Tribhuvan University
Faculty of Humanities and Social Science (FoHSS)
Micro-Syllabus

Course Title	**: DotNet Technology (3 Cr.)**
Course Code	**: CACS302**
Nature of Course	**: Theory + Practical**
Year/Semester	**: III/V**
Marks	**: 60+20+20 [24 + 8 + 8]**
Class Load	**: 6 Hrs. /Week (Theory: 3hrs. Practical: 3 Hrs.)**

Course Description:
This course covers different concepts of .NET framework. It also covers basic to advanced features of C# language including language basics, creating types and inheritance, delegates, events, lambda expressions, LINQ, working with databases, and developing web applications using ASP.NET.

Course Objectives:
The primary objective of this course is to provide concepts of .NET framework and different concepts of C# programming language and make students familiar with their uses and applications.

Course Contents

Unit 1: Introducing C# and the .NET framework **7 Hrs.**

Object Orientation; Type Safety; Memory Management; Platform Support; C# and CLR; CLR and .NET Framework; Other Frameworks; Framework Overview; .NET Standard 2.0; Applied Technologies

---

- Introduction to C# Language
- Object Orientation
  - Unified Type System
  - Classes and Interfaces
  - Properties, methods and events
  - Functions can be treated as values
  - C# supports patterns for purity
- Type Safety
- Memory Management
- Platform Support
- C# and CLR
- Introduction and Overview of .NET Framework
- Common Language Runtime (CLR)
  - Features of Common Language Runtime
- Framework Class Library (FCL)
  - Features of Framework Class Library
- Other Frameworks

- Universal Windows Platform (UWP)
- .NET Core with ASP.NET Core
- Xamarin

- Framework Overview
- .Net Standard 2.0
  - Overview
  - What is new in the .NET Standard 2.0?
  - Older .NET Standards
- The CLR and Core Framework
  - System Types
  - Text Processing
  - Collections
  - Queries
  - XML
  - Diagnostics
  - Concurrency and Asynchrony
  - Streams and I/O
  - Networking
  - Serialization
  - Assembled, Reflection and Attributes
  - Dynamic Programming
  - Security
  - Advanced Threading
- Applied Technologies
  - Descriptions of .NET Implementations
  - User-Interface APIs
  - ASP.NET
  - ASP.NET Core
  - Windows Presentation Foundation (WPF)
  - Windows Forms
  - Xamarin
  - Universal Windows Platform (UWP)
  - Silverlight
  - Backend Technologies
    - ADO.NET
      - Provider Layer
      - DataSet Model
      - LINQ to SQL
      - Entity Framework Core (EF Core)
    - Windows Workflow (.Net Framework only)
    - COM+ and MSMQ (.NET Framework only)
  - Distributed System Technologies
    - Windows Communication Foundation (WCF)
    - Web API
    - Remoting and .ASMX Web Services(.NET Framework Only)
  - Scope of .Net Technology

- Features of Object Oriented Programming
- Procedure-Oriented Vs. Object-Oriented Programming

Unit 2: The C# Language Basics **12 Hrs.**

Writing Console and GUI Applications; Identifiers and Keywords; Writing Comments; Data Types; Expressions and Operators; Strings and Characters; Arrays; Variables and Parameters; Statements (Declaration, Expression, Selection, Iteration, and Jump Statements); Namespaces

---

- A First C# Program
- Compilation
- Identifiers and Keywords
- Avoiding Conflicts
- Contextual Keywords
- Literals, Punctuators, and Operators
- Comments
- Type Basics
  - Predefined Type Examples
  - Custom Type Examples
- Conversions
- Value Types Versus Reference Types
  - Value Types
  - Reference Types
  - Generic Type Parameters
  - Pointer Types
- Numeric Types
- Numeric Conversions
  - Converting between floating-point types
  - Converting between floating-point and integral types
- Decimal Conversions
- Operators in C#
  - Arithmetic Operators
  - Relational Operators
  - Logical Operators
  - Bitwise Operators
  - Assignment Operators
  - Miscellaneous Operators
- Conditional Operators (Ternary)
- Strings and Characters
- Char Conversions
- String Type
  - String Concatenation
  - String Interpolation
  - String Comparisons
- Arrays

  - Multidimensional Arrays
  - Rectangular Arrays
  - Jagged Arrays
  - Simplified Array Initialization Expressions
  - Bounds Checking
- Variable and Parameters
- The Stack and the Heap
  - Stack
  - Heap
- Definite Assignment
- Default Values
- Parameters
  - Passing arguments by value
  - The ref modifier
  - The out modifier
  - The params modifier
  - Optional Parameters
- Operator Precedence and Associativity
  - Left-associative Operators
  - Right-associative Operators
- Null Operators
  - Null Coalescing Operator
  - Null-Conditional Operator
- Statements
  - Declaration Statements
  - Local Variables
  - Expression Statements
  - Conditional Operator
  - Control Statements
    - C#'s Selection Statements
    - If
    - Nested ifs
    - The if-else-if Ladder
    - Switch
  - Iteration Statements
    - For loop
    - While loop
    - Do while loop
    - For each loop
    - Nested Loop
  - Jump Statements
    - Using break
    - Using break to exit a Loop
    - Using Continue
    - Using return
    - The goto statements
    - The throw statements
- Namespaces

- Using Directive
- Using static
- Rules within a namespace
    - Name Scoping
    - Name Hiding
    - Repeated namespaces
    - Nested using directive
- Advanced Namespace Features
    - Extern

<u>Unit 3:</u> Creating Types in C# **12 Hrs**

Classes; Constructors and Deconstructors; this Reference; Properties; Indexers; Static Constructors and Classes; Finalizers; Dynamic Binding; Operator Overloading; Inheritance; Abstract Classes and Methods; base Keyword; Overloading; Object Type; Structs; Access Modifiers; Interfaces; Enums; Generics

---

- Classes
- Object
- Fields
    - The read-only modifier
    -
- Methods
    - Expression-bodied methods
    - Overloading methods
- Constructors
    - Default Constructor
    - Instance Constructors
    - Overloaded Constructors
- Destructor
    - Static Constructor
- this reference
- Properties
    - Automatic Properties
- Indexers
- Difference between Indexers and Properties
- Static Classes
- Static Members of a class
- Finalizers
- Inheritance
    - Terms used in Inheritance
    - Syntax
    - Types of Inheritance in C#
        - Single Inheritance
        - Multilevel Inheritance
        - Hierarchical Inheritance
        - Multiple Inheritance

- Interface in C#
- Abstract Classes
- Abstract Members
- Polymorphism
- Method Overloading
  - Number of parameters
  - Data type of parameters
  - Sequence of data type of parameters.
- Method Overriding
- Virtual Method
  - Features of Virtual Method
- Upcasting and Downcasting
- The as Operator
- The is Operator
- Operator Overloading
  - Overloading Unary Operators
  - Overloading Binary Operators
- Sealing Functions and Classes
  - C# Sealed Class
  - Sealed Method and Properties
- The base Keyword
- The object type
- Boxing and Unboxing
- The GetType Method and typeof Operator
- Structs
- Access Modifiers
  - Restrictions on Access Modifiers
- Enums in C#
- Generics
  - Generic Classes
  - Generic Methods
- Dictionary
- Queues
- Stacks
- List
- Array List

Unit 4: Advanced C# **14 Hrs.**

Delegates, Events, Lambda Expressions, Exception Handling, Introduction to LINQ, Working with Databases; Web Applications using ASP.NET

- Delegates
  - Declaring Delegates
  - Instantiation and Invocation of Delegates
  - Multicast Delegates

  - Delegates Mapping with Instance and Static Method
  - Delegates Vs. Interfaces in C#
  - Delegate Compatibility
    - Type Compatibility
    - Parameter Compatibility
    - Return type Compatibility
  - Generic Delegate Types
  - Func and Action Delegates
- Events
  - Declaring Events
  - Implementing Event in a Button Click
- Anonymous Method in C#
- Lambda Expressions
- Exception Handling
  - The catch clause
  - The finally Block
  - Throwing Exception
  - Re-throwing Exception
  - Common Exception Types
- Introduction to LINQ
  - LINQ Method
  - Use of Lambda Expression in LINQ
  - LINQ Operators
  - Various Examples
- Working with Databases
  - Comparison between ADO and ADO.NET
  - Working with Connection, Command
  - DataReader, DataAdaper, Dataset and Datatable
  - Connect C# to MYSQL
  - Complete CRUD Operations
- Writing Windows Form Application
  - Introduction to Win Forms
  - Basic Controls
- Web Applications using ASP.NET
  - Elements of ASP.NET Web Applications
  - Different types of form controls in ASP.NET
  - Launch Another form on button click
  - Validation Controls in ASP.NET

Laboratory Works

The laboratory work includes writing console and/or GUI programs in C#

- To implement basic language features.
- To create classes and objects and to implement different object oriented features.
- To implement inheritance.
- To implement advanced features like delegates, event handling, lambda expressions, exception handling.
- To implement LINQ and database applications

Text Books

1. *C# 7.0 in a Nutshel (7th Edition), the Definitive Reference*, Joseph Albahari & Ben Albhari, O'Reilly.
2. *Microsoft Visual C# Step by* Step (9th Edition), John Sharp, Pearson Education.

Reference Books

1. *C# 7.0 All-in-One For Dummies (1st Edition)*, John Paul Mueller, Bill Sempf, Chuck Sphar, John Wiley & Sons, Inc.
2. *Professional C# 7 and .NET Core 2.0 (7th Edition)*, Christian Nagel, John Wiley & Sons, Inc.