12SDD - Assessment #3

Task Name:	Major Project Part B
Weighting:	30%
Marked out of:	100
Outcomes Assessed:	1.1, 1.3, 4.2, 4.3, 5.1, 5.2, 5.3, 6.1, 6.2, 6.3, 6.4
Areas of Assessment:	Implementing the Solution Testing the Solution Modifying the Solution Developing a Software Solution

DETAILS ABOUT TASK:

The HSC SDD course requires a mandatory major software development project. It is the vehicle through which the theory of the Software Development Cycle is applied.

In Part A of your major project, you identified the problem to be solved, planned the solutions and identified the key features that you application would possess.

This part of the major project involves putting all of that planning into action. You are to build your application as close to the specifications from your proposal as possible. Any major deviations from this will need to be cleared by your teacher or Mr Howse.

Your application must be able to function on a test machine other than the school computers, which means that it will need to able to operate without hard coded URL's. Any minor problems will be fixed by the teacher in order to get the application to work. If, however, they proved to be serious errors, marks will be deducted.

Guidelines

	Task Component	Top Band Descriptor	Mark
Defining the Problem			
H	System Specs - Software	Complete spec sheet of software requirements – must be realistic and particular to the chosen project.	2
H	System Specs - Hardware	Complete spec sheet of hardware requirements – must be realistic and particular to the chosen project.	2
Planning the Solution			
H5.1	Build Log	Thorough bug report and build descriptions. Descriptions should exist for each minor build and major build outlining the features added, and the bugs removed. This component should be completed via **commit comments** in BitBucket. Build log should show **regular** commits to the code base and be detailed in their content.	10
H	User Manual	The user manual covers all the processes performed by the user, and contain screen shots and trouble shooting. Must also include installation instructions.	10
H	GUI Design Internal Help	Professionally designed GUI with all the appropriate features of good design. (Familiar, forgiving, robust, consistent, etc) Suitable prompts, tutorials and help given to the user throughout the application.	5 3
Building the Solutions			
H	Bug Free	Program operates machine to the de	Mark: 5
	Features of the Application	Design of at least three features within the application. The elegance and simplicity of the solution will be taken into consideration as will the complexity of the feature being attempted. (5 Marks per each feature)	15

H	Modular Programming	Design of application follows good modular design practices. Evidence should exist that common processes are found in only one place and referenced throughout the code. · Appropriate use of sub-routines/functions/modules (5) · One task per sub-routine/function (3) · Simple command line (2)	10
H	Intrinsic Documentation	All variables are clearly defined and their meaning is easily apparent	3
	Internal Documentation	Code is well documented – both the procedures and functions and the processes.	5
Testing the Solution			
H	Test Data	List of tests completed while designing the application demonstrating the selection of test data and the modules tested.	5
	Peer Report – Choose a classmate to desk check and comment on their code.	Peer report is thorough, including a desk check and concise summary of the good and bad. The desk check must test at least two of the algorithms within the code.	5
Modifying the Solution			
H	Justification of coding	Code is justified and well explained with examples from the program itself.	5
	Online Class Contributions Reflection	Online participation in class activities such as blogging, commenting, writing tutorials, and answering questions on the forums. The reflection will be an honest and thorough account of the project, highlighting the things done well, and where improvement is required.	10 5