Software Engineering Group Project

Stage 1 (“Final Product”)

This document describes Stage 1 of the Software Engineering Group Project and should be read in conjunction with the Group Project Guide, the Project Specification, and the Group Project information posted on Canvas. Stage 1 will count for 30% of the Group Project mark.

Overview

The group’s first task is to simulate making a bid for the project. The group bid consists of the production of the Requirements Specification, Project Decision and Planning, Risk Analysis, and Project Costing. You must also create mock-ups of your user interface and do a short usability evaluation. The expected content of the documents is given below. There will be Software Engineering lectures on each of the topics where more information is available. Groups should also be guided by the expectations of their Manager and the project requirements. Examples of good documentation from previous years will be available on Canvas.

Stage 1 specification

For Stage 1, you should produce a single, integrated report consisting of a number of sections as described below

1. Requirements Specification

The requirements specification should provide an introduction to the project, including its purpose and scope, and aims and objectives. A complete, comprehensive, prioritised description of the software requirements, both functional and non-functional, should be produced and organised into logical groups and numbered. Relevant UML and/or other diagrams could be included.

2. Risk Analysis

Identify the risks that your group is likely to encounter during the year and give details of a suitable strategy to manage these risks. Groups should focus on the risks that they as students might realistically encounter, not those that their pretend company might encounter. Risks should be considered in conjunction with project decisions and the plan (see below).

3. Project Decisions and Plan

In this stage, the group must decide on key issues such as which software to use, which software process to use, how the complete product will be divided into sub-systems (top level design), how the software will be assessed (for usability, technical correctness, and customer expectations), what each student expects to contribute to the project, and how the group intends to collaborate. Provide details of these key issues. Explain when the various tasks will be done, how long they will take, and who will do them. The plan should cover both the development and evaluation of the product and the production of all documents. Don't forget to allow time for other coursework, examinations, and holidays! All groups are asked to plan their product development using iterations, so at this stage you should decide on the software development methodology, number of iterations, their duration, and which top level tasks will be included in each iteration. Show dependencies between tasks. For the first iteration (which may have already started in Semester 1), break the tasks down into detail. Later iterations do not have to be planned in too much detail, but any tentative details around their planning could be included. The plan should be presented clearly, possibly using tables, lists, or graphics such as a Gantt chart.

4. Project Costing

The team should produce an overall budget for the project plus a breakdown of all relevant costs in each section. Consider all development, evaluation, and deployment costs. Any additional costs to the customer for hardware, proprietary software, etc. should also be specified. For each section of the budget, explain how the costs were arrived at. For example, for hardware or software hosting, give details of current prices from potential suppliers that you have discovered. For staff, explain how you have divided the group into roles, how much time each person will spend in that role, and what salary and overheads you have taken into account. Include references to where the numbers in your costing come from. Most of the staff costing should be based on the time that you actually have available and the roles that you take, but you can invent a few other future details such as training and maintenance.

5. Usability Evaluation of Mock-ups

Once the requirements are finalised, groups should design a mock-up of the interface for most of the system. This can be as simple as hand-drawn screens, PowerPoint with links, or using more sophisticated interface design software.

Once you have completed the mock-ups, test how usable the design is. Create an experimental protocol where you show participants in your study screenshots and ask them what they think certain components are for, to get feedback on overall usability. For this stage, just collect subjective data.

- **Subjective qualitative data**: ask participants open-ended questions on how they found the experience, what they liked/disliked. If you show them two different designs, you can ask them

which one they liked best. See the example testing protocol on Canvas for an example on how to do this. Be careful not to use leading questions (e.g., "I think this is a great feature, what do you think?").

- **Subjective quantitative data:** ask some questions on Likert or rating scales. Examples will be provided on Canvas, or you can create your own similar version specifically for your interface.

You are not expected to calculate complex statistics in this study since you may not have enough participants, but any summary statistics you can give will be helpful.

The following checklist may help you organise your usability study:

1. **Consent form:** Use the consent form template provided on Canvas, clearly stating what is involved in participating in this study.
2. **Ethics**: The project coordinator will obtain ethics for the evaluation on behalf of the groups. **Do NOT** have vulnerable people as participants (including children under the age of 18) or actively deceive your participants.
3. **Usability testing plan**: create a test plan (see an example on Canvas).
4. **Testing protocol**: create a detailed testing protocol, including screenshots if you ask participants to go to particular screens. Make sure everyone evaluating the interface follows this protocol so that results are consistent.
5. **Questionnaires (initial and exit)**: develop an initial questionnaire, querying the participants' demographics (e.g., age, occupation, experience with computers as necessary); to gather an overview of the participant's experience also give them an exit questionnaire. Base questionnaires on standardised ones, such as SUS, that use Likert or rating scales. Make sure to gather qualitative feedback.
6. **Pilot**: some members of the team should run through the evaluation as a participant to discover any obvious problems with it. Correct the protocol if necessary.
7. **Recruiting participants**: try to get at least 6-8 participants for your study. Try to get participants from a variety of backgrounds and ages, but relevant to the project topic.
8. **Perform study**: get each participant to (electronically) sign a consent form. Save these forms. Go through the testing protocol making detailed notes. Ask the participant to complete the exit questionnaire.
9. **Anonymising data:** all data must be anonymised, i.e., use participant numbers on any documents such as questionnaire answers. Have a master sheet with names/subject numbers kept safely by one group member. Remember any GDPR requirements with respect to the use of personal/sensitive data

You will find a number of resources on Canvas that may help you with your usability study (e.g., consent form, template for usability test plan, etc.).

- An introduction including aims and objectives
- Design decisions for your mock-ups/final product. Include some screenshots to illustrate your approach and add more information in an appendix if necessary.

- Provide a description of your experimental method, referring to your plan and protocol, i.e., what are you testing?
- Details of participants and their responses (e.g., number of participants), age, amount of computer use). Use descriptive statistics, e.g., percentage of males/females/other.
- Results analysis, which includes summarising qualitative feedback with lots of pretty graphs and descriptive statistics for the Likert/rating scale questions. If there are particularly low responses for any questions, this will highlight improvement areas.
- Conclusions, including recommended changes
- Appendix (don't dump anything here without referring to it in your text!):
    - Usability testing plan
    - Mock-up testing protocol
    - Sample blank questionnaire if used
    - Sample blank consent form
    - More screenshots if required. This is not needed for every screen but for every style of screen.

Stage 1 report structure

When writing the Stage 1 report, decide on a suitable style for this and all subsequent reports that you will submit throughout the project. Always use the same title page and include the document subject, your group name, Manager name, and names of students in the group. Include a table of contents in your report. Remind yourselves of what you learned in previous courses and make sure you get the terminology right. For Stage 1, you are meant to submit a single integrated report rather than a collection of reports on each of the topics. Refer to the F29SO lecture on Reporting for additional guidance on writing your reports. Keep the length of your report to a maximum of 50 main pages, not including appendices.

Deadlines and submission

Submit your completed Stage 1 report on the Canvas site for F29SO, with only one report submitted for each group. Check Canvas for the deadline for Stage 1, keeping in mind that the submission time is campus dependant.

Assessment

Software Engineering (F29SO) and Professional Development (F29PD) are both evaluated by coursework only (i.e., no final exam). Your F29SO and F29PD marks will each include individual coursework from the particular course (weighted at 34%) and your individual Group Project mark (weighted at 66%).

This coursework contributes to the Group Project mark and will make up 30% of the group's overall mark (before individual adjustments). Stage 1 will be marked by both the Project Coordinator and your

Manager. It will be marked out of 70 marks using the following mark distribution and assessment criteria.

Criteria	Poor	Adequate	Excellent
Requirements (out of 20 marks)	**0 to 7 marks** No clear aims and objectives. A list of software requirements but not prioritised, some missing and no use of the terminology of functional and non-functional requirements; requirements not numbered.	**8 to 13 marks** A brief introduction to the project. A list of software requirements but perhaps not prioritised or some missing. Use of terms functional and non-functional requirements; requirements numbered.	**14 to 20 marks** A comprehensive introduction to the project, including clear purpose and scope, aims and objectives. A complete, comprehensive, prioritised description of the software requirements, with both functional and non-functional requirements that are organised well, logically grouped, and numbered. Relevant UML

			diagrams are included.
Risk Analysis (out of 10 marks)	**0 to 3 marks** A limited list of the risks the group will likely encounter during the year. No strategies given to manage these risks. Risks are not mentioned in terms of likelihood and effect.	**4 to 6 marks** A reasonable list of the risks the group will likely encounter during the year. Some strategies given to manage these risks. Risks are described in terms of likelihood and effect.	**7 to 10 marks** A comprehensive list of the risks the group will likely encounter during the year. Sensible strategies given to manage these risks. Risks considered in conjunction with project decisions and plan (see next section). Risks described in terms of likelihood and effect. Other aspects such as risk type given also. Presented in an easy to understand manner (e.g., in a table).
Project Decision and Planning (out of 10 marks)	**0 to 3 marks** Very few decisions included from the following: software to use; which software process to use; how the complete product will be divided into sub-systems (top level design); how the software will be assessed for usability, technical	**4 to 6 marks** Most issues listed but some missing from the following: software to use; which software process to use; how the complete product will be divided into sub-systems (top level design); how the software will be assessed for usability, technical correctness	**7 to 10 marks** Key issues listed such as which software to use; which software process to use; how the complete product will be divided into sub-systems (top level design); how the software will be assessed for usability, technical correctness and customer

	correctness and customer expectations; what each student expects to contribute to the project; and how the group intends to collaborate. A poor project plan given not covering all stages or not allocating appropriate timespans. No graphical representation of plan.	and customer expectations; what each student expects to contribute to the project; and how the group intends to collaborate. A fair project plan given but perhaps not covering all stages or not allocating appropriate timespans. A graphical representation (e.g., Gantt/PERT) but perhaps no dependencies or not clearly presented. Perhaps lack of detail for the first iteration.	expectations; what each student expects to contribute to the project; and how the group intends to collaborate. A realistic plan given covering both the development and evaluation of the product and the production of all documents. A graphical representation of the plan (e.g., Gantt/PERT) showing dependencies between the tasks. The first iteration is broken down in detail with later ones described at least at a high level.
Project Costing (out of 10 marks)	**0 to 3 marks** A budget given but many missing details, e.g., costs not broken down or details on development, evaluation, and deployment costs Missing. Customer costs not considered and details missing for the explanations	**4 to 6 marks** An overall budget given for the project but perhaps not a comprehensive breakdown of all development, evaluation, and deployment costs considered in the costing. Not all additional costs to the customer are	**7 to 10 marks** An overall budget given for the project plus a breakdown of the costs of each section. Development, evaluation, and deployment costs considered in the costing. Any additional costs to the customer for

	on how the costs were arrived at. For staff, little information on the following: how the group have divided into roles, how much time each person will spend in that role, and what salary and overheads. Limited/no cost references included.	considered for hardware, proprietary software, etc. Some details may be missing for the explanations on how the costs were arrived at. For staff, missing information from some of the following: how the group have divided into roles, how much time each person will spend in that role, and what salary and overheads. Additional cost references could be included	hardware, proprietary software, etc. are also specified. Explanations on how the costs were arrived at are given. For staff, explanation given on how the group have divided into roles, how much time each person will spend in that role, and what salary and overheads are taken into account. Appropriate cost references included.
Usability Evaluation of Mock-ups (out of 20 marks)	**0 to 7 marks** A study evaluating interface mock-ups but with some aspect which is either unethical or the mock-ups have not been given adequate thought. The experimental protocol is flawed in some way or did not include the use of a consent form. Questionnaires may contain questions that are leading,	**8 to 13 marks** A reasonable study evaluating interface mock-ups but perhaps not completely thought through with respect to design. 6-7 subjects recruited with a good experimental protocol followed including the use of a consent form. Questionnaires may include some questions that are	**14 to 20 marks** A well-run, ethical study evaluating well-thought-through interface mock-ups. 6-7 participants recruited with a sound experimental protocol followed, including a valid consent form. Questionnaires are either a standard (e.g., SUS) or include questions that are well thought through and without leading

	ambiguous or unethical. Limited discussion of findings and recommendations given. No use of descriptive statistics. Data not anonymised.	leading or ambiguous. Some findings and recommendations given. Limited use of descriptive statistics.	or ambiguous questions. A table of findings is given, along with a discussion on which ones to take forward. Some descriptive statistics used to describe the participants' demographics and subjective data

Feedback

Written feedback will be provided to the group approximately three working weeks after the submission/completion of Stage 1. Your Manager may also provide additional verbal feedback to the group.

Learning Objectives, Late Submission of Coursework, Mitigating Circumstances, and Plagiarism

Please refer to the **Group Project Guide** for information about the learning objectives for the Group Project and details on the relevant course/university policies. These policies are also posted on Canvas. You are responsible for reading and understanding these policies for completing the project.