41129 Software Innovation Studio

Subject information	Detail
Delivery	Spring Session
Total workload hours	300 hours
Credit points	12
Result type	Grade and marks

Subject description

The Software Innovation Studio subject blends field research and user observation with cutting-edge algorithm application to drive innovation in software design and service enhancement. It cultivates skills in idea generation, logical reasoning, systematic execution, and agile project management. You'll experience the full software engineering cycle in a practice- and project-based setting, with the goal of developing valuable software solutions, enhancing existing platforms, or resolving complex issues.

Subject learning outcomes (SLOs)

1. Evaluate stakeholder requirements and translate into software project (B.1)

2. Develop and improve innovative software that fits users' needs. (C.1)

3. Apply systematic and logical data analytics for software engineering. (D.1)

4. Apply agile and systematic approaches to manage software engineering projects, as an efficient team. (E.1)

5. Reflect on skills learned as the final step in the development of one's career plan. (F.1)

Teaching and learning strategies

This studio-based subject provides a learning environment to leverage big data analytics for innovative software development. Students will practice the implementation of big data analytics and logical thinking. The students will also form small self-managed teams to practice agile project planning for a proof of concept, prototype, problem solve and/or minimum viable product with innovative ideas or concepts. Teams are aided and guided by mentors (tutors, industry partners and academic researchers).

To encourage high technical standards and peer learning, all teams develop the software based on specified requirements but are free to decide how those requirements can be implemented to achieve greater satisfaction(s) or fit users' need(s). The team must have at least one discussion every week to review the project scope, progress, achievements and risks/hazards. Students are also expected to demonstrate their collaboration as well as communication skills during the teamwork and weekly discussions. Students will participate in peer evaluations of teamwork by SparkPLUS as well.

Students are encouraged to use the outside of workshop time to read latest articles and work with their team members. Lecturers, tutors and mentors will provide feedback regarding student achievements, results of meeting, logical thinking, teamwork and hazards in the workshops.

Formal assessments of the innovative concept, project achievements and deliverables occur at the early, middle and end of the semester. The lecturers, mentors, tutors and peers will all involve in the formal assessment and provide feedback through online tools and workshops.

Content (topics)

In this subject, students will experience the following:

1. Develop innovative software for users' needs
2. Data analytics
3. Logical Thinking
4. Configuration management including source and version control
5. Agile software management
6. Team communication and collaboration
7. Reflective software design practice
8. Software development and unit testing

Required and recommended texts

All required and recommended texts for this subject are contained in the **Reading List (https://canvas.uts.edu.au/courses/36580/external_tools/7229)** which you can access from the Canvas site menu.

Other required resources

If you have concerns about affording the course-related costs for this subject, **UTS Financial Assistance (https://www.uts.edu.au/current-students/support/financial-help/financial-assistance-service)** can help you plan ahead or apply for grants or interest-free loans.

Questions and concerns

Assessment general information

For a project-based subject, students can document their progress and reflection before weekly workshop. They can highly engage in stand-up and retrospective meetings with their team members during the whole semester to achieve the innovative software. Students are also expected to regularly arrange and attend meeting with their mentor(s) for demonstration, problem discussion, performance review and guidance.

Students can demonstrate high level design, logical analysis and synthesis of the function they work. All clear evidence can be used to highlight the testing, scaling, security, time complexity, alternative ways of implementation and/or drawback of your implementation.

The students will also participate in peer evaluations of teamwork via SparkPLUS.

The tutors and lecturers will carefully assess students' engagement and contribution through all evidence from weekly meetings, version control, logical thinking and SparkPLUS.

Students can demonstrate the software which is fully testing, workable, secure and scalable. The mentors and/or clients will carefully assess students' process and product.

Assessment tasks submitted via electronic upload will be subject to similarity detection software. Where routinely used, students will be given the opportunity to review their own work prior to submission.

Minimum requirements

In order to pass the subject, a student must achieve an overall mark of 50% or more.

Attendance requirement

3hpw, on campus, weekly

Assessment feedback

This subject follows agile software development principles and practices. The work will be assessed and discussed during or following each sprint. The feedback will be provided by lecturers, tutors and mentors during the workshop or assessment review within 2 weeks.

Assessment task 1: Project Pitch

Assessment   **Detail**
Information

Description	The purpose of this task is how well the students are learning to

1. deliver innovative idea(s)
2. do comprehensive field research and background analysis
3. do proper project planning

Each team will give a presentation to address

1. clear project scope with innovative ideas and concepts
2. detail achievements with reasonable timeline
3. clear task assignments and timeline

Learning Outcome	• SLO1 Evaluate stakeholder requirements and translate into software project (B.1) • SLO2 Develop and improve innovative software that fits users' needs. (C.1) • SLO3 Apply systematic and logical data analytics for software engineering. (D.1) • SLO4 Apply agile and systematic approaches to manage software engineering projects, as an efficient team. (E.1)
Type	Presentation
Groupwork	Group, group assessed
Weight	10%
Length	3 –5 minutes presentation
Assessment feedback	

Assessment task 2: Implementation and logical thinking

Assessment Information	Detail

Description	The purpose of this task is to

1. reflect their work progress and plan weekly,
2. design system/software,
3. leverage data analytics,
4. make proper decisions

Students must prepare programming progress and attend the sprint meeting every week. All progress should be synchronised to online tools (MS Teams, Git and Trello).

In the middle and end of semester, every student needs to have one document to demonstrate

1. critical individual reflection,
2. high-level systematic software engineering synthesis with logical thinking process,
3. individual contributions and members' inputs during decision making

Learning Outcome	• SLO1 Evaluate stakeholder requirements and translate into software project (B.1) • SLO2 Develop and improve innovative software that fits users' needs. (C.1) • SLO3 Apply systematic and logical data analytics for software engineering. (D.1) • SLO4 Apply agile and systematic approaches to manage software engineering projects, as an efficient team. (E.1)
Type	Report
Groupwork	Individual
Weight	35%
Length	1000 – 3500 words
Assessment feedback	

Assessment task 3: Engagement and teamwork assessment

Assessment Information	Detail

Description	The purpose of this task is to demonstrate and understand how well the communication and collaboration in the team. Every student must involve in peer review to assess individual team member's engagement and performance via SPARKPlus in the middle and end of semester. The task for students is to demonstrate 1. efficient communication, closed collaboration and high engagement in the team, 2. reflection of the teamwork by clear and detail record of stand-up and/or retrospective meeting, 3. ability to give fair rates and suggestions to other team members

Learning Outcome	• SLO4 Apply agile and systematic approaches to manage software engineering projects, as an efficient team. (E.1) • SLO5 Reflect on skills learned as the final step in the development of one's career plan. (F.1)
Type	Reflection
Groupwork	Group, individually assessed
Weight	10%
Length	500 – 2000 words
Assessment feedback	

Assessment task 4: Product assessment

Assessment Information	Detail
Description	The purpose of this task is to learn how to deliver a software/system against the users' needs. The task for students is to showcase that the software can satisfy the requirements from clients or end users. The outcomes specified for the proposed project in assessment 1 have been well achieved. Every team must prepare presentation (slides and videos) and live demo to have closed interactions with all classmates.
Learning Outcome	• SLO1 Evaluate stakeholder requirements and translate into software project (B.1) • SLO2 Develop and improve innovative software that fits users' needs. (C.1) • SLO3 Apply systematic and logical data analytics for software engineering. (D.1) • SLO4 Apply agile and systematic approaches to manage software engineering projects, as an efficient team. (E.1) • SLO5 Reflect on skills learned as the final step in the development of one's career plan. (F.1)
Type	Demonstration
Groupwork	Group, individually assessed
Weight	35%
Length	3-5 minutes presentation followed by exhibition

Assessment task 5: Technical reading and competitive market analysis

Assessment Information	Detail

Description	The purpose of this task is to gain the tech knowledge as well as deeper analysis to cover the most potential competitors in the same or similar market
	Some state-of-the-art approaches will be introduced. Before the weekly workshop, the students must read assigned articles to gain general concepts of specific algorithm. The students will investigate the current market and shape the innovative project.
	The task for students is to
	1. Investigate the current market
	2. learn the state-of-the-art tech especially big or AI data analytics
	3. practice programming and implementation
	4. shape the innovative and unique project
Learning Outcome	• SLO1 Evaluate stakeholder requirements and translate into software project (B.1) • SLO2 Develop and improve innovative software that fits users' needs. (C.1)
Type	Presentation
Groupwork	Group, group assessed
Weight	10%
Length	presentation(s) in workshop

Academic Integrity

Academic integrity is about demonstrating honesty, trust, fairness, respect, responsibility and courage in your studies and assessments. Studying at UTS and being part of our community means maintaining

these values and acting with academic integrity at all times. **Find out everything you need to know about academic integrity at UTS** (https://www.uts.edu.au/current-students/support/academic-support/academic-integrity) .

Assessment procedures

All staff and students involved in the assessment of coursework subjects at UTS are subject to the **Coursework Assessments Policy** (https://www.uts.edu.au/about/uts-governance/policies/uts-policy/coursework-assessments-policy) . The policy applies to the assessment of all coursework subjects. This policy does not apply to thesis subjects that are taken by students enrolled in research degrees but does apply to any coursework subjects undertaken by research degree students. It does not describe policy that relates to academic progression through a course of study. The policy should be read with the **Coursework Assessments Procedure** (https://www.uts.edu.au/about/uts-governance/policies/uts-policy/coursework-assessments-procedure) and **Student Rules.** (https://www.uts.edu.au/about/leadership-governance/governance/rules/student-rules/)