# COURSE OUTLINE: CSD207 - INTRO C# .NET APP

Prepared: Bazlur Rasheed
Approved: Corey Meunier, Chair, Technology and Skilled Trades

| | |
|---|---|
| **Course Code: Title** | CSD207: INTRO TO C# & .NET DESKTOP APPLICATIONS |
| **Program Number: Name** | 2090: COMPUTER PROGRAMMER |
| **Department:** | COMPUTER STUDIES |
| **Semesters/Terms:** | 20F |
| **Course Description:** | This course introduces the student to the C# programming language and the .NET framework. Students will design, develop, test and debug applications demonstrating practical knowledge of C# language constructs and the .NET framework and libraries. Desktop applications including Windows Forms and console based applications will be written in the Visual Studio Integrated Development environment.<br>Students will write applications that build on concepts and language constructs developed in this and other courses including structured programming techniques, basic language syntax, data types, file I/O, variable scope, arrays, collection classes, references, sequence, selection, repetition and object oriented programming techniques such as encapsulation, inheritance, polymorphism and UML syntax.<br>This is a lab oriented course with emphasis on practical hands on exercises. Students will be introduced to and gain practical knowledge in the use of git, git clients and cloud based repositories. |
| **Total Credits:** | 4 |
| **Hours/Week:** | 4 |
| **Total Hours:** | 60 |
| **Prerequisites:** | CSD102 |
| **Corequisites:** | There are no co-requisites for this course. |
| **This course is a pre-requisite for:** | CSD331 |
| **Vocational Learning Outcomes (VLO's) addressed in this course:**<br>**Please refer to program web page for a complete listing of program outcomes where applicable.** | **2090 - COMPUTER PROGRAMMER**<br>VLO 1 Identify, analyze, develop, implement, verify and document the requirements for a computing environment.<br>VLO 5 Communicate and collaborate with team members and stakeholders to ensure effective working relationships.<br>VLO 6 Select and apply strategies for personal and professional development to enhance work performance.<br>VLO 9 Support the analysis and definition of software system specifications based on functional and non-functional requirements.<br>VLO 10 Cntribute to the development, documentation, implementation, maintenance and testing of software systems by using industry standard software development methodologies based on defined specifications and existing technologies/frameworks. |

In response to public health requirements pertaining to the COVID19 pandemic, course delivery and assessment traditionally delivered in-class, may occur remotely either in whole or in part in the 2020-2021 academic year.

| | | |
|---|---|---|
| | VLO 11 | Apply one or more programming paradigms such as, object-oriented, structured or functional programming, and design principles, as well as documented requirements, to the software development process. |
| **Essential Employability Skills (EES) addressed in this course:** | EES 1 | Communicate clearly, concisely and correctly in the written, spoken, and visual form that fulfills the purpose and meets the needs of the audience. |
| | EES 2 | Respond to written, spoken, or visual messages in a manner that ensures effective communication. |
| | EES 3 | Execute mathematical operations accurately. |
| | EES 4 | Apply a systematic approach to solve problems. |
| | EES 5 | Use a variety of thinking skills to anticipate and solve problems. |
| | EES 6 | Locate, select, organize, and document information using appropriate technology and information systems. |
| | EES 7 | Analyze, evaluate, and apply relevant information from a variety of sources. |
| | EES 8 | Show respect for the diverse opinions, values, belief systems, and contributions of others. |
| | EES 9 | Interact with others in groups or teams that contribute to effective working relationships and the achievement of goals. |
| | EES 10 | Manage the use of time and other resources to complete projects. |
| | EES 11 | Take responsibility for ones own actions, decisions, and consequences. |
| **Course Evaluation:** | Passing Grade: 50%, D<br><br>A minimum program GPA of 2.0 or higher where program specific standards exist is required for graduation. | |
| **Other Course Evaluation & Assessment Requirements:** | To successfully pass this course, the student must receive passing grades for both the Test and Evaluation portion of the class and the Laboratory portion.<br><br>Grade<br>Definition Grade Point Equivalent<br>A+ 90 - 100% 4.00<br>A 80 - 89%<br>B 70 - 79% 3.00<br>C 60 - 69% 2.00<br>D 50 - 59% 1.00<br>F (Fail) 49% and below 0.00<br><br>CR (Credit) Credit for diploma requirements has been awarded.<br>S Satisfactory achievement in field /clinical placement or non-graded subject area.<br>U Unsatisfactory achievement in field/clinical placement or non-graded subject area.<br>X A temporary grade limited to situations with extenuating circumstances giving a student additional time to complete the requirements for a course.<br>NR Grade not reported to Registrar`s office.<br>W Student has withdrawn from the course without academic penalty. | |
| **Books and Required Resources:** | Beginning C# 7 Programming with Visual Studio 2017 by Benjamin Perkins, Jacob Vibe Hammer and Jon D. Reid<br>Publisher: John Wiley & Sons, Inc., Wrox a Wiley Brand. Edition: 7th Edition | |

In response to public health requirements pertaining to the COVID19 pandemic, course delivery and assessment traditionally delivered in-class, may occur remotely either in whole or in part in the 2020-2021 academic year.

ISBN: 978-1119458685, 1119458684

**Course Outcomes and Learning Objectives:**

| **Course Outcome 1** | **Learning Objectives for Course Outcome 1** |
|---|---|
| Introduction to Visual C# and .NET Framework | 1.1 What is C#?<br>1.2 What is the .NET Framework? |
| **Course Outcome 2** | **Learning Objectives for Course Outcome 2** |
| Introduction to Visual Studio and Visual Programming | 2.1 The Visual Studio 2017 Integrated Development Environment<br>2.2 Navigating the Visual Studio IDE<br>2.3 Create a simple application (Command Line Programming)<br>2.4 Create a simple application (Visual Programming) |
| **Course Outcome 3** | **Learning Objectives for Course Outcome 3** |
| Introduction to C# Console and Windows Application Programming | 3.1 Console Applications<br>3.2 Desktop Applications<br>3.3 Web Applications |
| **Course Outcome 4** | **Learning Objectives for Course Outcome 4** |
| Variables and Expressions | 4.1 Basic C# Syntax<br>4.2 Basic C# Console Application Structure<br>4.3 Variables<br>4.4 Expressions |
| **Course Outcome 5** | **Learning Objectives for Course Outcome 5** |
| Flow Control | 5.1 Boolean Logic<br>5.2 Branching<br>5.3 Looping |
| **Course Outcome 6** | **Learning Objectives for Course Outcome 6** |
| More about Variables | 6.1 Type Conversion<br>6.2 Complex Variable Types<br>6.3 String Manipulation |
| **Course Outcome 7** | **Learning Objectives for Course Outcome 7** |
| Functions | 7.1 Defining and Using Functions<br>7.2 Variable Scope<br>7.3 The Main() Function<br>7.4 Struct Functions<br>7.5 Overloading Functions |
| **Course Outcome 8** | **Learning Objectives for Course Outcome 8** |
| Debugging and Error Handling | 8.1 Debugging in Visual Studio<br>8.2 Error Handling |
| **Course Outcome 9** | **Learning Objectives for Course Outcome 9** |
| Introduction to Object-Oriented Programming | 9.1 What Is Object-Oriented Programming?<br>9.2 OOP Techniques<br>9.3 OOP in Desktop Applications<br>9.4 Defining Classes<br>9.5 Class Definitions in C# |

| | |
|---|---|
| | 9.6 System.Object<br>9.7 Constructors and Destructors<br>9.8 OOP Tools in Visual Studio<br>9.9 Class Library Projects<br>9.10 Interfaces Versus Abstract Classes<br>9.11 Struct Types<br>9.12 Shallow Copying Versus Deep Copying |
| **Course Outcome 10** | **Learning Objectives for Course Outcome 10** |
| Defining Class Members | 10.1 Member Definitions<br>10.2 Additional Class Member Topics<br>10.3 Interface Implementation<br>10.4 Partial Class Definitions<br>10.5 Partial Method Definitions |
| **Course Outcome 11** | **Learning Objectives for Course Outcome 11** |
| Class Designer in Visual Studio using Unified Modeling Language (UML) Syntax | 11.1 Understand the physical structure of the software objects and their relationships<br>11.2 How to design UML class diagram<br>11.3 Generate C# code using visual studio |
| **Course Outcome 12** | **Learning Objectives for Course Outcome 12** |
| Basic Desktop Programming | 12.1 Windows Forms<br>12.2 Windows Presentation Foundations (WPF)<br>12.3 Events<br>12.4 Controls (Forms, Labels, TextBox, buttons, etc.) Properties and Layout<br>12.5 GroupBoxes and Panels<br>12.6 Checkboxes and RadioButtons<br>12.7 PictureBoxes |
| **Course Outcome 13** | **Learning Objectives for Course Outcome 13** |
| Advanced Desktop Programming | 13.1 Menus<br>13.2 MonthCalendar control<br>13.3 DateTimePicker control |
| **Course Outcome 14** | **Learning Objectives for Course Outcome 14** |
| Git, Git clients and cloud based repositories | 14.1 Git a version control system (VCS)<br>14.2 Git Basics<br>14.3 Git Clients<br>14.4 GitHub a Web-based Git |

**Evaluation Process and Grading System:**

| Evaluation Type | Evaluation Weight |
|---|---|
| Lab Assignments | 30% |
| Lecture Assignments and Attendance | 10% |
| Quizzes | 10% |
| Theory and Lab Tests | 50% |

**Date:** July 6, 2020

In response to public health requirements pertaining to the COVID19 pandemic, course delivery and assessment traditionally delivered in-class, may occur remotely either in whole or in part in the 2020-2021 academic year.

| | |
|---|---|
| **Addendum:** | Please refer to the course outline addendum on the Learning Management System for further information. |