Document Details			
Version Number	**Last Updated**	**Developed/Edited By**	**Validation Date**
002	30/01/2018	Stewart Godwin	**Dec 2017**

Training Package	ICT Information and Communications Technology Training Package
Qualification Title	ICT40515 AWF2 Certificate IV in Programming
Qualification Title	ICA50715 AWE6 Diploma of Software Development
Course/Cluster Title	Programming II
Assessment Title	Project AT2
Brief Description of Assessment Task	
The analysis, design, coding, testing and project documentation of a C#.NET application.	

Units of competency, elements to be assessed

National Code	SIN	Competency Title	Elements of Competency
ICTICT403	AUU46	Apply software development methodologies	1. Determine and select appropriate methodology for a given activity 2. Apply the selected development methodology 3. Adjust project to suit appropriate methodology
ICTPRG418	AUV62	Apply intermediate programming skills in another language	1. Code using user defined data structures 2. Code using standard algorithms 3. Debug code 4. Document activities 5. Test code 6. Create an application

Date of Assessment	Week 7-10	**Completed by**	Week 10
Instructions to Students	The analysis, design, coding, testing and project documentation of a C#.NET application as described on the following page.		
Resources Required	Reference books / Internet / Blackboard / Visual Studio 2017/ MS Project/ Excel		
Instructions to Lecturer/Assessor	Assess each project using the exemplars. Ensure that each student gives a demonstration of their fully functional program.		

Lecturer's Details (Add your lecturers details below)

Name	
Email	
Campus	

Students to sign this document when submitting an assessment

Date Submitted:	

STUDENT DECLARATION

- I have read and understand the details of the assessment.
- I have been informed of the conditions of the assessment and the appeals process.
- I agree to participate in this assessment.
- I certify that the attached is my own work.

Student ID	Student Name	Student Signature

Assessment Feedback (Lecturer and Student Copy)

Assessment Title	Project AT2		
Candidate name		**Attempt No**	
Assessor name			
Performance demonstrated by this assessment is	Satisfactory □	Not Yet Satisfactory □	
	Assessment outcome and feedback received on	Date	

Assessor Marking Guide and Comments:

Criteria	s/ns	Comment
Program reads data from 3 binary files and writes data into List<t>		
Add ITEM button function; checks textboxes for data checks for item type add item to List<T> has error message for incorrect input		
Add CUSTOMER function; checks textboxes for data has error message for incorrect input add unknown customer add customer to List<T>		
Add Transaction function checks textboxes for data has error message for incorrect input add transaction to List<T>		
Listbox selection populates corresponding textboxes Correct functionality for “Items” Correct functionality for “Customers” Correct functionality for populating Transactions Correct functionality for “Transactions”		
Program reads data from List<T> and writes data to 3 binary files		
All data is realistic (must represent real names, and game data)		
All major controls have tooltips		
Report must have; Introduction Analysis with diagram Project plan with Gantt chart Activity diagrams for click methods and binary file handling Test data/table and screen captures Code headers and comments		
Training guide with diagrams and explanations.		

Candidate signature: *(once feedback has been received)*		**Date**	
Assessor signature: *(once feedback has been provided)*		**Date**	

Instructions

Introduction

Analyse, design, code, test and project documentation of a C#.NET application as follows.

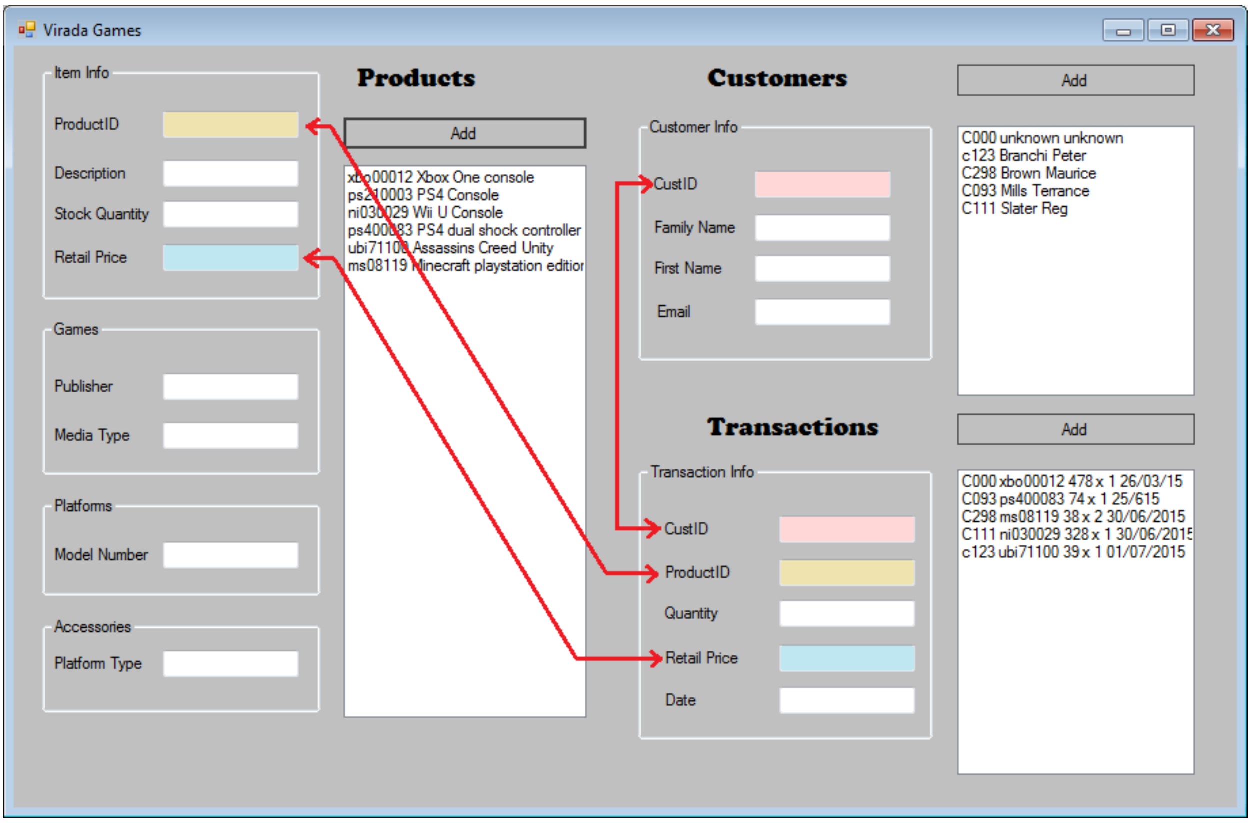
Case Study – Games Store

Virada Games is a company which specialises in the sale of electronic games for a variety of platforms. It also sells many other game devices and accessories. The company requires a stand-alone application which will be used in-store to keep track of all items and the sales to customers. The structure is as follows

- An Item class which has an ***itemID***, ***description***, ***stock quantity*** and ***retail price***, these attributes must be private, with associated assessor methods to store and retrieve information.
- There are three types of items as follows (these are derived classes of Item) with associated assessor methods to store and retrieve information,
  - Games which have a ***publisher*** and ***media type*** (DVD, Online, etc)
  - Platforms which have a ***model number***
  - Accessories which have a ***platform type*** (Console, PC, etc)
- Customers need to be tracked for marketing purposes if they give permission, create a class with private attributes and associated assessor methods to store and retrieve information,
  - ***custID***, ***surname***, ***firstname***, ***email address***.
  - If no details are provided they can be grouped as a single customer unknown with an ID of "C000"
- Transactions are a class with private attributes and associated assessor methods to store and retrieve information which contains the following info
  - ***custID***, ***itemID***, ***quantity***, ***retail price***, ***date***.

Design

The Graphical Interface is as follows



xbo00012 Xbox One console
ps210003 PS4 Console
ni030029 Wii U Console
ps400083 PS4 dual shock controller
ubi71100 Assassins Creed Unity
ms08119 Minecraft playstation editior

C000 unknown unknown
c123 Branchi Peter
C298 Brown Maurice
C093 Mills Terrance
C111 Slater Reg

C000 xbo00012 478 x 1 26/03/15
C093 ps400083 74 x 1 25/615
C298 ms08119 38 x 2 30/06/2015
C111 ni030029 328 x 1 30/06/2015
c123 ubi71100 39 x 1 01/07/2015

Functionality

The program functionality must satisfy the following criteria:

- The program loads the items information from a binary file called items.dat when the program starts into an appropriate List<T> structure.
- The program loads the customer information from a binary file called customers.dat when the program starts into an appropriate List<T> structure.
- The program loads the transaction information from a binary file called transactions.dat when the program starts into an appropriate List<T> structure.
- When the items add button is clicked
  - The information in the textboxes is checked to verify which type of item is to be entered
  - If information is in too many boxes then an error message popup is generated
  - If the information is correct then an item object of the correct type is created and the object is added to the List<T> structure and the ID and Description is added and displayed in the listbox.
- When the Customer add button is clicked
  - A customer object is created and the object is added to the List<T> structure and the custID, surname and first name is added and displayed in the listbox.
- When an item in either of the two upper listboxes (Item or Customer) is clicked, the information relating to that record is to be added to the textboxes to the left and also to the appropriate textboxes under Transaction, (refer: red arrows in Design Diagram).
- Before the Transaction add button is clicked the user must first select a customer and item. This action will populate the custID, productID and retail price textboxes in the transactions groupbox (ensure these textboxes are read only).
  - The information in the textboxes is checked to verify all data is present
  - If all data is present a transaction object is created and all the info is added as a single string to the listbox.
- When an record in the transaction listbox is clicked
  - The transaction info is filled into the remaining transaction text boxes
  - The appropriate item and customer are selected in the upper listboxes. This should autofill the upper textboxes.
- All data should be written back to the three binary files when the form closes
- Add realistic data to test the application
  - 4 customers,
  - 10 items (4 games, 3 platforms and 3 accessories),
  - 4 transactions
- Each of the controls should have tool tip text attached

Report

A report is also required, which has the following sections. Save in pdf format

- Introduction
    - a simple explanation of what the program is required to do
    - choose a development methodologies and explain why you believe that it would be the best to use in this situation (include a diagram)
- Analysis: a statement of
    - What data items need to be inputted
    - What processes need to be performed
    - What output is required
- Project plan: a statement of
    - List the tasks that need to be performed
    - List the job title for each task, when and in what order (UI designer, programmer, etc)
    - List the physical resources required for each task (PC, software, etc)
    - Draw a Gantt chart to show the above information (MS Project or Excel)
- Design
    - An activity diagram for each of the major code segments
        - Three add buttons
        - Read binary file
        - Write binary file
    - A class diagrams (UML or similar)
- Test Data
    - The data and activities that will be used to test the design. Include the test table of the data.
- Testing Evidence
    - Screenshots of the program functioning using the test data previously stated
- Code
    - A reference to the code (the first line of the method and associated comment)

Ensure that you include a reference to any resources you used to complete your responses.

Training Guide

Write a 2 page training guide and save in pdf format. The training guide should explain the usage of the application and each of the controls. There will need to be an image and appropriate details explaining various controls.

Demonstration

Before you can submit your final project and associated files you will need to demonstrate the fully functional program.

The following observation checklist will be used to verify your program

Observation Checklist		
Criteria	Yes	No
Loads “items.dat” binary file	□	□
Loads “customers.dat” binary file	□	□
Loads “transactions.dat” binary file	□	□
Correct functionality for Add Button in “Items”	□	□
Correct functionality for Add Button in “Customers”	□	□
Correct functionality for populating Transactions	□	□
Correct functionality for Add Button in “Transactions”	□	□
Correct functionality for transaction listbox	□	□
Save method to write data to binary files	□	□
Tool tips on all controls	□	□
Final demonstration outcome:		

Submission

Following is a checklist to help you check whether you have completed all requirements. Submit each as a separate file

Item	**Checklist**
Zipped solution folder	□
Report	□
Training Guide	□

End of Assessment Task