**Description: OC_Masterbrand_CMYK.eps**

in partnership with

****

ICT40515 Certificate IV in Programming

Develop Mobile Applications

**SP7/Module 10 Assessment**

ICTPRG409 Develop mobile applications

ICTPRG427 Use XML effectively

Assessment: 33131/01

*© Open Colleges Pty Ltd, 2016*

*All rights reserved. No part of the material protected by this copyright may be reproduced or utilised in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage and retrieval system, without permission in writing from the copyright owner.*

*All terms mentioned in this text that are known to be trademarks or service marks have been appropriately capitalised. Use of a term in this text should not be regarded as affecting the validity of any trademark or service mark.*

**Assessment task 33131/01**

## Introduction

This assessment will test your skills and knowledge gained through completing the learning and activities in **Module 10: Develop Mobile Applications.**

This assessment consists of three parts.

In the first part of this assessment, you are required to design and develop a mobile app that interacts with a local XML file.

In the second part of this assessment, you are first required to create a web service, and then to design and develop a mobile app that consumes this web service.

In the third part of the assessment, you will answer a number of questions which will test your knowledge on developing and building a mobile app and using an XML file.

**Note – you must successfully complete this assessments (33131/01) to achieve competency in:**

***ICTPRG409 Develop mobile applications***

***ICTPRG427 Use XML effectively***

**Part A**

**Tasks 1-6**

**Design and develop a mobile app that uses an XML file**

**Requirements**

You have been asked to develop a contact management app. The purpose of the app is to allow a user to manage their contacts by allowing users to view, modify and delete their contacts. It has been decided that the initial app should be developed for a Windows 10 phone using a Windows Universal app platform.

The app is to interact with an XML file at a local level, which will act as data storage for the contacts. The XML file must store several details about each of the contacts, including their:

* ID
* firstname
* lastname
* mobile
* email.

ID is the combination of their firstname and lastname.

The app should meet the following requirements:

* it should be usable from both a portrait and landscape orientation
* when the app is started, it should populate a listbox with the name (the combined firstname and surname) of every contact in the XML file
* when a user selects a contact from the listbox, all of the details of that contact should be displayed to the user in textboxes
* the user should be able to update the contact details by editing them and then clicking an **Update** button
* the user should be able to delete a contact by selecting that contact and clicking a **Delete** button.

Note that:

* you are not required to ensure that the XML file is saved once the emulator is switched off
* you can use any available emulator, but It is recommended that you select either a 5-inch or 6-inch screen size emulator, so that you can easily fit the requirement elements in different orientations on the screen.

|  |
| --- |
| **TASK 1: SELECT AND SET UP DEVELOPMENT ENVIRONMENT**  Based on the requirements for the app, provide answers and evidence for the following tasks:   1. Explain which target platforms the app should be developed for and why? 2. Based on the target platforms you selected from the previous question, detail which development tool(s) would you use to develop the app and why you selected this/these tools(s) rather than other development tools? 3. Based on the selected development tool(s) from the previous question, set up the tools and create a development project for the app.   **Submit:**   * **Written answers to the first two tasks** * **Screenshot evidence for the third task that the project has been set up in the development tool(s).** |
|  |
| **TASK 2: DESIGN MOCKUPS**  Based on the requirements for the app, develop app mockups using a design tool of your choice that:   * shows all the app interface elements in portrait view, and * shows all the app interface elements in landscape view.   Each view must allow for the elements to fit on the screen and provide a colour style for the app.  **Submit:**   * **A screenshot of the landscape and portrait mockup displaying the page layout and content. Input fields should show text in Lorem ipsum.** |
|  |
| **TASK 3: CREATE XML DOCUMENT**  Based on the app requirements, develop an XML file for the app and ensure that this is included in the project. Create three dummy contacts to serve for testing purposes.  The XML file should:   * be well-formed, incorporating appropriate structures and syntax * use Visual Studio’s inbuilt parser to ensure there are no validation (syntax) errors * be correctly formatted.   **Submit:**   * **The XML file.** |
|  |
| **TASK 4: DEVELOP UNIVERSAL WINDOWS APP**  Based on the requirements and the designs created in Task 2 above, develop a Universal Windows app so that:   * when the app is loaded, a local copy of the XML created in the previous activity is created * the code has been written to parse the xml file to populate the listbox with all the contacts contained in the XML file * when a contact is selected in the listbox, the textboxes are updated to provide details of that contact * contact details should be updated when the user clicks the **Update** button * contact details should be deleted when the user clicks the **Delete** button.   **Submit:**   * **Visual Studio project code** * **Screenshot of UI displaying a selected contact, an updated contact and a deleted contact. Additional screenshots should show selection of different contacts and the popup keyboard when entering data.** |
|  |
| **TASK 5: DEBUG AND TEST APP**  You are required to test your app to ensure it meets the required functionality. In recording this, you are to complete the [testing template](https://app.box.com/s/a6yxqxi95qzge6furu6f1bvajskvp5g6) provided. Rename the testing template document to TestCasePartA\_yourName\_studentNumber.docx. (For example, TestCasePartA\_JohnSmith\_17756433.docx).  For the testing document, include the name of the application, your name, the date of when the test was carried out and a brief description of what you are testing in “Test Specifications”. For each test, assign a sequential number to the ID column, what you are testing, how you are going to perform the test, the steps in performing the test, what you will be inputting, the expected and actual result and if the test passed or failed. You are to iterate the testing process until the application is valid and meets the requirements.  Create at least four test cases using the testing template to test the outcomes of the following:   * display all contacts from the XML file * display contact details when a contact is selected * modify a contact * delete a contact.   You can do other test cases.  **Submit:**   * **Testing document with results** |
|  |
| **TASK 6: DEPLOY APP**  Once the app has been fully tested, configure the app manifest file for deployment. The app manifest should:   * In the *Application* tab, provide create a description of the app and set the supported resolutions as Landscape and Portrait only * In the *Packaging* tab, set the version number to 1.0.0 and provide your name as the publisher.   **Submit:**   * **Submit the screenshots of the Application and Packaging tabs for the app manifest and the Package Creation Completed screen from the Create App Package wizard.** |

**Part B**

**Tasks 1-4**

**Create a web service and, design and develop a mobile app to consume this web service**

**Requirements**

Ultimate Gaming PC’s produces a range of high end computers designed for the discerning gamer. These gaming PC’s are distributed through a range of retailers. The retailers would like to access details of these gaming PC’s to add to their store front app for their buyers to gain information on the range of PC’s provided.

You have been asked to develop a web service that will accept a product code and return product details which includes the PC title, description and price. You are also to develop an app that will allow the customer to select a product code from a combo box, access the web service to retrieve and display the PC details.

|  |
| --- |
| **TASK 1: CREATE THE WEB SERVICE**  Based on the requirements, develop a web service in C# that accepts a product code and returns the product code, the PC’s title and description and the price. Once created, you are to deploy the web service to the localhost web server to test. Use the following three PC details in your web service:  Product Code: UG1  Title: UG Hunter  Description: Intel I7/3.4GHZ, 8GB RAM, 32GB SSD, 1TB HHD, Gigabyte GeForce GT710  Price: $1,999  Product Code: UG2  Title: UG Predator  Description: Intel I7/4GHZ, 16GB RAM, 128GB SSD, 2TB HHD, Gigabyte GeForce GT730  Price: $2,499  Product Code: UG3  Title: UG Beast  Description: Intel I7/4GHZ, 32GB RAM, 256GB SSD, 2TB HHD, Gigabyte GeForce GT750  Price: $2,999  **Submit:**   * **Visual Studio project code** * **Screenshot of the web services page displayed in your browser and a screen shot of the XML document returned when you invoke the web service from the web services page.** |

|  |
| --- |
| **TASK 2: DESIGN MOCKUPS**  Based on the requirements for the app, develop app mockups using a design tool of your choice that:   * shows all the app interface elements in portrait view * shows all the app interface elements in landscape view.   Each view must allow for the elements to fit on the screen and provide a colour style for the app.  **Submit:**   * **A screenshot of the landscape and portrait mockup displaying the page layout and content. Input fields should show text in Lorem ipsum.** |

|  |
| --- |
| **TASK 3: DEVELOP APP TO CONSUME WEB SERVICE**  Based on the requirements and the designs created in the previous task, develop a Universal Windows app so that:   * the product codes can be listed, from which the user can choose * based on the product code selected, access the web service developed in Task 1 to retrieve and display the Product code, title, description and price * an AppBar can be created at the bottom of the screen which changes the user interface theme.   **Submit:**   * **Visual Studio project code** * **Screenshot of UI displaying a selected product code and displayed data returned from the web service. Additional screenshots should show the display of different gaming PC details and the popup keyboard when entering data.** |

|  |
| --- |
| **TASK 4: DEBUG AND TEST APP**  You are required to test your app to ensure it meets the required functionality. In recording this, you are to complete the [testing template](https://app.box.com/s/a6yxqxi95qzge6furu6f1bvajskvp5g6) provided. Rename the testing template document to TestCasePartB\_yourName\_studentNumber.docx. (For example, TestCasePartB\_JohnSmith\_17756433.docx).  For the testing document, include the name of the application, your name, the date of when the test was carried out and a brief description of what will be tested in “Test Specifications”. For each test, assign a sequential number to the ID column, what you are testing, how you are going to perform the test, the steps in performing the test, what you will be inputting, the expected and actual result and if the test passed or failed.  Create at least two test cases using the testing template to test the outcomes of the following:   * access web service from web service page * selection of product code and return and display relevant data.   You can do other test cases.  **Submit:**   * **Testing document with results** |

**Part C**

**Questions 1-12**

**App development and XML written assessment**

**Requirements**

Answer each of the twelve (12) questions. Do not write more than 100 words for each question.

Copy and Paste written assessment for each of the questions below into a new Word Document – or – write your answers directly into this document. Name or rename the document to: StudentNumber\_MDXML\_33131-01.docx. For example, 665437\_MDXML\_33131-03.docx.

|  |
| --- |
| **QUESTION 1: SUMMARISE MOBILE APP CONSTRAINTS**  Provide a brief summary of at least three design constraints that should be taken into account when developing apps for hand-held devices.  Memory Management – Mobile devices (especially older ones) have limited memory so if your application requires a lot to run or it is memory inefficient it will perform poorly on a mobile.  Short Battery Life – Most mobiles have quite short battery life and if your application is very power intensive it can force users to stop using the application as it drains their battery far too fast and in turn makes their phone unusable for its main purpose.  Small Screen Size – Mobiles naturally have smaller screen sizes than a regular computer and therefore consideration needs to be given to the readability and usability of the application on a smaller screen. The app my need a different design completely for smaller displays.  Interruption – Since these applications are on mobile, and the primary purpose of a mobile is to /send and receive phone calls and text messages the applications needs to be able to be interrupted at any point during its use and then be resumes without having adverse effects on the functionality or performance of the application. |
| **QUESTION 2: SUMMARISE OBJECT-ORIENTED PRINCIPLES (OOP)**  Provide a brief summary of what the following OOP are:   * Encapsulation   Encapsulation is simply data hiding. It is the process of making sure that variables of a class are not able to be interreacted with by another class. It then makes use of Get methods which allow the user to read the value and Set methods which allow the user to assign a value to a variable.   * Polymorphism   Polymorphism is the name given when a bunch of subclasses all inherit from the same base class. This allows certain classes to use the same method but in their own way. An example of this is a dog and a cat class which both inherit from the animal class. The animal class might have the method talk. Both the cat and the dog class also have the method talk however when it is called from the cat class it Meows and when it is called from the dog class it Barks.   * Inheritance   Inheritance is when another class inherits all the properties and methods from another class. The class that inherits everything is called the subclass and that it inherits from is called the super class. This can be chained many times for example.  You could have an animal class and there are two subclasses which inherit from animal being mammal and reptiles. Mammal can have another class inherit from it for example Whale. This means that Whale has all the properties and methods of both the Animal and the Mammal class as well as any it might have of its own where as Mammal only has properties and methods from the Animal class. |
| **QUESTION 3: SUMMARISE VISUAL PROTOTYPING**  Describe the user-interface design principles. Include a brief summary of how you would use, wireframes, mockups and prototypes.  All three choices can either be sued separately or in conjunction with one another. Wireframing involves a very quick sometimes hand drawn sketch of the UI and its functionality. It is meant o convey the purpose of the application in the simplest of terms. A mock-up is a more realistic looking design of the applications GUI. It will often look very similar to the end product but at this stage is just normally screenshots of a non-functioning program. A Prototype generally looks almost exactly the same as the finished product and often has some actual functionality. It will most likely already have the code to go between different screens and may even have certain areas of the application full functional for demonstration purposes.  They are all used at different stages of development. A wireframe or mockup can most likely be used at the beginning to demonstrate to the development team what needs to happen and the prototype is often something shown to clients or potential shareholders as they are more impressed with a semi working product. |
| **QUESTION 4: SUMMARISE XML**  Provide a brief explanation of what XML is and what it is primarily used for.  XML is a mark up language similar to HTML. It uses code tags to wrap up information. XML on it owns does not actually do anything. XML is primarily designed to carry data and transfer data between applications. XML is very useful in sharing information in a standardised way throught he use of APIs, also known as web services. |
| **QUESTION 5: SUMMARISE Language integration for web design**  Research and explain how HTML, CSS, JavaScript and XML are used in web design.  HTML is the layout of the webpage, where things go, where they are getting their information and how it all comes together. CSS is how the webpage is styled. While you can do all of the styling on the HTML file itself it is much easier and neater to use a CSS (Cascading Style Sheet) to do all the styling. JavaScript is how things on the website do other things. When you go to a weather website most likely there is some JavaScript being run in the background using some form of API to retrieve that data. JavaScript is the only true programming language out of the 4 mentioned. Another example is filling out forms on a Webpage there is most likely some JavaScript code telling the website how to respond and what to do when the submit button is clicked.  XML is useful when you want to send data in a standardised way to another user. This is also generally how APIs operate. The user sends the request for data and expects it to come back in a certain format that they can manage. |
| **QUESTION 6: SUMMARISE MOBILE APP PLATFORMS**  Provide a brief summary of the following three mobile platforms:   * Android   Android is the most popular platform. The following manufactures all use Android for their mobile devices: Samsung, HTC, LG, Motorola, Sony Xiaomi and Asus. Android is written on a Linux Operating system. Android applications are mostly written in Java. Android apps are mostly developed using the follow IDEs: Android Studio, Eclipse with ADT and Visual Studio using Xamarin.   * iOS   iOS is developed by Apple. It is based on OS X from apple computers. All apple devices use the iOS. iOS apps have previously be primarily written in Objective C and C programming languages however recently the new programming language Swift has taken over and most developers have swapped to Swift now. The main IDEs for iOS development are: XCode and Visual Studio with Xamarin plugin.   * Windows 10 mobile   Windows 10 mobile is developed by Microsoft and based on the windows 10 operating system. It is the third most used behind Android and iOS. The following languages are used for windows 10 development: C#, XAML, HTML5, JavaScript and C++. The main Ide to use when developing a Windows 10 application is Visual Studio. |
| **QUESTION 7: SUMMARISE XML TREE HIERARCHY**  Briefly explain the XML tree and how elements can be related to one another within the tree structure.  The XML tree refers to how elements can either be a Parent of a Child in relation to another element. In this example, you can see that Room is nested inside house. This means that Room is a child of House and furthermore House is a Parent of Room. All properties that House has room will also have however House will not have any properties unique to Room.  <House>  <Room>  </Room>  </House> |
| **QUESTION 8: SUMMARISE ELEMENTS AND ATTRIBUTES**  Provide a brief summary of what an XML element and attribute are.  XML Elements are the containers that can hold other elements, text and attributes. An XML element may also be empty. XML attributes are designed to hold data related to a specific element. In this example <Dog bread=”Husky”> the attribute is husky. An attribute is basically a description of what that specific element is. Whereas an element is everything including the tags all the way to the end tag. |
| **QUESTION 9: SUMMARISE THE World Wide Web**  Research and describe the main features and services of the world wide web (WWW).  The World Wide Web is where documents can be retrieves using a Uniform Resource Locator (URL) via the internet. IT works across all platforms and generally uses a web browser to access the information. Some of the features include access to data from across the world in real time, ability to access banking and pay bill from a browser interface and the ability to connect and communicate with other people from all over the world in an instant. |
| **QUESTION 10: SUMMARISE COMMON XAML ELEMENTS**  Provide a brief overview of the main difference between HTML and XAML.  XAML and HTML are completely different mark-up languages. HTML is the standard language of the internet whereas XAML is derived from XML and is used to describe content presentation for parsers that render WPF and Silverlight. The main difference really is that they are not the same at all. XAML is not used for web pages at all and it is used to store data in a standardised way. |
| **QUESTION 11: Information System feature and source characteristics**  Explain briefly what an AppBar is used for in a Universal Windows app and the main XAML elements required to create the AppBar.  The AppBar represent the container control that holds app UI component for commanding and experiences. It has the generic windows App Bar features on it so it is kept uniform across applications. Not all applications will use this as it is a feature not many will need.  The main XAML element required is:  <AppBar>  Content  </AppBar> |
| **QUESTION 12: Data Modelling**  Explain Data Modelling.  Data Modelling is an easy way to represent or explain how an application will work or flow. It is often the first step in the designing of an OOP. It is the process of making sure that the model you have created will meet all the clients requirements and because it is only a model it can be easily changed without much work, this means that you cut down on redevelopment time if the application needs to be changed as all the changes should be made before the actual application has even started to be coded. Stakeholders are often interested in seeing some data modelling before investing so it is a good idea to have a firm grasp on how to use it effectively to portray your design. |

References:

<https://os.opencolleges.edu.au/E1161#/sub-topic/61945/10.4.1>

<https://www.w3schools.com/xml/xml_elements.asp>

<http://cscie12.dce.harvard.edu/lecture_notes/2007-08/20080130/slide13.html>

https://social.technet.microsoft.com/wiki/contents/articles/34645.uwp-understanding-appbar-and-command-bar.aspx

**Next steps for you:**

You have now reached the end of this assessment.

Ensure that you have completed all of the above tasks. Use the checklist to double check that you have submitted everything required for this assessment.

Zip all your files (including your Visual Studio project files) and upload to OpenSpace. Name your file 33131-01\_yourName\_studentNumber.zip. (For example, 33131-01\_JohnSmith\_17756433.zip).

Your trainer will provide you with feedback for this assessment – ensure that you read this feedback carefully.

**Checklist of items to submit as evidence for this assessment:**

|  |  |  |
| --- | --- | --- |
| **PART A** | | |
| TASK 1 | Written Answers to Task 1.1 and 1.2 | 🞏 |
| TASK 1 | Screenshot showing the project has been created in Visual Studio | 🞏 |
| TASK 2 | Screenshot of mockups in portrait and landscape modes | 🞏 |
| TASK 3 | XML file | 🞏 |
| TASK 4 | Project code | 🞏 |
| TASK 4 | Screenshot of UI displaying a selected contact, an updated contact and a deleted contact | 🞏 |
| TASK 5 | Testing Document with results | 🞏 |
| TASK 6 | Screenshots of the Application and Packaging tabs for the app manifest and the Select and Configure Packages screen from the Create App Package wizard | 🞏 |
| **PART B** | | |
| TASK 1 | Project code for web service | 🞏 |
| TASK 1 | Screenshot of the web services page displayed in your browser and a screen shot of the XML document returned when you invoke the web service from the web services page | 🞏 |
| TASK 2 | Screenshot of mockups in portrait and landscape modes | 🞏 |
| TASK 3 | Project Code for mobile app that consumes the web service | 🞏 |
| TASK 3 | Screenshot of UI displaying a selected product code and displayed data returned from the web service | 🞏 |
| TASK 4 | Testing Document with results | 🞏 |
| **PART C** | | |
|  | Document containing answers to 12 questions | 🞏 |
| **Name each assessment file appropriately and submit for marking.** | | |