Last Updated: 11/19/2021 11:56:46

SENG1031 Software Engineering Workshop 1

Course Outline

Session 2, 2013 [25 July 2013]

Contents

1 Course Information Summary
2 Course Philosophy
3 Course Aims
4 Learning Outcomes
5 Staff and Web Access
6 Lecture Times and Venues
7 Syllabus and Teaching Strategy
8 Communication with students
9 Assessment
10 Group Work
11 Conflict Resolution
12 References
13 Computing Resources
14 Academic Misconduct

1 Course Information Summary

Course Code	SENG1031
Units of Credit	6
Hours per week	5
Prerequisite	none
Excluded	SENG1010, SENG1020
Lectures	Tuesday 1100-1400
Location	Electrical Eng 418
Staff	Boualem Benatallah & Denis Weerasiri

2 Course Philosophy

Software Engineering is about designing and developing high quality software systems. People involved in software design and development must consider the full context of the proposed system including task requirements; user requirements and attributes; and technical possibilities and limitations. These considerations, as well as the ethical and legal implications, all form an integral part of the design process.

This workshop will offer students the opportunity to concentrate on software requirements analysis issues including artifacts produced as well techniques and tools to support this process. In addition, it aims to provide students with some of the skills, techniques and information that an engineer would use in the early stages of the development process.

3 Course Aims

This course will be project driven and students will have an opportunity to work on a significantly sized project that will tax your knowledge and skills acquired in this course.

In addition, the importance of people in technology, and the issues related to the quality of project management will also be emphasized. The role of managers, users, designers, programmers and analysts are all critical throughout the system's development process. Students will have an opportunity to exercise their project management skills and to develop project plans.

Aside from the primary task of building software systems, practitioners must also be able to critically review and evaluate such systems. In the latter part of the workshop, students will be required to critique the project report produced by other teams as well as your own. Our goal will be to improve your skills and techniques in being able to critically review software designs and to report on your findings.

4 Learning Outcomes

When students have successfully completed this course they will have a conceptual framework for:

- An appreciation of the role of Software Engineering in Software System development
- Experience with the development of project plans, design documents, status reports and results
- The recognition that production of quality software is a task demanding a disciplined approach to all stages of its development

- An understanding of the concepts and relevance of each stage of the software development life cycle
- An appreciation of the many and varied issues involved in the development of software systems
- The role and the importance that Software Engineering review processes play in producing quality systems

In particular, students will also learn and apply generic skills including interviewing techniques, the design review process, developing team skills, working creatively through group work and brainstorming and managing the varying levels of uncertainty involved in these processes. A primary goal of this course is to teach students the importance and process of group work.

Overall, students will learn that both behavioral (human based) and technical views (computer based) of design and implementation are equally important issues. In order to develop long-lasting and efficient software systems—in essence a quality product—Software Engineers need to be aware of and address both these factors in the development process.

Less tangible outcomes are for students to also see that specifications are not the panacea of correctness, specification can be wrong and more importantly they can be subtly wrong. In addition, students will see first hand how assumptions and bias can sometimes be embodied in a specification unintentionally.

In addition, students will also develop an awareness of the community of engineering professions and the importance of keeping current through life-long learning and through interacting with that community. Students will also be encouraged to develop their research skills as one of the means of acquiring the necessary knowledge and skills to solve engineering problems.

5 Staff and Web Access

Name	Location	Phone	Email
Boualem Benatallah (LiC)	K17 - 407	9385 4767	boualem@cse.unsw.edu.au
Denis Weerasiri	K17 - 412-03	9385 6531	denisw@cse.unsw.edu.au

The course website address is:

- http://www.cse.unsw.edu.au/~se1031

Important: You must check this site regularly for updates and announcements. If you miss an important announcement because you did not check the class website, then you will have to wear any consequences that may result. No special dispensation will be granted as a result.

6 Lecture Times and Venues

Three consecutive hours per week has been set aside for the course. It should be note this is a project-based course and hence lectures are only used to introduce software engineering techniques, methods, and tools, which are relevant to project deliverables. The lectures include briefs about project deliverables, specific techniques and tools. The course will also features guest lectures on specific Software engineering techniques, methods, and tools. Lectures will be 1 or 2 hours depending on the subject matter. Students should avoid scheduling tutorials, laboratory classes from other courses in this time slot.

Time:Tuesday 1100 - 1400
Location:Electrical Eng 418

No lectures will be held in week 13.

During Presentation Week, you are expected to make suitable arrangements to be present for the majority of the presentations. While every effort will be made to hold the presentations during the allocated Lecture slot, additional hour(s) may be required depending on the number of teams formed this semester.

7 Syllabus and Teaching Strategy

Unlike the majority of the courses in your degree, all the SENG workshops are project driven courses focusing on the practical application of software engineering skills acquired from other courses in the degree program.

The workshops are not designed to be lecture-based courses. Lectures in this course are there to provide students with support and guidance in developing their practical skills. Students are expected to engage themselves in research and self-study of the materials required for this course.

8 Communication with students

In addition to the UNSW email address (Z-mail), the School of CSE provides students with a login account and email. You are hereby advised that:

- It is the policy of the School that all contact and announcements will be made via your official CSE email address only; and
- The course web page will be the other main source of announcements and information.

Students who choose to use other email accounts do so at their own risk as they may miss important announcements. You are advised to redirect your CSE email to your preferred email account. You can redirect your CSE email by using the mlalias command. Click here for more information on using this command.

If you do not want to redirect your CSE email then it is your responsibility to check your CSE email account and the course web page on a regular basis.

9 Assessment

In order to pass this course, a satisfactory performance is required in each component of the assessment. The assessments will include the following delivarables.

- Brainstorming phase
- Various software specification artifacts (architecture diagrams, state diagrams, sequence diagrams, stories etc.)
- Software prototyping phase
- Group presentations
- Peer reviews

The overall marking scheme for the deliverables will be available during week 1. Details about deliverables' deadlines and requirements will be available during the semester.

Late Penalty: A late penalty of 10% per day applies to late submissions. The penalty applies to the assessed value of the deliverable; individual marks will then be calculated using the peer assessment formula after the penalty has been applied.

10 Group Work

For the project, students will be placed in teams. You are expected to spend, on average, half an hour with your group mentor once a week. It is up to the team to negotiate a convenient time to meet weekly with your mentor.

Important: Everyone in the team must be present at each and every mentor meeting so it is essential that you find a time that is convenient for all members of the team. Missing your mentor meeting without a valid reason may have an effect on your final mark.

All teams will have a uniform and fixed number of students, ideally 4 students per team but this will depend upon the population of the course. Each team will be responsible for producing and submitting the deliverables. Lecture and Mentor meetings will provide specific guidance for preparing each of the required deliverables. Once students have been allocated a team, they will not be allowed to change teams. Team formation will take place during week 1’s lecture.

Important: If you are unsure about being enrolled in this course, do not put yourself into a team as this will cause a significant disruption to the other members of the team, and the class as a whole, when you do decide to discontinue. Put your name on a pending list and inform the lecturer of your situation. The pending list will be handed out in week 1’s lecture.

11 Conflict Resolution

Students working in a team must endeavor to behave in a cooperative and professional manner. However conflicts may still arise from time to time and it is in your best interest to resolve the matter amicably while the matter is minor. Your mentor should be kept informed of such problems. If required, your mentor will help in the mediation process. There is no point waiting until the end of session to vent your dissatisfaction and frustration, the damage will have already been done. It is best to resolve the problem while it is small and manageable.

Should the conflict escalate beyond a minor disagreement, the LiC reserves the right to arbitrate on the matter and to make a decision that is final. Once again it is in your interest to resolve the matter before it escalates.

12 References

There will be no specific text book for this course. The following references are recommended:

Primary reference:

- Ian Sommerville, Software Engineering, 9th ed, Addison-Wesley Publishing Company, 2010.

Other main references:

- Leszek A Maciaszek, Requirements Analysis and System Design, 2e, Addison-Wesley, UK, ISBN 0321204646, 2005.
- Armando Fox and David Patterson, Engineering Software as a Service: An Agile Approach Using Cloud Computing, 2013.
- Scott Berkun, The art of project management, O'Reilly 2005, ISBN 978-0-596-00786-7, pp. I-XII, 1-488.

13 Computing Resources

Students will be able to make use of the computing resources in the CSE labs. CSE labs are located in the Electrical Engineering, Mechanical Engineering and the K-17 building, a map is available at:

- http://www.cse.unsw.edu.au/~helpdesk/cse_maps.html

For more information about the facilities consult the following link:

- http://www.cse.unsw.edu.au/help/computing/facilities/index.html

14 Academic Misconduct

Copying assignments is unacceptable. Assignments will be checked. The penalties for copying range from receiving no marks for the assignment, through to receiving a mark of 00 FL for the course. In some instances, expulsion from UNSW (for repeat offenders). Allowing someone to copy your work counts as plagiarism, even if you can prove that it is your work.

Further details you are directed to the CSE School Plagiarism Policy. You acknowledged receipt of these rules when you obtained your CSE computer account, and the link above is for your convenience so that you can review the rules now.

We are aware that a lot of learning takes place in student conversations, and we do not wish to discourage this activity. However, it is important, for both those helping others and those being helped, not to provide/accept any programming language code, designs or other written material in writing or electronically, as this is apt to be used exactly as is, and this can lead to plagiarism penalties for both the supplier and the copier of the codes. Write something on a piece of paper, by all means, but tear it up or take it away when the discussion is over.

If you are new to studying in Australia, be aware that attitudes to plagiarism at UNSW may be different from those in your home country. Make sure you are clear about the rules here at UNSW. In brief, and for the purposes of SENG1031, plagiarism includes copying or obtaining all, or a substantial part, of the material for your project, whether it is written or graphical, without written acknowledgment in your assignment from the following resources:

- a location on the internet;
- a book, article or other written document (whether published or unpublished) whether electronic or on paper or other medium;
- another student (past or presently enrolled or whether in your class or another class);
- a non-student (e.g. from someone who writes assignments for money).

Note that if you copy code or other material from another student or non-student with acknowledgement, you will not be penalised for plagiarism, but you are unlikely to get any marks for the copied code/material. If you use code or material found in a publication (on the internet or otherwise) then the marks you get for this will be at the marker’s discretion, and will reflect the marker’s perception of the amount of work you put into finding and/or adapting the code/material, and the degree to which you understand the code/material.

Note also that there is a big difference between being able to understand someone else’s code/material, and writing that code/material yourself from scratch. A software engineer must be able to derived their own work from scratch.