NET Centric Computing

Course Title: NET Centric Computing   **Full Marks:** 60 + 20 + 20
Course No: CSC378   **Pass Marks:** 24 + 8 + 8
Nature of the Course: Theory + Lab   **Credit Hrs:** 3
Semester: VI

Course Description:
The course covers the concepts of cross-platform web application development using the ASP.NET Core MVC framework using C# programming Language.

Course Objectives:
The objective of this course is to understand the theoretical foundation as well as its practical aspects of ASP.NET Core web application framework and C# language features.

Course Contents:
Unit 1: Language Preliminaries (8 Hrs.)
Introduction to .Net framework, Compilation and execution of .Net applications, Basic Languages constructs, Constructor, Properties, Arrays and String, Indexers, Inheritance, use of "base" keyword, Method hiding and overriding, applying polymorphism in code extensibility, structs and enums, abstract class sealed class, interface, Delegate and Events, Partial class, Collections, Generics, File IO, LINQ (Language Integrated Query) Fundamentals: Lambda Expressions, Try statements and Exceptions, Attributes: Attribute Classes, Named and Positional Attribute Parameters, Attribute Targets, Specifying Multiple Attributes, Asynchronous Programming: Principle of Asynchrony, Async/Await patterns in C#

Unit 2: Introduction to ASP.NET (3 Hrs.)
.NET and ASP.NET frameworks: .NET, .NET Core, Mono, ASP.NET Web Forms, ASP.NET MVC, ASP.NET Web API, ASP.NET Core, .NET Architecture and Design Principles, Compilation and Execution of .NET applications: CLI, MSIL and CLR, .NET Core in detail, .NET CLI: build, run, test and deploy .NET Core Applications

Unit 3: HTTP and ASP.NET Core (3 Hrs.)
HTTP, Request and Response Message Format, Common web application architectures, MVC Pattern, ASP.NET Core Architecture Overview, Projects, and Conventions, ASP.NET and ASP.NET MVC

Unit 4: Creating ASP.NET core MVC applications (10 Hrs.)
Setting up the Environment, Controllers and Actions: Create Controllers, Create Actions and Action Results Types, Rendering HTML with Views: Razor Syntax, Understanding Tag Helpers, Models: Binding and Validations, URL Routing and features, Web API Applications: API Controllers, JSON, Dependency Injection and IOC containers

Unit 5: Working with Database (6 Hrs.)
ADO.NET basics: Connection, Command, Reader and Adapter classes, Entity Framework (EF) Core, Object-Relational Mapper (ORM), Adding EF Core to an application: Choosing database provider, data models and data context, Querying and Saving data to database: Create, read, update and delete records

Unit 6: State Management on ASP.NET Core Application (4 Hrs.)
State Management on stateless HTTP, Server-side strategies: Session State, TempData, Using HttpContext, Cache Client-side strategies: Cookies, Query Strings, Hidden Fields

Unit 7: Client-side Development in ASP.NET Core (4 Hrs.)
Common client-side web technologies, JQuery, Forms and Validation, Single Page Application (SPA) Frameworks: Angular, React

Unit 8: Securing in ASP.NET Core Application (5 Hrs.)
Authentication: ASP.NET Core Identity, Adding authentication to apps and identity service configurations, Authorization: Roles, Claims and Policies, Securing Controllers and Action Methods, Common Vulnerabilities: Cross-site Scripting attacks, SQL Injection attacks, Cross -site Request Forgery (CSRF), Open Redirect Attacks

Unit 9: Hosting and Deploying ASP.NET Core Application (2 Hrs.)
App Servers and Hosting models: IIS, Nginx, Apache, ASP.NET Core Module, Kestrel, Docker and Containerization, Publish to Azure cloud

Laboratory works:
The laboratory work includes writing programs covering most of the concepts of above units using C# and .NET core SDK (3.0 or above)

Text / Reference Books:
1. C# 8.0 and .NET Core 3.0 – Modern Cross-Platform Development, Fourth Edition, by Mark J. Price, 2019
2. ASP.NET Core in Action, by Andrew Lock, 2018
3. Learning ASP.NET Core 2.0, Michel Bruchet, Jason De Oliveira, 2017