COMP4920 Management and Ethics

Semester 2, 2016

Course Outline

Contents

- Course Staff
- Course Details
- Preamble
- Course Aims
- Student Learning Outcomes
- Assumed Knowledge
- Teaching Rationale
- Teaching Strategies
- Assessment
- Academic Honesty and Plagiarism
- Course Schedule
- Resources for Students
- Course Evaluation and Development

Course Staff

Contact Details

Name	Role	E-Mail	Phone
Wayne Wobcke	Lecturer in charge	w.wobcke@unsw.edu.au	9385 6475
Bruno Gaeta	Lecturer	bgaeta@unsw.edu.au	9385 7213

Course Details

Units of Credit: 6

Web Site: http://www.cse.unsw.edu.au/~cs4920/

Timetable:

Lecture Time	Room	Lecturers
Wed 12.00-2.00	Rex Vowels	Wayne Wobcke, Bruno Gaeta

Seminar Time(**)	Room	Facilitator	E-Mail
Mon 10.00-12.00	Square House 208	Sandeepa Kannangara	s.kannangara@unsw.edu.au
Mon 12.00-2.00	Square House 208	Bruno Gaeta	bgaeta@unsw.edu.au
Tue 10.00-12.00	Mathews 123	Rob Everest	robertce@cse.unsw.edu.au
Tue 12.00-2.00(*)	Mathews 125	Wayne Wobcke	w.wobcke@unsw.edu.au
Tue 3.00-5.00(*)	Square House 207	Wayne Wobcke	w.wobcke@unsw.edu.au
Wed 10.00-12.00	Quad G054	Sandeepa Kannangara	s.kannangara@unsw.edu.au
Wed 2.00-4.00	Mathews 226	Rob Everest	robertce@cse.unsw.edu.au
Wed 4.00-6.00	Mathews 309	John Calvo-Martinez	jcalvo@cse.unsw.edu.au
Thu 9.00-11.00	Ainsworth G01	John Calvo-Martinez	jcalvo@cse.unsw.edu.au

Note: (*) Class for SENG students only

Note: (**) Seminars may be cancelled depending on enrolments

Preamble

COMP4920 is taken by two groups of students: **SENG students** who have completed software project management as part of Software Engineering workshops, and **COMP students** (Computer Science, Computer Engineering and Bioinformatics students) who have not previously studied software project management. Lectures are common, but seminars and assessment differ between the two groups. Differences are highlighted in blue.

Course Aims

COMP4920 covers practical aspects of both *software project management* and *professional issues and ethics*, and as such is critical preparation for graduates about to enter the workforce, in addition to being essential for accreditation of the Software Engineering, Computer Science, Computer Engineering and Bioinformatics degree programmes. **Students enrolling should be in the final year of study or nearing completion of their degree.**

There are two specific themes and objectives.

- *Software Project Management.* To gain practical experience in all phases of the planning and execution of a software project, including requirements scoping, choice of software process methodology, project planning and scheduling, teamwork and communication, and risk and change management. SENG students study project management only in seminars: there is no practical component.
- *Professional Issues and Ethics.* To appreciate the responsibilities of a professional software engineer and understand the ethical dimensions of the IT industry as applied to specific issues such as software correctness, privacy and security, intellectual property and legal obligations of IT practitioners. SENG students study professional issues to a greater depth and breadth than COMP students.

Student Learning Outcomes

On successfully completing this course, students should be able to:

- *Software Project Management.* (COMP students only)
  - Understand the range of activities involved in a large software project.
  - Be able to plan and successfully execute a team-based software project.
  - Take on different roles in a team to contribute to team success.
  - Understand and appropriately apply software engineering processes.
  - Understand the importance of teamwork and communication in a software project.
- *Professional Issues and Ethics.*
  - Understand the responsibilities of a professional software engineer.
  - Appreciate and apply ethical frameworks to make professional judgements.
  - Develop critical thinking in relation to professional issues.
  - Gain in-depth understanding of several topical professional issues.
  - Appreciate different ways of managing intellectual property.
  - Understand the societal context of technology developments.
  - Improve communication skills needed to present reasoned arguments.

This course contributes to the following UNSW graduate attributes.

- *The skills involved in scholarly enquiry.* The course emphasizes professional codes of ethics and conduct that oblige engineers to keep up to date and rigorously apply the latest methods and technology.
- *The capacity for analytical and critical thinking and for creative problem solving.* Critical thinking is developed through analysis of professional issues and case studies in seminars and through writing an analytical essay involving in-depth research on a topical issue.
- *Capacity for enterprise, initiative and creativity.* The course covers management of intellectual property and innovation including aspects specific to the software industry, such as software patents and open source software.
- *An appreciation and respect for diversity.* Societal benefits and drawbacks of software engineering are discussed explicitly.
- *Skills required for collaborative and multidisciplinary work.* The software project requires the ability to work within technical teams in the development of a product that requires teamwork and management of a project. (COMP students only)
- *Respect for ethical practice and social responsibility.* Explicit discussion of ethical theories and professional codes provides students with frameworks to make ethical judgements and knowledge of what society expects of professional engineers.
- *Skills of effective communication.* Effective communication skills are encouraged through preparation and delivery of a student-led seminar and through writing a critical essay on a topical issue.

Assumed Knowledge

Students are assumed to be in their final year of study (or nearing graduation) **and** completed around half of their Stage 3 or 4 courses, so are assumed to have reasonable knowledge and maturity in Software Engineering, Computer Science, Computer Engineering or Bioinformatics.

COMP students only: Students are assumed to be competent computer programmers with knowledge of agile software processes and experience of working in agile teams based on the Scrum methodology. The level of experience is assumed to be sufficient to undertake (from conception and planning to completion) a moderately large software development project. **Note that seminar facilitators play the role of clients, and cannot provide technical advice on programming languages or platforms.**

Teaching Rationale

COMP4920 is an important course to help students become "job ready". The course covers both software project management from a practical point of view, and professional issues and ethics as related to the IT industry. Students benefit by having the opportunity to interact with industry experts, hence **attendance at lectures is very important**,

especially as the course focuses on "soft skills".

For *software project management*, the teaching philosophy is that much of this knowledge cannot be "taught" but rather is gained through experience and reflection. Hence the approach taken in this course is that students learn about software project management through working in and managing a team-based software project, encompassing all phases of the project from requirements scoping, application of a software process methodology, project planning and scheduling, teamwork and communication, risk management and change management. Students develop a project plan in the first half of semester **and will be held to that plan for the second half of semester** (subject to any agreed modifications).

For *professional issues and ethics*, teaching is based on seminar-style discussion groups, encouraging students to express their ideas and form their own judgements on specific issues relating to the IT industry on the basis of rational arguments. Seminars also promote team organization and presentation skills through a team-based student-led seminar. Critical thinking is developed through researching and writing an in-depth analytical essay on a specific professional issue of relevance to the industry. SENG students also conduct a debate and have an essay-style written exam.

Time management is an important aspect of this course. It is expected that each student attends **all** lectures and seminars, prepares for each seminar by reading the relevant material **in advance**, contributes actively to seminar discussion, and spends roughly 10-15 hours on the student-led seminar and roughly 15-20 hours on the essay and 15-20 hours on the lecture summaries.

SENG students only: Students should spend 10-15 hours on the debate and a further 25-35 hours on exam preparation.

COMP students only: The software project is a major component of the course and should take 40-50 hours **per person** for planning and execution.

Teaching Strategies

The course has a mixture of guest lectures on topics of interest and seminars focusing on particular professional issues, case studies and the software project.

Lectures provide an overview of one particular aspect of software project management or a professional or ethical issue. Guest lecturers are able to provide expertise in a variety of areas and the lectures provide an essential foundation to apply in seminars and essays. **Note that due to the commercially sensitive nature of the material, it is not possible to provide lecture recordings.**

Seminars provide students an opportunity for more in-depth discussion on particular topics and, in student-led seminars, enable students to develop skills in expression, critical analysis and presentation.

Essays enable students to study in-depth a topic of interest and promotes the development of critical thinking and analytical abilities and written communication skills.

Software projects are the means through which students develop an understanding of software project management and the ability to apply software processes in all phases of planning and executing a software project.

Assessment

The assessment for this course consists of the following weighted components.

- Seminar participation (10%)
- Student-led seminar (10%)
- Essay (20%)
- Lecture summaries (10%)

SENG students only:

- Debate (10%)
- Written examination (40%)

COMP students only:

- Software project plan – presentation and document (20%)
- Software product (30%)

The student-led seminar and debate combine an individual and team mark, and in addition include a peer assessment component. Seminar participation and the essay, lecture summaries and written examination are assessed individually.

The *seminar participation* mark is based on *active* and *relevant* contributions over all seminars (including student-led seminars), which requires attendance at all lectures and reading the seminar material **in advance**. It is not an attendance mark. See the UNSW Teaching website for more details on why and how seminar participation is assessed.

The *student-led seminar* mark is based on coverage of the chosen topic, including identification of relevant ethical issues, presentation style and engagement of the audience in discussion.

The *essay* mark is based on the originality and depth of ethical analysis applied to the chosen topic, drawing on the main ethical theories discussed in the course, and on the quality of the writing and appropriate use of evidence in developing a reasoned argument.

The *lecture summaries*, due in Week 13, should consist of summaries *and short reflections* on any 5 guest lectures relating to professional issues and ethics (i.e. not software processes). Each summary (at most 1 page) should contain an overview of the main points of the lecture, identify a professional/ethical issue discussed in the lecture, and include a short reflection on how that issue can be addressed by applying ethical reasoning.

The assessment of the *software project* includes the presentation and written submission of a project plan (in Week 8) and a presentation/demonstration of the final system (in Week 13). Two seminars will be devoted to the informal presentation and class discussion of the projects (sprint reviews) to enable the group to provide feedback on the progress of each project. The mark for the software product consists of an individual component and a team component; each team member generally receives the same mark for the team component. The individual mark will in part be based on a project diary shown to the facilitator at various intervals. **Part of the assessment is based on project management and teamwork, including the appropriate use of project management tools.**

The final mark for the course is determined by adding together these component marks according to the above weighting to give a result out of 100, which is subject to further scaling.

Late submission policy for assignments: Assignments (project plan document, essay and lecture summaries) submitted late are subject to the penalty that the mark reduces by 20% per (calendar) day late, for up to three days, after which a mark of 0 is received. Projects submitted late will receive a 0 mark.

Academic Honesty and Plagiarism

UNSW has instituted severe penalties for academic plagiarism. Therefore it is important for students to understand what is acceptable and unacceptable in the presentation of work for assessment.

You should **carefully** read the UNSW policy on academic integrity and plagiarism. Note, in particular, that *copying* (taking ideas and/or text from other students or the Internet and presenting them as your own), and *collusion* (working together on an assignment, or sharing parts of assignment solutions) are forms of plagiarism. That is, giving your assignment solution to another student counts as collusion, regardless of the originality of your work. In COMP4920, this applies particularly to the essay, which must be written in your own words, and with properly cited sources. General expectations of students at UNSW, and the procedures for handling student misconduct (including plagiarism), are set out in the UNSW student code.

In essence, as applied to COMP4920, copying or sharing material from the Internet such as slides, text or program code (that is, without proper citation) counts as plagiarism and is unacceptable. In a student-led seminar, essay or debate, all material must be in the student's own words (except quotations, which should be kept to a minimum and must be clearly identified as such). References must be from primary sources (i.e. do not cite *Wikipedia* articles or the like). Essays will be run through *turnitin* for plagiarism detection. In the software project, sharing code *amongst team members* is acceptable, but sharing code between (members of) different teams is unacceptable.

The penalties for plagiarism range from receiving 0 marks for the assignment, through receiving a mark of 00 FL for the course, to expulsion from UNSW (for repeat offenders). The school maintains a register of students with confirmed plagiarism offences. Note that allowing someone else to copy your work counts as academic misconduct, and makes you liable to a penalty, even if you can prove that the work is yours originally.

Course Schedule

The following is the rough sequence of lecture topics by week. Broadly speaking, there are two lectures on each of (i) ethics, (ii) agile methods, (iii) legal perspectives, (iv) software licensing, and (v) business perspectives. However, as most lectures are given by guest lecturers, this schedule is subject to change. More details will be provided during the semester as the lectures are confirmed.

Week	Topic
1	Introduction to Project Management and Ethics
2	Theoretical Underpinnings of Ethics
3	Moral Reasoning and Professional Ethics
4	Agile Software Processes in Practice
5	Legal Perspectives on System Development
6	Agile Product Management and User Experience
7	Data Privacy and Uberveillance
8	Intellectual Property and Software Patents
9	Open Source Software
10	Innovation and Entrepreneurship
11	Employment Conditions and Contracts

Resources for Students

References

Reference material for lectures and seminars will be provided on the course web pages throughout the semester. There is no set textbook for the course, however some of the introductory lecture on software project management uses material from the following book, which is an excellent reference for software engineering generally (though not specifically for agile methods).

Sommerville, I. *Software Engineering.* Tenth Edition. Pearson Education, Upper Saddle River, NJ, 2015.

Course Evaluation and Development

Computer Science and Engineering courses are evaluated by student survey each time they are taught. The survey includes standard questions asked of all comparable courses so that it is possible to compare a course with other relevant UNSW courses, and also includes space for free-form comments. Survey responses are anonymous. The completed survey forms are analysed statistically by someone independent of the course staff, and the results, including free-form comments, are made available to the lecturer in charge *after* grades have been reported and released.

For COMP students, COMP4920 was offered for the first time in 2012 as a replacement for COMP3711 Software Project Management (taught by the School of Information Systems, Technology and Management) and a previous 3 unit course COMP2920 Professional Issues and Ethics. Feedback from these previous courses is primarily that the offering of COMP3711 treats project management at too abstract a level for Computer Science and Engineering students, which is the reason for the more practical approach to software project management taken in COMP4920. For SENG students, COMP4920 replaces and largely builds on the previous course SENG4921 Professional Issues and Ethics.

Students are generally satisfied with the current course. However, a few students commented in 2015 that the requirements for the lecture summaries were not clear. To address this concern, a more precise template for lecture summaries has been provided.

Wayne Wobcke, Bruno Gaeta
Last updated: 19 November 2021
CRICOS Provider No. 00098G