MAKERERE UNIVERSITY BUSINESS SCHOOL

Bachelor of Business Computing

BUC 2227: Business Application Programming

Course Description

- This course aims to improve students programming skills through application and extension of programming theory and problem solving skills within the business domain.
- The course will reinforce the Object-oriented programming concepts and constructs to analyze, design, and implement Windows-based business applications that fully utilize the Graphical User Interface tools and techniques.
- Develop a fully functioning enterprise information system that utilizes advanced programming techniques for interacting with the database.
- The .NET environment and MySQL database server will be used in this period of study.

Objectives of the course

- To equip learners with problem-solving skills and logic to solve business and education applications.
- To provide students with the knowledge needed to plan graphical-user business applications using pseudo code, flowcharts, and/or IPO (Input-Process-Output) charts to produce user-friendly computer programs.
- To introduce students to the understanding of OOP analysis, design issues and techniques to evaluate business applications both from the standpoint of the developer and the business user
- To introduce students to understanding the debugging techniques and program errors in business applications.

Learning Outcomes

Upon completion of this course, a student will be able to:

- Use problem-solving and logic skills to solve business and education applications.
- Plan graphical-user business applications using pseudo code, flowcharts, and/or IPO charts to produce user-friendly computer programs.
- Apply the knowledge of OOP analysis, design issues and techniques in evaluating business applications both from the standpoint of the developer and a business user.

Detailed Course outline

Detailed Course outline …

- **Mode of delivery**
  - Lectures (face to face and online)
  - Practical
  - Tutorial
- **Mode of assessment**
  - Course work 30%
    - Test
    - Project work
  - End of semester examination 70%

MAKERERE UNIVERSITY BUSINESS SCHOOL

Business Application Programming

TOPIC 1

An Introduction to .NET development environment

BUC 2227

Topic 1 break down

- An Introduction to .NET
- An Introduction to Visual Studio Environment
- Creating Applications & Writing Code
- Object-Oriented Application Development using .NET
- Visual program design and development process

The .NET Framework: Overview

- The .NET Framework is a robust and versatile software development platform created by Microsoft.
- It provides a comprehensive environment for building, deploying, and running applications, particularly on Windows.
- It supports multiple programming languages, including Visual Basic .NET (VB.NET), C#, and F#.

Key Features of the .NET Framework

1. Common Language Runtime (CLR)

- It is the runtime environment of the .NET Framework. It manages the execution of .NET programs by providing:
  - Memory management.
  - Garbage collection.
  - Exception handling.
  - Security.

Key Features of the .NET Framework…

- **2. Base Class Library (BCL)**
- This is a collection of reusable classes, interfaces, and value types. It simplifies development by providing pre-built functionality for:
  - File I/O.
  - Data structures.
  - Networking.
  - Database connectivity.

Key Features of the .NET Framework…

- **3. Language Interoperability:** The .NET Framework supports multiple programming languages. Code written in VB.NET can interact seamlessly with code written in other .NET-supported languages like C#.
- **4. Windows Forms and WPF**
- **Windows Forms:** For building desktop applications with GUIs.
- **Windows Presentation Foundation (WPF):** For creating rich and interactive UIs.

Key Features of the .NET Framework…

- **5. ADO.NET:** A powerful library for database connectivity and operations. It allows VB.NET applications to interact with relational databases like SQL Server, MySQL, or Oracle.
- **6. ASP.NET:** A framework for building web applications and services.
- **7. Deployment and Versioning**
- Simplifies the deployment of applications through tools like ClickOnce and support for side-by-side versioning.

Why .NET Framework for Business Application Programming?

1. Ease of Development

- The .NET Framework provides a rich development environment with extensive libraries, making it easier to develop business applications efficiently.

2. Scalability

- Applications built on the .NET Framework can scale from small utilities to enterprise-level solutions, supporting the growth of business needs.

Why .NET Framework for Business Application Programming?

3. Rapid Application Development

- With tools like Visual Studio and the extensive support of the BCL, developers can quickly create prototypes and fully functional applications.

4. Database Integration

- ADO.NET simplifies the integration of VB.NET applications with business databases, enabling the creation of data-driven applications such as inventory management or payroll systems.

Why .NET Framework for Business Application Programming?

5. Cross-Language Integration

- The .NET Framework allows you to integrate components written in other .NET-supported languages, offering flexibility in building complex systems.

6. Support for Windows Applications

- Since most businesses use Windows-based systems, the .NET Framework is ideal for developing applications that run seamlessly on these platforms.

Why .NET Framework for Business Application Programming?

7. Security

- Built-in features like code access security (CAS) and role-based security make .NET applications reliable for handling sensitive business data.

How .Net will be used in the course

- **Practical Application Development**
  - You will learn to create practical business applications like:
    - Accounting systems.
    - Inventory management tools.
    - Employee tracking systems.
- **Database Connectivity**
  - Using ADO.NET, you can develop applications that interact with databases, a crucial skill for modern business programming.

How .Net will be used in the course

- **Graphical User Interface (GUI) Design**
  - Through Windows Forms, you will be able to build user-friendly interfaces for business tools.
- **Problem-Solving**
  - The structured approach of VB.NET combined with .NET libraries will help you develop problem-solving skills by building real-world applications.
- **Career-Ready Skills:** Knowledge of the .NET Framework is highly sought after in industries, preparing you for roles in software dev't, especially for Windows-based applications.

Introduction to Visual Studio Development Environment

- **Visual Studio** is an Integrated Development Environment (IDE) developed by Microsoft.
- It is widely used for creating applications across multiple platforms, including desktop, web, mobile, and cloud. It supports various programming languages, including VB.NET, C#, Python, and more.

Key Features of Visual Studio

1. Code Editor

- Provides syntax highlighting, code completion (IntelliSense), and error detection.
- Offers features like code snippets to speed up coding.

2. Debugger

- A powerful tool for identifying and fixing errors in code.
- Allows step-by-step execution, variable inspection, and breakpoints.

Key Features of Visual Studio...

3. Project Templates

- Predefined templates for different types of applications (e.g., Console Application, Windows Forms, WPF, ASP.NET).

4. Solution Explorer

- Organizes files and resources in a hierarchical view.
- Provides quick access to project components.

Key Features of Visual Studio…

5. Toolbox

- Contains controls and components (e.g., buttons, text boxes) for designing user interfaces.

6. Designer View

- A drag-and-drop interface for designing application GUIs.
- Includes Windows Forms Designer and WPF Designer.

Key Components of the Visual Studio Interface

Component	Description
Menu Bar	Provides access to commands like File, Edit, View, and Debug.
Toolbar	Offers quick access to common actions like saving files or starting debugging.
Solution Explorer	Displays the structure of your project and its files.
Code Editor	The main area where you write and edit code.
Properties Window	Displays properties of selected items (e.g., controls on a form).
Toolbox	Contains controls for designing GUIs.
Output Window	Displays build, debug, and error messages.
Error List	Shows errors, warnings, and suggestions in your code.
Debug Toolbar	Provides tools for controlling debugging (e.g., stepping through code).

Practicals (Lab Sessions)