

University of Wollongong
School of Computing and Information Technology
CSCI222/CSCI822/MCS9222
System Development
Autumn 2016

Assignment 1 - Group Assignment (15 marks)

TASKS

Your tasks are to:

- Form and structure your group, allocating roles and responsibilities to your members;
- Elicit and specify requirements for the system, including all **detailed** *functional and non-functional* requirements for the new system, the data that the system will need to manage, interfaces to other systems, and interfaces and functionalities for different classes of users. The lecturer and tutor will act as clients for this project. You would also need to explore existing similar systems that are available online.
- Complete the planning and the requirements analysis for the system that is described in the attached informal specification document.
- Use a versioning system of your choice.
- Produce a report detailing the group's work.

The book “*The rational unified process: an introduction*” by P. Kruchten¹ contains explanations of the Rational Unified Process (RUP) roles. Details include lists of artefacts that are the responsibility of a person holding a particular role, and workflow diagrams. Students should check these outlines to gain guidance as to what they are to do in the project.

Each group will hold formal minuted meetings. The group member holding the "manager" role is responsible for these meetings, in particular preparing agendas and keeping meetings on track, and is also responsible for the reports on meetings. The meeting structure should be along the following lines:

- Reports from individuals who were allocated "urgent action items" in the previous meeting; discussion of issues related to these "urgent action items";
- Manager reviews his/her perception of the current status of the project; this review is to use data from bug reports and feature tests as created by the systems integrator;
- Scheduled agenda items are discussed - in the early meetings this will involve resolution of open design issues, later it will involve review of progress on implementation elements and integration;

¹ The Kruchten book is available as an electronic resource through the University library's subscription to Safari.

- Reports by individual members on progress in their areas of responsibility ("code walkthroughs" may be used) and identification of any problems;
- Identification of any "urgent action items" and allocation of responsibility for these;
- Proposals for agenda items for the subsequent meeting.

Each group member will maintain a work diary. This should cover: planned work schedule, actual work times, summary of work completed, report on "defects" in own work. The purpose of a work diary is really to provide data that identify ineffective work practices so that remedial action can be taken². Often you will find planned and actual work times differ significantly; such differences may point to poor work management practices. Every developer has his/her own blind spots that lead to defects in code. Analysis of defect patterns can help identify weaknesses and thus lead to a pre-emptive approach where the developer checks for their typical errors prior to running code.

REPORT

The work of the group is to be presented in a report. The report is the basis for your assessment. The report should briefly present the product that you are developing and provide evidence for a competently handled development process. It should use grammatically correct English and should not contain innumerable spelling errors. You should use the spelling and grammar checkers in your word processor application.

The report should be structured as below:

- Business case: What is this system that you are to build? Who will it serve? What will it do for them?
- Detailed plans for the whole project (from requirements until implementation).
- Risks and counter measures.
- Software Requirements Specification
- Use cases (including use case diagrams and detailed description for each use case)
- A domain model with detailed description of each class.

The "meta-report" part should include:

- A tabular summary of the group structure identifying group members, the roles that they filled, the artefacts that they successfully delivered.
- Group meeting records and individual diaries:
 - There should be a summary detailing the work done at each group meeting.
 - There should be agendas and reports from all group meetings.
 - There should be work diaries of all members of the group.
- Provide evidence for the appropriate use of version control software; this would typically take the form of excerpts from subversion's logs of commit operations. Subversion (or

² Take a look at Humphrey's Personal Software Process; it will give you helpful suggestions as to what data to record, and how to use such data to improve your work practices.

equivalence) statistical reports, showing overall contributions by different members, should be included in the report.

SUBMISSION

Your group is to submit a **hardcopy** of the report to the lecturer in **the Wednesday lecture of Week 7 (13th April 2016)**. *The exactly same softcopy of the report must also be submitted to eLearning by one member of your group by the same date.* The report **must** contain a “member contribution assessment” section. This should identify the member contributions on the ordinal scale “contributed”, “some contribution”, “almost no contribution”, or “no contribution”. This section **must be signed by all group members**.

MARKING SCHEME

- Overall presentation of report (2 marks)
- Requirements elicitation and analysis efforts as evidenced by your report (9 marks)
- Project management (4 marks)

These marks are based on evidence for the use of an effective software development process. Factors contributing to this part of the assessment will include: evidence for an effective group structure and adoption of roles; effective group collaboration as evidenced by meaningful group meetings; well-planned and disciplined work processes of individuals; and effective use of a versioning system.

Individual marks are based on the group mark. Individual marks are adjusted in accord with the “member’s contribution assessment” document submitted by the group. Individual marks *cannot* be greater than the group mark.

APPENDIX: QUESTION-ANSWERING SYSTEM

You are asked to develop a question-answering system (written in C++ or Java). Using this system, users can ask and answer questions, vote questions and answers, accept an answer, provide comments and gain reputations (see StackOverflow <http://stackoverflow.com> for an example).

This system consists of the following subsystems:

- A Profile Subsystem that manages users, memberships, reputations and topics of interest.
- A Question Subsystem that manages questions, (correct) answers, their votes and associated comments.
- A Search Subsystem that support recommending questions and query for a specific question.
- A Reporting Subsystem to generate various reports such as the most interesting questions, weekly questions, monthly questions, the top users (in terms of reputations), etc.

You must elicit further requirements from the lecturer and the tutor, and specify them in much more detail.

The client has also provided **some sample data** (in the enclosed file) that you should utilise.

Note: The first part of this project (Assignment 1) involves planning and eliciting and analysing requirements. The second part (Assignment 2) will involve designing, implementing and testing the system.