

Software Engineering with Objects and Components 1

Group Tutorial Project: Handout 1

September 2005

This project should be undertaken in *teams of 3–4 people*. The aim of the project is to give a small-scale case study within which to gain experience Software Engineering using UML (and Java) in collaborative software development.

The remaining sections in this handout outline the project's organisation, overview and deliverables. Subsequent handouts will provide details on deliverables and assessment. You should read *all* of this handout before commencing any activity on the project.

Organisation

This section describes the organisation of the project. You should read it carefully and ask if you do not understand anything.

Communication

Notices relating to the project will be distributed to the class via the tutorials and archived on the course web page. Messages relating to the project will be posted to the SEOC1 mailing list and newsgroup: `eduni.inf.course.seoc1`. *It is your responsibility to keep in touch by attending tutorials and reading the newsgroup and course web-page.*

Deliverables

The work you are required to do is packaged into two deliverables. Each contributes to your final assessment. A deliverable consists of: (i) A piece of work done by the team; together with (ii) An assessment of the effort contributed by each member of the team (this is expected to be 100% each.)

How you decide to partition the work which contributes to a deliverable is a matter for the team. At the end of each deliverable report there is a section which allows the team to assess the level of contribution of each team member to the deliverable. If any team member is dissatisfied with the working of the team they are entitled to submit a “minority report” on a separate copy of the deliverable report form for that deliverable. Assessments of contributions and “minority reports” will be used in determining the individual mark allocated to a team member. The phasing of the deliverables is:

Deliverable 1: Final deadline **5pm Monday 31st October.**

Deliverable 2: Final deadline **5pm Monday 5th December.**

Submit your work to the ITO, JCMB, Room 1502.

Part of the work of this project is deciding on the phasing of work such that you can meet the above deadlines. The precise content of the deliverables should be decided in consultation with your tutor. You should keep in mind that you have only limited time to devote to the project.

This document, together with the supporting details on deliverables, provides enough information that you can decide how to go about tackling the problem. It may well be that

you should plan to submit work some time before the deadlines to allow for slippage and for other work to dominate close to the end of term.

Overview

The overall aim of the project is to give you experience of collaborative activities of the kind typically undertaken in software development and the rôle of modelling languages such as UML in those activities.

The School of Informatics is in the process of developing a Secure Coursework Submission System (SCSS). This has been underway for some time and a partial requirements analysis and design has been completed (see the links on the practical page). Your tutorial group is one of four (competing) software companies who are bidding to win this lucrative development contract. As a tutorial group, you will be asked to develop the specifications for SCSS. Bearing in mind the limited resources you have to devote to this project, your group will split into three teams, each of whom will produce the specification for a part of the overall system. This will give you some awareness of the issues involved in the development of systems by structured teams, and of the use of UML in expressing and communicating models of systems.

Deliverable 1 concentrates on constructing a suitable requirements document (incorporating Use Case Models) for SCSS, providing the specification (incorporating Class Models) for an initial development iteration of the system, and providing supporting evidence validating that specification (e.g. using CRC cards). This contributes 50% of the final coursework mark.

Deliverable 2 concentrates on the design and test of a small “proof of concept” prototype of SCSS. This contributes 50% of the final coursework mark.

Problems and Grievances

There are many potential risks in this project. For example: you may find UML difficult; preparing for tutorials may take more time than you can afford; you may have difficulties working with certain teams in your group, or even with certain members of your own team.

If you are finding difficulties with the work, speak firstly to the other members of your team to see if you can solve the problem. If this does not work, speak next to other teams in your group. Finally, if this does not solve the problem then your team/group should present their problem to the tutor.

Your tutors’ job is to ensure that the tutorials run (relatively) smoothly and that they cover the necessary work. The preparation for the tutorials is your responsibility. You should expect to spend several hours a week on preparation for tutorials (including deliverables). If you find that you are spending considerably longer than this, then you are probably going into too much detail in your specification. If this happens, consider taking a more abstract view of the system, or restricting to a smaller part of the system.

Finally, if you find that you are having irreconcilable difficulties with other members of your team, you should (politely) inform your tutor as soon as possible.

MASSIMO FELICI
SEPTEMBER 27, 2005