.	Demonstrate connection and functioning of LED and Switch.
5.	Display numbers from 0 to 99 on seven segment display board.
6.	Demonstrate functioning of humidity sensor.
7.	Connect and demonstrate light sensor to raspberry pi.
8.	Connect and demonstrate use of ultra sonic sensor using raspberry pi.
9.	Connect and demonstrate use of soil moisture sensor using raspberry pi
10.	Demonstrate use of buzzer program with different time interval.

**Course Outcomes**

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

1. To Introduce the terminology technology and its applications
2. To introduce the concept of machine to machine with necessary protocols
3. To introduce the python scripting language used for IoT devices
4. To introduce RaspberryPI platform that was used widely in IoT applications
5. To implement a various components and sensors services on IoT devices.

<b>23PCA303C</b>	<b>Web 2.0</b>	<b>Credits: 03</b>
<b>Hrs/Week: L:T:P:S</b> <b>3:0:0:0</b>		CIEMarks:50
Total Hours/Week: 40Hrs		SEEMarks:50

<b>UNIT-I</b>	<b>13 Hrs.</b>
<p>Rich Internet Applications With Ajax: Limitations of Classic Web application model, AJAX principles, Technologies behind AJAX, Examples of usage of AJAX; Asynchronous communication and AJAX application model. Ajax with XMLHttpRequest object: Part 1 :Creating Ajax Applications: An example, Analysis of example ajax.html, Creating the JavaScript, Creating and opening the XMLHttpRequest object, Data download, Displaying the fetched data, Connecting to the server,</p> <p>Adding Server-side programming, Sending data to the server using GET and POST. Module</p>	
<b>UNIT-II</b>	<b>13 Hrs.</b>
<p>Ajax with XMLHttpRequest object: Handling multiple XMLHttpRequest objects in the same page, Using two XMLHttpRequest objects, Using an array of XMLHttpRequest objects, Using inner functions, Downloading JavaScript, connecting to Google Suggest, Creating google.php, Downloading from other domains with Ajax, HTML header request and Ajax, Defeating caching, Examples. Working with XML DOM in Ajax :Building XML and working with XML in JavaScript, Getting the document element, Accessing any XML element, Handling whitespace in Firefox, Handling cross-browser whitespace, Accessing XML data directly, Validating XML, Further examples of Rich Internet Applications with Ajax.</p>	
<b>UNIT-III</b>	<b>13 Hrs.</b>
<p>Working with PHP and Ajax :Working with PHP server variables, Getting the data in to array format, Wrapping applications in to a single PHP page, Validating input from the user, Validating integers and text, DOM, Appending new elements to a web page using the DOM and Ajax, Replacing elements using the DOM, Handling timeouts in Ajax, Downloading images with Ajax, Example programs. Introduction to Bootstrap: What Is Bootstrap? Bootstrap File Structure, Basic HTML Template, Global Styles, Default Grid System, Basic Grid HTML, Offsetting Columns, Nesting Columns, Fluid Grid System, Container Layouts, Responsive Design. Typography, Emphasis Classes, Lists, Code, Tables, Optional Table Classes, Table</p>	

Row Classes, Forms, Buttons, Images, Icons.	
<b>UNIT-IV</b>	<b>13 Hrs.</b>
<p>Bootstrap Layout components: Dropdown Menus, Button Groups, Button Groups as Radio Buttons and Checkboxes, Buttons with Dropdowns, Split Button Dropdowns, Dropup Menus, Navigation Elements, Tabular Navigation, Basic Pills Navigation, Stackable Navigation, Dropdowns, Navigation Lists, Tabbable Navigation, Navbar, Forms, Navbar Menu Variations, Breadcrumbs, Pagination, Pager, Labels, Badges, Typographic Elements. Bootstrap Javascript Plugins Programmatic API, Transitions, Modal, Dropdown, Dropdown Usage via JavaScript, Scrollspy, Toggleable Tabs, Tooltips, Popover, Alerts, Buttons, Collapse, Carousel, Typeahead, Affix.</p>	
<b>Text Books</b>	
<ol style="list-style-type: none"> <li>1. Professional AJAX – Nicholas C Zakas et al, Wrox publications, 2008.</li> <li>2. Steven Holzner: Ajax: A Beginner’s Guide, Tata McGraw Hill, 2014.</li> <li>3. Jake Spurlock: "Bootstrap: Responsive Web Development", O'Reilly Media, 2014</li> </ol>	

<b>Course Outcomes</b>
<p><b>After completion of the course student will be able to:</b></p> <ol style="list-style-type: none"> <li>1. Understand the rich internet concepts and applications.</li> <li>2. Analyze the working of development models in web designing.</li> <li>3. Illustrate appropriate component lifecycle techniques using frameworks.</li> <li>4. Evaluate and implement state based systems using data models and data binding.</li> <li>5. Design &amp; Implement effective and responsive web applications</li> </ol>



<b>23PCA108L</b>	<b>OOMD Laboratory</b>	<b>Credits: 2</b>
<b>Hrs/Week: L:T:P:S</b> <b>0:0:3:0</b>		CIEMarks:50
Total Hours/Week: 40 Hrs		SEEMarks:50

1. Design the Library system: Identify the use cases of the system. (Suggestive use cases: borrow book, return books, read newspapers, reference, and digital library). Develop the use case diagram, Packages and documentation for the same. Preferable use of uses & Extends relationships expected.
2. Design the Examination system: Identify the use cases. (Suggestive use cases:– Form filling, Get Hall Ticket, Write exam, get result ,Verify Hall Ticket) Develop the use case diagram, Packages and documentation for the same. Preferable use of uses & Extends relationships expected.
3. Analyze and design the system for ATM Transaction: Identify the use cases. (Suggestive use cases : Transaction, Approval process, Invalid PIN, Deposit Amount, Deposit savings, Deposit checking, withdraw Amount, withdraw checking, saving, withdraw saving denied, checking Transaction History, saving Transaction History). Develop the use case diagram, Packages and documentation for the same. Draw the essential class diagrams.
4. Analyze and design the system for voting system (The actors are presiding officer, 1<sup>st</sup> polling officer, 2<sup>nd</sup> polling officer,Election officer, voter, candidate; Processes : Voting, counting, and announcement of results). Develop the use case diagram, Packages and documentation for the same. Draw the essential sequence diagrams and state chart diagrams.
5. Analyze and design the system for Results section of autonomy, mainly responsible for CGPA, SGPA Calculation, Grade card generation. Develop the class diagrams and packages.
6. Analyze and design the system for Employee reference. (The Process HR Manager contacts Employees of his company and HR manager of other company to publicise about the vacancy. The person who has referred the right candidate, will be given bonus. Interview, Short listing, selection list announcement, Bonus for referred employees are all parts of the process.). Develop the use case diagram , sequence diagrams and state chart diagrams.
7. Analyze and design the system for autonomous education system. (Classes: students Teacher, courses, subjects, core, Electives, Labs). Develop the class diagrams, sequence diagrams and packages.
8. Analyze and design the system for Railway reservation, which includes the details of Boarding

- a. station, classes of reservation, fare, Time table, concessions, No. of Seats, Seat No, State of reservation ( confirmed, waiting, RAC). Develop the class diagrams, packages, component diagrams and deployment diagrams.
9. Analyze and design the system for Income Tax assessment. Develop the class diagrams, collaboration diagrams, state chart diagrams.
10. Analyze and design the system for Online shopping system. Develop the class diagrams, collaboration diagrams, state chart diagrams and packages.
  - a. a.Check for the product.b.Place order.
  - b. c.Track the order.
  - d.Cancel the order.
  - e.Feedback

<b>Professional Elective-III</b>		<b>Professional Elective-IV</b>	
<b>Subject Code</b>	<b>Subject</b>	<b>Subject Code</b>	<b>Subject</b>
23PCA308E	Big Data Analytics	23PCA312E	Introduction to Machine Learning
23PCA309E	Cyber Security	23PCA313E	Block Chain Technologies
23PCA310E	Flutter Framework	23PCA314E	Android Programming Concepts
23PCA311E	Software Testing	23PCA315E	Software Project Management

<b>23PCA308E</b>	<b>Big Data Analytics</b>	<b>Credits: 03</b>
<b>Hrs/Week: L:T:P:S</b> <b>3:0:0:0</b>		CIEMarks:50
Total Hours/Week: 40 Hrs		SEEMarks:50

<b>UNIT-I</b>	<b>10 Hrs.</b>
<p>Types of digital data: Classification of Digital Data, Structured Data, Semi-Structured Data, Unstructured Data. Introduction to Big Data: Characteristics, Evolution, Definitions and Challenges of big data, other characteristics of data which are not definitional traits of big data, Why big data? Are we just an information consumer or do we also produce information? Traditional Business Intelligence (BI) versus Big data, A typical Data Warehouse environment, A typical Hadoop environment, What is changing in the realms of big data? Big data analytics Where do we begin? What is big data analytics? What big data analytics isn't? Why this sudden hype around big data analytics? Classification of analytics, Top challenges facing big data, why is big data analytics important? Greatest challenges that prevent businesses from capitalizing on big data, what kind of technologies are we looking towards to help meet the challenges posed by big data? Data science, Data Scientist, Terminologies used in big data environment, BASE, Few top Analytics tools</p>	
<b>UNIT-II</b>	<b>10 Hrs.</b>
<p>The big data technology landscape: NoSQL, Where is it used? What is it? Types of NoSQL databases, Why NoSQL? Advantages of NoSQL, What we miss with NoSQL? NoSQL Vendors, SQL Versus NoSQL , NewSQL, Comparison of SQL, NoSQL and NewSQL, Hadoop: Features of Hadoop, Key advantages of Hadoop, Versions of Hadoop, Overview of Hadoop Ecosystems, Hadoop Versus, SQL, Integrated Hadoop systems offered by leading market vendors, Cloud based Hadoop solutions. Hadoop: Introducing Hadoop, Why not RDBMS, Distributed Computing Challenges, History of Hadoop, Hadoop Overview, Hadoop Components, High Level Architecture of Hadoop, Hadoop Distributed File System(HDFS), HDFS Architecture, Daemons Related to HDFS, Working with HDFS Command, Special Features of Hadoop, Processing Data With Hadoop, Introduction, How Map Reduce Works? Map Reduce Example, Word Count Example using Java. Managing Resources and Applications with YARN, Introduction, Limitation of Hadoop 1.0, Hadoop 2: HDFS, Hadoop 2: YARN, Interacting with Hadoop EcoSystem, Hive,Pig, HBASE, Sqoop, Business Intelligence on Hadoop.</p>	

UNIT-III	10 Hrs.
<p>NoSQL - MongoDB: What is MongoDB? Why MongoDB? Using JSON, Creating or generating a unique key, Support for dynamic queries, Storing binary data, Replication, Sharding, Updating information in-place, Terms used in RDBMS and MongoDB, Data types in MongoDB, MongoDB - CRUD (Insert(), Update(), Save(), Remove(), find()), MongoDB- Arrays, Java Scripts, Cursors, Map Reduce Programming, Aggregations. NoSQL - Cassandra: What is Cassandra? Why Cassandra? Peer to peer network, Gossip and Failure detection, Anti-Entropy &amp; Read Repair, Writes in Cassandra, Hinted handoffs, Tunable consistency, Cassandra- CQLSH - CRUD, Counter, List, Set, Map, Tracing.</p>	
UNIT-IV	10 Hrs.
<p>Hadoop Hive: Introduction to Hive - The Problem, Solution - Hive Use Case, Data Growth, Schema Flexibility and Evolution, Extensibility, What is Hive, History of Hive and Recent Releases of Hive, Hive Features, Hive Integration and Work Flow, Hive Data Units, Hive Architecture, Hive Primitive Data Types and Collection Types, Hive File Formats, Hive Query Language - Statements, DDL , DML, Hive Partitions, Bucketing, Views, Sub Query, Joins, Hive User Defined Function, Aggregations in Hive, Group by and Having, Serialization and Deserialization, Hive Analytic. Functions. Hadoop - Pig: Introducing Pig, History and Anatomy of Pig, Pig on Hadoop, Pig Features, Pig Philosophy, Word count example using Pig, Use Case for Pig, Pig Primitive Data Types, Collection Types and NULL, Pig Latin Overview, Pig Latin Grammar - Comments, Keywords, Identifiers, Case sensitivity in Pig, Common Operators in Pig, Pig Statements, LOAD, STORE, DUMP, Interactive Shell -GRUNT, FILTER, SORT, GROUP BY, ORDER BY, JOIN, LIMIT, Pig Latin Script, Local Mode, Map Reduce Mode, Running Pig Script, Working with Field, Tuple, Bag, User Defined Function, Parameters in Pig.</p>	
Reference Books	
<ol style="list-style-type: none"> <li>1. Seema Acharya, Subhashini Chellappan, Big Data and Analytics, Wiley Publications, 2nd Edition, 2015, ISBN:978-81-265-5478-2.</li> <li>2. Raj Kamal, Preethi Saxena, Big Data Analytics, Introduction to Hadoop, Spark and Machine Learning, McGraw hill Education.</li> <li>3. Cindi Howson, Successful Business Intelligence, McGraw-Hill Publications, E-ISSN:0-07-149851-6.</li> <li>4. Frank J Ohlhorst, “Big Data Analytics: Turning Big Data into Big Money”, Wiley and SAS Business Series, 2012.</li> <li>5. Michael Berthold, David J. Hand, Intelligent Data Analysis, Springer, 2007.</li> </ol>	

6. Paul Zikopoulos, Dirk deRoos, Krishnan Parasuraman, Thomas Deutsch , James Giles, David Corrigan, “Harness the Power of Big data – The big data platform”, McGraw Hill, 2012.

#### **Course Outcomes:**

##### **After completion of the course student will be able to:**

1. Differentiation of digital data and to define business intelligence, big data and analytics.
2. Apply different techniques for big data analytics.
3. Comprehend HDFS architecture and programming environment.
4. Device NoSQL statement to process unstructured data.
5. Understanding programming in Hive and Pig technologies.

<b>23PCA309E</b>	<b>Cyber Security</b>	<b>Credits: 03</b>
<b>Hrs/Week: L:T:P:S</b> <b>3:0:0:0</b>		CIEMarks:50
<b>Total Hours/Week:</b> <b>40 Hrs</b>		SEEMarks:50

<b>UNIT-I</b>	<b>10 Hrs.</b>
---------------	----------------

**Introduction to Cybercrime:** Definition and Origins of the Word, Cybercrime and Information Security, Who are Cybercriminals? Classifications of Cybercrimes, An Indian Perspective, hacking and Indian Laws., Global Perspectives.

<b>UNIT-II</b>	<b>10 Hrs.</b>
----------------	----------------

**How Criminals Plan Them:** Introduction, How criminals plan the attacks, Social Engineering, Cyber Stalking, Cybercafe & cybercrimes. **Botnets:** The fuel for cybercrime, Attack Vector. Tools and Methods used in Cybercrime: Introduction, Proxy Servers and Anonymizers, Phishing, Password Cracking, Key Loggers and Spywares

<b>UNIT-III</b>	<b>10 Hrs.</b>
-----------------	----------------

**Different Forms of attacks in Cybercrime:** Virus and Worms, Trojan Horses and Backdoors, Steganography, DoS and DDoS Attackers, Attacks on Wireless networks. Phishing and Identity Theft: Introduction, methods of phishing, phishing, phishing techniques, spear phishing, types of phishing scams, phishing toolkits and spy phishing, counter measures, Identity Theft

<b>UNIT-IV</b>	<b>10 Hrs.</b>
----------------	----------------

**Understanding Computer Forensics:** Introduction, Historical Background of Cyber forensics, Digital Forensics Science, Need for Computer Forensics, Cyber Forensics and Digital Evidence, Digital Forensic Life cycle, Chain of Custody Concepts, network forensics.

<b>Reference Books</b>
------------------------

1. Sunit Belapure and Nina Godbole, "Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives", Wiley India Pvt Ltd, 2011, First Edition.
2. Rajkumar Singh Rathore, Mayank Bhushan, "Fundamentals of Cyber Security", BPB; 2017, First Edition.
3. Anand Shinde, "Introduction to Cyber Security", 2020, Notion Press, First Edition.
4. Nilakshi Jain and Dhananjay R. Kalbande, "Cyber Security and Cyber Laws", Wiley India Pvt Ltd., 2020.

## Course Outcomes

### After completion of the course student will be able to

1. Understand cybercrime terminologies and laws.
2. Illustrate tools and methods used on Cybercrime.
3. Describe the different forms of attacks, Phishing and Identity Theft.
4. Comprehend cyber offences and Botnets.
5. Justify the need of computer forensics.



<b>23PCA310E</b>	<b>Introduction to Flutter</b>	<b>Credits: 03</b>
<b>Hrs/Week: L:T:P:S</b> <b>3:0:0:0</b>		CIEMarks:50
Total Hours/Week: 40 Hrs		SEEMarks:50

<b>UNIT-I</b>	<b>10 Hrs.</b>
<b>Features of Flutter-</b> Advantages of Flutter- Disadvantages of Flutter. Flutter Installation in Windows, Creating Simple Application in Android Studio - Architecture of Flutter Applications.	
<b>UNIT-II</b>	<b>10 Hrs.</b>
<b>Flutter Basics:</b> Widgets- Gestures- Concept of State- Layers- Introduction to Dart Programming- Variables and Data types- Decision Making and Loops. Functions- Object Oriented Programming. Introduction to Widgets- Widget Build Visualization.	
<b>UNIT-III</b>	<b>10 Hrs.</b>
<b>Introduction to Layouts:</b> Type of Layout Widgets- Single Child Widgets- Multiple Child Widgets- Advanced Layout Application-Introduction to Gestures- Statement Management in Flutter. Ephemeral State Management-Application State - scoped model- Navigation and Routing.	
<b>UNIT-IV</b>	<b>10 Hrs.</b>
<b>Animation on Flutter:</b> Introduction to Animation Based Classes-Work flow of the Flutter Animation- Working Application- Android Specific Code on Flutter- Introduction to Package- Types of PackagesUsing a Dart Package- Develop a Flutter Plugin Package- Accessing Rest API- Basic Concepts- Accessing Product service API. <b>Database Concepts:</b> SQLite- Cloud Fire store- Internalization on Flutter- Using intl Package-Testing on FlutterTypes of Testing- Widget Testing- Steps Involved- Working Example-Deployment- Android Application- IOS Application- Development Tools- Widget Sets- Flutter Development with Visual Studio Code- Dart DevTools- Flutter SDK	
<b>Reference Books</b>	
<ol style="list-style-type: none"> <li>1. Richard Rose , “Dart and Flutter”, Oreilly.</li> <li>2. Reto Meier, “ Professional Android 4 Application Development “ , Wrox Publications 2012.</li> </ol>	
<b>Course Outcomes</b>	
<b>After completion of the course student will be able to:</b>	
<ol style="list-style-type: none"> <li>1. Understand concepts of Flutter architecture.</li> <li>2. Build simple Flutter application using simple widgets and layouts</li> <li>3. Build Animation on Flutter</li> <li>4. Develop Flutter applications using Dart packages.</li> <li>5. Construct Flutter application using database.</li> </ol>	

<b>23PCA311E</b>	<b>Software Testing</b>	<b>Credits: 03</b>
<b>Hrs/Week: L:T:P:S</b> <b>3:0:0:0</b>		CIEMarks:50
Total Hours/Week: 40 Hrs		SEEMarks:50

<b>UNIT-I</b>	<b>10 Hrs.</b>
<p><b>BASICS OF SOFTWARE TESTING:</b> Human Errors and Testing; Software Quality; Requirements, Behavior and Correctness; Correctness versus Reliability; Testing and Debugging; Test Metrics. Testing and Verification; Defect Management; Execution History; Test-generation Strategies, Static Testing. Types of Testing.</p>	
<b>UNIT-II</b>	<b>10 Hrs.</b>
<p><b>TEST GENERATION FROM REQUIREMENTS:</b> Introduction; The Test-Selection Problem; Equivalence Partitioning; Boundary Value Analysis; Category-Partition Method. Cause-Effect Graphing, Test generation from predicates.</p>	
<b>UNIT-III</b>	<b>10 Hrs.</b>
<p><b>STRUCTURAL TESTING:</b> Overview; Statement testing; Branch testing; Condition testing, Path testing; Procedure call testing; Comparing structural testing criteria; The infeasibility problem.</p> <p><b>DEPENDENCE, DATA FLOW MODELS, AND DATA FLOW TESTING:</b> Definition-Use pairs; Data flow analysis; Classic analyses; From execution to conservative flow analysis; Data flow analysis with arrays and pointers; Inter- procedural analysis; Overview of data flow testing; Definition-Use associations; Data flow testing criteria; Data flow coverage with complex structures; The infeasibility problem.</p>	
<b>UNIT-IV</b>	<b>10 Hrs.</b>
<p><b>TEST CASE SELECTION AND ADEQUACY:</b> Overview; Test specification and cases; Adequacy criteria; Comparing criteria; <b>PROCESS:</b> Integration and component-based software testing: Overview; Integration testing strategies; Testing components and assemblies. System, Acceptance and Regression Testing: Overview; System testing; Acceptance testing; Usability; Regression testing; Regression test selection techniques; Test case prioritization and selective execution.</p>	
<b>Reference Books</b>	
<ol style="list-style-type: none"> <li>1. Foundations of Software Testing Aditya P Mathur, Pearson Education, 2008. (chapter 1, 2)</li> <li>2. Software Testing and Analysis Process Principles and Techniques Mauro Pezze, Michal Young, Wiley India,2008.</li> <li>3. Software Testing Principles and Practices Srinivasan Desikan, Gopaldaswamy Ramesh, 2ndEdition, Pearson, 2007.</li> </ol>	

## Course Outcomes

### After completion of the course student will be able to

1. Identify errors, faults, failures, test process, correctness, reliability, oracles.
2. Comprehend the various testing models, defect management, quality attributes and test generation strategies.
3. To generate requirement-based test cases (black box testing) and structural testing (white box testing).
4. Design the test cases to check data flow for the given program.
5. Develop test cases by using various adequacy criteria. 6. Apply various testing processes to test a given small application.

23PCA312E	<b>Introduction to Machine Learning</b>	<b>Credits: 03</b>
<b>Hrs/Week: L:T:P:S</b> <b>3:2:0:0</b>		CIEMarks:50
Total Hours/Week: 40 Hrs		SEEMarks:50

<b>UNIT-I</b>	<b>10 Hrs.</b>
---------------	----------------

**Basics of Machine Learning;** Introduction to Artificial Intelligence and Machine Learning, Types of Machine Learning and its comparisons, Applications of Machine Learning, Issues in Machine Learning

<b>UNIT-II</b>	<b>10 Hrs.</b>
----------------	----------------

**Preparing to Model:** Introduction, Machine Learning Activities, Types of Data in Machine Learning, Exploring structure of data, Data Pre-processing (Dimension Reduction and Feature subset selection), Model Selection.

<b>UNIT-III</b>	<b>10 Hrs.</b>
-----------------	----------------

**Supervised Learning:** Introduction, Classification (Introduction, classification model, learning steps, Common classification algorithm), Regression (Linear Regression, Multivariable Regression, Logistic Regression).

<b>UNIT-IV</b>	<b>10 Hrs.</b>
----------------	----------------

**Unsupervised Learning:** Introduction and its applications, Techniques in Unsupervised Learning (Clustering, K-Means). **Neural Network:** Introduction, Architecture of Artificial Neural Network.

<b>Reference Books</b>
------------------------

1. Saikat Dutt, Machine Learning, Pearson, 2019.
2. Tom Mitchell, Machine Learning (First Edition), McGraw- Hill, 1997
3. Anuradha Srinivasa Raghavan, Vincy JOdeph, Machine Learning, Wiley, 2019.
4. Ian Goodfellow, Yoshua Bengio, Aaron Courville, Deep Learning, MIT Press. 2016.

<b>Course Outcomes</b>
------------------------

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

1. Recognize major programming languages.
2. Identify potential applications of machine learning in practice.
3. Select the suitable machine learning tasks for given application.
4. Implement feature extraction and selection to represent data as features to serve as input to machine learning models.

<b>23PCA313E</b>	<b>Block Chain Technologies</b>	<b>Credits: 03</b>
<b>Hrs/Week: L:T:P:S</b> <b>3:0:0:0</b>		CIEMarks:50
Total Hours/Week: 40Hrs		SEEMarks:50

<b>UNIT-I</b>	<b>10 Hrs.</b>
<b>Introduction to Blockchain:</b> Introduction to Blockchain, How Blockchain works, Blockchain vs Bitcoin, Practical applications, public and private key basics, pros and cons of Blockchain, Myths about Bitcoin.	
<b>UNIT-II</b>	<b>10 Hrs.</b>
<b>Blockchain:</b> Architecture, versions, variants, use cases, Life use cases of blockchain, Blockchain vs shared Database, Introduction to cryptocurrencies, Types, Applications.	
<b>UNIT-III</b>	<b>10 Hrs.</b>
<b>Concept of Double Spending:</b> Concept of Double Spending, Hashing, Mining, Proof of work. Introduction to Merkel tree, Privacy, payment verification , Resolving Conflicts , Creation of Blocks.	
<b>UNIT-IV</b>	<b>10 Hrs.</b>
<b>Introduction to Bitcoin and Ethereum:</b> Introduction to Bitcoin, key concepts of Bitcoin, Merits and De Merits Fork and Segwits, Sending and Receiving bitcoins, choosing bitcoin wallet, Converting Bitcoins to Fiat Currency. Introduction to Ethereum, Advantages and Disadvantages, Ethereum vs Bitcoin, Introduction to Smart contracts, usage, application, working principle, Law and Regulations. Case Study.	
<b>Reference Books</b>	
<ol style="list-style-type: none"> <li>1. Beginning Blockchain: A Beginner's Guide to Building Blockchain Solutions by Arshdeep Bikramaditya Signal, GautamDhameja (PriyansuSekhar Panda., APress.</li> <li>2. Blockchain Applications: A Hands-On Approach by Bahga, Vijay Madiseti.</li> </ol>	
<b>Course Outcomes</b>	
<b>After completion of the course student will be able to:</b> <ol style="list-style-type: none"> <li>1. Demonstrate the basics of Block chain concepts using modern tools/technologies.</li> <li>2. Analyze the role of block chain applications in different domains including cybersecurity.</li> <li>3. Evaluate the usage of Block chain implementation/features for the given problem.</li> <li>4. Exemplify the usage of bitcoins and its impact on the economy.</li> <li>5. Analyze the application of specific block chain architecture for a given problem.</li> </ol>	

<b>23PCA314E</b>	<b>Android Programming Concepts</b>	<b>Credits: 03</b>
<b>Hrs/Week: L:T:P:S</b> <b>3:0:0:0</b>		CIEMarks:50
Total Hours/Week: 40 Hrs		SEEMarks:50

<b>UNIT-I</b>	<b>10 Hrs.</b>
<b>Overview</b> - Basics of Android, MVC , GUI, Components, Events, Layout Manager, Multiple Activities, Passing Data between Activities, Transitions, Persistent.	
<b>UNIT-II</b>	<b>10 Hrs.</b>
Creating Menus, SQLite, Managing Device Orientation, Touches and Swipe, Graphics, Animations, Sound and Gaming	
<b>UNIT-III</b>	<b>10 Hrs.</b>
Fragments, Using Libraries and their APIs, Using GPS and Location Services, Using Another App within the App	
<b>UNIT-IV</b>	<b>10 Hrs.</b>
XML and Content Apps, Android Widget, In-App Advertising, Security and Encryption	
<b>Reference Books</b>	
Herve Franceschi, "Android Application Development ", Jones and Bartlett Learning. Trish Cornez and Richard Cornez, "Android Programming Concepts", Jones and Bartlett Learning.	
<b>Course Outcomes</b>	
<b>After completion of the course student will be able to:</b>	
<ol style="list-style-type: none"> <li>1. Demonstrate the Understanding of fundamental of Android Programming. (Understand)</li> <li>2. Build their ability to develop software with reasonable complexity on mobile platform. (Apply)</li> <li>3. Discover the life cycles of Activities, Applications, intents and fragments. (Evaluate)</li> <li>4. Design the Android apps by using Java Concepts. (Create)</li> </ol>	

<b>23PCA315E</b>	<b>Software Project Management</b>	<b>Credits: 03</b>
<b>Hrs/Week: L:T:P:S</b> <b>3:0:0:0</b>		CIEMarks:50
Total Hours/Week: 40 Hrs		SEEMarks:50

<b>UNIT-I</b>	<b>10 Hrs.</b>
<p><b>INTRODUCTION TO SOFTWARE PROJECT MANAGEMENT:</b> The importance of Software Project Management, Project Definition, Software Project versus Other Types of Project, Contract Management and Technical Project Management, Activities Covered By Software Project Management, Plans, Methods, and Methodologies, Stakeholders, Setting objectives, Management Control, Overview of Project Planning – Stepwise Project Planning.</p>	
<b>UNIT-II</b>	<b>10 Hrs.</b>
<p><b>PROJECT EVALUATION:</b> Project Portfolio Management, Evaluation of Individual Projects: Technical Assessment, Strategic Assessment, Cost Benefit Analysis, Cash Flow Forecasting, Cost Benefit Evaluation Techniques, Risk Evaluation. Programme Management, Managing the Allocation of Resources within Programmes, Strategic Programme Management, Creating a Programme, Aids to Programme Management and Benefits Management.</p>	
<b>UNIT-III</b>	<b>10 Hrs.</b>
<p><b>ACTIVITY PLANNING AND RISK MANAGEMENT:</b> Objectives, Project Schedules, Projects and Activities, Sequencing and Scheduling Activities, Network Planning Models, Formulating a Network Model, Forward Pass, Backward Pass, Identifying the critical path, Activity Float, Shortening the Project Duration, Activity on Arrow Networks, Risk Management – Introduction, categories of Risk, Risk Management Approaches, Risk identification, Risk Assessment, Risk Planning, Risk Management, Evaluating Risks to the schedule, Boehm’s Top 10 Risks and Counter Measures.</p>	
<b>UNIT-IV</b>	<b>10 Hrs.</b>
<p><b>MONITORING AND MANAGING CONTRACTS:</b> Introduction, Creating Framework, Collecting the Data, Review, Visualizing Progress, Cost Monitoring. <b>Managing Contracts</b> – Introduction – Types of Contract – Stages in Contract Placement – Typical terms of a Contract – Contract Management – Acceptance. <b>MANAGING PEOPLE:</b> Introduction – Understanding Behavior – Selecting The Right Person For The Job – Instruction in the Best Methods, Motivation, The Oldham-Hackman Job Characteristic Model, Stress, Stress Management, Health and Safety, Some ethical and Professional Concerns.</p>	
22	

## Reference Books

1. Bob Hughes, Mike Cotterell, and Rajib Mall: Software Project Management – Sixth Edition  
Tata McGraw Hill, New Delhi, 2006.
2. Royce, “Software Project Management”, Pearson Education, 1999.
3. Jalote, “Software Project Management in Practice”, Pearson Education, 2002.
4. Gopaldaswamy Ramesh, “Managing Global Software Projects” – McGraw Hill Education (India)  
Fourteenth Reprint 2

## Course Outcomes

### After completion of the course student will be able to

1. Analyze the success of a software project in a real world environment.
2. Evaluate appropriateness of projects against Strategic, Technical, and Economic criteria.
3. Identify the risk factors and their effects in a software projects.
4. Design various charts and reports for visualizing project progress status.
5. Use the right human resources and their role in the software project management.