A POLYTECHNIC INSTITUTION

School of Computing
Program: Part-time Studies

Course Outline
COMP 2691

Intermediate Windows Application Development in .NET

Start Date:	April 14, 2009					**End Date:**	June 30, 2009
Total Hours:	36	**Total Weeks:**	12			**Term/Level:**	2 **Course Credits:** 3
Hours/Week:	3	**Lecture:**	1.5	**Lab:**	1.5		

Prerequisite:

COMP 2690 Windows Application Development in .NET

COMP 2691 is a Prerequisite for:

COMP 3691 Advanced Windows Application Development in .NET

■ Course Description

This hands-on intermediate level course follows-on from COMP 2690. Participants will use **MS Visual Studio 2008** in the **.NET Framework 3.5** to create rich Windows Forms programs with advanced windows controls. Students learn Microsoft industry standards and are exposed to .NET features such as Delegates, Attributes and Reflection. Topics include: I/O System, Object Serialization, Common Windows Controls, Advanced Windows Controls, Programming with ADO.NET, Multithreading, Asynchronous Programming Techniques, COM Interop, XML.NET Document Object Model, XML.NET Query Language, XML.NET Navigation, and Reflection.
Upon successful completion participants will be able to design, debug and deploy stand-alone Windows applications. This course, along with COMP 2690, covers most of the topics from the Microsoft .NET Framework exam 70-536, plus topics from MCAD exams (3 Credits).

Prerequisite: COMP 2690.

■ Evaluation

Lab Participation	10%
Assignments	30%
Midterm Exam	25%
Final Exam	35%
TOTAL	100%

Comments: In order to pass the course

1) your overall course mark must be at least **60%**

and

2) the average between the midterm and final exam mark must be at least **50%**

■ Course Learning Outcomes/Competencies

Upon successful completion, the student will be able to:

- Work with Input and Output (I/O) System
- Serialize and Deserialize Objects in .NET Framework
- Explain Simple Delegates, Multi-Cast Delegates, and Asynchronous Delegates
- Publish and Subscribe to Events
- Use Common Windows Controls such as LinkLabel, Scrollbar, Provider, MonthCalendar, DateTimePicker, DomainUpDown, NumericUpDown, TrackBar, ProgressBar, Timer, and MaskedTextBox Controls
- Use Advanced Windows Controls such as ImageList, TreeView, ListView, TabControl, DataGrid, DataGridView, MenuStrip, ToolStrip, StatusStrip, and FlowLayoutPanel Controls
- Develop Windows Forms Custom Controls
- Design Multithreaded Applications
- Explain Thread Synchronization Techniques such as Mutex, Semaphore, and Event Techniques
- Use Asynchronous Programming Model (APM) Techniques
- Call Legacy COM Objects (COM Interop) in .NET framework
- Create and Manage .NET Components and Assemblies
- Use the ADO.NET Classes to Build Data-bound Windows Forms
- Read, Write, Process, and Search the XML Data
- Use Reflection Techniques to Load an Assembly, and Dump its Type Information

■ Verification

I verify that the content of this course outline is current.

Amir Ahani April 1, 2009 a.ahani
Authoring Instructor Date

I verify that this course outline has been reviewed.

Kevin Cudihee
Program Head/Chief Instructor Date

I verify that this course outline complies with BCIT policy.

Kim Dotto
Dean/Associate Dean Date

Note: Should changes be required to the content of this course outline, students will be given reasonable notice.

■ Instructor

Amir Ahani	Office Location:	Office Phone: (604) 376 - 1164
	Office Hrs.:	E-mail Address: amirahani@gmail.com

■ Learning Resources

Required Textbook:

1. C# 2008 for Programmers by Harvey Deitel and Paul Deitel (ISBN: 0-13-714415-6)

Recommended References:

2. MCTS Self-Paced Training Kit (Exam 70-536): Application Development Foundation (ISBN: 0-7356-2277-9)
3. Programming Microsoft Visual C#: The Base Class Library by Francesco Balena (ISBN: 0-7356-2308-2)
4. Programming Microsoft Visual C# 2008: The Language by Donis Marshall (ISBN: 0-7356-2540-9)
5. Programming in the Key of C# by Charles Petzold (ISBN: 0-7356-1800-3)
6. Microsoft Visual C# .NET Core Reference by Mickey Williams (ISBN: 0-7356-1290-0)
7. Inside C#, Second Edition by Tom Archer, Andrew Whitechapel (ISBN: 0-7356-1648-5)
8. Microsoft Visual C# .NET Language Reference (ISBN: 0-7356-1554-3)
9. Programming Microsoft Windows with C# (ISBN: 0-7356-1370-2)
10. Windows Forms Programming in C# by Chris Sells (ISBN: 0-321-11620-8)
11. Applied XML Programming for Microsoft .NET by Dino Esposito (ISBN: 0-7356-1801-1)
12. Microsoft Visual C# by John Sharp (ISBN: 0-7356-2129-2)
13. Mastering COM and COM+ by Ash Rofail and Yasser Shohoud (ISBN: 0-7821-2384-8)
14. Microsoft ADO.NET by Rebecca Riordan (ISBN: 0-7356-2164-0)
15. Programming Microsoft ADO.NET Core Reference by David Sceppa (ISBN: 0-7356-2206-X)
16. Pro C# 2008 and the .NET 3.5 Platform by Andrew Troelsen (ISBN: 978-1590598849)
17. Professional Visual Studio 2008 by Nick Randolph and David Gardner (ISBN: 978-0470229880)
18. Professional C# 2008 by Christian Nagel, Bill Evjen, and Jay Glynn (ISBN: 978-0470191378)
19. C# Class Design by Richard Conway, Teun Duynstee, and Roger Rowland (ISBN: 1-86100-828-7)
20. Practical Guidelines and Best Practices for Microsoft Visual C# by Francesco Balena (ISBN: 978-0735621725)

■ Information for Students

By attending this course and receiving this course outline, you have been made aware of the following policies. Please follow the links provided as each student is responsible for reading and complying with these policies.

The following statements are in accordance with the *BCIT Student Regulations Policy 5002*. To review the full policy, please refer to http://www.bcit.ca/files/pdf/policies/5002.pdf.

Attendance/Illness:
In case of illness or other unavoidable cause of absence, the student must communicate as soon as possible with his/her instructor or Program Head or Chief Instructor, indicating the reason for the absence. Prolonged illness of three or more consecutive days must have a BCIT medical certificate sent to the department. Excessive absence may result in failure or immediate withdrawal from the course or program.

Academic Misconduct:
Violations of academic integrity, including dishonesty in assignments, examinations, or other academic performances are prohibited and will be handled in accordance with the *Violations of Standards of Conduct* section of Policy 5002.

The School of Computing and Academic Studies expects the highest level of professional conduct and ethical behaviour from all students enrolled in part time studies courses and programs. All students are reminded of the BCIT policy related to the *Responsible Use of Information Technology*. Read the full policy here: http://www.bcit.ca/files/pdf/policies/3501.pdf.

The Computing and IT knowledge and skills acquired by students in the course of their studies confers upon them, as with all professionals, a special responsibility to use their knowledge in a responsible, professional and ethical manner. Further, given that misuse of computer facilities at BCIT can have significant legal and/or economic impacts, upon evidence of any such misconduct, the School may recommend immediate suspension, even for first offences.

Attempts:
Students must successfully complete a course within a maximum of three attempts at the course. Students with two attempts in a single course will be allowed to repeat the course only upon special written permission from the Associate Dean. Students who have not successfully completed a course within three attempts will not be eligible to graduate from their respective program.

Accommodation:
Any student who may require accommodation from BCIT because of a physical or mental disability should refer to BCIT's Policy on Accommodation for Students with Disabilities (http://www.bcit.ca/files/pdf/policies/4501.pdf), and contact BCIT's Disability Resource Centre (SW1-2300, 604-451-6963, http://www.bcit.ca/drc/) at the earliest possible time. Requests for accommodation must be made to the Disability Resource Centre, and should not be made to a course instructor or Program area.

Any student who needs special assistance in the event of a medical emergency or building evacuation (either because of a disability or for any other reason) should also promptly inform their course instructor(s) and the Disability Resource Centre of their personal circumstances.

Assignments:

Assignments must be done on an individual basis unless otherwise specified by the instructor. Late assignments, lab reports or projects will **not** be accepted for marking. There are six assignments. Save each of your assignments as your assignment number, lastnamefirstname, BCITID.
For example **AssignmentNumberOne_AhaniAmir_A00559977.zip**

Labs:

Students are encouraged to help each other out and share ideas. Lab exercises are to be done *in class* during the lab period and your participation in these exercises is required in order to be marked as present for that class and to receive the lab mark. You will *not* be handing in the labs. Simply finishing the lab exercise in class *or* working on it until the end of the lab time is sufficient for the lab mark. If you finish early and I am satisfied with your work, you may leave early.

Makeup Tests, Exams or Quizzes:

If you miss a test, exam or quiz, you will receive zero marks. Exceptions may be made for **documented** medical reasons or extenuating circumstances. In such a case, it is the responsibility of the student to inform the instructor **immediately**.

Ethics:

BCIT assumes that all students attending the Institute will follow a high standard of ethics. Incidents of cheating or plagiarism may, therefore, result in a grade of zero for the assignment, quiz, test, exam, or project for all parties involved and/or expulsion from the course.

COMP 2691 Schedule

Wk #		Course Topics	References	Due Date
Week 1	**Lecture**	**Review of COMP 2690** **Input / Output (Part One)** ***Files, Directories, and Drives*** • The FileSystemInfo Class • The FileInfo Class • The DirectoryInfo Class • The DriveInfo Class • The Path Class • The FileSystemWatcher Class	**Ref 1** Ref 2 Ref 3 Ref 9 Ref 12 Ref 16 Ref 17 Ref 18 Ref 19 Ref 20	
	Class Activity	Lab 1: How to Get Information about a File Lab 2: How to Copy a File Lab 3: How to Enumerate Files in a Directory Lab 4: How to Enumerate Drives Lab 5: How to Change a File Extension Lab 6: How to Monitor a Directory for Changes Guided Practice 1: Enumerating Files and Watching for Changes		
	Home Activity	**Assignment 1** Building a Hot-Plugging Application		**Week 2**
		Optional		
		Research 1: Describe Several Advanced Uses of FileSystemWatcher Class by Visiting the Following Websites: (a) Visual C# Developer Center (b) The C# Language (c) CodeProject C# cs (d) C# Corner		

Wk #		Course Topics	References	Due Date
Week 2	**Lecture**	Review of Week 1 **Input / Output (Part Two)** ***Reading and Writing Files*** • The File Class • The Directory Class • The FileStream Class • The StreamReader Class • The StreamWriter Class • The StringReader Class • The StringWriter Class • The MemoryStream Class • The BufferedStream Class	Ref 1 Ref 2 Ref 3 Ref 9 Ref 12 Ref 16 Ref 17 Ref 18 Ref 19 Ref 20	
	Class Activity	Lab 1: How to Read from a File Lab 2: How to Search inside a File Lab 3: How to Write to a File Lab 4: How to Write to an existing File Lab 5: How to Read an in-memory string using StringReader Lab 6: How to Write an in-memory string using StringWriter Lab 7: How to Use a MemoryStream Guided Practice 1: Reading and Writing Files		
	Home Activity	**Assignment 2** Reading Data from a Sequential-Access Text File		**Week 3**
		Optional		
		Research 1: How to Read/Write Data from/to USB Port Research 2: How to Read/Write Data from/to Serial Port Research 3: What is Luhn's Formula for Validating Credit Card Numbers? Research 4: What are the Several Advanced Uses of Regular Expressions?		

Wk #		Course Topics	References	Due Date
Week 3	**Lecture**	**Review of Week 2** **GUI Concepts (Part One)** ***Event Handling*** • Delegates and the Event-Handling Mechanism • Mouse-Event Handling • Keyboard-Event Handling ***Windows Forms Controls*** • The LinkLabel Control • Scrollbar Controls • Provider Controls (ToolTip Control and ErrorProvider Control) • The MonthCalendar Control • The DateTimePicker Control • The DomainUpDown Control • The NumericUpDown Control • The TrackBar Control • The ProgreeBar Control • The Timer Control • The MaskedTextBox Control ***Custom Control Creation*** • Inheriting from an Existing Control • Improving the Custom Control • Composing Multiple Controls • Creating a Control from Scratch	**Ref 1** Ref 3 Ref 4 Ref 5 Ref 6 Ref 10 Ref 18 Ref 19 Ref 20	

Wk #		Course Topics	References	Due Date
Week 3	**Class Activity**	Lab 1: How to Draw the Mouse's Moving Path Lab 2: How to Display the Character Presses using KeyChar Lab 3: How to Use LinkLabel Control Lab 4: How to Use the ScrollBar Controls Lab 5: How to Use ToolTip Control Lab 6: How to Use ErrorProvider Control Lab 7: How to Use DateTimePicker Control Lab 8: How to Use DomainUpDown Control Lab 9: How to Use NumericUpDown Control Lab 10: How to Use TrackBar Control Lab 11: How to Use MaskedTextBox Control Guided Practice 1: Creating and Using Simple Delegates Guided Practice 2: Using the Timer Control Guided Practice 3: Creating a Composite Control with Visual Studio		
	Home Activity	**Assignment 3** Creating a Composite Control for Validating Data		**Week 5**
		Optional		
		Research 1: How to Create Custom Provider Controls Research 2: How to Handle Excel Events from a Client that is Developed with Visual C# Research 3: How to Establish a COM Event Sink		

Wk #		Course Topics	References	Due Date
Week 4	**Lecture**	**Review of Week 3** **ADO.NET and XML (Part One)** ***A Quick XML Prime- Optional*** • XML Building Blocks • DTD Construction Basics • Pondering Elements and Attributes • Attributes Types • Understanding Namespaces • XML Formatting Strategies ***Serialization*** • Serializing / Deserializing Objects • Binary Serialization • Soap Serialization • XML Serialization ***ADO.NET*** • The ADO.NET Object Model • .NET Data Providers • The Connection Object • The Command Object • The Parameter Object • The DataReader Object • The DataAdapter Object • The DataSet Object • The DataTable Object • The DataRow Object • The DataColumn Object	**Ref 1** Ref 11 Ref 14 Ref 15 Ref 17 Ref 18	

Wk #		Course Topics	References	Due Date
Week 4	Class Activity	Lab 1: How to Create a Simple XML File Lab 2: How to Serialize an Object Lab 3: How to Deserialize an Object Lab 4: How to Create Classes that Can be Serialized Lab 5: How to Disable Serialization of Specific Members Lab 6: How to Provide Version Compatibility during the Serialization Lab 7: How to Use XML to Serialize an Object Lab 8: How to Use XML to Deserialize an Object Lab 9: How to Generate a Class based on a Schema by Using XSD tool Lab 10: How to Serialize a DataSet Lab 11: How to Establish a Connection to MS Access Lab 12: How to Establish a Connection to MS SQL Server Lab 13: How to Establish a Connection to MS Excel Lab 14: How to Read the Connection String from app.config file Guided Practice 1: Serialize and Deserialize Objects Guided Practice 2: Using XML Serialization Guided Practice 3: Working with Connected Data Guided Practice 4: Working with Disconnected Data		
	Home Activity	**No Assignment**		
		Optional		
		Research 1: What are the Advantages of XSD Schema? Research 2: What is the SQL Server Express 2005 User Instance? Research 3: How to attach DB files to a SQL Server database instance? Research 4: What is the Provider Factory Model? Research 5: Explain the three types of T-SQL statements: DDL/DCL/DML		

Wk #		Course Topics	References	Due Date
Week 5	**Lecture**	**Review of Week 4** **GUI Concepts (Part Two)** ***Advanced Windows Forms Controls*** • The ImageList Control • The TreeView Control • The ListView Control • The TabControl Control • The DataGrid Control • The DataGridView Control • The MenuStrip Control • The ToolStrip Control • The StatusStrip Control • The FlowLayoutPanel Control	**Ref 1** Ref 3 Ref 5 Ref 6 Ref 10 Ref 18	
	Class Activity	Lab 1: How to Use ImageList Control Lab 2: How to Use TreeView Control Lab 3: How to Populate a TreeView Control from a DataSet Lab 4: How to Use TreeView Control to Display Directory Structure Lab 5: How to Use ListView Control Lab 6: How to Use ListView to Display Directories and their Contents Lab 7: How to Use TabControl Control Lab 8: How to Insert a Tab in Particular Position in a TabControl Lab 9: How to Use DataGrid Control Lab 10: How to Use DataGridView Control Guided Practice 1: Providing Standard Menu Items to a Form Using MenuStrip Guided Practice 2: Using ToolStrip Control Guided Practice 3: Using StatusStrip Control Guided Practice 4: Arranging Controls on Forms Using a FlowLayoutPanel		
	Home Activity	**Assignment 4** Creating an Explorer Style Interface		**Week 7**
		Optional		
		Research 1: What is the Data Access Layer (DAL)? Research 2: View the "ARCast -Threading and Async IO" show from Channel 9 Microsoft and explain which of the following models perform faster: (a) Single threaded or (b) Multi-threaded model.		

Wk #		Course Topics	References	Due Date
Week 6		**Midterm Exam** (No Lecture, No Class Activity)		

Wk #		Course Topics	References	Due Date
Week 7	Lecture	**Review of Week 5** **Multithreading** • Threading Fundamentals • The Thread Class • Thread Priorities and Thread Scheduling • Creating and Executing Threads • Sharing Date • Thread Synchronization • The Monitor Class • The Interlocked Class • The Mutex Class • The ThreadPool Class • Multithreading with GUIs • Asynchronous Programming Models (APM Models)	**Ref 1** Ref 2 Ref 6 Ref 7 Ref 10 Ref 12 Ref 16 Ref 17 Ref 18	
	Class Activity	Lab 1: How to Create a Thread Lab 2: How to Use Multiple Threads Lab 3: How to Pass Data to Threads Lab 4: How to Avoid Collisions in the Multithreaded Environment Lab 5: How to Synchronize Access to Objects Using Lock Keyword Lab 6: How to Synchronize Access to Objects Using Monitor Class Lab 7: How to Avoid Deadlocks in the Multithreaded Environment Lab 8: How to Synchronize Access to Objects Using ReaderWriterLock Lab 9: How to Synchronize Access to Objects Using WaitHandle Class Lab 10: How to Synchronize Access to Objects Using Mutex Class Lab 11: How to Synchronize Access to Objects Using Semaphore Class Lab 12: How to Synchronize Access to Objects Using Event Class Lab 13: How to Use Asynchronous Programming Model (APM) Lab 14: How to Use APM- "Wait-Until-Done" Model Lab 15: How to Use APM- Polling Model Lab 16: How to Use APM- Callback Model Lab 17: How to Use the Timer Class Guided Practice 1: Using the Thread Class to Demonstrate Multithreading. Guided Practice 2: Using a Mutex to Create a Single-Instance Application. Guided Practice 3: Using the ThreadPool to Queue Work Items. Guided Practice 4: Using the BackgroundWorker Component to Run an Operation Asynchronously Guided Practice 5: Running a Long Time Operation in the Background		
	Home Activity	**Assignment 5** Calculating Digits of Pi using Multithreading Model		**Week 9**
		Optional		
		Research 1: What are the Best Practices for Implementing the Event-based Asynchronous Pattern?		

Wk #		Course Topics	References	Due Date
Week 8	**Lecture**	Review of Week 7 **Component Object Model** ***COM and COM+*** • Migrating from Legacy COM Components to .NET Components • Using ActiveX Controls • Using "Legacy" COM Components (RCW Process) • Using PInvoke to Call the Windows API • Create and Manage .NET Components • Exposing .NET Component to Legacy Component (CCW Process) • COM and COM+ Interoperability	Ref 2 Ref 13	
	Class Activity	Lab 1: How to Use Windows Forms ActiveX Importer Tool Lab 2: How to Use the Type Library Importer Tool (tlbimp.exe) Lab 3: How to Use Direct Reference with a COM Library Lab 4: How to Use the Type Library Exporter Tool (tlbexp.exe) Lab 5: How to Use PInvoke to Call GetComputerName API function Lab 6: How to Use API Viewer Tool		
	Home Activity	**No Assignment**		
		Optional		
		Research 1: How Late Binding Can be Used for Loading ActiveX Controls? Research 2: What are the Custom, Dispatch, and Dual InterfacesInterfaces? Research 3: What is Marshaling? Research 4: What is COM Registration? How to Install a COM Library into the Global Assembly Cache?		

Wk #		Course Topics	References	Due Date
Week 9	**Lecture**	**Review of Week 8** **ADO.NET and XML.NET (Part Two)** ***XML Objects*** • XMLDocument, and XMLDataDocument • XPathDocument • XMLConvert • XPathNavigator • XMLReader • XMLWriter • XMLTextReader • XMLNodeReader • XMLTextWriter • XSLTransform	Ref 11 Ref 14 Ref 15 Ref 17 Ref 18	
	Class Activity	Lab 1: How to Create a New XMLDocument from Scratch Lab 2: How to Parse XMLDocument Using the DOM Lab 3: How to Parse XMLDocument Using the XPathNavigator Lab 4: How to Search the XMLDocument Using the DOM Lab 5: How to Search the XMLDocument Using the SelectSingleNode Lab 6: How to Search the XMLDocument Using the GetElementByTagName Lab 7: How to Search the XMLDocument Using the SelectNodes Guided Practice 1: Using XML with OLE DB Data Provider Guided Practice 2: Loading Huge XML Document Asynchronously Guided Practice 3: Building a Hot-Plugging XML DOM		
	Home Activity	**Assignment 6** Creating a Custom CSV-to-XML Reader		**Week 12**
		Optional		
		Research 1: Explain the XMLReadMode Options such as Auto, ReadSchema, IgnoreSchema, InferSchema, DiffGram, and Fragment. Research 2: What is the XPath Language?		

Wk #		Course Topics	References	Due Date
Week 10	**Lecture**	**Review of Week 9** **Reflection** • The Assembly Class • The Module Class • The Common Attribute Classes • The Type Class	Ref 2 Ref 3 Ref 8 Ref 18	
	Class Activity	Lab 1: How to Get the Current Executing Assembly. Lab 2: How to Get the Assembly Modules Lab 3: How to Get the Assembly Attributes Lab 4: How to Get the Assembly Types Lab 5: How to Use Reflector Tool Lab 6: How to Use Assembly Documenter Tool Guided Practice 1: Using the .NET Tools to Examine an Assembly Guided Practice 2: Setting Assembly Attributes and Display them at runtime Guided Practice 3: Loading an Assembly, and Dumping its Type Information Guided Practice 4: Invoking Members through Reflection		
	Home Activity	**No Assignment**		
		Optional		
		Research 1: Explain the Builder Classes of System.Reflection.Emit Namespace.		

Wk #		Course Topics	References	Due Date
Week 11	**Lecture**	**Review of Week 10**		
		ADO.NET and XML.NET (Part Three)		
		Define XML Data	Ref 11	
		Format and Display XML Documents	Ref 14	
		XML Processing with .NET	Ref 15	
		XML Readers and XML Writers	Ref 17	
		XML Document Object Model (DOM)	Ref 18	
	Class Activity	Guided Practice 1: Populating a DataSet Object from a Database		
		Guided Practice 2: Moving around in DataSet Objects and Retrieving Data		
		Guided Practice 3: Using Strongly Typed DataSet Objects		
		Guided Practice 4: Using DataSet Objects with Multiple Tables		
		Guided Practice 5: Finding and Sorting Data in DataSet Objects		
		Guided Practice 6: Updating Data with ADO.NET		
		Guided Practice 7: Adding Data with ADO.NET		
		Guided Practice 8: Deleting Data with ADO.NET		
		Guided Practice 9: Editing Data with the DataGrid Control		
		Guided Practice 10: Displaying the Contents of an XML Document		
		Guided Practice 11: Modifying an XML File by Using Code		
		Guided Practice 12: Reading an XML Document into a DataSet Object		
	Home Activity	**No Assignment**		
		Optional		
		Research 1: What is the LINQ to XML?		

Wk #		Course Topics	References	Due Date
Week 12		**Final Exam** (No Lecture, No Lab)		