Curtin College-Bentley
ISEN1000 **Introduction to Software Engineering**
Trimester 3, 2023

Exercise Submission 1

Weight: 40% of the unit

General Information. Include your name and student ID at the start (add a title page/header footer).

Timeframe. You have 14 days (336 hours) to complete and submit your answers, from **05:00pm on 8th Nov 2023** until **05:00pm on 22th Nov 2023** (UTC+8). You may schedule your work however you wish within this period. **However, late submissions are not allowed** (LAP students considered and will be dealt with accordingly).

Submission. Submit your answer document to the "Exercise Submission 1" on Moodle under **"Assessments Tab"** (1st Tab after unit Intro) . You must verify that your submission was successful. **Correctly submitting is entirely your responsibility.**

Reference Material. This is an OPEN BOOK and OPEN COMPUTER exercise. You may refer to any written material, including your notes, course materials, books, websites, Unit Moodle page recordings etc. However:

- You must complete this exercise entirely on your own. Use of Generative AI is prohibited.
- During the exercise, you may not communicate with any other student regarding the test.
- During the exercise, you may not communicate with any other person in order to seek or receive an answer to any test question.
- Your answer document will be checked by text matching software for signs of cheating, collusion and/or plagiarism.
- The exercise questions have been designed such that any two students, working independently, should not produce the same answers.

*****GO THROUGH THE COMPLETE DOCUMENT BEFORE YOU START*****

1. Overall Exercise Description

In this assessment you will do a complete software requirement analysis and will submit detailed planning as your assessment. Detailed description on each task and steps required to be performed in Section 3.

2. Exercise Problem Description (Question)

You are preforming a requirement analysis and planning work for software system (Name: SkyFly) used for airline reservation (e.g., Qantas Reservation System). The scope of the system includes overseeing all operation being performed for passenger reservations and seats management. You have been included in the team at a time where the requirements are still updating/changing from the client.

Flight reservation and seats management systems usually work having multiple passengers for different classes (economy/business) in a plane. The system usually incorporates passengers record (names, IDs and contacts/emails), travelling dates, travelling destination(s), planes arrival and departure time and scheduled dates, maximum seating capacity, important communication (e.g., booking confirmation/cancellation, amended bookings/expected delays), etc. (Note: NO CONNECTING FLIGHTS CAPACITY IS CONSIDERED, YET).

- New flights, destinations and routes can be added to the current reservation system, each added plane will obviously have all related operation described above.
- There can be multiple planes (by same/different airline) leaving/arriving at the same airport for same destination, but they should be managed in a way that system should consider them separately for all related operations (as described above).
- The system should have a capability to assign/remove new flight to/from a destination if required for some specific cases (e.g., higher passenger density on a festival).
- The systems should also be able to manage higher density of passenger in case of increased travelling to/from a destination.
- The systems should have an updated schedule (24/7) for passengers to and management staff.
- Passenger should be able check the updated flight schedule, seats availability for each scheduled flight booked (for a particular passenger).
- The system should be able to deal with online/on-desk payments for booking, providing confirmation emails/receipts and tax invoice.
- The systems should send reminder email(s) to each passenger travelling in next 12 hours, clearly providing, terminal and gate details, also provide instructions on special arrangements if any (e.g., No/limited food/drinks available on the terminal due to extra passenger density.)

*****The following should be considered for solving this Question*****

3.1 Requirements and Planning

Your main task is to perform requirements analysis and project planning for the problem description below. The documents you must create (and submit) are as follows:

- A plan for the development of the system, including:
  - A Work Breakdown Structure (WBS). Each task should be broken down into a reasonable set of sub-tasks.
  - An AON or AOA graph incorporating duration estimates and dependencies.
  - Identification of the critical path and estimated overall project duration, by clearly mentioning the Early Start (ES), Late Start (LS), Early Finish (EF), Late Finish (LF), and Slack time. Use PERT Chart to estimate.

Note
You are not expected to perform planning poker to derive the estimates, as this is *not* a group exercise. Just make some reasonable guesses on your own. Also consider the parallel tasks/activities in the WBS.

- An overview of the requirements, including:
  - Actor identification, identify at least 5 actors (think about the different between stakeholder and actors, while listing actors)
  - At least 10 (in total) user stories that summaries the key functional requirements of the system.
  - At least 4 (in total) use cases, each corresponding to a particular user story. Pick the more complex user stories for this purpose. Show all parts of each use case including extensions.
  - One (in total) UML use case diagram, showing which actors are involved in each function of the system. This one diagram should incorporate all the features covered by the user stories, not just your 3 fully-written-out use cases.
  - At least 3 (in total) usability requirements, 3 performance requirements, and 3 reliability requirements. These must be specific to the system at hand.

Note
The lower limits (described above) used this assessment will probably aren't sufficient to completely describe the requirements of the complete system, neglect that at this stage. Just describe a representative subset of the system's requirements.

- A plan for the use of version control throughout the project, by the development team; specifically:
  - What branches will the development team need, and what for?
  - When should they be created, and when should they be merged and pushed? (This is asking about time, measured in days or weeks after the start of the project, assuming everything goes smoothly. A simple hint is when you reach a milestone in the WBS, you may need branching and/or project staging)

Note
This is about what will happen in the future, once all the requirements and planning work are done. The version control branching can be used when there are parallel components after milestone/stage in the WBS. Merging can be planned when multiple components connect on a milestone.

3.2 Version Control

Based on the plan described in the above bullets, you need to describe the appropriate git commands, (as per your planned used of VCS) e.g., initiating local git repository (in the folder containing the plans).

Note
You can paste the screenshots of the actual instructions used, for storing, editing, and/or committing the plan documents for the local repository. You can use the planning document(s) i.e., components of this assignment as Git repo component (as codes used in the tutorial) or you can add some dummy files for your demo.

4. What to submit

All planning related document e.g., a word file describing actors, user stories, AON/AOA graph, PERT Chat, UML diagrams, etc. Also submit your local git repositories along with all trackable commits and file.

5 Submission Guidelines

The unit Moodle page will accept documents in .pdf, .odt, doc, .docx or .zip/ format. It is recommended to combine all files in a archive file and submit as one.

For the usecase diagrams and AOA/AON graph, include these as .pdf, .png or .jpg (if required or you can paste them in your actual word document or can submit in a zip file). you can use appropriate diagramming tool, e.g., draw.io, Dia, Visio; or (for UML specifically) Umlet, PlantUML, etc. You must save/export your work to a .pdf, .png or .jpg file though.

Zip your entire directory, being very careful to include the. git/ subdirectory (which contains all repository details, including all commits (if any)). Submit your .zip/ file to the“Exercise Submission 1” area on unit Moodle page.

Please verify that your submission is correct and not corrupted. You may make multiple submissions, only your last one will be marked. **However, late submissions are strictly not allowed** (LAP students considered and will be dealt with accordingly)**.**

6 Marking Criteria

You will be marked out of 40 marks; breakdown is as follows:

1. **Planning**
    - **3 marks –** Work Break Down Structure (WBS)
    - **3 marks –** AOA/AON graph
    - **2 marks –** Identification of critical path and project duration
2. **Functional Requirement Analysis**
    - **4 marks –** User stories
    - **4 marks** – Use case diagrams
    - **5 marks** – Written use case(s), including FOE and extensions
3. **Non-Functional Requirement Analysis**
    - **3 marks** – Usability requirements
    - **3 marks** – Performance requirements
    - **3 marks** – Reliability requirements
4. **Version Control**
    - **5 marks** – Actual use of version control
    - **5 marks** – Plan for using version control

7 Academic Integrity

Please see the Coding and Academic Integrity Guidelines on unit Moodle page.

In summary, this is an assessable task. If you use someone else's work or assistance to help complete part of the assignment, where it's intended that you complete it yourself, you will have compromised the assessment. You will not receive marks for any parts of your submission that are not your own original work. Further, if you do not reference any external sources that you use, you are committing plagiarism and/or collusion, and penalties for academic misconduct may apply.

Curtin college also provides general advice on academic integrity at https://www.curtincollege.edu.au/content/dam/navitas/upa/curtin/pdfs/academic-integrity-policy.pdf

The unit coordinator may require you to provide an oral justification of, or to answer questions about, any piece of written work submitted in this unit. Your response(s) may be referred to as evidence in an academic misconduct inquiry.