

Student Practical Assessment (Exercise)

Business Unit/Work Group	IT Studies		
Qualification Code	National Code: ICT50715	Qualification Title	Diploma of Software Development
Unit Code/s	ICTPRG501	Unit Title/s	ICTPRG418 Apply intermediate programming skills in another language TAADE
Assessment Task Title	File Handling and Serialization		
Student Name	Submit your solution via your LEARN account	Student SIS ID	Submit your solution via your LEARN account
Assessor Name	You have been added to a LEARN group which defines your assessor. This is normally your Course Registration Number (CRN) lecturer.	Date	2018 Semester 1

Student Guide for Written Assessment	
Overview of Assessment	This assessment will require you to interpret a set of requirements specifications to design, develop, build, deploy and document a C#.NET UWP application to save and retrieve data by serializing/reserializing the data to/from a XML file on the device
Task/s to be assessed	You will be assessed on the successful completion the Application Development Specifications
Time allowed	The assessment must be completed and uploaded by the end of the first term of the semester
Location	You could complete this assessment in class or at another suitable site (e.g. at home)
Decision making rules	To receive a satisfactory outcome for this your solution must meet all the Application Development Specifications
Assessment conditions	<ul style="list-style-type: none"> You <u>may</u> be asked to explain your solution to the instructor in class You are not allowed copy or use code that is not yours without acknowledgement. You have till the end of the term from the time of release of this assessment to complete and upload your solution on the subject LEARN site
Resources required	To complete this assessment, you will need to use Visual Studio 2015 with the C#>NET Windows 10 SDK Toolkit and access to the subject Learn resources. Visual Studio 2015/Windows 10 SDK based machines are provided during this assessment. You can use a Mac if you prefer but these are not provided.
Results/Re-assessment	You will be provided with feedback for the assessment and be given the opportunity to resubmit any required corrections only once.
Submission Instructions	<ul style="list-style-type: none"> Create a folder called yourName_4CSM_Assessment3 on the desktop and add your completed NetBeans project inside this folder. Add your name to the comments at the beginning of each file.

File Handling and Serialization Project

IT Works has hired you in the role of a developer to work in a team of other analyst, to build an Windows based application for their client TafeSA. The client requires a simple multi page Windows based app to capture and store information from various Contacts. The application should target a variety of Windows based device families and use the devices local file storage capabilities.

The systems analyst has come up with the following application development specifications

Application Development Specifications

The applications should have the following UWP pages:

- Login page (*Login.xaml*)
- A Main page (*Main.xaml*) that has a list of menu options such as:
 - View all Contacts
 - Add a new Contact
 - Change Password.
- Page to List all Contact Names (*Contacts.xaml*) from isolated storage
- Page to display a Contacts details (*Details.xaml*) -when you select a Contact from *Contacts.xaml*, with the option of editing a contact and saving its details.
- Change password page (*ChangePassword.xaml*) to enable a user to changes the password.
- Page to add new Contacts and save them in isolated storage.

Business Rules

1. You have to use devices local storage to store all Contact and Login information.
2. Access to any of the pages requires the user to be logged in successfully. If the user tries to access any of the above pages directly, he/she should be redirected to the *Login.xaml* page
3. In keeping with good OO practices it is recommended you create a class to encapsulate a Contact's details such as full name, home telephone number, mobile telephone number and email.
4. All contact objects created must be held in a suitable data structures such as a List.
5. Use an appropriate control to inform the user of any actions (for example if the login is unsuccessful a Message Dialog Box could be used to inform the user of the issue).
6. Proper error handling and data validation strategies must be in place.
7. To keep the app simple, it is recommended that you hard code the users login credentials in local storage and maintain the users login status using a Boolean property in the *App.xaml.cs* class

