Software Engineering with Objects and Components
Group Tutorial Project Handout

This project should be undertaken in *teams of 3–5 people*. The aim of the project is to give a small-scale case study within which to gain experience in Software Engineering by using UML (and Java) in collaborative software developments. The project aims to give you experience of collaborative activities of the kind typically undertaken in software development and the role of modelling languages such as UML in those activities. The remaining sections in this handout outline the project 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.

Project Organisation

This section describes the organisation of the project - read it carefully and ask any relevant clarification.

Communications

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 SEOC mailing list and wiki. *It is your responsibility to keep in touch by attending tutorials, contributing to discussions and reading the newsgroup and course web page.*

Deliverables

The work, you are required to do, consists of two deliverables. Each one contributes to your final assessment. A deliverable consists of: (i) A piece of work done by the team, together with (ii) a deliverable assessment of the effort contributed by each member of the team (this is expected to be 100% each.) and the deliverable itself. Each deliverable assessment has a section which allows the team to assess the level of contribution of each team member to the deliverable. Assessments of contributions and deliverables will be used in determining the individual mark allocated to a team member. How you decide to partition the work which contributes to a deliverable is a matter for the team.

Deadlines

The deadlines for the deliverables are:

Deliverable 1: 2pm Friday, 29th October 2010

Deliverable 2: 2pm Friday, 26th November 2010

Submit your work to the ITO.

Part of the work of this project is deciding on the phasing of work such that you can meet the above deadlines. 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.

Case Study Overview

P&G has commissioned the School of Informatics to design and develop a *Supply Chain Logistics System (SCLS)*. Your tutorial group acts as a software company, who is bidding to win this lucrative development contract. As a tutorial group, you will be asked to develop the specifications for the 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.

Deliverables

Deliverable 1 concentrates on constructing a suitable requirements document (incorporating Use Case Models) for the SCLS, 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 the SCLS. 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. If this does not solve the problem then your team/group should present their problem to the tutor. 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.

Your tutor's 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. The precise content of the deliverables should be decided in consultation with your tutor.

Massimo Felici
September 30, 2011