GUJARAT TECHNOLOGICAL UNIVERSITY

Bachelor of Engineering
Subject Code: 3174906
. Net Technology

Type of course: Professional Elective VI

Prerequisite: Concepts of Object oriented programming approach

Rationale: Object-oriented programming (OOP) has gained popularity due to its support for object reuse, enhancing code efficiency and maintainability. The .NET framework provides a powerful, object-oriented environment with a comprehensive Base Class Library (BCL) that supports web development tasks like graphic rendering, file handling, and data manipulation. .NET simplifies development with a consistent programming model, built-in security, and easy deployment and maintenance, making it ideal for both small and large-scale applications. It allows developers to focus on functionality while reducing complexity and development time.

Teaching and Examination Scheme:

Teaching Scheme			Credits	Examination Marks				Total Marks
				Theory Marks		Practical Marks		
L	T	P	C	ESE (E)	PA	ESE (V)	PA (I)	
3	0	2	4	70	30	30	20	150

Sr. No.	Content	Total Hrs.
1	**Introduction To .Net Architecture:** Overview of .NET framework, MSIL (Microsoft Intermediate Language), CLR (Common Language Runtime), CLS (Common Language Specification), CTS (Common Type System), Namespaces, Assemblies, Garbage Collection, Managed Execution, Side-by-Side Execution, Debugging, and the end of DLL Hell.	05
2	**Object Oriented Programming in C#:** Creating classes, declaring variables and methods, using access modifiers, constructors, abstract classes, partial classes, inheritance, method overloading, method overriding, anonymous methods, properties, indexers, and exception handling.	07
3	**Building GUI with C# and Database Connectivity using ADO.NET:** **GUI Development**: Working with C# Windows applications, common form controls, visual inheritance, creating MDI forms, and event handling. **ADO.NET**: Overview of ADO.NET framework, working with SQL Server database, managed providers, datasets, data sources, connected and disconnected architectures, binding data with DataGrid, and Crystal Reports.	08
4	**Web Development and ASP.NET Controls:** Understanding Web Server and HTTP/HTTPS protocols, ASP.NET benefits, ASP.NET page layout, lifecycle, HTML server controls, web server controls, validation controls, and introduction to AJAX for dynamic web functionality.	07

5	**Master Page, Theme, and State Management in ASP.NET:** Using Master Pages and Themes in ASP.NET for consistent layouts and styling. Different methods of state management in ASP.NET (view state, session state, cookies, query strings, and hidden fields). Creating and consuming web services in ASP.NET.	05
6	**Getting Started with ASP.NET MVC:** Introduction to MVC (Model-View-Controller) architecture. Understanding the components: Model, View, and Controller. Advantages of using MVC. Working with application configuration files in an ASP.NET MVC application.	06
7	**Basics of Cloud Computing:** Understanding the concept of cloud computing, cloud models, and the cloud-optimized stack. Introduction to Microsoft Azure and using C# to create a storage container in Azure. Developing an ASP.NET web application to interact with Azure storage.	04

Suggested Specification table with Marks (Theory):

Distribution of Theory Marks					
R Level	U Level	A Level	N Level	E Level	C Level
20	**30**	**25**	10	10	05

Legends: R: Remembrance; U: Understanding; A: Application, N: Analyze and E: Evaluate C: Create and above Levels (Revised Bloom's Taxonomy)

Note: This specification table shall be treated as a general guideline for students and teachers. The actual distribution of marks in the question paper may vary slightly from above table.

Reference Books:

1. Professional C# .Net, Christian Nagel, Wrox Publication
2. ASP.NET Complete Reference, Matthew Macdonald and Robert Standefer, TMH
3. C# The Basics, Vijay Mukhi, BPB Publications
4. Beginning C# and .NET, Benjamin Perkins & Jon D. Reid, Wrox publication.

List of Practical's / Tutorials:

1. Implement method overloading and overriding, as well as constructors and destructors in C#.
2. Develop a Windows Forms Application to implement an arithmetic calculator.
3. Create a Windows Forms application using controls like menus, dialogs, tooltips, dropdowns, radio buttons, and selection buttons.
4. Implement inheritance, visual inheritance, and interfaces in a Windows Forms application.
5. Use Dataset, DataReader, XMLReader, and various data sources (SQL, Object, and XML) in both Windows and Web applications.
6. Utilize data controls like DataList, GridView, DetailsView, Repeater, and ListBound controls in a Windows/Web application.
7. Develop an ASP.NET web application using web controls and validation controls.
8. Create a web application illustrating the use of themes, master pages, and site maps.
9. Implement state management techniques (cookies, sessions, etc.) in a web application.
10. Write code in ASP.NET to create and consume a web service within any web application.

11. Develop an ASP.NET web application using the MVC (Model-View-Controller) architecture.
12. Create a Microsoft Azure storage container and develop an ASP.NET application to consume it.

Mini Project: Build a mini project (either Desktop or Web-based) that demonstrates practical use of learned concepts.

Course Outcome: After learning the course the students should be able to:

Sr. No.	CO Statement	Marks % Weightage
CO-1	Understand the core concepts of C# and the .NET Framework, including the contents and organization of namespaces.	15
CO-2	Develop and implement console and GUI applications using C# and .NET technologies.	20
CO-3	Apply various navigation techniques for integrating web pages within a site.	15
CO-4	Design dynamic web pages using ASP.NET controls that interact with databases.	25
CO-5	Implement state management techniques such as cookies and sessions in web applications. Develop and deploy basic cloud-based applications.	25

List of Open-Source Software/learning website:

http://www.c-sharpcorner.com
https://www.w3schools.com/
http://www.csharphelp.com/index.html
http://www.codeproject.com
https://msdn.microsoft.com
https://github.com/