

1 Introduction

The goal of this project is to develop a quality software architecture for a web based personal organiser. This must be done by establishing and managing a process for creating the architecture and for working collaboratively in teams to achieve this aim. The project will be conducted throughout the semester and it is expected that you will apply the lectures and readings to your project work where appropriate.

There are three key submissions which determine the major milestones of the project. Apart from the submissions which are at set dates, you are expected to establish processes, choose software tools and apply software engineering principles to the development of the software architecture independently as part of a *development team*.

You will be assessed on the quality of your submissions, and on how well you exhibit an understanding of the practice of software engineering.

1.1 Aims

Software engineering is a practical discipline concerned with *engineering* software. The aim of this project is to give you some of the experiences necessary to understand the issues in software development, and the difficulties encountered in software development, for example, in managing complexity. Certainly, we expect that you will experience some of the issues of:

- Working in teams and managing teams;
- Establishing and managing processes;
- Dealing with project risk;
- Understanding the role that clients play in the software development process;
- Managing clients, team members and cooperating with other teams; and
- Controlling the quality of artifacts in a software development process.

The project was formulated from the idea that personal preferences for something like a personal organiser can vary and that the requirements of one client are unlikely to match completely the requirements of another. Your team will be independently gathering requirements from a number (most likely three) clients. These will then be reviewed and merged into a single software requirements specification.

2 The Project Brief

The project brief for 433-341 is to develop a quality assurance plan, elicit requirements and develop an SRS and then produce a Software Architecture for a personal organiser (see Section 2.2). Exactly what type of web-based personal organiser is up to the three students who are acting as the clients.

This project is different from past 341 projects in that three members of the team will be acting as the clients (at least in the early stages of the requirements). This has several advantages.

It provides an opportunity to experience requirements elicitation from the point of view of the client.

The specific requirements of three different individuals for a personal organiser are likely to be different enough to provide you with some interesting decisions when it comes to deciding what to build.

The group becomes effectively “self-contained” removing the risks associated with access to clients for 341 (a large class that lasts only 12 weeks).

Provided that you meet the submission deadlines you are free to design a process that will achieve this brief. To get you started, there are a number of key steps to take.

But we don't know what to build!

Three of you will be acting as clients for this project. You get to decide what to build. The only guidance that we will give you is that it should be a web based personal organiser. There will be an example provided in the lecture to show you what we mean. Remember that you must develop an architecture for it and the more sophisticated and complex that you make it the harder it will be to design. For example whilst it might be nice to include an SMS messaging service that notifies the user of share price changes it is probably well beyond the scope of the project. If you *must* get carried away, then it might be best to have a wide scope for the initial three SRS and then to cut scope for the final one. Again it is up to you but remember that you will be assessed on your ability to solve problems, apply processes, and to document your designs and not on the actual functionality that your personal organiser might provide. Making it so simple that the requirements are trivial will not impress us however. Put simply, make the scope as large as you believe your processes can reasonably deal with.

Step 1

You will need to get into your teams and settle your team structure, roles and responsibilities and your team goals fairly rapidly. It will be necessary to meet as a team outside of lectures and tutorials in order to do this. Make sure you elect a project manager!

Step 2

You will need to create a process fairly rapidly. Some examples of what your process should take into account are, how you are going to collaborate with the other teams working on your project, the "*interfaces*" between teams, repository management, review pass/fail criteria and managing risk factors. There are other factors that you need to take into account as well.

You will also need to think about how you are going to assure quality, that is, that the architecture can be implemented and that it meets the client's needs.

There will be a template on the web that you can use as a guide, **BUT** filling in the template without tailoring it to your project or without justification *will* suffer penalties.

Your processes are submitted in the form of a modified SQAP.

Step 3

Split into pairs 2 and commence the requirements elicitation process. This should be done independently and the "clients" should attempt to genuinely play the part of clients and not simply treat it as a collaborative task to write an SRS. That said we expect that both students in the pair will be involved in writing the SRS and the project manager might be involved to ensure that processes are followed and that the three groups use a standard template.

2.1 Teams

Where possible the team should be composed of 3 students who are enrolled in 340 and 4 students who are not. Class numbers indicate that there will be a slight oversupply of non-340 students and so groups with non-340 students only are free to form.

Unlike other years there is no relationship between the 340 and 341 projects so there is no requirement that the 341 group reflect any 340 project group structure. Nor is there a requirement that you form groups based on your tute groups.

Regardless of the origins of your team you will be expected to act professionally in your team and to manage it responsibly.

You will need to assign a number of roles to people in your team. In turn you are expected to fulfil those roles and not let other team members do all of the work. The team should at least elect a *Project Manager* with the responsibility of ensuring that tasks are allocated and that adequate documentation is kept throughout the project.

A more comprehensive list of roles will be placed on the subject webpage as a guide.

Your team should agree on the roles and responsibilities of team members as soon as possible and write them into your project SQAP. Having a written record can sometimes help in disputes or help you to see where you have missed something.

2.2 Submissions

The submissions in Table 1 are, in effect, milestones in your project plan. The submissions in Table 1 are for each team, that is, only a *single* submission for each team should be made.

Submission	Due Date	Percentage of Final Mark
The Quality Assurance Plan (SQAP)	Monday, 31st March, 05:00pm	10%
The 3 SRS	Monday, 14th April, 05:00pm	
The 3 SRS Reviews and final SRS	Monday, 5th May, 10:00pm	20%
The Software Architecture Design Document (SADD)	Friday, 26th May, 10:00pm	20%

Table 1: Submissions for the project.

There SRS documents consist of 3 software requirements specs (one for each pair), and 3 SRS reviews. A fourth *unified* SRS that represents a complete, merged, version of the three individual SRS with a justification of the inclusion or exclusion of the requirements. It is this document that will be used as the guide for the software architecture.

The SRSs will be swapped within the group and reviewed by another pair. It makes sense to include the Project Manager in these reviews and create a draft of the merged SRS as the reviews are undertaken but you may adopt any process that you like. Whilst we expect that the final SRS will be of high quality we don't expect you to spend a large amount of effort producing it. If you can justify it, it will be acceptable to have the final SRS very similar to one of the 3 originals. You might decide that one of the SRS is pretty good and it simply needs to include a few of the requirements or ideas from the other two.

As well as the team submissions, every person is asked to submit a brief report estimating the percentage of the final submission made by each team member. A template will be provided for individual submissions. It is also necessary to submit a brief record of all meetings held. These minutes will allow us insight into the processes that you follow.

The rule for assessment is that you will be marked as a team, but if there are serious discrepancies in the workload or in the assessments made of workload in the individual reports, we may mark you as individuals instead of as a team.

2.3 Process Models for 433-341

The key milestones in the form of submissions for 433-341 are given and can't be changed. How you achieve those milestones is not!

Advice for how to establish and manage a process will be available from several sources. This introduction to the project cannot list all of the sources of help that are available, but here are some possibilities.

- Firstly, some of the lectures will deal with problems that often face software development groups.
- Secondly, bring up problems that you are having in tutorials and see if others in your group have some workable suggestions.
- Check back to the project page on the web frequently. We will try and provide you with a number of good examples and project resources. These will include templates and checklists for your submissions.
- The references contain a wealth of useful information and background to the issues. Where possible, use the references to plan out what you are going to do before you actually do it.

3 Assessment

The percentage weighting for each of the submissions is given in Section 2.2. However, here is what we expect from your submissions.

As part of your project work you will be expected to

- develop, manage and document your processes for obtaining and critiquing the requirements, and
- develop, manage and document your process for developing the SADD from the requirements.

In submissions you need to demonstrate your ability to:

- understand and *critically* analyse new and unfamiliar situations;
- understand the issues and trade-offs in developing good solutions to client's problems; and
- exercise control over the quality of the artifacts that are developed.

In this subject you are expected to demonstrate these qualities by documenting:

- the rationale behind your choice of process;
- the issues, trade-offs and rationale for your designs; and

- the reasons for your confidence in the suitability of your requirements, and the correctness of your designs.

In short, you are expected to justify your choices and to argue a case for why we should believe that your analyses and designs are appropriate and correct for your problem.