Government of Western Australia South Metropolitan TAFE	Assessment Coversheet Assessment 3– Project	Form no: F1020110 Issue date: 30/05/2019 Review date: 19/05/2020

Qualification details			
Training Package code and title:	ICT Information and Communications Technology Training Package		
National Qualification Code & Title:	ICT70515 Diploma of Software Development	**State code:**	AWE6

Student Name			
Student Declaration	I declare that the evidence submitted is my own work: ……..		
Assessor Name	Andrew Samway		
Date Due	Session 19	**Date Received**	Session 8

Unit of Competency			
National Code & Title	ICTICT509 Gather data to identify business requirements	**State code:**	
National Code & Title	ICTPRG523 - Apply advanced programming skills in another language	**State code:**	AUW10
Assessment Tool	**Assessment 3**- Project		
Demonstration Tasks	**Objective:** To demonstrate your ability to gather requirements and show advanced skills in programming you will document and deliver a project using some of the techniques you have used during this Semester. You may create anything; however it must be sufficiently complex to meet the requirements listed in Part 3. This may be an external client from industry or otherwise your client is your lecturer.		

Demonstration Tasks	**Question 1** – You need to create a document that outlines what you are planning to make that comprises the following:- 1. What data structures are you using? 2. Where are you using hashing techniques? 3. What sorting algorithm are you using how this is different from selection and bubble sort? 4. What search technique are you using? 5. What third party libraries are you using? 6. Where can I find the documentation for this? 7. A mock-up of the GUI. 8. What source control are you using? 9. What are your coding standards you are enforcing? 10. What tests are you going to run? **Question 2** – Create a Product Specification Design document. A sample has been provided. Your lecturer must sight this before you begin programming and sign off on it being appropriate. **Question 3** – Implement your solution - Must contain dynamic data structures - Must contain hashing techniques - Must contain sorting algorithm - Must contain searching technique - Must contain 3rd party library - Must have a GUI - Must adhere to coding standards **Question 4** – Complete test documentation which must include a test table, debugging screenshots

Materials and Equipment	Students will need a USB drive to save their work on.
Skills being assessed	ICTICT509 1.1 Identify information repositories across the business 1.2 Review current organisational documentation 1.3 Develop critical questions to elicit information from key stakeholders using a mixture of open and closed questions 2.1 Use a wide range of information gathering techniques

2.2 Review reports and other data sources for business information
2.3 Confirm with stakeholders business critical factors relating to current and future directions of the organisation

3.1 Analyse group and individual responses to clearly define business priorities
3.2 Document data analysis for review according to organisational standards

ICTPRG523

1.1 Design dynamic data structures
1.2 Implement and use dynamic data structures, including double-linked lists and binary trees for coding

2.1 Code using hashing techniques
2.2 Consider and record, the advantages and disadvantages, of at least three sorting algorithms
2.3 Code at least one sorting algorithm
2.4 Code advanced searching techniques for use with complex data structures

3.1 Use features of the language that enable inter-process communication through at least one mechanism
3.2 Use features of the language that allow for operating system 'signals' to be captured and responded to

4.1 Use a third-party library in the construction of an application
4.2 Reference third-party documentation
4.3 Use procedural techniques to write an application, to work within a graphical user interface (GUI) environment

5.1 Use integrated development environment (IDE) debugging facilities, or a stand-alone debugger
5.2 Use program debugging techniques to detect, and resolve, errors of syntactical, logical and design origin

6.1 Use source-code version control
6.2 Adhere to guidelines for developing maintainable code, and to a set of provided coding standards
6.3 Apply suitable internal documentation to all the code created, using the tools available in the target language

7.1 Design and document the tests to be undertaken
7.2 Undertake limited testing of the produced code, to ensure that it complies with the program specification
7.3 Capture the test results

8.1 Develop a solution, according to program-specification design document
8.2 Design the algorithm, and construct, and test, the application in response to a problem description and language

Assessment Decision	☐ Satisfactory	☐ Not Yet Satisfactory	
Assessor Signature		**Date**	Click here to enter a date.
Feedback to student			
Feedback from student			
Student signature		**Date**	