



ASSESSMENT 4 BRIEF	
Subject Code and Title	SDM404 Software Development and Management
Assessment	Final Project Demonstration
Individual/Group	Group/Collaborative
Length	Presentation Duration – maximum of 12 minutes Individual contribution report 750 words +/-10%
Learning Outcomes	This assessment addresses the Subject Learning Outcomes outlined at the bottom of this document.
Submission	Submission due 11:55pm Sunday (Module 12). You will also be asked to demonstrate your project in your scheduled class time during Module 12.
Weighting	35%
Total Marks	100 marks

Task Summary

In Assessment 3 you need to follow the project plan that you have developed in Assessment 2 part B to develop the software application that you have proposed in Assessment 2 part A. The produced software needs to be of high quality and follow the coding standards that the group agreed on in the project plan. You also need to prepare a presentation about the process you have followed to produce the software product you have developed. Your presentation should not be more than 12 minutes. As part of that assessment you are also required to write an individual contribution report showing the tasks you have worked on, the problems you have faced while working on these tasks, the solutions you have proposed for those problems and the final solution you have adapted (with reasons for choosing that specific solution). The contribution report is expected to be 750 words (+/- 10%).

Context

This assessment requires you to demonstrate the complete and working software project have proposed and prepared the plan for. It assesses your programming skills, ability to follow a software plan, analyse and manage risks and the ability to follow a quality management plan as well. The demonstration assesses your ability to communicate your ideas to other stakeholders.

The contribution report measures how you can reflect on the work you have been doing and how you can analyse and solve the problems that you face.



Task Instructions

1. Follow the software plan that you have prepared to develop the software application
2. Make sure that the software produced is of high quality (E.g., no bugs, no errors, features are complete and correct) and that it follows the coding standards the team agreed on.
3. Make sure to follow your test plan and prepare a list of the tests that you will run.
4. Make sure to keep a bug report.
5. Prepare an individual contribution report:
 - a. Tasks you have worked on (task details/date/..)
 - b. Problems that you faced while working on these tasks (if any).
 - c. Proposed solutions to these problems and the final solution that you have chosen.

Referencing

It is essential that you use appropriate APA style for citing and referencing research. Please see more information on referencing here http://library.laureate.net.au/research_skills/referencing

Submission Instructions

1. Submit Assessment 4 via Assessment link in the main navigation menu in SDM404 Software Development Management by 11:55pm Sunday (Module 12):
 - a. Source codes and applicable libraries.
 - b. Individual contribution report.
2. The Learning Facilitator will provide feedback after your presentation and also via the Grade Centre in the LMS portal. Feedback can be viewed via My Grades.



Assessment Rubric

Assessment Attributes	High Distinction (Exceptional) 85-100%	Distinction (Advanced) 75-84%	Credit (Proficient) 65-74%	Pass (Functional) 50-64%	Fail (Yet to achieve minimum standard) 0-49%
Software Application meets project objectives 20%	<p>The project objects (functional and non-functional) are all addressed and show all aspects have been explored.</p> <p>The program meets all of the project specifications with no bugs.</p>	<p>The project objectives (functional and non-functional) are met with few cases that may be questionable and/or additional aspects need to be explored.</p> <p>The program produced the correct results and displayed them correctly for almost all project specifications with no bugs.</p>	<p>The project objectives (functional and non-functional) are met but clearly not in entirety and/or there were some clear deficiencies.</p> <p>The program produced correct results for most project specification and/or had few bugs.</p>	<p>Some of the project objectives (functional and non-functional) are met.</p> <p>The program produced some correct results and/or had several bugs.</p>	<p>Very little if any of the project objectives appear to be met in any clear way.</p> <p>The program did not work or had many bugs</p>
Quality of source code and Error Handling 10%	<p>High standard of coding and commenting was consistently applied. The code is easy to read and maintain.</p> <p>The program checked for all error conditions and handled them appropriately.</p>	<p>Good standard of coding and commenting were evident. Code is easy to read with some expressions needing improvement. Commenting is evident and code is somewhat easy to maintain.</p> <p>The program checked for most error conditions and handled them appropriately.</p>	<p>Reasonable clarity and code can be followed with some effort. Comments helped to understand the code. Logical organisation is lacking in some ways and there were a number of questionable expressions.</p> <p>The program checked for some error conditions and handles them appropriately</p>	<p>Code is somewhat readable but needed a lot of effort to understand. Some formatting is applied but not consistent. Very little comments were provided.</p> <p>The program checked for few error conditions and does handle them appropriately.</p>	<p>Extremely hard to read, little or no attention to code formatting, lack of logical organisation, little or no appreciation of coding and commenting practices.</p> <p>The program does not check error conditions.</p>



<p>Software process plan was followed</p> <p>10%</p>	<p>The process plan is followed in an exceptional way with a very good eye on each detail of the process.</p> <p>The schedule milestones are all met without putting any extra pressure/stress on the development team.</p> <p>Project risks are exceptionally anticipated and managed in a very good way.</p> <p>Quality measures are strictly followed and the resultant software produced reflects that.</p>	<p>The process plan is followed during the whole project development cycle.</p> <p>Almost all the schedule milestones are met.</p> <p>Almost all of the project risks are anticipated and managed.</p> <p>Almost all of the quality measures are followed.</p>	<p>The process plan is followed in most parts of the project.</p> <p>Most of the schedule milestones are met while a few are not.</p> <p>Most of the project risks are anticipated and managed.</p> <p>Most of the quality measures are followed.</p>	<p>The process plan is slightly followed.</p> <p>Few of the schedule milestones are met while most are not.</p> <p>Few risks are anticipated and managed.</p> <p>Few quality measures are followed.</p>	<p>The developed process plan is not followed</p> <p>None of the schedule milestones are met</p> <p>Risks are not analysed or managed.</p> <p>Quality plan is not followed/met as well.</p>
<p>Application of programming knowledge</p> <p>10%</p>	<p>Excellent choices of third-party libraries or systems and clearly established ability to use them with few if any limitations.</p> <p>Consistently showed application of programming skills and knowledge in terms of data structures, algorithms and other applicable theory. Where appropriate, showed</p>	<p>Appropriate use of some third-party libraries or systems with very little choices could have been better in hindsight.</p> <p>Showed reasonable demonstration of programming skills and knowledge in terms of data structures, algorithms and other applicable theory in most parts and showed some</p>	<p>Reasonable use of some third-party libraries or systems but lacked the capacity to make use of them and/or some choices could have been better in hindsight.</p> <p>Showed reasonable demonstration of programming skills and knowledge in terms of data structures, algorithms and other applicable theory, but only in some parts. Lacked</p>	<p>Used some third-party libraries or system appropriately.</p> <p>Showed some basic programming skills but some inappropriate programming approaches are used.</p>	<p>Demonstrated no or little ability to make use of appropriate third-party libraries or systems.</p> <p>Demonstrated no or little programming skills. Inappropriate programming approaches are used frequently.</p>



	sophistication and a high-level appreciation of programming intricacies.	sophistication in programming skills.	sophistication in programming.		
Test plan & Bug reports 20%	Test cases are thorough and systematic, well documented with expected and actual output.	Test cases are thorough and systematic, known bugs are documented	Tests cover most representative cases, tests and known bugs are adequately documented	Test cases miss significant scenarios, and are poorly documented; bugs are poorly documented but mostly incomplete and some are incorrect.	Test cases are absent or very few, and are poorly documented or undocumented; bugs not documented
Effective Communication (Presentation/Oral) 15%	<p>Engages and sustains audience interest. Expertly presented; the presentation is logical, persuasive, and well-supported by evidence, demonstrating a clear flow of ideas and arguments.</p> <p>Discerningly selects and precisely employs a wide range of specialised language and terminology.</p> <p>Clear, confident and persuasive delivery.</p> <p>Dynamic, integrated and</p>	<p>Engages audience interest. Information, arguments and evidence are very well presented; the presentation is logical, clear and well-supported by evidence.</p> <p>Accurately employs a wide range of specialised language and terminology.</p> <p>Clear and confident delivery.</p> <p>Confidently and</p>	<p>Presentation is easy to follow. Information, arguments and evidence are well presented, mostly clear flow of ideas and arguments.</p> <p>Accurately employs specialised language and terminology.</p> <p>Correct, but occasionally stilted or awkward delivery.</p> <p>Uses engaging presentation techniques</p>	<p>Presentation is sometimes difficult to follow. Information, arguments and evidence are presented in a way that is not always clear and logical.</p> <p>Employs some specialised language and terminology with accuracy.</p> <p>Correct, but often stilted or awkward delivery.</p>	<p>Difficult to understand for audience, no logical/clear structure, poor flow of ideas, argument lacks supporting evidence.</p> <p>Specialised language and terminology is rarely or inaccurately employed.</p> <p>Stilted, awkward and/or oversimplified delivery.</p> <p>Limited use of engaging</p>



	<p>professional use of a wide range of engaging presentation techniques (e.g. posture; eye contact, expression; gestures; volume, pitch and pace of voice; stance; movement)</p> <p>Employs succinct, creative and engaging presentation aids that effectively integrate a wide range of elements (graphics, multi-media, text, charts, etc.).</p>	<p>consistently uses a range of engaging presentation techniques (e.g. posture; eye contact, expression; gestures; volume, pitch and pace of voice; stance; movement)</p> <p>Employs succinct, styled and engaging presentation aids that incorporate a range of elements (graphics, multi-media, text, charts, etc.).</p>	<p>(e.g. posture; eye contact; gestures; volume, pitch and pace of voice)</p> <p>Employs clear and somewhat engaging presentation aids as directed. A few aspects require further refinement (e.g. amount of information, styling, editing, etc.).</p>	<p>Sometimes uses engaging presentation techniques (e.g. posture; eye contact; gestures; volume, pitch and pace of voice)</p> <p>Employs basic, but generally accurate presentation aids as directed. A number of aspects require further refinement (e.g. amount of information, styling, editing, etc.).</p>	<p>presentation techniques. (e.g. posture; eye contact; gestures; volume, pitch and pace of voice)</p> <p>Presentation aids are not employed or developed as directed.</p>
--	--	--	--	--	--



Contribution report 15%	<p>Report is exceptional in organisation and structure.</p> <p>Contribution to the process is described in an exceptional way with emphasis on the important contributions made to the process. Very good analysis of the problems faced, good discussion of the proposed solutions and good reasoning for the choice of the applied solution.</p> <p>Sentences are grammatically correct and free from spelling errors.</p>	<p>Report in total is well structured and organised.</p> <p>Clearly identifies own contribution to the whole planning process with detailed description of the problems faced, solutions adapted and reasons for the choice of these solutions.</p> <p>Free from spelling and grammatical mistakes</p>	<p>Report is well structured and organised for the most part.</p> <p>Detailed description of own contribution to the planning process, problems faced and solutions</p> <p>Few spelling/grammatical mistakes that does not affect the flow of the document.</p>	<p>Some parts of the report are structured well, but others are not.</p> <p>A good level of details is given about own contribution, problems faced and solutions.</p> <p>Report contains some grammatical/ spelling mistakes that would interrupt the flow of the document.</p>	<p>Report is poorly structured</p> <p>The report lacks a lot of details about own contribution to the preparation of the plan and the problems faced and solved.</p> <p>Report contains a lot of spelling and grammatical mistakes</p> <p>Word count is not met.</p>
--	--	--	---	--	--



The following Subject Learning Outcomes are addressed in this assessment

SLO a)	Explain the basic concepts of software project management, software development and project planning.
SLO b)	Evaluate different software processes, and assess their suitability in a given context.
SLO c)	Identify, analyse, and manage software project risks.
SLO e)	Plan an effective way of communication between the different stakeholders.
SLO f)	Apply a chosen methodology for managing a Software project throughout its life cycle.