COMP3900/9900
Project Deliverables and Assessment

An overview of the assessment items is described below.

Work Diary

Each student should maintain a work diary (call it z0001234.txt if your student zId is z0001234) and check the file into GitHub.

From Week 1 onwards, we expect that you will update this file every week with your new contributions to the team and the project. Check the file into GitHub at least once a week (can be more frequent).

The content of the file should be brief but clear. For example, it should be like:

```
Week 1

Group formed. I created the Trello & GitHub accounts for the team. Together
with John Smith, I wrote the Epic section of the proposal. I also found and
discussed with the team all available software tools and libraries that we
can use for the project.

Week 2

I wrote the first version of hello.c, api.h, api.c and drafted a design for
the Web service API between the client and the server – all by myself.
```

You should also include either in this diary or in the GitHub (or Trello) account the following information (if applicable) about the project progress:

- what was planned for the period since the last work diary entry
- what was finished

- what were the main technical and non-technical obstacles and how they were overcome (or what was tried and did not work)
- what was not finished, why this happened, how this affects the overall project progress and what adjustments (if any) are needed so the success of the project is not endangered
- what is planned for the next period (e.g., next week)

We expect that other members of the team may access this file to know where others are up to, but you should not modify your peer's work diary.

The Proposal

- **Project Proposal** (due Friday Week 3 @ 23.59) (10%): Each trimester, students will choose a project from a couple of possible projects with fixed domain and scope. Students may propose a project by modifying one of a given sample projects or propose their own project as far as it is within the given scope and of similar technical depth.

- **Group Formation:** A group will be ideally of four (4) students in size. Groups will be formed by students within the same lab with the help of the mentor. However, an extra member may be added to a group by the mentor if the group has only three (3) members (and there are still students yet to form a group within the lab). Please do record your group in WebCMS3 course website under Groups page and specify group members as well as the Scrum Master (as Admin in WebCMS3 Groups terminology).

- **Roles:** Each group should have a Scrum Master and three (or two) Developers. Their responsibilities are discussed in the lecture. Note that these roles are for accountability. We expect that all members should be involved in coding. Scrum Master may contribute marginally less coding efforts, e.g., 10% less, if he/she will administrate GitHub (having a Maintainer role) and Trello accounts for the team.

- The proposal should be self-contained, and should:
  - Include a **title page** containing: project title; a nominated group name; each member's name, email, student ID, role; proposal submission date.
  - Be **at least 4 pages long** (at most 12pt font with reasonable margins & spacing), not including the title page, and be in **PDF format** that is readable with Acrobat on the CSE workstations.
  - The marking criteria for the proposal are based on the:
    - Technical depth of the project and its scale (50%),
    - Background (The problem domain, the existing work/system and its drawbacks) (10%),
    - Epics (40%):

- A list of at least three (3) epics and their brief descriptions
- The scope of each epic (what will you be doing? and what is out of scope?)
- The size of each epic
- The final selection of the epics (e.g., based on sizing) for the final deliverables, and their justifications.

The selected epics will be delivered and demonstrated during the final phase of the project.

Epics (and later user stories) are expected to be captured using GitHub Project Board or Trello.

Joint group project: *More developers tend to be capable to deploy software of a larger scale. In this regard, a joint group project (i.e., a project that is taken by multiple groups together* ***within the same lab****) is encouraged. If you aim for a joint group project, please state in the proposal: the names of other groups; and the overall idea (then the selected epics will be the parts that your group is responsible for).*

When marking the joint group projects, three (3) extra factors are considered:

(i) the extra resources (e.g., 8 people instead of 4);
(ii) the more challenging project management (e.g., inter-group communications and management);
(iii) the (larger) scope/scale of the project.

❖ Proposal is marked out of 10 and contributes 10% towards the final mark for the project.

Final Deliverables

- **Project Demonstration/Presentation** (During Week 10 Lab) (15%): Each group should prepare a 15/20-minute presentation about the final outcome of the project. This could be a demonstration of the system built (preferred), and/or a presentation of the work highlights (optional).

  Presentation can be extended to 30-minute if multiple groups joining together to target a project of a larger scale.

  This will take place in the lab of Week 10. You may decide on what presentation style is the best for your group, e.g., one member presents everything, each member presents one part, or Apple WWDC sort of presentations (i.e., one person does the talking and the rest of the team runs/controls the demo).

  Please note that UNSW community, but also outside experts (e.g., from prospective employers) may be invited to project demos.

  We adopt the guidelines from the CSE Honours thesis system which have been adjusted to better fit COMP3900/9900 course which is project focused.

  The adjusted guidelines are listed below. Note that the grades below (FL, PS, CR, DN, HD) indicate the range of marks that your group will likely obtain.

	Technical Quality and Completeness of the project as demonstrated (70%)	Structure and Delivery of the Demo/Presentation (30%)
FL	Lack of understanding of the technical work required. No/little outcome demo	No structure. Poorly prepared or delivered presentation
PS	Close to complete, reasonable effort, some level of engineering practice. Demo of reasonable tech skills used & good team work	Generally appropriate demo showcase/materials. Questions not handled well
CR	Mostly complete and functional work demonstrated (some scope for improvement). Satisfactory team effort, a good level of engineering practice & tech skills employed	Good structure and appropriate demos/materials. Q/A handled well
DN	Complete work (requiring minor refinements), good team effort, significant effort invested, solid technical quality, showing a good level of engineering practice & tech skills employed	Well structured demos and effective materials/examples. Q/A handled well. Good interaction with audience
HD	Complete and fully functional or correct, coherent piece of work (by all team members). Effort invested is Impressive. Of highest quality, demonstrating the best engineering practice, solid tech/research methodology	Excellent structure demos and well designed materials. Confident and professional delivery. Presentation/demo aides effortlessly integrated with the delivery

- **Project Report** (Friday Week 10 @ 23.59) (15%): The project report should be prepared according to the instructions below and should at least include the following information:

  - Using the similar format as the project proposal, your report should be at least 12 pages long, excluding the title page, and be in PDF format that is readable with Acrobat on the CSE workstations.
  - Title page similar to that of the project proposal, with the project submission date.
  - Overview - Architecture / design of the overall system & functionalities (*for a joint group project, this section will be common to each group*).
  - Descriptions of the functionalities developed by the team (and why not – the justifications of the functionalities that are missing/different from the project proposal).
  - Proper references and brief descriptions of ALL third-party functionalities (clouds/services/APIs/libraries/code) used by the team, with justification for their use and discussion how their licensing terms impact results of this project.
  - Implementation challenges: descriptions of any tricks, non-trivial algorithms, special architecture/design, etc.
  - User documentation/manual: how to build, setup, configure, and use your system and functionalities. (*For a joint group project, this section can be common for each group, if they merge all functionalities together in one code repositories for final submission; or otherwise, each group shall have a different description in this section.*)
  - Use either APA (https://student.unsw.edu.au/apa) or Harvard (https://student.unsw.edu.au/harvard-referencing) referencing style.

  The marking guidelines for the Report are as follows:

	Overview (10%)	Functionalities & Implementation Challenges (50%)	User doc/manual (30%)	Document presentation, title page, references (10%)
FL	Fail to present the overall picture (design & architecture) of the project	Clearly deficient, lack of any useful details	Insufficient / incorrect instructions to compile, build, setup or use the software	Impedes document reading; or missing sections
PS	Provides vague/insufficient design and architecture descriptions	"Thin" results, lacking intellectual engagement, lack of justifications	Unclear instructions but can still follow to build and run the software	Poor formatting and document structure
CR	Clear design and architecture, but has weakness or technical issues with them	Several functionalities of the software not coherently linked	Easy to follow to build and setup. Not enough information to cover all the functionality usages	Poor judgement with respect to layout, possible padding
DN	Clear and correct design and architecture	Solid, coherent work, linking all the functionalities together into a consistent story. Good technical depth of implementations	Complete and correct instructions	Minor issues, but overall high quality
HD	Concise and professional presentation	Solid, coherent and consistent functionalities. Good description on solving difficult technical, research, or implementation issues	Professional and concise instructions (correct and complete)	Professional, easy to read and high quality presentation (layout & design, etc)

- **Software Quality** (Friday Week 10 @ 23.59) (40%): This is mainly for the scale and technical depth of the delivered implementation; the correctness of the implementation; its value or its novelty (e.g., will it add any value to its intended users or just re-implement the existing feature but in a different way); its performance (e.g., is it too slow for its intended usages); clarity of your code, its design, its structure and its organization; interface design; and ease of use.

The marking guidelines for the Software Quality are as follows:

	Technical depth & Novelty (50%)	Correctness & Performance (30%)	Code style, structure, readability (10%)	Interface & Usability (10%)
FL	Implementation far from completion	Unacceptable performance, buggy even with a few tests	Messy code structures, difficult to read	Primitive interface and difficult to use
PS	Complete implementation according to the scope of the proposal	Overall correct but slow	Readable but not organized	Poor interface design but still usable
CR	Complete implementation; solving some technical challenges	Overall correct and efficient	Code is well structured and readable with some documentation	Generally good design with usability issues on some use cases
DN	Completed with some degree of technical novelty	No issues during demo and project testing (by the assessors)	Well structured; and easy to read with ample documentation	Generally good design and ease to use in all aspects
HD	Completed, with good degree of functional and technical novelty	Robust & excellent performance	Easy to read, well documented, demonstration of excellent coding style & practice	Professional interface design, excellent usability

- **Participation Records, Peer Assessment** (Sunday Week 10 @ 23.59) (20%): Each member's contributions to the project are evaluated based on 3 components: (1) the participation records from GitHub, Trello, and Lab progress/sprint demo; (2) your claimed contributions in your GitHub diary; (3) a rating of each member from their peers (i.e., Peer Assessment).

For Peer Assessment, each member may **optionally** submit a text file (namely peers.txt). Each line of this file contains the zID of your group member, followed by a space, followed by an integer score from 0 to 10 (full mark). Please **do not** include yourself in the file. The score is to indicate the relative efforts and contributions to the effort, from your perspective. For example, a sample file looks like:

```
z1000123 8
z1234123 10
z2468123 4
```

if you believe that the first member contributes well to the project, and the second member is the critical contributor, whilst the last person has done unacceptable work for the project. Remember, you are not one of them on the list.

Note that the scores received for a group will be aggregated and consolidated. Furthermore, the deviation of the scores matter (whilst the actual scores do not). For example, every member having the same score of 7 is the same as every member having the same score of 3. Similarly, it will have the same effect to the score of every member of the group if the above sample file is changed to:

```
z1000123 4
z1234123 5
z2468123 2
```

If none of the group members submit this file, it will be the same as if all members have submitted this file and specifying that everyone has the same score.

If extreme scores are obtained within a group, records on GitHub and Trello will be used to substantiate these scores. Therefore, please keep the GitHub and Trello accounts for at least a few weeks after you receive the course grade from COMP3900/9900.

Note:

Each group shall create a GitHub account and Trello account during Week 1 and add cs9900@cse.unsw.edu.au (for COMP3900, please also use this cs9900 email) as a member to all the repositories and boards in each of these accounts respectively.