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

Exercise Submission 2

Weight: 40% of the unit

General Information. Include your name and student ID at the start.

Timeframe. You have 14 days (336 hours) to complete and submit your answers, from **4:30pm on 08th Jan 2024** until **4:30pm on 22nd Jan 2024 (UTC+8)**. You may schedule your work however you wish within this period. **However, late submissions are strictly not allowed**.

Submission. Submit your answer document to the **"Exercise Submission 2" on Moodle under Assessments Tab**. You must verify that your submission was successful. Correctly submitting is entirely your responsibility. It is recommended to put all your files in a zip/compressed file for submission (name zip file as: **studentID_Exercise2_ISE**).

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.

- 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 receivean 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.

Git Repository Requirements: Questions 3 and 4 require you to initiate a git repository, please follow the following guidelines in this regard:

- The git repository should be name as "**studentID_Exercise2_ISE**"

- The global git repository (if online) should be private.

- The link for the global repository (if online) should also be submitted.

Hint: Initiate git repository at the start of the exercise submission and add files as you develop.

Question 1: Planning, and Agile Software Project Management [Total Marks: 20]

Consider the following (Fig.1) 13 nodes AON and along with the Table 1 project descriptions.
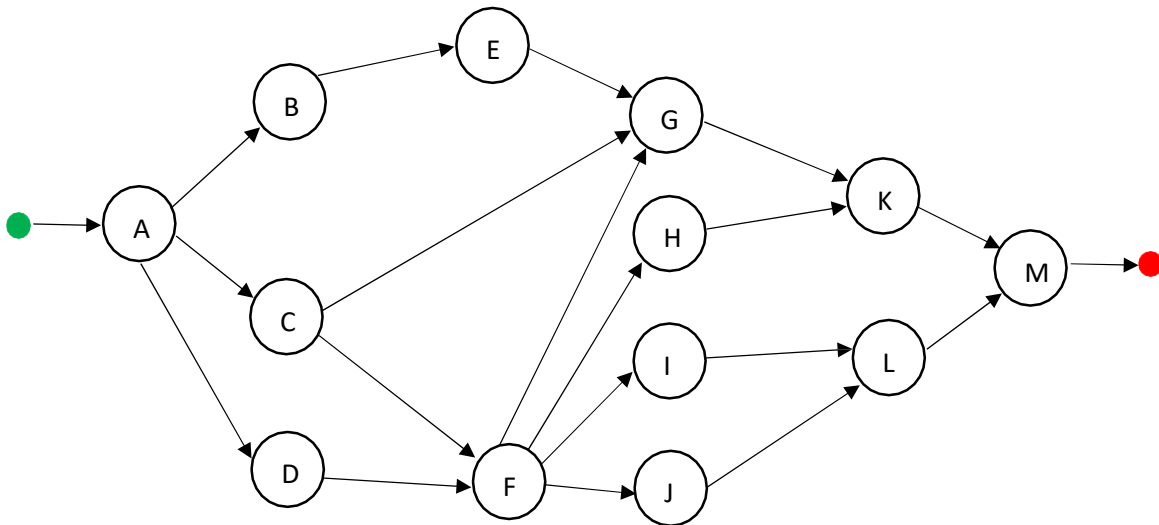

Fig. 1: 12 Nodes AON

Table 1: AON Graph Description

Sr.	Task	Estimated Duration (weeks)	Dependencies
1.	A	5	-
2.	B	3	-
3.	C	7	-
4.	D	15	-
5.	E	7	-
6.	F	3	-
7.	G	5	-
8.	H	6	-
9.	I	8	-
10.	J	9	-
11.	K	4	-
12.	L	2	-
13.	M	1	-

A. List the missing dependencies in Table 1. **(1 mark)**

B. Estimate the Early Start (ES), Late Start (LS), Early Finish (EF) and Late Finish (LF) for each task in the AON, present your answer in a tabular form. **(2 marks)**

C. Find out the task(s) which have slack time, and clearly mention the slack time of each identified task in tabular format. **(2 marks)**

D. Show your complete working (make a PERT chart) on estimation of overall optimistic project duration, along with critical path estimation. **(2 marks)**

E. You project is in at week 9 and you are using Scrum based project management, there is a sprint in 2 weeks, where client wants to add three activities (say N, O, P, in WBS) as backlog, what necessary action would you consider in this case, and describe the responsibilities of each team to deal with the situation **(4 marks)**

F. You has completed activity "H" by week 20 and all dependencies of "H". However, all other activities are not complete. Estimate actual performance on the Burn-up Chart and describe the overall project progress in text. **(4 marks)**

G. Show a Kanban Board when both **(5 marks):**

1. "G" and "H" are under review and "I" and "J" are done/completed with all their dependencies.

2. When activity "K" and "L" are under review.

Question 2 is on Next Page

Question 2: Functional and Non-functional requirements [Total Marks: 20]

Consider the following applications need to be automated with the help of a software:

1. Online baking application.

2. Mobile application for weather updates.

3. Online application for rental accommodation booking.

Select any two of the above applications and

A. Identify at least three stakeholders and three actors for each application (human/non-human). **(3 marks)**

B. Write one user story for each identified actor for (particular application). The user stories should be significantly different from each other. **(3 marks)**

C. Pick any two user stories from part B and give a full use case description for the selected user stories (considering them as functional requirements). The use cases must include at least 2 extensions for each use case. **(6 marks)**

D. Identify which one would be the best reliability metric (MTTF, MTBF, ROCOF, POFOD or Availability) for each identified user story in part B. Justify you answer by providing reason of selection. **(4 marks)**

E. Identify 3 usability requirement for 3 user stories (select any 3, from part B) **(4 marks)**

Question 3 is on Next Page

Question 3: Unit Testing [Total Marks: 25]

*Following considerations should be made while answering this question

- Name the file (python file) appropriately e.g., "MyCalculator.py" (for a calculator software)
- Name the functions appropriately e.g., "MySum" (for sum operation)
- All code should be executable on college provided VMWare based Linux environment.
- This question uses the git repository follow the guidelines.

A. Design a black box test (making table for all test cases) for the Equivalence Partitioning testing and write equivalent Python based test code. **(7 marks) [submit the test code as python files]-Production code provided (see codes folder Q3PartA.py).**

> A function takes four integer inputs (num1, num2, num3, num4), and returns the mean four numbers if the numbers meet the basic requirement and -1 otherwise. The basic requirement is, if num1 is grater then num2 and num3 is greater than or equal to num4, the result is mean of all number.

B. Design a test for the following for the Boundary Values Analysis (BVA) testing and write equivalent Python based test code. **(5 marks) [submit test code as python files] - Production code provided (see codes folder Q3PartB.py)**

> A function estimates the eligibility for COVID vaccine based on the entered age (integer value) of individuals. If the age is less than 6 the output will be "Not Eligible" for age 6 to 12 shows "Eligible for 1 dose" for age up to 16 "Eligible for 2 doses", for age up to 25 "Eligible for 3 doses" for age up to 45 "Eligible for optional dose" for age up to 60 output is "Must take optional dose" for age up to 85 output will be "See your GP for eligibility". For any invalid age ranges (e.g., <0 or >85) system shows an output "Invalid age".

C. Consider the following python code to find the median of 3 numbers, develop a test design for white box testing. **(7 marks) [Production code is provided with question (see codes folder Q3PartC.py), submit both test code and edited code (if any)]**

```python
def median():
    '''
    The funcation takes three values as input from user and calculates the median of the values.
    You are supposed to write a white box test for the code checking all paths in the code.
    Also, test the input and output on console (i.e. use test fixture concepts).
    '''
    value1 = int(input("Enter 1st number: "))
    value2 = int(input("Enter 2nd number: "))
    value3 = int(input("Enter 3rd number: "))
    median  = -1
    if (value1-value2) * (value3-value1)>0:
        median = value1
    elif (value2-value1) * (value3-value2)>0:
        median = value2
    else:
        median = value3
    print(median)
```

D. Implement your test design in parts A, B and C using python (unittest) modules, run show the screenshots in the report and submit the implemented python codes. **(6 marks) [Hint: Convert the codes in class-based implementation.]**

 *The online uploaded codes should be a private repository.

Question 4 is on Next Page

Question 4: Modularity [Total Marks: 20]

*Consider the provided code file (**Q4.py**) with the exercise document and perform the following on the provided code. This question uses the git repository follow the guidelines.

    A. Identify the modularity issues in all functions, list (make a table, see example below) the function name, identified issue, category of issue, comments on how it is an issue, proposed solution. **(5 marks)**

Table 2: Example details expected

Sr.	Function name	Identified issue	Category of issue	Comments	Proposed Solution
1.	Func_1	Global variables used	Coupling issue	The global variables increase the dependency of Func1 and Func3	Remove global variables by parameters

    A. Demonstrate how the flag(s) used in the code provided are affecting the modularity, what specific kind issues are introduced using flag(s) in the code. **(5 marks)**

    B. Refactor the code provided by remove the modularity issues identified in part A. change all functions and function calls in main function and other functions in the code. **(7 marks)**

    C. Use version control to put all files initial and refactored file, making sure to meet the basic functions after refactoring (code running after refactoring) **(3 marks) [provide the git repository and online git link (if uploaded to global repository i.e., online), Name repository as: studentID_Exercise2_ISE]**

    *The online uploaded codes should be a private repository.

Question 5 is on Next Page

Question 5: Ethics and Professionalism [Total Marks: 15]

For each of the following sets of software systems:

- Pick one of the systems listed.

- Pick one general area of software engineering from among the following: planning, functional requirements, non-functional requirements, project management, and testing.

   (Your choice here should be different for parts A, B and C)

- Describe a specific way in which a lack of professional conduct in this area, for this system, could lead to a harmful outcome.

Note: marks for this question will depend on the specificity (degree of detail) and plausibility of your answers. You will not receive marks for rewriting examples already given in the unit material.

A. Pick one from the following: **(5 marks)**

   a. Online medical record management software

   b. Online booking software for bus tickets

   c. An online software for tracking flights (google flightradar24)

B. Pick one from the following: **(5 marks)**

   a. An online chat server

   b. Video calling software.

   c. Bus management software for a city (e.g. Transperth)

C. Pick one form the following: **(5 marks)**

   a. Patient record management software for flu vaccine.

   b. Mobile food delivery app

   c. Online stock market application.

Submission Guide on Next Page

Submission Guide

Consider following for submission for this assessment

1. Main word/Pdf document consisting of answers for all questions (i.e., report)

2. Submit all figures, image, screenshots, graphs, and charts

3. Submit all production and test cases, also git repository (both local, global (if any, as link of repo))

Use of Git VCS

Make one folder for complete assessment, put everything in that folder and initiate a git repository in the same folder. Add (stage) all files (specifically codes), make commits when required (specifically for code). Zip/compress the main folder including all files and git repository and submit the zip/compressed file on Moodle page.

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.