Course Outline

Lecturer

Edit Resource (/COMP9900/19T3/resources/32974/edit)

Contents

- Course Details
- Course Summary
- Assumed Knowledge
- Student Learning Outcomes
- Teaching Strategies
- Teaching Rationale
- Student Conduct
- Assessment
- Course Schedule
- Resources for Students
- Course Evaluation and Development

Course Details

Course Code	COMP3900 / COMP9900
Course Title	Computer Science / Information Technology Project
Lecturer in Charge	Tatjana Zrimec
Course Admin	Tatjana Zrimec
Tutors/Mentors	Ali Darejeh Ka Yee Wong
Classes	**Lectures** : Tue 12:00 - 14:00 Science Theatre (K-F13-G09) Thu 16:00 - 18:00 Keith Burrows Theatre (K-J14-G5) Timetable for all classes (https://webcms3.cse.unsw.edu.au/COMP9900/19T3/timetable)
Consultations	Contact Lecturer to arrange for a consultation
Units of Credit	6
Course Website	http://cse.unsw.edu.au/~cs3900 (https://webcms3.cse.unsw.edu.au/COMP9900/19T3) http://cse.unsw.edu.au/~cs9900 (https://webcms3.cse.unsw.edu.au/COMP9900/19T3)

Handbook Entry	http://www.handbook.unsw.edu.au/undergraduate/courses/current/COMP3900.html (http://www.handbook.unsw.edu.au/undergraduate/courses/current/COMP3900.html) http://www.handbook.unsw.edu.au/postgraduate/courses/current/COMP9900.html (http://www.handbook.unsw.edu.au/postgraduate/courses/current/COMP9900.html)

Course Summary

A capstone software project. Students work in teams to define, implement and evaluate a real-world software system. Most of the work in this course is team-based project work, although there are some introductory lectures on software project management and teamwork strategies. Project teams meet fortnightly with project mentors to report on the progress of the project. Assessment is based on a project proposal, a final project demonstration and report, and on the quality of the software system itself. Students are also required to reflect on their work and to provide peer assessment of their team-mates' contributions to the project.

Assumed Knowledge

Before commencing this course, students should be able to:

- produce correct software programs in Python, Java or C/C++, i.e., compilation, running, testing, debugging, etc.
- produce readable code with clear documentation.
- basic knowledge of database programming, Web programming and/or script programming (such as Python, PHP, and Javascript).

Note:
For COMP9900, students must be in their final semester of study, and have completed at least 72 UOC towards MIT program 8543.
For COMP3900, students have completed COMP1531, and COMP2521 or COMP1927, and are enrolled in a BSc Computer Science major with completion of 120 UOC.

Student Learning Outcomes

After successfully completing this course, students will be able to:

1. work as an effective member of a team to develop a software project (with a real-world application in mind)
2. develop strategies leading to the development of high-performing, self-managing team
3. develop skills to handle team issues during different phases of software development
4. communicate effectively, orally and in writing, with peers, mentor (as supervisor) and lecturer (as stakeholder)
5. appreciate and be familiar with the role of project management in software development
6. produce clear and comprehensive software project proposal and report
7. participate in a group presentation, demonstration, and liaising with mentor and stakeholder
8. use relevant software tools to support effective software management.

This course contributes to the development of the following graduate capabilities:

Graduate Capability	Acquired in
Scholars capable of independent and collaborative enquiry, rigorous in their analysis, critique and reflection, and able to innovate by applying their knowledge and skills to the solution of novel as well as routine problems	Project
Entrepreneurial leaders capable of initiating and embracing innovation and change, as well as engaging and enabling others to contribute to change	Project
Professionals capable of ethical, self- directed practice and independent lifelong learning	Project
Global citizens who are culturally adept and capable of respecting diversity and acting in a socially just and responsible way	Project

Teaching Strategies

- Lectures: the main way to introduce and overview software project management; and discuss various teamwork strategies.
- Labs: for the team to fortnightly meet up and interact with the mentor.
- Project: give students the hands-on experience on a real-world software system, in a teamwork environment.

Teaching Rationale

The learning focus in this course is primarily a team-based software project (for students to practice their skills and knowledge in a real-world, teamwork setup). The course will have an emphasis on the practical software development skills in a teamwork environment. Students will learn the basic contents of software project management through introductory lectures.

Student Conduct

The **Student Code of Conduct** (Information (https://student.unsw.edu.au/conduct) , Policy (https://www.gs.unsw.edu.au/policy/documents/studentcodepolicy.pdf)) sets out what the University expects from students as members of the UNSW community. As well as the learning, teaching and research environment, the University aims to provide an environment that enables students to achieve their full potential and to provide an experience consistent with the University's values and guiding principles. A condition of enrolment is that students *inform themselves* of the University's rules and policies affecting them, and conduct themselves accordingly.

In particular, students have the responsibility to observe standards of equity and respect in dealing with every member of the University community. This applies to all activities on UNSW premises and all external activities related to study and research. This includes behaviour in person as well as behaviour on social media, for example Facebook groups set up for the purpose of discussing UNSW courses or course work. Behaviour that is considered in breach of the Student Code Policy as discriminatory, sexually inappropriate, bullying, harassing, invading another's privacy or causing any person to fear for their personal safety is serious misconduct and can lead to severe penalties, including suspension or exclusion from UNSW.

If you have any concerns, you may raise them with your lecturer, or approach the School Ethics Officer (mailto:ethics-officer@cse.unsw.edu.au) , Grievance Officer (mailto:grievance-officer@cse.unsw.edu.au) , or one of the student representatives.

Plagiarism is defined as (https://student.unsw.edu.au/plagiarism) using the words or ideas of others and presenting them as your own. UNSW and CSE treat plagiarism as academic misconduct, which means that it carries penalties as severe as being excluded from further study at UNSW. There are several on-line sources to help you understand what plagiarism is and how it is dealt with at UNSW:

- Plagiarism and Academic Integrity (https://student.unsw.edu.au/plagiarism)
- UNSW Plagiarism Procedure (https://www.gs.unsw.edu.au/policy/documents/plagiarismprocedure.pdf)

Make sure that you read and understand these. Ignorance is not accepted as an excuse for plagiarism. In particular, you are also responsible that your assignment files are not accessible by anyone but you by setting the correct permissions in your CSE directory and code repository, if using. Note also that plagiarism includes paying or asking another person to do a piece of work for you and then submitting it as your own work.

UNSW has an ongoing commitment to fostering a culture of learning informed by academic integrity. All UNSW staff and students have a responsibility to adhere to this principle of academic integrity. Plagiarism undermines academic integrity and is not tolerated at UNSW. Plagiarism at UNSW is defined as using the words or ideas of others and passing them off as your own.

If you haven't done so yet, please take the time to read the full text of

- UNSW's policy regarding academic honesty and plagiarism (https://student.unsw.edu.au/plagiarism)

The pages below describe the policies and procedures in more detail:

- Student Code Policy (https://www.gs.unsw.edu.au/policy/documents/studentcodepolicy.pdf)
- Student Misconduct Procedure (https://www.gs.unsw.edu.au/policy/documents/studentmisconductprocedures.pdf)
- Plagiarism Policy Statement (https://www.gs.unsw.edu.au/policy/documents/plagiarismpolicy.pdf)
- Plagiarism Procedure (https://www.gs.unsw.edu.au/policy/documents/plagiarismprocedure.pdf)

You should also read the following page which describes your rights and responsibilities in the CSE context:

- Essential Advice for CSE Students (https://www.engineering.unsw.edu.au/computer-science-engineering/about-us/organisational-structure/student-services/policies/essential-advice-for-cse-students)

Assessment

Item	Due	Weighting
Proposal	Week 3 @ 23.59	10%
Project Demo	Week 10 and Week 11 - Project Demo Schedule	15%
Project Report	Friday Week 10 to Tuesday Week 11 @ 23.59	15%
Software Quality	Friday Week 10 to Tuesday Week 11 @ 23.59	40%
Participation Records & Peer Assessment (Optional)	Sunday Week 10 to Wednesday Week 11 @ 23.59	20%

Course Schedule

Week	Lectures	Labs	Notes
1	Course introduction; teamwork strategies; software tools	Group formation, project brainstorming	-
2	Agile software development; software management overview	Work on the project, progress report	-
3	Software development methods overview	Work on the project, preparing the **project proposal**	-
4	-	Work on the project, progress **demo**	-
5	-	Work on the project	-
6	-	Work on the project, progress **demo**	-
7	-	Work on the project	-
8	-	Work on the project, progress **demo**	-
9	-	Work on the project	-
10	Final wrap-up lecture - revision of the project outcomes	Final Project **demo**	-

Resources for Students

There are no specific texts and recommended readings for COMP3900/COMP9900. Programming language specific texts may be useful as references, depending on the programming language(s) used in the project. Other online resources and/or documentation related to this trimester's real-world projects will be provided in the course website during the trimester.

Course Evaluation and Development

This course is evaluated each term using the myExperience system at the end of the term.

Based on the feedback from previous terms, the formula to calculate the peer assessment marks is adjusted. Your feedback is important and will be considered to improve future offerings of this course.

Students are also encouraged to provide informal feedback during the term, and let the lecturer in charge and mentors know of any problems, as soon as they arise. Suggestions will be listened to very openly, positively, constructively and thankfully, and every reasonable effort will be made to address them as soon as possible.

Resource created 8 months ago (Monday 09 September 2019, 04:23:13 PM), last modified 3 months ago (Tuesday 04 February 2020, 01:30:42 AM).

Comments

(/COMP9900/19T3/forums/search?forum_choice=resource/32974)

(/COMP9900/19T3/forums/resource/32974)

There are no comments yet.