Semester - 4

Course Code: 401

Course Title: Organizational Soft-skills in Software Industry

Course Code	401
Course Title	Organizational Soft-skills in Software Industry Ability Enhancement Course – 04 [In option to this course, the course will be selected by the student and required 2 credits can be opted from the list of courses mentioned in Table-6 (Page number 51 – 52) from NEP-2020 S.O.P. of Gujarat State implementation handbook for NcrFr. The credits can be acquired through any valid MOOC, online courses recognized and approved by UGC or from courses offered by college/institute out of the course basket offered by the University under the Ability Enhancement courses]
Credits	2
Course Category	**Ability Enhancement Course (AEC-04)**
Level of Course	200-299 (Intermediate Level)
Teaching per Week	2 Hours
Minimum weeks per Semester	15 (Including class work, examination, preparation etc.)
Review / Revision	-
Implementation Year:	A.Y. 2024-2025
Purpose of Course	Computer Science professionals work at different levels in the hierarchy of various jobs in IT. It is essential to understand the Organization Structure and behavior. - Integration of Knowledge and Skills: One objective of a multidisciplinary course is to foster the integration of knowledge and skills from different disciplines. By combining various areas of study, students can gain a holistic understanding of a particular topic or problem. This objective aims to break down the traditional boundaries between subjects and encourage students to see connections and relationships across different fields. - Promoting Critical Thinking and Problem Solving: Another objective is to enhance students' critical thinking and problem-solving abilities. Multidisciplinary courses often involve complex real-world issues that require a multifaceted approach. By engaging with diverse perspectives and methodologies, students develop the capacity to analyze problems from multiple angles, think creatively, and propose innovative solutions. - Enhancing Collaboration and Communication Skills: Collaboration and effective communication are essential skills in today's interconnected world. Multidisciplinary courses aim to cultivate these skills by providing opportunities for students to work collaboratively with peers from different disciplines. Through group projects, discussions, and presentations, students learn how to articulate their ideas, listen actively to others, and collaborate effectively to achieve common goals. This objective prepares students for interdisciplinary work environments and encourages the exchange of ideas across disciplinary boundaries.
Course Objective	These courses are designed as combination of Indian Languages (from the Eighth Schedule of the Indian Constitution) and English language courses, with a specific focus on enhancing language and communication skills. The primary objective of these courses is to help students acquire and demonstrate essential soft-skills in discipline specific(software industry), linguistics skills, including critical reading, expository writing and academic writing.

	HEIs have flexibility to introduce courses that are tailored to specific disciplines or are applicable across all undergraduate programmes. A list of a few AEC courses is provided in Table-6 (3.3.4) of Implementation of NEP-2020 for the state of Gujarat S.O.P.
Pre-requisite	Knowledge of English at H.Sc.(10^{th}) Level
Course Outcomes	**CO1:** After completion of the course the student will be aware about the Structure of an organization **CO2:** Also, will have better understanding of human behaviour in an organization **CO3:** Students will understand and develop their attitude **CO4:** Students will learn the importance of motivation **CO5:** Students will be able to understand the leader, skills of leader and leadership styles **CO6:** students will have idea about BPO and call centers
Mapping between Course Outcome(CO) and Program Specific Outcome (PSO):	
Course Content	**Unit 1: Introduction to Software development Organization Structure** 1.1 What makes an organization 1.2 Overview of software organizational structure and its importance in software development 1.3 Structure of organization: 1.4 Traditional vs. Agile organizational structures in software development 1.5 Roles and responsibilities within software development teams 1.6 Management in Software Organization : Scope and Role of Management **Unit 2: Writing Skills for Effective Communication in Organizations** 2.1 Importance of writing skills in software organizations 2.2 Principles of effective written communication (clarity, conciseness, coherence) 2.3 Techniques for writing professional emails, reports, and documentation 2.4 Best practices for writing technical documents and user manuals in software development **Unit-3 : Software Organizational Hierarchy and team building** 3.1 Hierarchy in software development organization and roles of Project manager, System Analyst, System Architect, Business Model Developer, Team Leaders, Coders, Debuggers. 3.2 Managerial Skills (Technical Skills, Human Skills, Conceptual Skills) 3.3 Importance of verbal communication skills in software development teams 3.3.1 Effective communication in meetings, stand-ups, and presentations 3.3.2 Active listening techniques for better understanding and collaboration 3.3.3 Strategies for conveying technical concepts to non-technical stakeholders **Unit 4: Communication Strategies for Collaboration** 4.1 Importance of communication in team collaboration and project management. 4.2 Strategies for resolving conflicts and addressing disagreements in software teams.

	PS01	PS02	PS03	PS04	PS05	PS06	PS07	PS08
CO1	■		■					
CO2	■							
CO3					■			
CO4	■		■	■				■
CO5	■		■	■	■	■		
CO6		■		■		■		■

	4.3 Effective communication techniques for remote and distributed teams. 4.4 Building rapport and fostering team cohesion through effective communication practices. 4.5 Opportunities for automation, intelligent decision-making, and impact on software development teams.
Reference Books	1.) Title: "Software Engineering at Google: Lessons Learned from Programming Over Time", Author: Titus Winters, Tom Manshreck, Hyrum Wright, Publisher: O'Reilly Media, ISBN: 978-1492082798 2.) Title: "The Elements of Style", Author: William Strunk Jr., E.B. White, Publisher: Pearson, ISBN: 978-0205309023 3.) Title: "Writing That Works: How to Communicate Effectively in Business", Author: Kenneth Roman, Joel Raphaelson, Publisher: HarperBusiness, ISBN: 978-0060956431 4.) Title: "Technical Communication: A Reader-Centered Approach", Author: Paul V. Anderson, Publisher: Cengage Learning, ISBN: 978-1305667884 5.) Title: "Crucial Conversations: Tools for Talking When Stakes Are High", Authors: Kerry Patterson, Joseph Grenny, Ron McMillan, Al Switzler, Publisher: McGraw-Hill Education, ISBN: 978-0071771320 6.) Title: "Nonviolent Communication: A Language of Life", Author: Marshall B. Rosenberg, Publisher: Puddledancer Press, ISBN: 978-1892005038. 7.) Title: "The Silent Language", Author: Edward T. Hall, Publisher: Anchor, ISBN: 978-0385055499 8.) Title: "Emotional Intelligence 2.0", Authors: Travis Bradberry, Jean Greaves, Publisher: TalentSmart, ISBN: 978-0974320625 9.) Title: "Leadership and Self-Deception: Getting Out of the Box", Authors: The Arbinger Institute, Publisher: Berrett-Koehler Publishers, ISBN: 978-1576759776 10.) Title: "Difficult Conversations: How to Discuss What Matters Most" Authors: Douglas Stone, Bruce Patton, Sheila Heen, Publisher: Penguin Books, ISBN: 978-0143118442.
Teaching Methodology	Class Work, Discussion, Self-Study, Case-study, Seminars and/or Assignments
Evaluation Method	50% Internal assessment. 50% External assessment. External Assessment: Each student will be given a case-study of software industry to study organizational structure, hierarchy of the employee structure, environment and interpersonal communication among the teams. Tools and techniques used to interact within the organization and with the clients. The students will create a report/document based on the given case study and give presentation at the end of the semester for final evaluation. The examiner panel will consist of two examiners including one faculty member/resource person who handled the course and one person from the software industry. (Incase the person from software industry is not available, both examiners can be faculty members/resource person of the institute.) Assessment : - Writing skills and report/documentation abilities (20%) - Oral presentations evaluating verbal communication skills (20%) - Viva-voce (20%) - Case study analysis and problem-solving exercises focusing on communication strategies in software organizations (40%)

Course Code : 402-01 Course Title: IoT (Internet of Things)	
Course Code	**402-01**
Course Title	**Internet of Things (IoT)**
Credit	4
Course Category	Minor Course
Level of Course	200-299 (Intermediate Level)
Teaching per Week	4 Hrs
Minimum weeks per Semester	15 (Including Class work, examination, preparation etc.)
Last Review / Revision	A.Y. 2023-2024
Implementation Year:	A.Y. 2024-2025
Medium of Instruction	English
Purpose of Course	The purpose of the "Introduction to IoT" course is to provide students with a foundational understanding of the Internet of Things (IoT) ecosystem. Through this course, students will gain insight into the concepts, technologies, and applications that underpin IoT networks and devices. They will explore the interconnected nature of IoT systems, learn about sensors, actuators, and connectivity protocols, and understand how data is collected, transmitted, and analyzed in IoT environments. Ultimately, the course aims to equip students with the knowledge and skills to comprehend the potential of IoT in various industries, and to critically evaluate IoT solutions for addressing real-world challenges.
Course Objective	To understand the concepts and protocols related to Internet of Things. To get an idea where the application areas are available for the Internet of Things to be applied.
Pre-requisite	Basic Knowledge of Networking
Course Out come	**CO1:** Understand the Concept of IoT: Students will be able to define the Internet of Things (IoT) and explain its significance in connecting physical devices, sensors, and actuators to the internet to enable data exchange and automation. **CO2:** Identify IoT Components and Technologies: Students will be able to identify and describe the key components of IoT systems, including sensors, actuators, microcontrollers, communication protocols, and cloud platforms. **CO3:** Explain IoT Communication Protocols: Students will be able to explain various communication protocols used in IoT networks, such as Wi-Fi, Bluetooth, Zigbee, and MQTT, and understand their strengths, weaknesses, and applications. **CO4:** Analyze IoT Applications and Use Cases: Students will be able to analyze real-world IoT applications and use cases across different industries, such as smart homes, healthcare, transportation, agriculture, and industrial automation. **CO5:** Design and Implement Simple IoT Solutions: Students will be able to design and implement simple IoT solutions using hardware components, microcontrollers, sensors, actuators, and basic programming languages. **CO6:** Evaluate IoT Security and Privacy Considerations: Students will be able to identify and assess security and privacy challenges in IoT

	systems, understand common vulnerabilities and threats, and explore strategies for securing IoT devices and data.
Mapping between Course Outcomes (CO) and Program Specific Outcomes (PSO):	

	PS01	PS02	PS03	PS04	PS05	PS06	PS07	PS08
CO1								
CO2								
CO3								
CO4								
CO5								
CO6								

Course Content

Unit 1: Introduction to Internet of Things

1.1 Definition & Characteristics of IoT
1.2 Introduction to IoT Architecture
1.3 Physical Design of IoT
  1.3.1 Things in IoT
  1.3.2 IoT Protocols (Ethernet , WIFI , WIMAX, LR-WPAN(Wireless personal area network), 2G/3G/4G Mobile Communication, IPV6,6LOWPAN,MQTT, WEB SOCKET)
1.4 Logical Design of IoT
  1.4.1 IoT Functional Blocks
  1.4.2 IoT Communicational Models
- Request – Response
- Publish –Subscribe
- Push –Pull
- Exclusive Pair

Unit 2. IoT and M2M

2.1 Introduction M2M
2.2 Introduction to Sensor Technology
2.3 Difference between IoT and M2M,
2.4 Security for IoT
2.5 IoT Enabling Technologies
  2.5.1 Wireless Sensor Networks
  2.5.2 Big Data Analytics,
  2.5.3 Embedded Systems.

Unit 3.Sensors and Actuators in IoT

3.1 Definition of Sensors
3.2 Types of sensors and its usage (Temperature, Humidity, Gas Detector, Ultrasonic, Fire detector, Light, Sound, IR, Water Level)
3.3 Introduction to Actuators
3.4 Types of Actuators
3.5 Difference between Sensors & Actuators

Unit 4.Introduction to Raspberry pi and Arduiano

4.1 Introduction on IoT Devices
4.2 Basic Building blocks of an IoT Device
4.3 Introduction to Raspberry pi (Concepts, purpose, Application areas)
4.4 Components of Raspberry pi
4.5 Introduction to Arduiano (Concept, purpose and Application areas)
4.6 Difference between Raspberry pi and Arduiano

Unit 5. Case Study

5.1 IoT for Smart city applications
5.2 IoT for Smart Home

	5.3 IoT for Health & Lifestyle
Reference Books	1. Internet of Things , A Hands – On Approach, Arshdeep Bahga, Vijay Madisetti published by Arshdeep Bahga& Vijay Madisetti 2. Internet of Things architecture and Design Principles, Raj Kamal, McGrawhill Education private limited, 2017 3. Learning Internet of Things, Peter Waher, / Packt Publishing Limited, 2015 4. The Internet of Things, Hakima Chaouchi, Wiley,2017 5. Getting started with the Internet of Things: by CunoPfister, O"Reilly Media. 6. The Internet of Things: Enabling Technologies, Platforms, and Use Cases", by Pethuru Raj and Anupama C. Raman (CRC Press) 7. "Building Arduino Projects for the Internet of Things: Experiments with Real-World Applications", Author: Adeel Javed, Publisher:Apress, ISBN:978-1484219393 8. "Understanding the Internet of Things: A Conceptual and Pragmatic Approach", Author: David Evans,Publisher: O'Reilly Media, ISBN: 978-1491924565 9. "Designing Connected Products: UX for the Consumer Internet of Things", Author: Claire Rowland, Elizabeth Goodman, Martin Charlier, and Ann Light, Publisher: O'Reilly Media, ISBN: 978-1449372569 10. "IoT Inc: How Your Company Can Use the Internet of Things to Win in the Outcome Economy", Author: Bruce Sinclair, Publisher:McGraw-Hill Education, ISBN: 978-1260025899
Teaching Methodology	Class Work, Discussion, Self-Study, Seminars and/or Assignments
Evaluation Method	50% Internal assessment. 50% External assessment.

Course Code: 402-02
Course Title: User Interface and User Experience Design (UI/UX Design)

Course Code	402-02
Course Title	User Interface and User Experience Design (UI/UX Design)
Credits	4
Course Category	Minor Course
Level of Course	200-299 (Intermediate Level)
Teaching per Week	4 Hrs
Minimum weeks per Semester	15 (Including class work, examination, preparation etc.)
Review / Revision	-
Implementation Year:	A.Y. 2024-2025
Purpose of Course	This course introduces UI/UX design principles, methodologies, and practical skills, preparing students for further exploration and specialization in the field. The purpose of a UI/UX course is to equip students with the knowledge, skills, and techniques necessary to design user interfaces and experiences that are intuitive, engaging, and effective. Through a combination of theoretical understanding and practical application, students learn to create user-centric designs that enhance usability, accessibility, and user satisfaction. The course covers topics such as user research, information architecture, interaction design, visual design, and usability testing, providing a comprehensive foundation in the principles and best practices of UI/UX design. By mastering these skills, students are prepared to pursue careers in various industries, contributing to the creation of seamless and enjoyable digital experiences for users.
Course Objective	i) Understand the Basics of UI/UX Design: Introduction to the fundamental principles and concepts of user interface (UI) and user experience (UX) design, including the difference between UI and UX, the importance of user-centered design, and the role of UI/UX in product development. ii) Learn User Research Methods: Familiarize with basic user research methods, such as user interviews, surveys, and observation techniques, to understand user needs, behaviors, and preferences. iii) Create Wireframes and Prototypes: Learn how to create low-fidelity wireframes and prototypes using simple design tools or pen and paper to visualize the structure and layout of digital interfaces. iv) Explore Interaction Design Principles: Introduction to interaction design principles, including affordances, feedback, and user flows, to design intuitive and responsive user interfaces that facilitate user interaction and navigation. v) Conduct Usability Testing: An overview of usability testing methods and techniques, such as heuristic evaluations and user testing sessions, to evaluate the effectiveness and usability of UI designs and gather feedback for iteration and improvement.
Pre-requisite	-
Course Outcomes	**CO1:** Provide students with a foundational understanding of user interface (UI) and user experience (UX) design principles, including usability, accessibility, and user-centered design.

CO2: Familiarize students with basic user research methodologies, such as user interviews, surveys, and personas, to identify user needs, behaviors, and preferences.

CO3: Develop students' ability to create low-fidelity wireframes and prototypes using industry-standard tools or pen and paper, enabling them to visualize and communicate design concepts effectively.

CO4: Introduce students to interaction design principles, including affordances, feedback, and user flows, to design intuitive and responsive digital interfaces that facilitate user interaction and engagement.

CO5: Explore fundamental principles of visual design, such as typography, color theory, and layout, to create aesthetically pleasing and visually coherent UI designs that enhance user experience.

CO6: Teach students how to plan and conduct usability testing sessions, analyze feedback, and iterate on designs to improve usability and user satisfaction, ensuring that designs meet user needs and expectations.

	PS01	PS02	PS03	PS04	PS05	PS06	PS07	PS08
CO1								
CO2								
CO3								
CO4								
CO5								
CO6								

Course Content

Unit 1: Introduction to UI/UX Design:

1.1 Overview of UI/UX Design and understanding the role of UI/UX design in product development.

1.2 Introduction to user-centered design principles and methodologies.

1.3 Exploring the significance of UI/UX in enhancing user satisfaction and product success.

Unit 2: User Research and Analysis:

2.1 Importance of user research in informing design decisions.

2.2 Creating user personas to represent target users and their needs.

2.3 Techniques for conducting effective user interviews to gather insights and feedback.

2.4 Overview of usability testing methods and techniques for evaluating design prototypes.

Unit 3: Interaction Design and Information Architecture:

3.1 Principles of Interaction Design (affordances, feedback, and user flows).

3.2 Understanding information architecture and organizing content for intuitive navigation.

3.3 Techniques for creating low-fidelity wireframes and interactive prototypes to visualize design concepts.

3.4 Understanding designing effective navigation systems to facilitate user interaction and exploration.

Unit 4: Visual Design Essentials:

4.1 Basics of Visual Design (typography, color theory, and layout).

4.2 Visual hierarchy to guide user attention and emphasize important content.

4.3 Iconography and Imagery to enhance user understanding and engagement.

4.4 Importance of branding and maintaining consistency across UI elements for a cohesive user experience.

Unit 5: Usability Testing , Iteration and case study:

5.1 Usability Testing Process (planning, conducting, and analyzing usability testing sessions).

5.2 Iterative design process and User feedback for continuous improvement.

	5.3 Designing for accessibility 5.4 Case study
Reference Books	1. "Don't Make Me Think, Revisited: A Common Sense Approach to Web Usability", Author: Steve Krug, Publisher: New Riders, ISBN: 978-0321965516 2. "The Design of Everyday Things: Revised and Expanded Edition", Author: Don Norman, Publisher: Basic Books, ISBN: 978-0465050659 3. "100 Things Every Designer Needs to Know About People", Author: Susan Weinschenk, Publisher: New Riders, ISBN: 978-0321767530 4. "About Face: The Essentials of Interaction Design", Author: Alan Cooper, Robert Reimann, and David Cronin, Publisher: Wiley India, ISBN: 978-8126556744 5. "The Elements of User Experience: User-Centered Design for the Web and Beyond", Author: Jesse James Garrett, Publisher: Pearson India, ISBN: 978-8131707918 6. "Universal Principles of Design, Revised and Updated", Author: William Lidwell, Kritina Holden, and Jill Butler, Publisher: Rockport Publishers India, ISBN: 978-1631596226 7. "The UX Book: Process and Guidelines for Ensuring a Quality User Experience", Author: Rex Hartson and Pardha S. Pyla, Publisher: Pearson India, ISBN: 978-9332518320 8. "Lean UX: Designing Great Products with Agile Teams", Author: Jeff Gothelf and Josh Seiden, Publisher: Wiley India, ISBN: 978-8126561977 9. "Designing for Interaction: Creating Innovative Applications and Devices", Author: Dan Saffer, Publisher: Pearson India, ISBN: 978-8131705648 10. "Designing Interfaces: Patterns for Effective Interaction Design", Author: Jenifer Tidwell, Publisher: O'Reilly India, ISBN: 978-8184045881 11. "Designing Web Interfaces: Principles and Patterns for Rich Interactions", Author: Bill Scott and Theresa Neil, Publisher: O'Reilly India, ISBN: 978-8184045799
Teaching Methodology	Class Work, Discussion, Self-Study, Case-study, Seminars and/or Assignments
Evaluation Method	50% Internal assessment. 50% External assessment.

Course Code: 403
Course Title: Java Programming Language

Course Code	403
Course Title	Java Programming Language
Credits	4
Course Category	Major Course
Level of Course	300-399 (Higher Level)
Teaching per Week	4 Hrs. (3 Hours Theory + 2 Hours Practical work)
Minimum weeks per Semester	15 (Including class work, examination, preparation etc.)
Review / Revision	2023-2024
Implementation Year:	A.Y. 2024-2025
Purpose of Course	To teach Object Oriented Programming (OOP) concepts through Coding using Java as programming language.
Course Objective	1. To make students understand the syntax and Object Oriented Programming (OOP) concepts using Java. 2. To make students understand various inbuilt Java classes and their working. 3. To make students understand the importance of OOP methodology. 4. To make students understand various types of OOP techniques.
Pre-requisite	Prior Knowledge object oriented concepts.
Course Outcomes	**CO1:** Understand the core principles of object-oriented programming (OOP) and apply them proficiently in Java, including classes, objects, inheritance, polymorphism, and encapsulation. **CO2:** Develop the ability to design, implement, and test Java applications, employing OOP concepts to create modular, reusable, and maintainable code. **CO3:** Demonstrate competence in utilizing Java's built-in libraries and frameworks to solve real-world problems efficiently, leveraging object-oriented design patterns where applicable. **CO4:** Analyze and debug Java programs effectively, employing best practices in error handling, exception handling, and debugging techniques to ensure robustness and reliability. **CO5:** Collaborate with peers in team-based Java projects, effectively communicating ideas, contributing to code reviews, and integrating individual contributions into cohesive software solutions.
Mapping between Course Outcomes(CO) with Program Specific Outcomes(PSO)	

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8
CO1								
CO2								
CO3								
CO4								
CO5								

Course Content	**Unit 1. Introduction to Java** 1.1 Properties of Java 1.2 Comparison of java with C++ 1.3 Java Compiler, Java Interpreter 1.4 Identifier, Literals, Operators, Variables, Keywords, Data Types 1.5 Branching: If – Else, Switch 1.6 Looping: While, Do-while, For

	1.7 Type Casting **Unit 2. Classes and Objects** 2.1 Simple Class, Field 2.2 Access Controls, Object creation 2.3 Construction and Initialization 2.4 Inheritance and Polymorphism in Java 2.4.1 Data encapsulation, overriding and overloading methods 2.5 this and super keywords 2.6 Static members, static block, static class 2.7 Interfaces: 2.7.1 Introduction to Interfaces, Interface Declaration. 2.7.2 Inheriting and Hiding Concepts. 2.7.3 Inheriting, Overloading and Overriding Methods and constructors. 2.7.4 Interfaces Implementations. **Unit 3. Basic Concepts of Strings and Exceptions :** 3.1 Strings 3.1.1 Basic String operations, String Comparsion 3.1.2 String methods (charAt(), concat(), equals(), indexOf(), isEmpty(), join(), lastIndexOf(), length(),split(), substring(),trim()) 3.1.3 StringBuffer class and its constructors. 3.1.4 StringBuffer methods : (append(),insert(),update(), delete(), reverse(),capacity()) 3.2 Introduction to Exceptions: 3.2.1 Exception Types, User defined Exception 3.2.2 Throw, Throws 3.2.3 Try, Catch and Finally **Unit 4. Threads and Packages:** 4.1 Thread 4.1.1 Introduction to Threads, Thread Model 4.1.2 Priority of Threads 4.2 Package Naming, Type Imports 4.2.1 Package Access, Package Contents 4.2.2 Package Object and Specification **Unit 5. Data Structure Implementation using Java Class** 5.1 Implementation of Data Structure using Java Class: 5.1.1 Concepts of singly and singly circular link-list 5.1.2 Singly Link List : Create, traverse, insert, delete node 5.1.3 Singly circular link list: create, traverse, insert, delete node.
Reference Books	1. Java Programming Language – Ken Arnold James Gosling, David Holmes: –Addison Wesley (Pearson Education) 2. Java – The complete reference, – Herbert Schildt: – Tata McGrawHill 3. Java 2 From Scratch: – Steven Haines: –PHI. 4. Programming in Java – E-Balaguruswamy: – Tata McGraw Hill 5. Java: How to Program: – Deitel & Deitel: – PHI
Teaching Methodology	Class Work, Discussion, Lab work, Self-Study, Seminars and/or Assignments
Evaluation Method	50% Internal assessment. 50% External assessment.

Course Code: 404
Course Title: .NET Programming

Course Code	404
Course Title	.NET Programming
Credits	4
Course Category	Major Course
Level of Course	300-399 (Higher Level)
Teaching per Week	4 Hrs. (2 Hours Theory + 4 Hours Practical work)
Minimum weeks per Semester	15 (Including class work, examination, preparation etc.)
Review / Revision	2023-2024
Implementation Year:	A.Y. 2024-2025
Purpose of Course	This syllabus has been prepared for the beginners to help them understand basic .Net programming. After completing this, students will get a moderate level of expertise in .Net programming from where, they can take themselves to next levels.
Course Objective	- To make students understand .Net as simple, modern, object- oriented computer programming language developed by Microsoftto combine the power of .NET Framework and the CLR with the productivity benefits. - To make students understand basic .Net programming and will also take through various advanced concepts related to .Net programming language.
Pre-requisite	Students are expected have concepts related to Programming techniques using Object Oriented.
Course Outcomes	**CO1:** Understand the fundamentals of .NET framework: Students will gain a solid understanding of the .NET framework, including its architecture, components, and how it supports various programming languages such as C# and Visual Basic.NET. **CO2:** Develop basic programming skills in C#: Students will learn the syntax, data types, control structures, and object-oriented programming concepts in C#, one of the primary languages used in .NET development. **CO3:** Create and manipulate .NET applications: Students will be able to create, compile, debug, and run basic .NET applications using Visual Studio IDE, including console applications, Windows Forms applications, and simple web applications. **CO4:** Utilize .NET framework libraries and APIs: Students will learn to leverage the vast array of libraries and APIs provided by the .NET framework for tasks such as file I/O, database access, error handling, and networking. **CO5:** Gain familiarity with modern software development practices: Students will be introduced to essential software development practices, including version control with Git, debugging techniques, unit testing, and documentation, to build robust and maintainable .NET applications. These outcomes aim to provide beginners with a foundational understanding of .NET programming technology and equip them with the skills needed to start developing simple applications using the .NET framework.

Mapping between Course Outcomes(CO) with Program Specific Outcomes(PSO)	

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8
CO1								
CO2								
CO3								
CO4								
CO5								

Course Content

Unit 1. Overview of Microsoft .NET Framework

1.1. The .NET Framework

1.1.1. Managed Code MSIL, Metadata and JIT Compilation - Automatic Memory Management.

1.2. The Common Language Runtime (CLR)

1.3. The .NET Framework class Library

Unit 2. Programming in Visual basic .net

2.1. IDE

2.2. Variables and Data Types

2.2.1. Boxing and Unboxing

2.2.2. Enumerations

2.2.3. Data Type Conversion Functions

2.2.4. Statements

2.3. String & Date Functions and Methods

2.4. Modules, Procedures and Functions

2.4.1. Passing variable number of arguments

2.4.2. Optional arguments

2.5. Using Arrays and Collections

2.6. Control Flow Statements

2.6.1. Conditional Statements

2.6.2. Loop Statements

2.6.3. MsgBox and InputBox

Unit 3. Introduction to Windows controls

3.1. Working with Tool Box Controls

3.1.1. Common controls - Label, Text Box, Button, Check Box, Radio Button, Date Time Picker, List Box, Combo box, Picture Box, Rich Text Box, Tree View, Tool Tip, Progress bar, Masked Text box, Notify Icon,Link Label, Checked List box

3.1.2. Container Controls

3.1.3. Data - Data Set, Data Grid

3.1.4. Component - Image list, error provider, Help provider,Timer

3.2. Working with Menus and Dialogue Boxes

3.3. Exception Handling

3.3.1. Structured Error Handling

3.3.2. Unstructured Error Handling

Unit 4. Object Oriented Programming

4.1. Creating Classes, Object Construction & Destruction

4.1.1. Properties, Methods, Events

4.1.2. Access Specifiers: Public, Private, Protected, ProtectedFriend

4.1.3. Me, MyBase and MyClass keywords

4.2 Abstraction, Encapsulation & Polymorphism

4.3 Interfaces & Inheritance

Unit 5. Database access using ADO.NET

5.1. Visual Database Tools

5.2. ADO .NET Object Model

5.3. ADO .NET Programming

Reference Books	1. Visual Basic .NET Programming (Black Book) - By Steven Son Holzner, DreamTech Publication 2. Mastering Visual Basic.NET by Evangelos Petroutsos BPB Publication 3. Moving to VB.NET: Strategies, Concepts, and Code - by Dan Appleman – Apress Publication 4. Microsoft Visual Basic .NET Step by Step - by Michael Halvorson,PHI Publication 5. Database Programming with Visual Basic.NET and ADO.NET - by F. Scott Barker – Sams Publication 6. Beginning .NET Web Services Using Visual Basic .NET - by JoeBustos and Karlli Watson, Wrox Publication 7. .NET – Complete Development Cycle - by G. Lenz, T. Moeller, Pearson Education. 8. Professional VB.NET, 2nd Edition - by Fred Barwell, et al – Wrox Publication
Teaching Methodology	Class Work, Discussion, Lab work, Self-Study, Seminars and/or Assignments
Evaluation Method	50% Internal assessment. 50% External assessment.

Course: 405-01: Web Designing-2

Course Code	405-01
Course Title	Web Designing-2
Credit	4
Course Category:	Major Course
Level of Course:	300- 399 (Higher Course)
Teaching per Week	4 Hrs (2 Hours Theory + 4 Hours of Lab. Work)
Minimum weeks per Semester	15 (Including class work, examination, preparation etc.)
Review / Revision	2023-2024
Implementation Year:	2024-2025
Purpose of Course	Web Design requires designers to create graphics, typography as well as images which are used only on the World Wide Web. While creating any design, web designers need to maintain balance between creating a good design as well as the speed and efficiency for the webpage/ website. This course deals with server-side communication.
Course Objective	To make students aware of web terminology and website designing tools. Student can understand and implement the real functions of website development.
Pre-requisite	Knowledge of HTML5, Bootstrap, JavaScript
Course outcome	**CO1:** Students will be able to create, organize and design websites. **CO2:** Students gain formal understanding of XML-based technologies which are used in Web-service. **CO3:** Students will be able to make dynamic changes to a web pages as well as respond to user and browser events through JQuery **CO4:** Students will be able to learn cross-browser supports via Ajax and Jason **CO5:** Students will be able to write asynchronous code using various techniques through Node.js

Mapping between Course Outcome(CO) and Program Specific Outcome (PSO):

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8
CO1								
CO2								
CO3								
CO4								
CO5								

Course Content

Unit-1 : Introduction of XML:

1.1 Characteristic and Use of XML
1.2 XML syntax (Declaration, Tags, elements)
1.3 root element, case sensitivity
1.4 XML document:
  1.4.1 Document Prolog Section
  1.4.2 Document element section
1.5 XML declaration and rules of declaration.

Unit-2: jQuery Fundamentals:

2.1 Introduction and basics:
  2.1.1 Advantage of jQuery and Syntax
  2.1.2 jQuery Selectors:
  2.1.3 jQuery Events (ready(),click(), keypress(),focus(),blur(),change())

	2.2 jQuery Effects: 2.2.1 Show/Hide, Fade, Slide, Stop, Chaining, Callback 2.3 jQuery Manipulation methods: 2.1.1 Get/Set methods (text(), attr(), html(), val()) 2.1.2 Insert methods: (append(), prepend(),text(), before(), after(), wrap()) 2.1.3 Remove element methods : (remove(),empty(),unwrap()) 2.3.4 Query Get and Set CSS properties using css() method. **Unit-3: JSON: (JavaScript Object Notation)** 3.1 Concept and Features of JSON 3.2 Similarities and difference among JSON and XML 3.3 JSON objects(with string and Numbers)) 3.4 JSON Arrays and their examples : 3.4.1 Array of string, Array of Numbers, Array of Booleans 3.4.2 Array of objects, Multi-Dimensional Arrays 3.4.3 JSON comments **Unit-4: AJAX (Asynchronous JavaScript and XML):** 4.1 Fundamentals of AJAX technology: 4.1.1 Difference between Synchronous and Asynchronous web application 4.1.2 XMLHttpRequest technology 4.2 XMLHttpRequest 4.2.1 Properties :(onReadyStateChange, readyState, responseText, responseXML) 4.2.2 XMLHttpRequest Methods : (Open(), send(), setRequestHeader()) 4.3 Working of AJAX and its architecture **Unit-5: Node.js :** 5.1 Concepts, working and Features 5.1.1 Downloading Node.js 5.2 Setting up Node.js server(HTTP server) 5.2.1 Installing on window 5.2.2 Components 5.2.2.1 Required modules, Create Server(http.createServer()) 5.2.2.2 Request and response 5.3 Built-in Modules 5.3.1 require() function 5.3.2 User defined module: create and include 5.3.3 HTTP module 5.4 Node.js as Web-server: 5.4.1 createServer() , writeHead() method 5.4.2 Reading Query String, Split Query String 5.5 File System Module: 5.5.1 Read Files (readFile()) 5.5.2 Create Files(appendFile(),open(),writeFile()) 5.5.3 Update Files(appendFile(),writeFile()) 5.5.4 Delete Files(unlink()) 5.5.5 Rename Files(rename())
Reference Books	1) JavaScript and JQuery (Interactive Front-End Web Development) by Jon Duckett 2) JavaScript and JQuery (The missing manual) by David Sawyer MCFarland 3) Essential ASP.NET Web Forms Development, Full Stack Programming with C#, SQL, Ajax, and JavaScript, Robert E. Beasley, Publisher: Apress

	4) Foundations of Ajax, Ryan Asleson, Schutla, Publisher: Apres 5) Ajax: The Complete Reference, By Thomas Powell, ISBN: 978-0-07-149216-4 6) Head First Ajax , Author: Rebecca M.Riordan, publisher: O'Reilly 7) Practical Node.js, Author: Azat Mardan,ISBN:978-1-4842-3038-1, Publisher: Apress 8) Node.JS Guidebook, BPB Publication, ISBN: 9789387284432, Author: Dhruti Shah. 9) JavaScript for Modern Web Development, ISBN: 9789389328721, eISBN: 9789389328738, Authors: Abhilasha Sinha, Ranjit Battewad, Alok Ranjan 10) Mastering HTML, CSS & Javascript Web Publishing, Authors:by Laura Lemay,Rafe Colburn, BPB Publication 11) JavaScript by Example, Author: Elitle Quigley, Publication: Prentice Hall, ISBN: 9780137054893, 9780137054893. 12) XML in easy steps, Publication: Tata McGraw Hill 13) XML crash course, Publisher: Tata McGraw Hill, ISBN: 9780071815161, 9780071815161 14) Beginning jQuery: From the Basics of jQuery to Writing your Own Plug-ins, by Jack Franklin Russ Ferguson,978-1484230268
Teaching Methodology	Class Work, Discussion, Self-Study, Seminars and/or Assignments
Evaluation Method	50% Internal assessment. 50% External assessment.

Course: 405-02: Mobile Application Development - 2

Course Code	**405-02**
Course Title	**Mobile Application Development – 2**
Credit	4
Course	Major Course
Level of Course	300 – 399 (Higher Course)
Teaching per Week	4 Hrs
Minimum weeks per Semester	15 (Including class work, examination, preparation etc.)
Review / Revision	2023-2024
Implementation From	2024-2025 A.Y.
Purpose of Course	Mobile application development is the process of creating software applications that run on a mobile device, and a typical mobile application utilizes a network connection to work with remote computing resources. Mobile device is used for different purposes ranging from email to online shopping and multiple apps for different reasons. Hence, the mobile development process involves creating installable software bundles, implementing backend services such as data access with an API, and testing the application on target devices. Knowledge about mobile application development on Android platform and gradually on hybrid platform is need of the current era.
Course Objective	1) To understand concepts of Mobile Technology 2) Understand the development process and have edge over mobile user interface (UI) design. 3) Understand various UI development tools, Application design interfaces and creating basic app on Android platform. 4) Concepts of DART and introduction of FLUTTER.
Pre-requisite	Basics of Mobile Application Development and designing concepts.
Course outcome	**CO1:** Students will be able to understand the internal concepts of Android. **CO2:** Students will have concepts of important Android Widgets(UI) **CO3:** To learn concepts of DART. **CO4:** To work on Flutter. **CO5:** To gain edge over Basic Flutter Widgets.
Mapping between Courses Outcome(CO) and Program Specific Outcome(PSO):	

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8
CO1								
CO2								
CO3								
CO4								
CO5								

Course Content	**Unit-1 : Basic Attributes and Events of Important Android Widgets(UI)** 1.1 ListView, Custom ListView 1.2 DatePicker, TimePicker, ProgressBar 1.3 Horizontal and Vertical ScrollView 1.4 AutoCompleteTextView, TextWatcher to EditText 1.5 ImageSlider, ImageSwitcher, SearchView 1.6 TAbLayout and FrameLayout

	Unit-2: Working with DART: 2.1 DART overview, concept, features and installation 2.2 Online editor DartPad and dart2js tool 2.3 Executing Dart basic code using Command line, DartPad and IDE 2.4 Understanding DART syntax: 2.4.1 Identifiers, Datatypes, variables, comments 2.4.2 Decision making (if, if..else, if..else if..., switch..case) 2.4.3 Iterative statements (for, for...in loop, while, do..while) 2.4.4 break, continue, label 2.5 DART function : 2.5.1 Calling function, deleting function 2.5.2 Passing arguments to function, lexical scoping. **Unit-3: Introduction of Flutter:** 3.1 Fundamentals of Flutter: 3.1.1 Installation and Architecture of Flutter 3.1.2 Features of Flutter 3.1.3 Creating basic flutter project using Android Studio 3.2 Flutter Widget: 3.2.1 Types of flutter widget: 3.2.1.1 Visible and Invisible 3.2.1.2 StatelessWidget, StatefulWidget 3.2.1.3 Single child widget and Multiple child widget **UNIT-4: Flutter basic widgets:** 4.1 Visible widget(Constructor and Properties): Text, Image, Button, Icon 4.2 Invisible widget(Constructor and Properties): column, row, center, padding, scaffold, stack 4.3Text, TextField 4.4 Buttons, Slider 4.5 Checkbox, RadioButton **Unit-5: Flutter widget (Constructor, attributes and Properties)** 5.1 Progress Bar, Stack 5.2 Lists 5.3 Alert Dialogbox , Tooltip 5.4 Toast, Switch 5.5 Charts, Flutter Form.
Reference Books	1) Android Application Development (With Kitkat Support), Author: Pradeep Kothari, Publisher: DreamTech Press.,ISBN:978-9351194095 2) Android Studio 3.0 Development Essentials: Android 8 Edition , Author: Neil Smyth, ISBN:978-1977540096 3) Flutter for Beginners: An introductory guide to building cross-platform mobile applications with Flutter and Dart 2, Author: Alessandro Biessek, Packt Publishing House,ISBN:978-1788996082 4) Beginning Flutter: A Hands On Guide to App Development, Author: Marco L. Napoli, Publisher: Wrox, ISBN:978-1119550822 5) Android Programming for Beginners - Second Edition, Author:John Horton, Publisher: Image Short ISBN: 978-1789538502 6) Android 9 Development Cookbook, Author: Rick Boyer, Publisher: Packet Publishing, ISBN:978-1788991216 7) The Dart Programming Language, Author:Bracha, Publisher:Pearson

	Education India, ISBN:978-9332570368 8) Google Flutter Mobile Development Quick Start Guide: Get up and running with iOS and Android mobile app development, Author: Prajyot Mainkar, Publication:Packt Publishing, ISBN:978-1789344967 9) Practical Flutter: Improve your Mobile Development with Google's Latest Open-Source SDK ,Author: Frank Zammetti, Publisher: Apress, ISBN:978-1484249710
Teaching Methodology	Class Work, Discussion, Self-Study, Seminars and/or Assignments
Evaluation Method	50% Internal assessment. 50% External assessment.

Course code: 406
Course Title: Skill Enhancement Course (SEC-04)

Course Code	406
Course Title	Skill Enhancement Course - IV (SEC – 04)
Credit	2
Category of Course	Skill Enhancement Course
Level of Course	200-299 (Intermediate)
Teaching per Week	2 Hrs (Any or Combination of Theory/Practical/Fieldwork/Internship/Project)
Minimum weeks per Semester	15 (Including class work, examination, preparation etc.)
Review / Revision	-
Implementation Year:	A.Y. 2024-2025
Purpose of Course	- As per NEP(National Education Policy-2020), it is mandatory for students to select a 2 credit Skill Enhancement Course out of the choices given by the college/institute. - It will be mandatory for the student to opt minimum one 2-credit Skill Enhancement Course from the course baskets of Skill Enhancement courses approved by the university or from any recognized MOOC or from recognised university through online mode subject to transfer of credit through ABC during semester-1 to semester-5. - The student can start an alternative career in the field by obtaining higher degree of knowledge in the area. - It's aimed at imparting practical skills, embedded internship, hands-on training, soft skills, life skills, such approved online courses etc. to enhance the employability of students. This may also include courses as per the need of new evolving technology.
Course Objective	Obtaining skill in particular field along with the regular curriculum of the selected program is essential. It not only enhance the skill but also provide an opportunity to develop skill in particular area where one can pursue career in future. Skill enhancement provides the opportunity and knowledge for an individual to develop and strengthen the necessary skills to gain, maintain, and advance in a chosen area. Skill enhancement programs are focused around training that combines the best practices from varieties of areas as described in NEP-2020 SOP by Gujarat State Higher education Department's SOP. Skill enhancement or training typically uses a combination of cognitive and behaviour problem solving approaches, both of which are used to strengthen a person's positive skill develop.
Pre-requisite	-
Course Content and Implementation road-map.	(i) University has categorised and prepared the basket of the courses including approved online courses that can be offered as Skill Enhancement Course. (ii) The institute/college/department can design and implement skill enhancement course by getting approval from the relevant apex body of the university considering the SOP of the certificate course policies of the University. (iii) The institutes/college/departments can select more than one course out of the given sets of courses and offer them to their students. (iv) The students can select any of the courses offered by the institute/college/department from the given choices and enrol for the course.

	(v) The institute/college/department will arrange appropriate resource person(s) for the course. (vi) The course evaluation will be taken place at the college/institute/department level based on the nature of the course. (vii) The institute/college/department will assess the student based on the nature of the course. The student will be granted the credits on successful completion of the course.
Reference Books	- The reference materials and books will be decided by the Institutes/Colleges/Departments based on the selected Courses. - Minimum five copies of relevant topics are recommended to keep in the library.
Teaching Methodology	Class Work/ Discussion/ Self-Study/ Seminars/ field works/ practical training/ field work and/or Assignments.
Evaluation Method	50% Internal assessment. 50% External assessment. (Evaluation and Assessment will be carried out based on the nature of the course. On successful completion of the course, the student will be granted 2 credits.)

Course code: 407
Course Title: Value Addition Course-IV (VAC-04)

Course Code	407
Course Title	Value Addition Course - IV (VAC – 04)
Credit	2
Category of Course	Value Addition Course
Level of Course	200-299 (Intermediate)
Teaching per Week	2 Hrs (Any or Combination of Theory/Practical/Fieldwork/Internship/Project)
Minimum weeks per Semester	15 (Including class work, examination, preparation etc.)
Review / Revision	-
Implementation Year:	A.Y. 2024-2025
Purpose of Course	As per NEP(National Education Policy-2020), it is mandatory for students to select a 2 credit Value Addition Course out of the choices given by the college/institute. It will be mandatory for the student to opt minimum one 2-credit Value Addition Course out of the list of offered courses recognised by the University during semester-1 to semester-4. The student can start an alternative career in the field by obtaining higher degree of knowledge in the area.
Course Objective	Obtaining knowledge in all or any of the components/fields like (i) Understanding India (ii) Environmental Science/Education (iii) Digital/Technological solutions (iv) Health & Wellness, Yoga education, sports, and fitness are essential for holistic development (v) Indian Knowledge system(IKS). The course components should be among these five categories/fields and as per the Curriculum and Credit Framework for Undergraduate Programmes of the UGC (Page-22 of the document). The purpose is to impart knowledge and understand the necessities of these aspects in life to make the healthy society and nation. It help in development of a dedicated and responsible citizen of the country by adding value to the life.
Pre-requisite	-
Course Content and Implementation road-map.	(i) The university has categorised and prepared the list of the courses that can be offered as Value Addition Course. (ii) The institute/college/department can design and implement skill enhancement course by getting approval from the relevant apex body of the university considering the SOP of the certificate course policies of the University. (iii) The institutes/college/departments can select more than one course out of the given sets of courses and offer them to their students. (iv) The students can select any of the courses offered by the institute/college/department from the given choices and enrol for the course. (v) The institute/college/department will arrange appropriate resource person(s) for the course. (vi) The evaluation will be taken place at the college/institute/department based on the nature of the course. (vii) The institute/college/department will assess the student based on the nature of the course. The student will be granted the credits on successful completion of the course.
Reference Books	- The reference materials and books will be decided by the Institutes/Colleges/Departments or as per the university guidelines based on the selected Courses. - Minimum five copies of relevant topics are recommended to keep in the library.

Teaching Methodology	Class Work/ Discussion/ Self-Study/ Seminars/ field works/ practical training/ field work and/or Assignments.
Evaluation Method	50% Internal assessment. 50% External assessment. Maximum Marks: 50 (Evaluation and Assessment will be carried out based on the nature of the course. On successful completion of the course, the student will be granted 2 credits.)

Internship: Student willing to exit the program at the end of the two semesters and to avail the Certificate in Computer Application or exit the program at the end of the first four semesters and to avail the Diploma in Computer Application, it is essential to acquire four credits from internship. A key aspect of the internship is induction into actual work situations. Internships involve working with local industry, government or private organizations, business organizations, artists, crafts persons, and similar entities to provide opportunities for students to actively engage in on-site experiential learning. In option to these internships, the student can avail such four credits by availing two 2-credit university approved courses during any of these semesters. The student is required to enroll and avail these 4-credits and produce the evidence in process to opt the multi-level exit option after successfully completion of first year (two semester) or second year(four semesters).