Bus 410-01: Software Design & Development - Fall 2012
TR; 8:00 – 9:30 AM; IS Lab, Sage 2235

Instructor: Dr. George Philip
Office: Sage 2448; Telephone: 424-3152; Email: Philip@uwosh.edu;
Office hours: M,T,W,R: 1:15 – 3:00; or, by appointment


Description:
This course deals with software design and development in the event-driven (visual or object-based) programming environment. Visual Basic (2010) will be used to gain an understanding of the event driven environment. Basic, intermediate and some advanced features of Visual Basic will be covered in this course. This course also introduces you to Csharp (C#) language.

The objectives of the course are:
1. Students will gain an understanding of software design and development in the event driven programming environment
2. Students will be able to use Visual Studio (2010) and Visual Basic for developing business applications with special emphasis on database applications.
3. Students will understand GUI development.
4. Student will understand basic object-oriented programming in Visual Basic
5. Students will develop problem solving and creative thinking skills.

College of Business Administration Assessment Goals (applicable to this course)
Business Knowledge: The COBA graduate will be competent in a functional area (major field).
Analytical Thinking & Problem Solving Skills: COBA graduates will analyze situations and make decisions, using problem solving techniques, as well as creative and advanced critical thinking skills.

The objectives of the MIS major that are addressed in this course are:
1. MIS graduates will have an understanding of information systems analysis and design
2. MIS graduates will be able to develop software using a visual development tool
3. MIS graduates will be able to model and develop database applications
4. MIS graduates will have the basic knowledge/skills to evaluate, select and use appropriate information technology to solve business problems in the various functional areas.

The course involves extensive hands-on work involving Visual Basic 2010. The primary method of learning is through examples/exercises involving computer work. The course work includes four assignments and several exercises involving hands-on work with Visual Basic.Net. Details of these assignments are given on separate handouts.

Grading Scheme (subject to minor adjustments)

<table>
<thead>
<tr>
<th>Component</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exam I</td>
<td>60</td>
</tr>
<tr>
<td>Exam II</td>
<td>40</td>
</tr>
<tr>
<td>Exam III</td>
<td>40</td>
</tr>
<tr>
<td>Assignments</td>
<td>100</td>
</tr>
<tr>
<td>Practice Exercises</td>
<td>25</td>
</tr>
<tr>
<td>In-class exercises, attendance/participation/professionalism, &amp; Professional Development</td>
<td>35</td>
</tr>
<tr>
<td>Total</td>
<td>300</td>
</tr>
</tbody>
</table>

Exams must be taken at the specified times unless prior permission is obtained from the instructor for an alternate time.
Students are expected to attend all class and lab sessions. If you miss a lab session, you must complete the missed lab work before coming to the next session, since most lab exercises build from what you did in previous sessions.

The table on the next page shows the course outline. Column 3 shows the VB and C# project folders and forms (.vb & .cs files) that contain examples to be discussed in class. These folders/files can be found in the VB & C# folders under Bus410 in TitanFiles, or in D2L.

Column 3 also shows the assignments and practice exercises that need to be handed in. These are handed out in class or posted in D2L along with other class materials.

All assignments/exercises are due at the beginning of the class, by 8:02 AM. Late assignments/exercises will have a penalty of 1 point/day. No late assignment will be accepted after the graded assignment is returned to students (Typically, practice exercises are returned the next class, and assignments returned the following Tuesday). Special consideration will be given to unforeseen circumstances. When you plan your work on assignments, you must allow for computer/printer/network problems in the lab. Having a printer down in a lab is not a valid excuse for handing an assignment late.

You may work in groups of maximum two on all assignments and exercises. Your individual grade for an assignment would depend on your contribution. Group members will be asked to provide peer evaluations.

Coding for the assignments must be done exclusively by your group, or by you if you work individually. Allowing another group/person to copy your code and copying any part of someone else’s work are considered academic dishonesty and will result in appropriate disciplinary action including failing grade. You are responsible to protect your code from being copied by others. Do not leave your code on the computers in the lab or any place where others can access it.

Practice exercises are exercises that need to be handed in. If time permits, some class time may be given to do these in class. You are required to hand-in any six of the seven practice exercises specified in column 3, for a total of 25 point. You may hand-in the other for participation points.

Class exercises are done in class depending on the availability of time. So, they are not scheduled in the syllabus. Class exercises missed due to an unexcused absence cannot be made up.

An excused absence may be granted for reasons related to university-approved event, bereavement, jury duty, or other personal exigencies, if the instructor is informed prior to the absence.

Students are awarded six points (out of 35 from the last group) for participation in a professional development activity. You may select one from the following two options.

Option I. Attend two professional IS meetings/events. For example,

1. MIS club meetings that present an IS topic (Make sure you write your name in the sign up sheet at the meeting; no report is necessary.)
2. Meetings of professional organizations in the IT field. (Assoc. of I.T. Professionals has monthly meetings in Appleton). Provide a brief report of the presentation.
3. Tour or other professional development activity arranged by the MIS club

Option II. Make a 10-minute presentation to the class on a current topic related to software development in VB. No written report is required.
<table>
<thead>
<tr>
<th>Week</th>
<th>Topics Covered</th>
<th>Exercises/Assignments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Week 1</td>
<td><strong>Introduction</strong></td>
<td>*<em>Bus410\VB\VBintro*</em> ConvertDollar1.vb, ConvertDollar2.vb, Date.vb, Strings.vb, DataTypes.Vb</td>
</tr>
<tr>
<td>Sep 6,11</td>
<td>Ch. 1: Intro to Visual Studio</td>
<td>*<em>Bus410\C#\CsharpIntro*</em> Convert1.cs. (These are examples to be discussed in class).</td>
</tr>
<tr>
<td></td>
<td>Ch 2: Design a Windows Form app.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ch 3: Code and Test a Windows Form app</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ch 4: Data types</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Label &amp; Text Boxes, Command buttons, Properties, methods, and events.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Displaying data in List Boxes.</td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Design Issues.</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Form Design guidelines</strong> (handout)</td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Ch 4, 9: Language essentials (Part 1)</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Numeric, Date, &amp; String data &amp; Variables. Formatting</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Built-in functions for Date &amp; String.</td>
<td></td>
</tr>
<tr>
<td>Week 2</td>
<td><strong>Ch 5: Control structures</strong></td>
<td><strong>Practice Exercise 1, due 9/13</strong></td>
</tr>
<tr>
<td>Sep 13,18</td>
<td>Selection: If-Then-Else, Select statements. Enumeration.</td>
<td>*<em>Bus410\VB\ProgramBasics*</em> Grade.vb, ConvertDollar3.vb, Validate.vb, DataEntry.Vb</td>
</tr>
<tr>
<td></td>
<td>Data validation: KeyPress, KeyUp, KeyDown, GotFocus &amp; Validate events.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Loops: For-Next, Do while/until, For Each-Next.</td>
<td>*<em>VB\Procedures*</em> GradeSubG</td>
</tr>
<tr>
<td></td>
<td><strong>Ch 6: Procedures, Event handlers</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Passing parameters, scope of variables, Functions, exception handling.</td>
<td>*<em>C#\Procedures*</em> GradeSubp.cs</td>
</tr>
<tr>
<td>Week 3</td>
<td><strong>Software Design Principles</strong></td>
<td></td>
</tr>
<tr>
<td>Sep 20,25</td>
<td>Coupling and cohesion of procedures</td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Ch 10: Windows Forms &amp; Controls</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Radio Button, ComboBox.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CheckBox, ListBox</td>
<td></td>
</tr>
<tr>
<td>Week 4</td>
<td><strong>GUI Design Guidelines(handout)</strong></td>
<td><strong>Practice Exercise 2, due 10/2</strong></td>
</tr>
<tr>
<td>Sep 27,</td>
<td><strong>Ch 12: Testing and Debugging.</strong></td>
<td>*<em>VB\Files*</em> FileInput.vb, Fileoutput.vb, Fileoutput_dialog.vb.</td>
</tr>
<tr>
<td>Oct 2</td>
<td><strong>Ch 21: Text Files, Common Dialog Box</strong></td>
<td></td>
</tr>
<tr>
<td>Week 5</td>
<td><strong>Exam I, October 4</strong></td>
<td><strong>Practice Exercise 3, due 10/9</strong></td>
</tr>
<tr>
<td>Oct 4,9</td>
<td><strong>Ch. 24: Menus &amp; Multi-form applications:</strong></td>
<td>*<em>Menus*</em> FileInput_menu.vb</td>
</tr>
<tr>
<td></td>
<td>MDI forms, Tab control, Toobar, ListViewItem</td>
<td>*<em>MultiFormApps*</em> Mortgage.vb, Pmts.vb, About.vb, MDIApp\frmMain.Vb, frmDocument.vb</td>
</tr>
</tbody>
</table>
Week 6  
Oct 11, 16

**Ch. 8: Arrays, Collections**

- **Sharing data & programs between procedures & forms (handouts)**
- **Sharing controls between forms**

Week 7  
Oct 18, 23

**Ch 13, 14, 15: Database and ADO.Net – bound controls:**
- Connect to a data source.
- Use of Datasets, Table Adapter, & Binding Navigator.
- Display data in DataGrid & TextBoxes, Complex Binding with ComboBoxes.
- Searching for records.
- Display data from multiple tables

Week 8  
Oct 25, 30

**Ch. 15: Database - bound Controls**
- Add/Edit/Delete database records.

**Ch. 15: Database – unbound controls:**
- Create data access objects using code at run time from classes generated at design time.
- Add/edit/delete records

Week 9  
Nov 1, 6

**Exam II, November 1**

**Ch. 16, 17: Database – unbound controls:**
- Create data access at run time from .Net classes.
- DataReader, Untyped Datasets, Transaction Processing.

Week 10  
Nov 8, 13

**Large databases**

**Accessing Oracle databases & Spreadsheets**

**Crystal Reports**

---

**Additional Controls**
- frmTabControl.vb, frmToolBar.vb

**Practice Exercise 4, due 10/16**
- VB\Arrays
  - frmArrayIntro1.vb, Array2D Intro.vb, Array2D.vb
- VB\Collections
  - frmList.vb, SortedList.vb
- C#\Collections
  - frmLists.cs

**ShareProcedures**
- AnalyseScore.vb, AboveAve.vb, FindScore.vb. (Versions 0 – 2)

**ShareData**
- Timer.Vb, StartTimer.Vb, StopTimer.Vb (Versions 0,1,2)

**ShareControls**
- frmTimer, frmStartTimer, frmStopTime, (Versions 0,1,2), ModTimer.vb

**Assignment 2, due Oct 23**

**Database_bound**
- Countries_Grid.vb, EmpDept.vb, EmployeeSearch_FilterBy.Vb’
- EmployeeSearch_ParamQry.Vb
- MasterDetail.vb

**Database_bound**
- EmployeeAddEdDel1.Vb,
- EmployeeAddEdDel2.Vb,

**Database_UsingCode**
- Employee_AddEdDel_UsingCode.vb

**VB\Database_UsingCode**
- Employee_UntypedDataSet.Vb,
- Employee_DataReader.Vb,
- TransactionProc.Vb’

**C\Database_UsingCode**

**Practice Exercise 5, due Nov 8**

**Practice Exercise 6, due Nov 13**
- DatabaseLargeUsingDr\AccountsDr.vb
- DatabaseLargeUsingTA\AccountsTA.vb
- Database_Orcacle\Location.vb
- Database_Orcacle\LocationUsingCode.vb
- Spreadsheet\ClubOfficers.vb
Week 11
Nov 15, 20
Ch 11, 18: Object Oriented Programming

THANKSGIVING BREAK

Week 12
Nov 27, 29
Ch 19, 20, 21: Object Oriented Programming

Week 13
Dec 4, 6
Language-Integrated Query (LINQ)
Creating Custom Controls

Week 14
Dec 11, 13
Windows API, Windows Registry
Read/Write XML
Exam III, Dec 13.

Assignment 3 due Nov 20
OOP\OopDemo\frmBookValue1.vb,
frmBookValue2.vb,
frmInvoice.vb
OOP\Invoice\frmInvoice.vb,
ValidData.vb

OOP\InvoiceLib\Invoice.vb
OOP\Invoice_Test\frmInvoice_TestLib.vb

Practice Exercise 7 due Dec 6
LINQ\CustInvoice_1.vb,
CustInvoice_2.vb
CreateControls\ctlClock.vb,
ctlLabelScrollBar.vb
CreateControls\TestControls\frmClock.vb
CreateControls\TextBoxLib\CustomTextBox.vb

Assignment 4 due Dec 11
API&Registry\API.vb, Registry.vb
OOP\InvoiceXML\frmReadXML.vb,
Invoice.vb

Grading Scale

<table>
<thead>
<tr>
<th>Grade</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>92.0% - 100%</td>
</tr>
<tr>
<td>A-</td>
<td>89.0% - 91.9%</td>
</tr>
<tr>
<td>B+</td>
<td>86.0% - 88.9%</td>
</tr>
<tr>
<td>B</td>
<td>82.0% - 85.9%</td>
</tr>
<tr>
<td>B-</td>
<td>79.0% - 81.9%</td>
</tr>
<tr>
<td>C+</td>
<td>76.0% - 78.9%</td>
</tr>
<tr>
<td>C</td>
<td>72.0% - 75.9%</td>
</tr>
<tr>
<td>C-</td>
<td>69.0% - 71.9%</td>
</tr>
<tr>
<td>D</td>
<td>60.0% - 68.9%</td>
</tr>
<tr>
<td>F</td>
<td>&lt;60</td>
</tr>
</tbody>
</table>