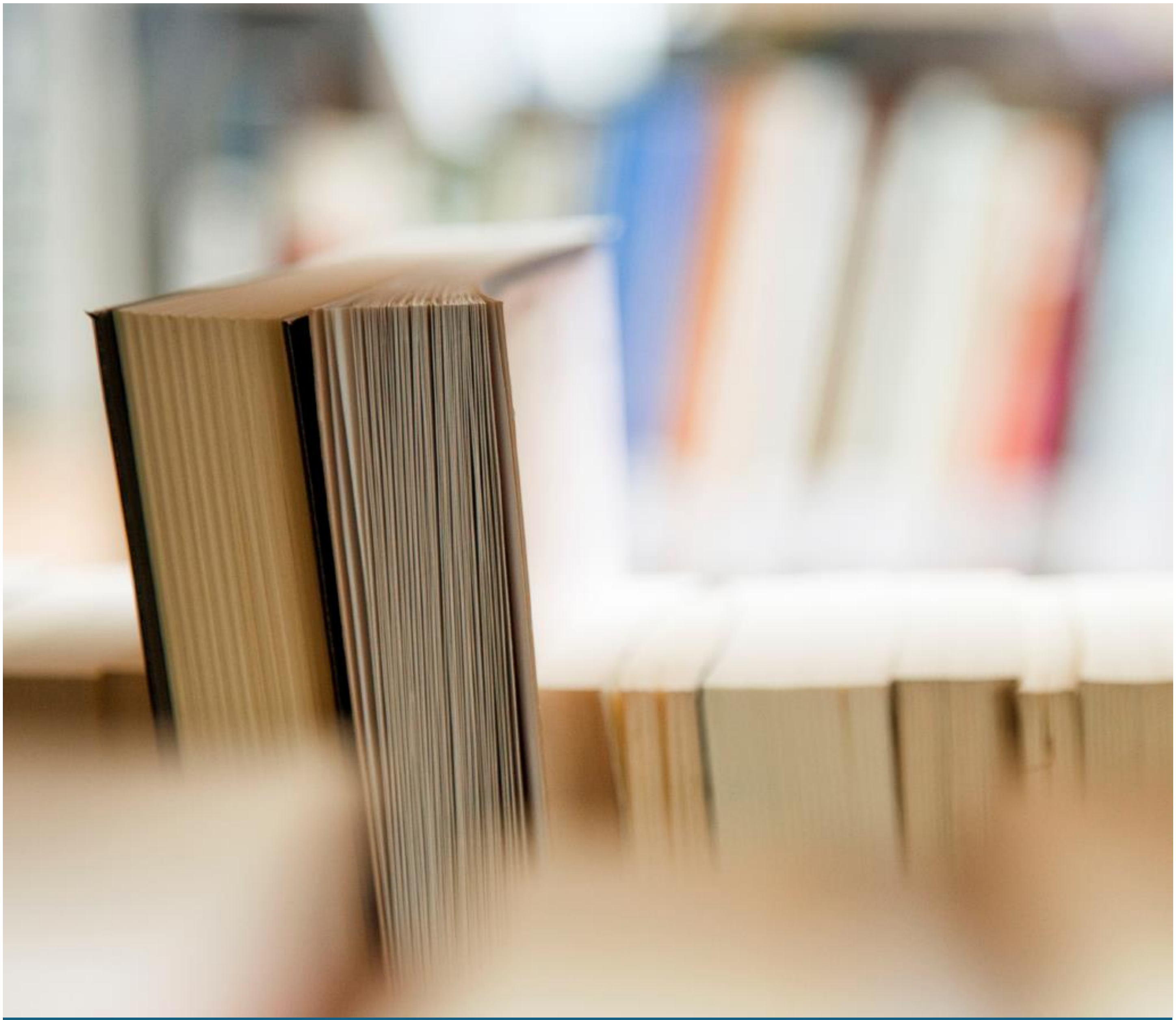

EDUPASS Academy
Value Added Programmes

Computer Science & Applications Syllabus 2024 – 25 onwards

www.edupassacademy.com

List of Courses

Total Hours : 45 Hours / 15 Days Credit : 02

Marks : 100 Marks Mode : Online / Offline

Syllabus

EDUCS01	**Python Programming**	**2 Credits | 45 Hours**

Course Description

It is a powerful programming language used in data science, web development, and creating software prototypes. Moreover, it acts as a “scripting language” and is currently used in software applications-one of the best and emerging computer science certification courses.

Course Objectives

1. Learn the fundamentals of Python programming, including syntax, data types, operators, and control flow.
2. Gain experience working with Python data structures and sets.
3. Be able to write Python functions to define reusable blocks of code.
4. Understand object-oriented programming concepts in Python classes and objects.
5. Get an introduction to working with Python libraries and modules.

Course Contents

Module I: Introduction to Python Programming

Setting up the Python environment, Python basics (syntax, data types, variables, operators), and Input and output

Module II: Control Flow and Functions

Conditional statements (if, elif, else), Loops (for, while), Functions (defining, calling, arguments, return values)

Module III: Data Structures

Lists, Tuples, Dictionaries and Sets

Module IV: Object-Oriented Programming (OOP)

Classes and objects, Inheritance, and Polymorphism:

Module V: Working with Python Libraries and Modules

Introduction to popular Python libraries (e.g., NumPy, Pandas, Matplotlib), Using libraries for data analysis, visualization, and other tasks

Course Outcomes

1. Students will be able to write basic Python programs to solve computational problems.
2. Students will be familiar with fundamental Python data structures and their usage.
3. Students will be able to define and use functions in Python programs.
4. Students will have an understanding of object-oriented programming concepts.
5. Students will be able to leverage Python libraries for various tasks.

EDUCS02	Artificial Intelligence & Machine Learning	2 Credits | 45 Hours

Course Description

This course introduces the fundamental concepts of Artificial Intelligence and Machine Learning, with a focus on implementing them using Python programming. You'll explore various AI techniques and machine learning algorithms to solve real-world problems.

Course Objectives

1. Gain a solid understanding of the core concepts of AI and ML.
2. Learn different AI problem-solving approaches like search algorithms, game playing, and knowledge representation.
3. Understand supervised and unsupervised learning paradigms in machine learning.
4. Implement popular machine learning algorithms using Python libraries like Scikit-learn and TensorFlow.
5. Evaluate and interpret the results of machine learning models.
6. Apply AI and ML techniques to solve practical problems in various domains.

Course Contents

- **Module I: Introduction to Artificial Intelligence**
  - What is AI? Different AI approaches
  - Search algorithms (Breadth-First Search, Depth-First Search)
  - Game playing (Minimax algorithm)
  - Knowledge representation (Logic Programming)
- **Module II: Introduction to Machine Learning**
  - Supervised vs. Unsupervised Learning
  - Machine Learning workflow (Data Preparation, Model Selection, Training, Evaluation)
  - Basic Statistics and Probability for Machine Learning
- **Module III: Supervised Learning Algorithms**
  - Linear Regression
  - Classification (Logistic Regression, Decision Trees)
  - Performance metrics (Accuracy, Precision, Recall, F1-score)
- **Module IV: Unsupervised Learning Algorithms**
  - K-Means Clustering
  - Principal Component Analysis (PCA)
- **Module V: Advanced Topics in AI & ML (Optional)**
  - Deep Learning (Introduction to Neural Networks)
  - Natural Language Processing (NLP) Fundamentals
  - Reinforcement Learning Concepts

Course Outcomes

1. Students will acquire a comprehensive understanding of AI and ML fundamentals.
2. Students will be able to apply search algorithms, knowledge representation techniques, and solve basic AI problems.

3. Students will be proficient at implementing various machine learning algorithms using Python libraries.
4. Students will have the skills to evaluate and interpret the performance of machine learning models.
5. Students will be able to apply AI and ML techniques to solve real-world data science problems.

EDUCS03	**Big Data Analytics**	**2 Credits | 45 Hours**

Course Description

This course equips you with the skills to manage, analyze, and extract insights from large and datasets (big data). You'll explore various tools and techniques to process, visualize, and disco valuable patterns from big data using Python.

Course Objectives

1. Understand the concepts and challenges associated with big data.
2. Learn different architectures and frameworks for handling big data.
3. Gain proficiency in data wrangling techniques using Python libraries (e.g., Pandas).
4. Explore data visualization methods for big data analysis.
5. Discover patterns and trends from large datasets using big data analytics techniques.
6. Leverage big data insights for data-driven decision making in various domains.

Course Contents

Module I: Introduction to Big Data

- What is Big Data? 4 V's of Big Data (Volume, Variety, Velocity, Veracity)
- Challenges of Big Data Management
- Big Data Analytics Applications

Module II: Big Data Architectures and Frameworks

- Distributed File Systems (HDFS)
- MapReduce Paradigm for Big Data Processing
- Introduction to Apache Spark and its functionalities

Module III: Data Wrangling with Python Libraries

- Using Pandas for data manipulation, cleaning, and transformation
- Handling missing values and data inconsistencies

Module IV: Data Visualization for Big Data

- Introduction to Data Visualization Techniques
- Creating effective charts and graphs with Python libraries (Matplotlib, Seaborn)

Module V: Big Data Analytics Techniques

- Exploratory Data Analysis (EDA) for big data
- Introduction to data mining and machine learning for big data
- Case studies of big data analytics applications

Course Outcomes

1. Students will acquire a solid understanding of big data concepts and challenges.
2. Students will be familiar with big data architectures and frameworks like Hadoop and Spark.
3. Students will be proficient in data cleaning and manipulation using Python libraries.
4. Students will be able to use data visualization tools to represent big data insights effectively.
5. Students will gain the ability to analyze big data using various techniques and discover valuable patterns.

EDUCS04	**Cyber Security**	**2 Credits | 45 Hours**

Course Description

This course delves into the ever-evolving world of Cyber Security, equipping you with the kno and skills to protect information systems and data from unauthorized access, use, disclosure, disruption, modification, or destruction. You'll explore core security concepts, threats, vulneral and countermeasures, gaining practical experience through hands-on labs and simulations.

Course Objectives

1. Understand the fundamental principles of Cyber Security (CIA triad - Confidentiality, Integrity, Availability).
2. Identify various cyber threats, vulnerabilities, and attack vectors.
3. Learn essential tools and techniques for network security, cryptography, and system hardening.
4. Gain proficiency in risk assessment and mitigation strategies for cyber security.
5. Understand security policies, procedures, and best practices for user awareness and incident response.
6. Develop skills for analyzing and detecting cyber security threats using practical exercises.

Course Contents

Module I: Introduction to Cyber Security

- Overview of Cyber Security landscape and its importance
- CIA Triad (Confidentiality, Integrity, Availability)
- Cybercrime and its impact on individuals and organizations

Module II: Cyber Threats and Vulnerabilities

- Types of Cyber Threats (Malware, Phishing, Social Engineering, etc.)
- Common System Vulnerabilities (Software vulnerabilities, Network vulnerabilities)
- Attack Vectors and Penetration Testing Techniques

Module III: Network Security Fundamentals

- Network Security concepts like Firewalls, Intrusion Detection/Prevention Systems (IDS/IPS)
- Network Segmentation and Access Control Measures
- Virtual Private Networks (VPNs) and secure communication protocols

Module IV: Cryptography and Data Security

- Encryption and Decryption techniques (Symmetric, Asymmetric)
- Digital Signatures and Hashing Algorithms
- Secure Key Management Practices

Module V: Security Policies and Incident Response

- User Awareness and Security Training Programs
- Security Policy Development and Implementation
- Incident Response Procedures and Disaster Recovery Plans

Course Outcomes

1. Students will acquire a comprehensive understanding of cyber security threats and vulnerabilities.
2. Students will be able to identify and implement network security best practices.
3. Students will gain hands-on experience with cryptography and data security techniques.
4. Students will develop skills to assess cyber security risks and implement mitigation strategies.
5. Students will understand the importance of security policies, procedures, and incident response plans.
6. Students will be better equipped to protect information systems and data from cyberattacks.

EDUCS05	**Fundamentals of Full Stack Development**	**2 Credits | 45 Hours**

Course Description

This course provides a comprehensive introduction to the world of full-stack development, equipping you with the foundational skills to build interactive web applications from the ground up. You'll explore both the front-end (user interface) and back-end (server-side) aspects of web development, gaining proficiency in essential technologies.

Course Objectives

1. Gain a solid understanding of the full-stack development workflow.
2. The fundamentals of HTML, CSS, and JavaScript for building dynamic web pages.
3. Undersatnd the Document Object Model (DOM) using JavaScript.
4. Explore client-side scripting with JavaScript frameworks like React or vanilla JavaScript.
5. Understand the basics of server-side development with a popular technology like Python or Node.js.
6. Learn how to connect a front-end application to a back-end database for data storage and retrieval.
7. Develop basic understanding of APIs and their role in web development.
8. Build a simple full-stack web application to showcase your acquired skills.

Course Contents

Module I: Introduction to Full Stack Development

- Overview of the full-stack development role and responsibilities
- Front-End vs. Back-End Development
- Full-stack development workflow (development cycle)

Module II: Front-End Development Essentials

- HTML: Structure and elements of web pages
- CSS: Styling web pages for visual appeal and responsiveness
- JavaScript: Introduction to programming and interactivity
- DOM Manipulation with JavaScript

Module III: Building Dynamic Web Applications

- Introduction to JavaScript Frameworks (e.g., React) or Vanilla JavaScript for building interactive user interfaces
- Understanding components, state management, and event handling
- Building reusable components for efficient development

Module IV: Introduction to Back-End Development

- Server-side programming fundamentals with a chosen technology (e.g., Python or Node.js)
- Introduction to databases and data storage concepts (e.g., SQL databases)
- Handling user requests and sending responses

Module V: Connecting Front-End and Back-End

- Introduction to APIs (Application Programming Interfaces)
- Building APIs to expose data and functionality to the front-end
- Making API calls from the front-end application to fetch and manipulate data
- Introduction to data persistence (storing data in a database)

Module VI: Building a Full-Stack Application (Project)

- Apply learned concepts to build a simple web application (e.g., to-do list, weather app)
- Integrate front-end and back-end components

Course Outcomes

1. Students will acquire a fundamental understanding of full-stack development principles.
2. Students will be able to build basic web pages with HTML, CSS, and JavaScript.
3. Students will understand how to manipulate the DOM and create dynamic user interfaces.
4. Students will have an introduction to server-side development concepts using a chosen technology.
5. Students will be able to connect front-end applications to back-end databases for data management.
6. Students will develop the ability to build simple full-stack web applications.

EDUCS06	**Advanced Full Stack Development**	**2 Credits | 45 Hours**

Course Description

This course builds upon your existing full-stack development foundation, delving into advanced concepts, frameworks, and best practices to create complex, scalable, and robust web applica You'll explore advanced front-end frameworks, server-side architectures, database optimizatio delve into areas like security and deployment.

Course Objectives

1. Gain proficiency in advanced features of a chosen front-end framework (e.g., React, Angular).
2. Understand and implement design patterns and best practices for front-end development.
3. Learn advanced state management techniques for complex web applications.
4. Explore back-end architectures like microservices and serverless functions (e.g., with Node.js or other technologies).
5. Optimize database queries and structures for performance and scalability.
6. Understand authentication and authorization mechanisms for secure web applications.
7. Learn continuous integration and continuous delivery (CI/CD) practices for deployment automation.
8. Build a real-world application showcasing advanced full-stack development skills.

Course Contents

Module I: Advanced Front-End Development

- Deep dive into advanced features of a chosen framework (e.g., routing, Redux, testing)
- Design patterns and best practices for clean and maintainable code
- Advanced state management techniques for complex applications
- Introduction to performance optimization techniques for front-end applications

Module II: Advanced Back-End Development

- Exploring back-end architectures like microservices and serverless functions
- Building and deploying scalable back-end services
- Database optimization for performance and scalability (e.g., indexing, query optimization)
- Introduction to NoSQL databases (optional)

Module III: Security and Best Practices

- Understanding common web vulnerabilities and security threats
- Implementing secure authentication and authorization mechanisms (e.g., JWT)
- Input validation and sanitization to prevent security vulnerabilities
- DevOps principles and best practices (version control, CI/CD)

Module IV: Building a Real-World Application (Project)

- Apply learned concepts to build a real-world full-stack application
- Focus on scalability, security, and performance considerations
- Utilize continuous integration and deployment practices for automation

Course Outcomes

1. Students will acquire advanced skills in a chosen front-end framework.
2. Students will be able to design and build complex, robust, and scalable web applications.
3. Students will understand advanced techniques for state management and front-end performance optimization.
4. Students will gain experience with advanced back-end architectures and database optimization.
5. Students will be able to implement secure authentication and authorization mechanisms.
6. Students will have a foundational understanding of DevOps principles and best practices.

EDUCS07	**Web Designing (HTML & CSS)**	**2 Credits | 45 Hours**

Course Description

This course introduces you to the fundamental building blocks of web design, focusing on HTM (Hypertext Markup Language) and CSS (Cascading Style Sheets). You'll learn how to create v appealing and user-friendly websites using these powerful tools.

Course Objectives

1. Understand the core principles of web design and its importance.
2. Gain proficiency in writing valid and well-structured HTML code.
3. Master fundamental CSS properties for styling web pages.
4. Learn how to structure web pages using HTML elements like headings, paragraphs, lists, tables, and images.
5. Apply CSS to control layout, typography, color schemes, and responsiveness.
6. Implement basic animations and interactive elements using CSS.
7. Validate and optimize your HTML and CSS code for best practices.
8. Design and build simple, static web pages using HTML & CSS.

Course Contents

Module I: Introduction to Web Design

- Importance of web design, user experience (UX), and user interface (UI)
- Web design tools and workflow
- Introduction to HTML and CSS

Module II: Building with HTML

- HTML syntax and basic elements (headings, paragraphs, lists)
- Creating hyperlinks for navigation
- Embedding images and multimedia content
- Understanding HTML forms for user input
- Validating and structuring HTML code

Module III: Styling with CSS

- CSS syntax and selectors (e.g., class, ID)
- Formatting text, colors, and backgrounds
- Laying out web pages with CSS (positioning, float)
- Responsive design basics for various screen sizes
- Introduction to CSS frameworks (optional)

Module IV: Advanced CSS and Design Principles

- CSS animations and basic interactive elements (e.g., hover effects)
- Applying design principles like typography, color theory, and whitespace
- Optimizing website performance and accessibility

Module V: Building a Simple Website Project

- Design and plan a simple web page or website layout
- Implement the design using HTML and CSS skills
- Validate and refine the website for optimal presentation

Course Outcomes

1. Students will acquire a solid understanding of HTML and CSS fundamentals.
2. Students will be able to write clean and valid HTML code for web page structure.
3. Students will be proficient in applying CSS for styling and layout purposes.
4. Students will be able to design and build basic, static web pages with a focus on user experience.
5. Students will have an understanding of web design best practices and accessibility considerations.

EDUCS08	Web Application Development	2 Credits | 45 Hours

Course Description

This course delves into the world of web application development, equipping you with the skills to build dynamic and interactive web applications. You'll explore various technologies and frameworks to create powerful web applications that interact with data and user input.

Course Objectives

1. Understand the fundamental principles of web application development.
2. Gain proficiency in server-side scripting languages (e.g., Python, PHP, Node.js) for handling user requests and responses.
3. Learn how to connect web applications to databases for data storage and retrieval.
4. Master essential web development frameworks (e.g., Django, Express.js) for efficient application building.
5. Implement user authentication and authorization mechanisms for secure access.
6. Understand web application architecture concepts (e.g., MVC, RESTful APIs).
7. Build functionalities like forms, user interaction, and data manipulation within your web application.
8. Deploy your web application to a live server for public access.

Course Contents

Module I: Introduction to Web Application Development

- Overview of client-server architecture and web application development workflow
- Different types of web applications (static, dynamic, single-page)
- Introduction to server-side programming concepts

Module II: Server-Side Scripting and Frameworks

- Hands-on experience with a chosen server-side scripting language (e.g., Python with Django or Node.js with Express.js)
- Learning fundamental syntax, data types, control flow, and functions
- Exploring web development frameworks to simplify application building
- Understanding routing, models, views, and controllers (MVC architecture)

Module III: Working with Databases

- Introduction to relational databases (e.g., MySQL)
- Writing SQL queries to interact with databases (CRUD operations)
- Connecting your web application to a database for data persistence

Module IV: Building Web Applications

- Implementing user interfaces using HTML and CSS (integration with server-side code)
- Handling user input (forms, data validation)
- Building dynamic functionalities like user registration, login, and data manipulation
- Introduction to front-end frameworks (optional) for advanced interactions

Module V: Advanced Topics and Deployment

- User authentication and authorization mechanisms (e.g., sessions, tokens)
- Introduction to web application security best practices
- Understanding RESTful APIs for data exchange with other applications
- Deploying your web application to a live server environment

Course Outcomes

1. Students will acquire a solid understanding of web application development principles.
2. Students will be proficient in server-side scripting to build dynamic web applications.
3. Students will be able to use frameworks to create web applications efficiently.
4. Students will understand how to connect web applications to databases for data management.
5. Students will be able to implement secure user authentication and interactive features.
6. Students will have the skills to deploy their web applications for public access.

EDUCS09	**Mobile Application Development**	**2 Credits | 45 Hours**

Course Description

This course introduces you to the exciting world of mobile application development using Flutter, a powerful open-source framework developed by Google. You'll learn how to create beautiful and user-friendly mobile applications for both Android and iOS platforms using a single codebase.

Course Objectives

1. Gain a solid understanding of mobile application development concepts and platforms (Android, iOS).
2. Master the fundamentals of the Dart programming language, the foundation of Flutter.
3. Learn how to design user interfaces (UIs) using Flutter's widget system.

4. Explore functionalities like state management, data handling, and user interaction in Flutter apps.
5. Understand the importance of navigation and routing in mobile applications.
6. Implement features like network calls and APIs for data access in your apps.
7. Integrate multimedia elements like images, videos, and sensors into your Flutter applications.
8. Build a basic mobile application using Flutter for Android and iOS.

Course Contents

Module I: Introduction to Mobile Application Development

- Overview of mobile app development landscape (Android, iOS)
- Benefits of cross-platform development with Flutter
- Setting up the Flutter development environment

Module II: Dart Programming Fundamentals

- Understanding the Dart programming language (syntax, data types, control flow)
- Working with variables, functions, and object-oriented concepts in Dart

Module III: Building User Interfaces with Flutter

- Exploring Flutter's widget system for building UI components
- Understanding layouts, widgets, and state management using Flutter's widgets
- Implementing dynamic UIs that respond to user interaction

Module IV: Advanced Flutter Concepts

- Navigation patterns in mobile applications
- Handling user input, forms, and data validation
- Working with APIs and network calls to fetch data
- Integrating multimedia elements (images, videos) and sensors in your apps

Module V: Building a Mobile Application with Flutter (Project)

- Design and plan a simple mobile application (e.g., to-do list, weather app)
- Develop the application using Flutter for both Android and iOS platforms
- Implement core functionalities and user interface using learned concepts
- Test and deploy the application to a virtual device or emulator

Course Outcomes

1. Students will acquire a foundational understanding of mobile application development.
2. Students will be proficient in the Dart programming language for building Flutter apps.
3. Students will learn how to design and build user interfaces using Flutter's widget system.
4. Students will be able to implement various functionalities in their Flutter applications.
5. Students will understand how to handle data, network access, and multimedia elements in mobile apps.
6. Students will build a basic mobile application showcasing their Flutter development skills.

EDUCS10	Data Science	2 Credits | 45 Hours

Course Description

This course delves into the fascinating world of data science, equipping you with the skills to extract valuable insights from large datasets. You'll explore various data analysis techniques, machine learning algorithms, and tools to solve real-world problems using the power of data.

Course Objectives

1. Understand the core principles of data science and its applications in various domains.
2. Gain proficiency in Python programming, a widely used language for data science.
3. Master data wrangling techniques to clean, manipulate, and transform datasets.
4. Learn exploratory data analysis (EDA) methods to understand data characteristics and patterns.
5. Explore various supervised and unsupervised machine learning algorithms for data analysis and prediction.
6. Gain practical experience with popular data science libraries like NumPy, Pandas, and scikit-learn.
7. Develop skills for data visualization to effectively communicate insights from data analysis.
8. Build and evaluate machine learning models for solving real-world problems.

Course Contents

Module I: Introduction to Data Science

- What is Data Science? The Data Science Workflow
- Applications of Data Science in different fields (e.g., finance, healthcare)
- Introduction to Python Programming for Data Science

Module II: Data Wrangling and Preprocessing

- Understanding data types, missing values, and inconsistencies
- Techniques for data cleaning, transformation, and feature engineering
- Hands-on practice with Python libraries for data manipulation (Pandas)

Module III: Exploratory Data Analysis (EDA)

- Statistical summaries and data visualization techniques
- Identifying patterns, trends, and relationships within data
- Using Python libraries like NumPy and Matplotlib for data exploration

Module IV: Introduction to Machine Learning

- Supervised Learning vs. Unsupervised Learning algorithms
- Understanding the Machine Learning workflow (model training, evaluation, deployment)

- Introduction to common supervised learning algorithms (e.g., Linear Regression, Decision Trees)

Module V: Machine Learning with Python Libraries

- Hands-on experience with scikit-learn library for implementing machine learning algorithms
- Model training, evaluation metrics (accuracy, precision, recall)
- Introduction to model selection and hyperparameter tuning

Module VI: Data Visualization and Communication

- Creating effective charts and graphs for presenting data insights
- Storytelling with data to communicate findings to a wider audience

Module VII: Case Studies and Project Work

- Applying data science techniques to real-world case studies in various domains
- Developing a data science project to showcase your learned skills

Course Outcomes

1. Students will acquire a solid foundation in data science concepts and methodologies.
2. Students will be proficient in Python programming for data analysis and manipulation.
3. Students will be able to clean, explore, and analyze data using various techniques.
4. Students will understand the principles of supervised and unsupervised machine learning.
5. Students will gain practical experience with popular data science libraries.
6. Students will develop skills for data visualization and communication of insights.
7. Students will be able to apply data science techniques to solve real-world problems through projects.

EDUCS11	**Robotics**	**2 Credits | 45 Hours**

Course Description

This course introduces you to the captivating realm of robotics, exploring the design, development, and operation of robots. You'll delve into the fundamental principles of mechanics, electronics, control systems, and programming to understand how robots function in the real world.

Course Objectives

1. Gain a comprehensive understanding of the history and evolution of robotics.
2. Understand the different types of robots (industrial, service, mobile, etc.) and their applications.
3. Explore core robotics concepts like kinematics, dynamics, and robot motion planning.
4. Learn about sensors and actuators that enable robots to perceive and interact with their environment.
5. Gain proficiency in basic robot programming languages and control systems.
6. Understand the fundamentals of robot electronics and hardware components.
7. Develop basic skills in robot simulation and virtual environments.
8. Design and build a simple robot using learned concepts (optional, project-based).

Course Contents

Module I: Introduction to Robotics

- History of Robotics and its impact on society
- Different classifications of robots and their applications
- Exploring robot anatomy - actuators, sensors, and control systems

Module II: Kinematics and Dynamics

- Understanding robot motion - degrees of freedom, coordinate systems
- Kinematic analysis of robot movements (forward and inverse kinematics)
- Introduction to robot dynamics (forces, energy, and motion)

Module III: Sensors and Perception

- Types of sensors used in robots (e.g., vision, proximity, touch)
- Sensor data acquisition and interpretation for robot environment awareness
- Introduction to sensor fusion for robust perception

Module IV: Robot Programming and Control

- Basic robot programming languages and control systems (e.g., ROS, Arduino)
- Programming robots to perform basic tasks and navigate their environment
- Understanding control strategies for robot behavior (PID control)

Module V: Robot Electronics and Hardware

- Introduction to robot electronics components (motors, power supplies)
- Interfacing sensors and actuators with control systems
- Basic troubleshooting and maintenance of robot hardware (optional)

Module VI: Robot Simulation and Project Work (Optional)

- Utilizing robot simulation software to model and test robot behavior
- Designing and building a simple robot (e.g., line follower, pick-and-place)
- Applying learned concepts in a project-based setting (may vary)

Course Outcomes

1. Students will acquire a foundational understanding of robotics principles and technologies.
2. Students will be able to differentiate between various types of robots and their applications.
3. Students will understand the fundamental concepts of robot motion, sensors, and control systems.
4. Students will gain basic skills in robot programming and control techniques.
5. Students will be able to use robot simulation tools for testing and development.
6. Students will develop problem-solving skills through potential robot design projects.

EDUCS12	**Internet of Things**	**2 Credits | 45 Hours**

Course Description

This course delves into the ever-expanding world of the Internet of Things (IoT), exploring how everyday objects are becoming connected and intelligent. You'll learn about the core technologies, applications, and security considerations of building and interacting with interconnected devices within the IoT ecosystem.

Course Objectives

1. Gain a comprehensive understanding of the Internet of Things (IoT) concept and its potential impact.
2. Explore various types of IoT devices, sensors, and communication protocols.
3. Understand the architecture of an IoT system, including data collection, processing, and analytics.
4. Learn about cloud platforms and their role in managing and connecting IoT devices.
5. Gain proficiency in basic programming for interacting with IoT devices.
6. Explore security challenges and best practices for securing IoT devices and data.
7. Understand real-world applications of IoT in various industries and domains.
8. Design and develop a simple IoT project (optional, project-based).

Course Contents

Module I: Introduction to IoT

- What is the Internet of Things? (IoT) - Definition, history, and evolution
- Applications of IoT in different sectors (e.g., smart homes, wearables, industrial automation)
- The impact of IoT on society and future trends

Module II: Building Blocks of IoT

- Types of IoT devices and sensors (e.g., temperature sensors, accelerometers)
- Communication protocols for data transmission (e.g., Wi-Fi, Bluetooth, cellular)
- Introduction to networking concepts for IoT devices

Module III: IoT System Architecture

- Data collection, processing, and analytics in IoT systems
- Role of cloud platforms in managing and connecting IoT devices
- Understanding data security and privacy considerations in IoT

Module IV: Programming for IoT

- Introduction to basic programming languages/environments for interacting with IoT devices (e.g., Python, Arduino)
- Sending commands and receiving data from IoT devices
- Building simple applications to control or monitor IoT devices

Module V: Security in IoT

- Potential security vulnerabilities in IoT systems (e.g., hacking, data breaches)
- Secure coding practices and authentication mechanisms for IoT devices
- Securing data transmission and storage in IoT systems

Module VI: Applications and Project Development (Optional)

- Exploring real-world applications of IoT in various domains (e.g., healthcare, agriculture, smart cities)
- Designing and building a simple IoT project (e.g., smart home device, environmental monitoring system)
- Integrating learned concepts of sensors, communication, and programming (may vary)

Course Outcomes

1. Students will acquire a solid understanding of the fundamental principles of IoT.
2. Students will be able to identify different types of IoT devices and their functionalities.
3. Students will understand the architecture and communication protocols used in IoT systems.
4. Students will gain basic programming skills for interacting with IoT devices.

5. Students will be aware of security challenges and best practices for securing IoT deployments.
6. Students will be able to analyze real-world applications of IoT in various industries.
7. Students will have the opportunity to develop practical skills through an optional project (if offered).

EDUCS13	**R Programming**	**2 Credits | 45 Hours**

Course Description

This course equips you with the powerful tools of R, a leading programming language for statistical computing and data analysis. You'll delve into R's functionalities, explore data manipulation techniques, statistical modelling, and data visualization to unlock valuable insights from your data.

Course Objectives

1. Gain a solid understanding of R programming syntax and data structures.
2. Master the art of data manipulation and cleaning in R with essential packages.
3. Learn statistical analysis techniques, including descriptive statistics, hypothesis testing, and regression analysis.
4. Explore powerful data visualization libraries in R to create compelling visualizations.
5. Understand the concept of reproducible research and implement best practices in R.
6. Gain proficiency in working with real-world datasets to solve practical problems.

Course Contents

Module I: Introduction to R Programming

- Setting up the R environment and exploring its basic functionalities
- Understanding data types in R (vectors, matrices, data frames)
- Working with operators, control flow statements, and functions

Module II: Data Manipulation with R

- Importing data from various sources (CSV, Excel, databases)
- Data wrangling techniques for cleaning, transforming, and organizing data
- Essential R packages for data manipulation (dplyr, tidyr)

Module III: Statistical Analysis in R

- Descriptive statistics - summarizing data characteristics
- Hypothesis testing - evaluating relationships and drawing conclusions
- Introduction to common statistical tests (t-tests, ANOVA, chi-square)
- Linear regression modeling for exploring relationships between variables

Module IV: Data Visualization with R

- Creating informative visualizations (histograms, scatter plots, boxplots)
- Powerful R libraries for data visualization (ggplot2, lattice)
- Customization of plots for clarity and effective communication of insights

Module V: Reproducible Research with R

- Importance of reproducible research and best practices in R
- Version control with Git for managing code and data
- R Markdown for creating literate documents that combine code, results, and explanations

Module VI: Working with Real-World Data (Project)

- Exploring and analysing a real-world dataset related to your interests
- Applying learned techniques for data manipulation, statistical analysis, and visualization
- Communicating findings through reports or presentations (optional)

Course Outcomes

1. Students will acquire a strong foundation in R programming for data analysis.
2. Students will be proficient in data manipulation and cleaning techniques.
3. Students will understand and apply various statistical analysis methods in R.
4. Students will be able to create informative data visualizations using R libraries.
5. Students will develop skills for reproducible research practices with R.
6. Students will gain experience in working with real-world datasets to solve practical problems.

EDUCS14	**Multimedia & Animation**	**2 Credits | 45 Hours**

Course Description

Multimedia and animation are two powerful tools that come together to create engaging and interactive experiences. This course delves into both these aspects, equipping you with the skills to bring your ideas to life through a combination of text, audio, images, video, and animation.

Course Objectives

1. Understand the core principles of multimedia and its various components (text, audio, images, video).
2. Gain proficiency in using multimedia authoring tools to create interactive presentations, websites, or applications.

3. Explore the fundamentals of animation and different animation techniques (2D, 3D, stop-motion).
4. Learn storyboarding techniques to visually plan your animation projects.
5. Master basic animation software to create simple animations.
6. Develop an understanding of design principles and visual communication for multimedia projects.
7. Apply your skills to create a multimedia project that combines animation and other multimedia elements (optional, project-based).

Course Contents

Module I: Introduction to Multimedia

- What is Multimedia? Benefits and applications of multimedia in various fields (education, entertainment, marketing)
- Exploring different multimedia components (text, audio, images, video) and their formats
- Understanding multimedia design principles and user experience (UX)

Module II: Multimedia Authoring Tools

- Introduction to popular multimedia authoring tools (e.g., Adobe Captivate, Articulate Storyline)
- Creating interactive elements like hyperlinks, buttons, & navigation multimedia projects
- Integrating audio, video, and animation elements within your projects

Module III: Introduction to Animation

- The history and evolution of animation
- Different types of animation (2D, 3D, stop-motion, motion graphics) and applications
- Principles of animation (timing, anticipation, etc.,) for creating realistic movement

Module IV: Storyboarding and Animation Software

- Storyboarding techniques for planning your animation projects
- Introduction to basic animation software (e.g., Adobe Animate, Toon Boom Harmony)
- Creating simple animations using keyframes, tweens, and basic animation principles

Module V: Design and Visual Communication

- Design principles for creating visually appealing and effective multimedia projects
- Applying color theory, typography, and layout principles
- Importance of visual storytelling and conveying messages through visuals

Module VI: Creating a Multimedia Project (Optional)

- Develop a concept for a multimedia project that combines animation and other multimedia elements (e.g., educational video, interactive presentation)
- Apply learned skills in storyboarding, animation, and multimedia authoring tools
- Integrate audio, video, and animation elements to create a cohesive final product

Course Outcomes

1. Students will acquire a solid understanding of multimedia concepts and its components.
2. Students will be proficient in using multimedia authoring tools for creating interactive projects.
3. Students will gain a foundational knowledge of animation principles and techniques.
4. Students will be able to use basic animation software to create simple animations.
5. Students will develop skills in storyboarding, visual communication, and design principles.
6. Students will be able to apply their skills to create a multimedia project that combines animation and other multimedia elements (if applicable).

EDUCS15	**React UI & UX**	**2 Credits | 45 Hours**

Course Description

This course dives into the exciting world of React, a powerful JavaScript library for building user interfaces (UI) and user experiences (UX) for web applications. You'll learn how to craft modern, responsive, and interactive web interfaces that not only look great but also provide a seamless user experience.

Course Objectives

1. Gain a solid understanding of React fundamentals and core components.
2. Master the concept of state management and data flow in React applications.
3. Learn how to build reusable and composable React components for efficient UI development.
4. Explore advanced UI design concepts like props, routing, and forms in React.
5. Understand best practices for user experience (UX) design in web applications.
6. Implement accessibility considerations to make your React applications usable by everyone.
7. Build a full-fledged React web application showcasing your learned skills (project-based).

Course Contents

Module I: Introduction to React and JSX

- What is React? Understanding the benefits & popularity of React for building web UIs.
- Introduction to JSX syntax - the combination of HTML and JavaScript within React.
- Creating your first React component and understanding component lifecycle methods.

Module II: Components and State Management

- Deep dive into building reusable React components - the building blocks of your UI.
- Working with React state to manage dynamic data and update the UI accordingly.
- Exploring different state management solutions like Redux or Context API for complex applications (optional).

Module III: Props and Data Flow

- Passing data between React components using props for efficient communication.
- Understanding unidirectional data flow in React for predictable application behavior.

Module IV: Advanced React Concepts

- Routing in React applications - handling navigation between different pages or views.
- Building interactive forms with React - handling user input and validation.
- Introduction to styling React components using CSS frameworks like CSS-in-JS solutions (optional).

Module V: User Experience (UX) Design for Web Applications

- User-cantered design principles for creating intuitive and user-friendly applications.
- Information architecture and navigation design for clear user journeys.
- Visual design considerations like typography, color palettes, and overall UI aesthetics.

Module VI: Accessibility Best Practices

- Understanding the importance of web accessibility and making your React apps usable for everyone.
- Implementing WCAG guidelines for accessibility features like screen readers and keyboard navigation.

Module VII: Building a React Web Application (Project)

- Develop a concept for a full-fledged React web application (e.g., portfolio website, e-commerce store).
- Design and structure your application using components, state management, and routing.
- Implement interactive features, user interface elements, and styling.
- Integrate accessibility considerations and test your application for usability.

Course Outcomes

1. Students will acquire a comprehensive understanding of React concepts and development.
2. Students will be proficient in building reusable and well-structured React components.

3. Students will gain expertise in managing state and data flow within React applications.
4. Students will be able to implement advanced UI features like routing and forms.
5. Students will develop an understanding of UX design principles for web applications.
6. Students will learn how to build accessible React applications for a wider audience.
7. Students will build a complete React web application showcasing their skills (if applicable).

EDUCS16	Digital Marketing	2 Credits | 45 Hours

Course Description

Welcome to the exciting world of Digital Marketing! This course delves into the strategies, tools techniques used to promote brands, products, and services through online channels. You'll ex various digital marketing tactics to reach your target audience, engage them effectively, and ul drive business growth.

Course Objectives

1. Gain a comprehensive understanding of the digital marketing landscape and its importance in today's business world.
2. Master essential digital marketing channels like search engine optimization (SEO), social media marketing, and pay-per-click (PPC) advertising.
3. Learn content marketing strategies to attract and engage your target audience.
4. Explore email marketing techniques for building customer relationships and driving sales.
5. Understand website analytics and data-driven marketing for measuring campaign performance.
6. Develop a strategic approach to digital marketing campaign planning and execution.
7. Gain practical experience by working on real-world digital marketing projects (optional, project-based).

Course Contents

Module I: Introduction to Digital Marketing

- What is Digital Marketing? The evolution of marketing in the digital age.
- Importance of digital marketing for businesses of all sizes and industries.
- Understanding the digital marketing funnel - attracting, engaging, converting, and retaining customers.

Module II: Search Engine Optimization (SEO)

- Optimizing your website and content to rank higher in search engine results pages.
- Keyword research and targeting to attract organic traffic.
- On-page and off-page SEO strategies for website optimization.

Module III: Social Media Marketing

- Leveraging the power of social media platforms (Facebook, Instagram, Twitter, etc.) for brand awareness and engagement.
- Creating engaging social media content strategies for different platforms.
- Social media advertising and community management techniques.

Module IV: Content Marketing

- The power of content marketing in attracting and nurturing leads.
- Content creation strategies - blog posts, infographics, videos, and other formats.
- Content distribution channels and content promotion techniques.

Module V: Email Marketing

- Building effective email marketing campaigns to nurture leads and drive conversions.
- Email list segmentation and targeting for personalized communication.
- Email marketing automation tools and best practices.

Module VI: Pay-Per-Click (PPC) Advertising

- Exploring paid advertising options like Google Ads and social media advertising.
- Keyword research and bidding strategies for PPC campaigns.
- Landing page optimization and conversion rate optimization (CRO) techniques.

Module VII: Website Analytics and Data-Driven Marketing

- Understanding website analytics tools like Google Analytics to measure campaign performance.
- Using data insights to optimize your digital marketing strategies and make data-driven decisions.

Module VIII: Digital Marketing Campaign Planning & Execution (Optional)

- Develop a digital marketing plan for a specific product, service, or brand (optional, project-based).
- Defining marketing goals, target audience, and key performance indicators (KPIs).
- Selecting the right digital marketing channels and tactics for your campaign.
- Implementing your digital marketing plan and measuring its results.

Course Outcomes

1. Students will acquire a comprehensive understanding of digital marketing principles and strategies.
2. Students will be proficient in various digital marketing channels like SEO, social media marketing, and email marketing.
3. Students will develop skills in content creation, content marketing, and data-driven marketing.
4. Students will be able to plan and execute effective digital marketing campaigns.

EDUCS17	**MATLab**	**2 Credits | 45 Hours**

Course Description

Welcome to the world of MATLAB, a powerful programming language and environment designed specifically for technical computing. This course will equip you with the skills to solve numerical problems, analyze data, and create models across various scientific and engineering disciplines.

Course Objectives

1. Gain a solid foundation in MATLAB syntax, data structures, and control flow statements.
2. Master essential techniques for data manipulation, analysis, and visualization in MATLAB.
3. Explore numerical methods for solving mathematical problems like linear algebra, calculus, and differential equations.
4. Learn how to create and utilize functions to make your MATLAB code more efficient and reusable.
5. Understand the concept of scripting and develop basic MATLAB scripts for automating tasks.
6. Explore toolboxes and libraries that extend MATLAB's functionalities for specialized applications (optional).

Course Contents

Module I: Introduction to MATLAB

- Getting started with the MATLAB environment - interface, workspace, and basic commands.
- Understanding data types in MATLAB (numbers, strings, matrices) and working with operators.
- Control flow statements (if-else, loops) for creating conditional and iterative programs.

Module II: Data Manipulation and Analysis

- Importing data from various sources (CSV, Excel) and exporting results.
- Data cleaning, transformation, and organization techniques using built-in functions.
- Performing statistical analysis on data (descriptive statistics, hypothesis testing).

Module III: Numerical Methods

- Solving linear systems of equations using matrix operations.
- Exploring numerical methods for solving derivatives, integrals, and differential equations.
- Understanding concepts of numerical errors and their mitigation.

Module IV: Creating Functions in MATLAB

- Defining and using functions to modularize your code and improve reusability.
- Passing arguments to functions and returning output values.
- Debugging techniques for identifying and fixing errors in your MATLAB code.

Module V: Scripting in MATLAB

- Scripting basics - writing and executing MATLAB code in script files.
- Scripting for automating repetitive tasks and calculations.
- Organizing your code with comments and proper formatting for readability.

Module VI: Toolboxes and Extensions (Optional)

- Introduction to MATLAB toolboxes for specific applications (e.g., Signal Processing, Control Systems).
- Utilizing toolbox functions to perform specialized tasks related to your field of interest.
- Exploring community-developed toolboxes and extensions for further functionality.

Course Outcomes

1. Students will acquire a strong foundation in MATLAB programming for scientific computing.
2. Students will be proficient in data manipulation, analysis, and visualization techniques.
3. Students will understand and apply various numerical methods for solving mathematical problems.
4. Students will be able to create reusable functions and automate tasks using scripts.
5. Students will gain exposure to MATLAB toolboxes and their potential applications (if applicable).

EDUCS18	**Database Administration**	**2 Credits | 45 Hours**

Course Description

This course delves into the world of Database Administration (DBA), equipping you with the skills to manage, maintain, and optimize database systems. You'll explore the core principles of database management, delve into different database technologies, and learn best practices for ensuring the smooth operation, security, and performance of databases.

Course Objectives

1. Gain a comprehensive understanding of the fundamental concepts of database management systems (DBMS).
2. Explore different database models (relational, NoSQL) and their functionalities.
3. Master Structured Query Language (SQL) for data manipulation and querying in relational databases.

4. Learn essential database administration tasks like user management, security, & access control.
5. Understand database backup and recovery procedures for data protection.
6. Explore database performance optimization techniques for ensuring efficient data access.
7. Gain practical experience by working with real-world database scenarios.

Course Contents

Module I: Introduction to Database Management

- What is a database? Understanding the importance of databases in today's information-driven world.
- Different types of database models (relational, NoSQL) and their characteristics.
- Introduction to the concept of Entity-Relationship Modelling (ERM) for data design.

Module II: Exploring SQL

- Introduction to Structured Query Language (SQL) - the standard language for interacting with relational databases.
- Writing SQL queries for data retrieval (SELECT), insertion (INSERT), update (UPDATE), and deletion (DELETE).
- Utilizing advanced SQL functionalities like joins, functions, and subqueries for complex data manipulation.

Module III: Database Administration Fundamentals

- Understanding the role and responsibilities of a Database Administrator (DBA).
- User management - creating, assigning permissions, and managing user access to the database.
- Database security best practices - access control, encryption, and data privacy considerations.

Module IV: Database Backup and Recovery

- Importance of database backups for data protection in case of failures or errors.
- Different backup strategies (full, incremental, differential) and scheduling backups.
- Recovery procedures - restoring data from backups in case of data loss.

Module V: Database Performance Optimization

- Understanding database performance bottlenecks and techniques for identification.
- Database query optimization techniques - indexing, query tuning, and improving query efficiency.
- Database monitoring techniques for proactive performance management.

Module VI: Working with Real-World Databases (Optional)

- Hands-on experience with setting up and managing a database system (e.g., MySQL, PostgreSQL).
- Applying learned SQL skills for data manipulation and querying.
- Implementing security measures, user management, and backup strategies (optional, project-based).

Course Outcomes

1. Students will acquire a solid understanding of database management systems and their functionalities.
2. Students will be proficient in writing SQL queries for data manipulation and querying relational databases.
3. Students will develop essential database administration skills like user management, security, and backup procedures.
4. Students will understand database performance optimization techniques for efficient data access.
5. Students will gain practical experience working with a database system (if applicable).