Unit Outline*

INMT2239

Business Computing

Semester 2, 2010
Crawley

Assistant Professor Doina Olaru

* This Unit Outline should be read in conjunction with the Business School Unit Outline Supplement available on the Students web site http://www.business.uwa.edu.au/students
UNIT DESCRIPTION

Introduction

I am sure you will find this unit highly relevant and useful for your work/career and I will do my
best to make it a very enjoyable experience as well.

This unit is offered by the UWA Business School as an undergraduate unit within Business
Information Management major and minor for students in years 2-3.

Business Computing involves processing data to produce useful information in a timely and user
acceptable fashion, to satisfy specific business requirements. The objective of this unit is to equip
you with knowledge and hands-on skills to understand, manage and solve computing issues in the
business world. It reviews issues of understanding the business case and scoping the task, and
focuses on designing the user interface, writing and debugging source code and preparing user
documentation. Visual Studio 2010 and Excel Visual Basic are used in this unit.

This unit does not require previous programming knowledge and it is pitched for business
students. It lays the foundation for science and engineering students to further develop their
corresponding knowledge and skills to become professional programmers of Visual Studio.

It is of paramount importance that you continuously prepare for this unit throughout the
semester, as many of the topics covered in the unit are cumulative. The assessment is designed to
help you with this, including an assessable element every couple of weeks (exercises, a project,
and presentation). This permanent evaluation is better suited for the outcomes of this unit, as it
helps reinforcement of the concepts discussed during lecture and lab activities, ensures successful
application of the techniques, and assists you in the preparation for exam.

The material will be presented during lectures, followed by demonstrations/practice in the lab.
Sufficient time will be allocated for questions and discussion, however, please do not hesitate to
interrupt and ask questions at any moment. Timely feedback is most welcome, so please contact
me if you have any comments or concerns.
All materials are made available in WebCT, so I urge you to use it on a weekly basis.
During all teaching activities, I encourage participation of all students at discussions about what
we are learning, how it relates to past experiences, and how can we apply concepts and methods
to programming problems in business.

I wish you all the very best during this academic year and I look forward to seeing you all in
person on the 29th of July.

Unit content

This unit introduces you to the analysis of the business cases/problems and it includes designing
user interfaces, building, debugging, and testing, and finally documenting various applications for
deployment on computers, mobile devices, or Internet. You will be able to identify situations in
which automation is necessary to quickly and reliably solve business problems and decision-
making. You will also gain experience in designing Visual Studio applications suitable for solving
business problems.

Topics covered will include:

• Introduction to the use of computing concepts and techniques in business and
management;
• The VB development environment;
• Variables and calculations;
• Objects and object-oriented design;
• Programming structures and program style;
• Working with databases;
• Testing and debugging;
• Documentation;
• Making Microsoft VB more useful to us.

In this unit we begin to get comfortable with VB 2010 IDE, pre-defined classes, and basic VB syntax, building our skills one step at a time. With a strong foundation, we can then pursue more sophisticated techniques, including inheritance, polymorphism, generics, LINQ, collections, graphics and multimedia, etc.

The Goal of the unit

Through this unit, you will develop fundamental skills for analysing business problems and designing efficient rigorous solutions for information processing and scientific calculations.

Learning outcomes

On successful completion of this unit, you should be able to:

• develop an understanding of the conceptual and practical issues of programming in a wider business context;
• critically appraise the accuracy and relevance of data and the suitability of particular programming elements;
• use the most appropriate programming structures for a specific problem with confidence;
• examine the uses and limitations of various programming components;
• gain practical experience in the development of various applications using Visual Studio 2010 and Excel VBA;
• build your own computer application for a business case.

Educational Principles and Graduate Attributes

In this unit, you will be encouraged and facilitated to develop the ability and desire to:
• undertake business problem identification, analysis, and find the appropriate computer tools to describe and solve it;
• master the subject matter, concepts and techniques, apply and adapt the acquired knowledge to build your own business application;
• enhance competencies to work more effectively independently (through individual exercises) and in groups (through the completion of a group project);
• think and reason creatively about data and use computer applications; apply and adapt the acquired knowledge to facilitate automated data processing;
• develop more effective communication skills through a class presentation on the findings of your program application and by responding to any questions that arise following that presentation;
• develop competencies on the presentation of the business case and solution proposed for it through a project where results and information have to be presented clearly, concisely, and logically.

TEACHING AND LEARNING RESPONSIBILITIES

Teaching and learning strategies

Several types of activities are included in this unit to make your learning easier, engaging, stimulating. The aim is to provide you with the essential toolkit for business programming and solving real life problems using Visual Studio 2010.

Lectures will be combined with practical hands-on sessions to reinforce the elements acquired for each topic. Your accumulated knowledge and experience, your ideas will then get to fruit in your own projects and solved problems.
You will have the opportunity to work individually or to exchange ideas and collaborate in groups. You are encouraged to draw on your own work experiences and share ideas with your colleagues on particular use of programming tools and how to create applications.

Lecture activities

Hard copies of lectures’ slides are provided to you and made available in WebCT. Lectures are designed to consolidate the readings from the textbook and encourage critical appraise of programming tools. During lecture we will present concepts and work through examples.

Labs

The labs organised after lectures are important for your progress in this unit as the aim is to offer practical (“hands-on”) introduction to programming in Visual Studio. You will then be able to check your understanding of the course content and apply the concepts and methods from this unit to your own business problems.

Teaching and learning evaluation

You may be asked to complete two evaluations during this unit. The Student Perception of Teaching (SPOT) and the Students’ Unit Reflective Feedback (SURF). The SPOT is optional and is an evaluation of the lecturer and the unit. The SURF is completed online and is a university wide survey and deals only with the unit. You will receive an email from the SURF office inviting you to complete the SURF when it is activated. We encourage you to complete the forms as your feedback is extremely important and can be used to make changes to the unit or lecturing style when appropriate.

But more importantly, continuous communication and feedback from you (every day in class and via WebCT) is taken into consideration in updating and improving the unit.
In relation to the INMT2239 unit, previous student feedback has resulted in several changes over time, such as:

- Changes to the delivery of the unit where the focus is now on the hands-on experience. Demonstrations are made either in the lecture or lab, followed by your own applications/exercises. This is seen as a more suitable and useful approach to subject for students;
- Changes in the learning and teaching strategies, now including more team work. This is to further encourage active learning in teams, so that you not only learn by doing, but also sharing with your colleagues the learning experience;
- Changes to the assessment – a practical component and a presentation. This was done so that you are able to see the relevance of the material more clearly and to demonstrate easily your achievements in the programming and proudly exchange your experience with the class.

**Attendance**

Participation in class, whether it be listening to a lecture or getting involved in other activities, is an important part of the learning process. It is therefore important that you attend classes. More formally, the University regulations state that ‘to complete a course or unit students shall attend prescribed classes, lectures, seminars and tutorials’. Students should not expect to obtain approval to miss more than two classes per unit unless there are exceptional circumstances.

**CONTACT DETAILS**

We strongly advise students to regularly access their student email accounts. Important information regarding the unit is often communicated by email and will not be automatically forwarded to private email addresses.

<table>
<thead>
<tr>
<th>Unit coordinator/lecturer</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Name:</strong></td>
<td>Doina Olaru</td>
</tr>
<tr>
<td><strong>Email:</strong></td>
<td><a href="mailto:doina.olaru@uwa.edu.au">doina.olaru@uwa.edu.au</a></td>
</tr>
<tr>
<td><strong>Phone:</strong></td>
<td>+61864883908</td>
</tr>
<tr>
<td><strong>Consultation hours:</strong></td>
<td>Monday 12:30 – 2:30 pm or by appointment</td>
</tr>
<tr>
<td><strong>Lecture times:</strong></td>
<td>Thursday 8-9:45am</td>
</tr>
<tr>
<td><strong>Lecture venue:</strong></td>
<td>Don Voelte &amp; Nancy Keegan Case Study Room BUSN: 1.01</td>
</tr>
<tr>
<td><strong>Lab times:</strong></td>
<td>Thursday 10-11:45am, 12-1:45pm (starting week 2)</td>
</tr>
<tr>
<td><strong>Lab venue:</strong></td>
<td>G.85 UWA Business School</td>
</tr>
</tbody>
</table>
TEXTBOOK(S) & RESOURCES

Unit Website

This unit is supported in WebCT. All teaching materials – lecture notes, lab handouts, examples applications, snippets of code, etc. – are included in the unit folders in WebCT.

Recommended/required text(s)


Software requirements

This unit will use Visual Basic 2010 and Excel Visual Basic. Prior knowledge of Excel is desirable, but not essential. The textbook provides instructions on how to use specific features and the sessions in the lab are designed to make everyone comfortable with the software. As the unit makes much use of the computer, it is essential that you get a computer account. Moreover, for successful results this semester, you will need to spend several hours each week in the computer lab or at home, outside of the scheduled lab/tutorials.

Additional resources & reading material

The unit is supplemented in WebCT and all materials and additional handouts or examples will be included in WebCT.
### Key Dates

Key dates concern classes, tutorials and assessment. Consult the appropriate section for the key dates.

<table>
<thead>
<tr>
<th>Date</th>
<th>Assessment</th>
<th>Conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Continuous</td>
<td>Practical tutorial exercises</td>
<td>10% - submitted in class/lab (individually)</td>
</tr>
<tr>
<td>26th of August</td>
<td>Business case description</td>
<td>10% - submitted in WebCT</td>
</tr>
<tr>
<td>9th of September</td>
<td>In-class written quiz</td>
<td>15% - closed book class test</td>
</tr>
<tr>
<td>28th of October</td>
<td>Class presentations</td>
<td>5% - 10 min group presentation of the project (3-4 students/group)</td>
</tr>
<tr>
<td>29th of October</td>
<td>Programming project</td>
<td>25% - submitted in WebCT (3-4 students/group)</td>
</tr>
<tr>
<td>Exam period November 2010</td>
<td>Exam – 3 hours</td>
<td>35% - open book practical exam</td>
</tr>
</tbody>
</table>
# UNIT SCHEDULE

<table>
<thead>
<tr>
<th>Week #</th>
<th>Topic</th>
<th>Lecture</th>
<th>Seminar/tutorial/ questions</th>
<th>Independent activity (e.g. readings)</th>
</tr>
</thead>
</table>
| 1 29th Jul | **Course Introduction**  
Topic 1: Understanding programming concepts | What do we mean by business computing and why do professionals need to know about it? What might you need to know? In this session, we’ll introduce the course and discuss the .NET framework, the integrated development environment, and how we work with projects and solutions. | No lab | Chapters 1-3  
Del Sole (2010) |
| 2 2nd Aug  
(2pm – 201) | **Building Blocks**  
Topic 2: Using variables, constants and functions in calculations | We will cover in this session:  
- anatomy of a project;  
- data types and expressions;  
- operators.  
You will learn the main components of the projects, when and how to use various types of variables, and the differences between modules and classes.  
We will also get introduced to the debugging instrumentation during this week. | Lab 1 – first console and VB (GUI) application  
(4pm – G85, 6pm – G85) | Chapters 4-6, 13  
Del Sole (2010) |
<table>
<thead>
<tr>
<th>Date</th>
<th>Topic</th>
<th>Description</th>
<th>Lab/Final Project</th>
<th>Chapters</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 Aug</td>
<td><strong>Processing Decisions</strong></td>
<td>This session presents the decision and repetition structures with some simple but powerful examples to analyse the options available when preparing the algorithms to solve a programming problem. We will also cover in this session several controls (selection buttons, check boxes) and we will also discuss data validation.</td>
<td>Lab 2 – building applications for loan payment and shipping fees</td>
<td>4, 31-32</td>
</tr>
<tr>
<td>4 Aug</td>
<td><strong>Topic 3: Fundamentals of Processing Decisions (Cont’d)</strong></td>
<td>We continue the previous session with:  o repetition structures, lists, arrays.  o Controls suitable for looping structures.</td>
<td>Lab 3 – exiting a loop, rocket application, building mathematical series</td>
<td>16</td>
</tr>
<tr>
<td>5 Aug</td>
<td><strong>File Access</strong></td>
<td>This topic introduces the types of files used in projects (random, sequential, binary, etc.) and uses simple examples for accessing those files. We cover in this session:  o Visual Basic.NET functions for opening, closing files;  o Stream readers, stream writers;  o Structure variable;  o Dialog boxes.</td>
<td>Lab 4 – data base using sequential and random access, modifications loan payment</td>
<td>11, 19-20</td>
</tr>
<tr>
<td>Date</td>
<td>Topic</td>
<td>Notes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>------------</td>
<td>--------------------------------------------</td>
<td>----------------------------------------------------------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6 2nd Sep</td>
<td><strong>Sub and Function Procedures</strong></td>
<td>What happens when we have parts of the program that repeat? Can we make the program more efficient, reusable? YES - thanks to the amazing characteristics of subs and functions! We cover in this session: o Using sub procedures, function procedures; o Multiple forms and modules. Lab 5 – multiform savings analysis application, creating a text editor Chapters 7-8 Del Sole (2010)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
| 7 9th Sep  | **Arrays**                                 | We will cover: o Using a two-dimensional array to create applications;  
**In-class written test** Lab 6 – discussion mid-sem test Working with data bases Chapters 27 Del Sole (2010) |
<p>| 8 16th Sep | <strong>Web-based Applications</strong>                 | We dedicate this session to creating distributed Web applications by ASP.NET. We will work through examples and focus on alternatives ways to solve business-programming problems (e.g., converting Windows application to Web application). We also discuss in detail the types of errors in programming and strategies to correct/eliminate them (e.g., Intellisense for syntax, testing and validation for logical errors, etc.) Lab 7 – hotel booking, temperature conversion Chapters 37-39 Del Sole (2010) |
| 9 23rd Sep | <strong>Microsoft VB Applications</strong>              | In this session we focus on: o Making the MS Office package more useful for us, specifically working with Excel to create Excel VB applications; o Presenting VBA environment. Lab 8 – web (mobile phone) application Handout |</p>
<table>
<thead>
<tr>
<th>30\textsuperscript{th} Sep</th>
<th>Non-teaching break</th>
</tr>
</thead>
</table>
| 10 4\textsuperscript{th} Oct (8am 163) | Microsoft VB Applications (Cont’d)  
Topic 10: Programming with VBA in Excel (Part 2)  
In this session, you will learn to make full use of the charts, pivot tables, functions, macros in MS Excel.  
Lab 9 – VBA calculate perimeter and aria for various shapes (4pm – G85, 6pm – G85)  
Handout |
| 11 7\textsuperscript{th} Oct | Microsoft VB Applications (Cont’d)  
Topic 11: Programming with VBA in Excel (Part 3)  
More on:  
- Developing a VBA program that is independent from a workbook;  
- Learning about user forms, modules.  
Lab 10 – cost-benefit analysis  
Handout |
| 12 21\textsuperscript{st} Oct | Package and Deployment  
Topic 12: Wrapping up VB 2010  
Up until now, we have studied how to build various types of VB applications. This last topic in VB 2010 is dedicated to how to package and deploy the applications we built.  
We cover:  
- Converting applications;  
- Demonstration packaging.  
Lab 11 - deployment  
Chapters 40, 54-55, 57 Del Sole (2010) |
| 13 28\textsuperscript{th} Oct | Revision and Exam Preparation  
Revision of all 12 topics  
Class presentations  
Programming project due 29\textsuperscript{th} of October (WebCT)  
Revision |

Note: Changes are likely to occur in the lecture schedule with respect to material covered. No modifications are to be made for assessment.
ASSESSMENT MECHANISM

The purpose of assessment

There are a number of reasons for having assessable tasks as part of an academic program: providing opportunities for you to check your understanding of the course content and to explore data analysis and decision making relevance through real life examples, including your own experiences; encouraging you to explore and understand the subject fully and make connections with other disciplines or situations already known; foster collaboration with peers and enhance data analysis computing skills; acknowledge your efforts and progress in mastering the subject. The fact that we grade your work then gives you an indication of how much you have achieved – "Doing is demonstrating that you know it."

Providing feedback on your work also serves as part of the learning process.

In this unit we use a combination of individual and group assignments meant to continuously review your progress and facilitate the communication with your peers.

Assessment mechanism summary

<table>
<thead>
<tr>
<th>Item</th>
<th>Weight</th>
<th>Due date</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Practical tutorial exercises</td>
<td>10%</td>
<td>Ongoing</td>
<td>Exercises solved during the lab are submitted in WebCT (individual work)</td>
</tr>
<tr>
<td>Business case description</td>
<td>10%</td>
<td>26th of August</td>
<td>Lodged online in WebCT</td>
</tr>
<tr>
<td>In-class written quiz</td>
<td>15%</td>
<td>9th of September</td>
<td>Closed book test (written test)</td>
</tr>
<tr>
<td>Class presentations</td>
<td>5%</td>
<td>28th of October</td>
<td>10 min group presentation of the project (3-4 students/group)</td>
</tr>
<tr>
<td>‘Statistics in practice’ project</td>
<td>25%</td>
<td>29th of October</td>
<td>Group programming project – (groups of 3-4 students)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Lodged online in WebCT</td>
</tr>
<tr>
<td>Final practical exam</td>
<td>35%</td>
<td>Exams period – 3 hours</td>
<td>Students have to complete in the lab three exercises and submit them at the end of exam.</td>
</tr>
</tbody>
</table>

Note 1: Results may be subject to scaling and standardisation under faculty policy and are not necessarily the sum of the component parts. The school and Board of Examiners have the right to scale marks where it is considered necessary to maintain consistency and fairness.

Note 2: Your assessed work may also be used for quality assurance purposes, such as to assess the level of achievement of learning outcomes as required for accreditation and audit purposes. The findings may be used to inform changes aimed at improving the quality of Business School programs. All material used for such processes will be treated as confidential, and the outcome will not affect your grade for the unit.
Assessment components

Assessment item #1: Tutorial Exercises

Description
You will receive weekly exercises during the lab and you have to submit five sets of applications/exercises in WebCT. Your lecturer will assess your work based on the: a) inspection of the work done in the lab; b) response to questions to check understanding; c) your contribution to class.
If you are unable to attend the lab activity, inform your lecturer at the earliest possible time and make alternative arrangements (if possible). A reasonable explanation is needed and make-up arrangements must be agreed by your lecturer.

Assessment Criteria
Each set of tutorial exercises is worth 2% and the submissions are marked for the accuracy and completeness of the task. The solutions of the exercises will be submitted via WebCT.

Assessment item #2: Business Case Description

Description
The purpose of the case description (individual submission) is to help you understand the background/context and functionality requirements of a real case business application better and to prepare you for further development of the project. Each student needs to identify an opportunity for improving a task for an organisation or business by developing a Windows application in Visual Basic 2010.
You are encouraged to seek a real business case for which to justify the application development.

Actual program code is not required for this assignment. The assignment contributes 10% towards your final grade and it is due on the 26th of August 2010. The best developed business cases will be chosen by the lecturer for further developing the program in teams.

Assessment Criteria
The assessment is based on: a) the presentation of the business problem; b) description of the main functions of the program; c) design of a user-friendly graphical user interface; and d) presentation of the flow of programming.
The business case report will be submitted before 12:00pm (26th of August) via WebCT.

Assessment item #3: In Class Mid-Semester Test

Description
The purpose of the quiz is to help you evaluate your understanding of programming concepts before you start working on your project. The quiz is designed to provide you with helpful feedback on the issues associated with the design and use of the programming structures.
The quiz is worth 15% and it'll be held on the 9th of September. The quiz will take 60 min (excluding reading time). No textbooks or handouts are allowed for this test. Non-programmable calculators (approved by UWA) and statistical tables are permitted for use.
Assessment Criteria

Multiple choice questions (8%) and short answer/calculation questions (7%) are included in the test.

Assessment item #4: Class Presentation

Description

This assessment requires the project teams (in groups of 3-4) to present and discuss their application before submission of the project. The purpose is to facilitate discussion within the class, and provide feedback on the approaches used for solving the business case. It also provides a demonstration of the oral communication skills and application of knowledge to business computing applications.

The presentation is worth 5% and should last max 10 min (allowing for a max of 3-5 min of questions and discussions). The presentations will be organised on the 28th of October 2010.

Powerpoint slides (or any type of presentation), along with the demonstration of the application features need to be uploaded in WebCT before the day of presentation.

Assessment Criteria

The groups are assessed on the creativity, accuracy/correctness of their application, as well as their general presentation and answers to the questions.

Assessment item #5: Programming Project

Description

The group project is designed to develop your skills in: correct and efficient use of programming concepts and tools; appropriate design of graphical user interfaces; and in report and user manual writing. This is a group project where the main task is to build a VB 2010 application for the allocated business case and prepare a report based on the results of your application. In the project you are also required to add a user manual. The programming tasks should be shared evenly among the team members and ownership of the code should be stated in the program as remark statements.

The project contributes 25% towards your final grade and should be submitted in WebCT before 12pm (29th of October 2010).

Assessment Criteria

The report should include the following sections.

1. Title page. Includes your name, title of the report and its word length.

2. Executive summary. A single paragraph or page that describes the problem and the most important aspects of your application. This section is usually written last. Please note that this section is distinct from the Introduction.

3. Table of Contents. Gives an outline of the report together with the number of pages.

4. Introduction. Several paragraphs in which you describe the background of the business, question of interest to be addressed by your project. You should also include the motivation for the application and expected impacts of implementing it.
5. **Description of the application.** Summarize the tasks and the functionality of your application, as well as how each function is achieved. The key input variables and relevant assumptions should also be discussed here.

6. **Programming solution.** Use appropriate tools to solve the requirements of your application. Describe the programming structures and the flow of programming, present the GUI, discuss testing and debugging performed for your application and explain any limitations/problems encountered during the development of the application.

6. **Conclusions and Summary.** Summarise the key features of your applications and make recommendations for further development and/or deployment.

7. **References.** Harvard Style or Endnote (please consult the Referencing section and contact me for any unclear elements).

8. **Appendices.** Include any tables, graphs and other computer output nor directly referred to in the report. All tables and graphs should be numbered and should have a title and caption.

Please remember that this report needs to be accompanied by a user manual. Do not present a manual abounding on programming jargon; the focus of the user manual is to provide appropriate support/guidance to the user benefitting from your application.

When assigning your grades the School's generic grading scheme will be interpreted in the following way:
### Assessment item #6: Final Exam

**Description**

The final exam will cover ALL lecture material, and will be held in the exam period (November 2010).
The exam will be 180 min (3 hours) long, including reading time, and will contain three programming tasks using both VB 2010 and VBA for Excel. It will be open book and is worth 35% of your final mark.

Assessment Criteria

The assessment is based on: a) the design of a user-friendly graphical user interface for the application; b) presentation of the flow of programming; and c) the accuracy of the coding used in solving the tasks.

Submission of assignments

Assignments should be submitted on their due date on WebCT. All submissions should be accompanied by a signed Plagiarism Declaration Form indicating that “The work is original.”

Please remember to attach an Assignment Cover Sheet to the front of your assignment. You can download and print your Assignment Cover Sheet from the Students web page [http://www.business.uwa.edu.au/students/assessments](http://www.business.uwa.edu.au/students/assessments)

Late assignments will attract a penalty of 5% per day. The lecturer only in exceptional circumstances will waive this penalty. No marks will be awarded to assignments submitted after other students in the class have had their assignments returned.

Assignments will be returned in class or in WebCT. It is the intention that the marked assignments will be returned within two weeks of submission.

Student Guild

Phone: (+61 8) 6488 2295
Facsimile: (+61 8) 6488 1041
E-mail: enquiries@guild.uwa.edu.au
Website: [http://www.guild.uwa.edu.au](http://www.guild.uwa.edu.au)