TAYLOR'S UNIVERSITY

Wisdom · Integrity · Excellence

ITS66404 Software Engineering

ASSIGNMENT 2

Project Based Learning (PBL) with Purpose Learning

HAND OUT DATE: 29 October 2021 (Friday) (Week 10)

HAND IN DATE: 12 November 2021 (Friday) (Week 13)

WEIGHTAGE: 30%

Instructions to students:

- The assignment should be attempted in groups of 5 students.
- Complete this cover sheet and attach it to your assignment – this should be your first page.

Student declaration:	
I declare that: ▪ ***I understand what is meant by plagiarism*** ▪ ***The implication of plagiarism have been explained to us by our lecturer*** ***This project is all our work and I have acknowledged any use of the published or unpublished works of other people.***	
Names of Group Members	
Name	**Student ID**
1.	
2.	
3.	
4.	
5.	

Avoid copy and paste job in your report and it is considered as plagiarism. Plagiarism in all forms is forbidden. Students who submit plagiarised assignment will deserved a 0 marks.

1.0 Objective/Learning Outcomes:

This assignment has been designed for students to:

- Demonstrate subject specific skills with respect to applying modelling techniques and notations in the group-project coursework.
- Apply theory and foundations of software engineering.
- Apply some key aspects of a software engineering process.
- Determine requirements for a software system.
- Create and finalise models for analysis of a software system.
- Apply some key aspects of project management.
- Create test cases based on use case specification

Please note that assignment 2 is an extension from assignment 1.

2.0 Requirements:

You should complete the following:

1. Revised problem analysis based on your assignment 1, consisting of:
    a. **Modified use-case diagram** to include further observations/amendments where required. (If there are no amendments compared to your assignment 1 submission, an explanation is expected).
    b. Complete **use case descriptions and use case specifications** for ALL use cases. Use the template provided for use case descriptions and specifications.
    Note: At least FIVE (5) flows of events expected out of each use case.

2. Modified class diagram incorporating additional requirements that has been missed out in assignment 1.

3. Interface design proposal, consisting of **user-interface mock-ups** (sample screenshots – can be very basic) for 3-5 major user interaction cases should be produced.

4. A **test plan** for 3 major use cases, including at least FIVE (5) test cases per use case (follow use case specification). Use the template provided for testing.

5. At least THREE (3) **sequence diagrams** expected for any use case covering the happy path and one of the alternative pathways. *Hint: Drawing more than one diagram may be a good idea should the first diagram be incorrect.*

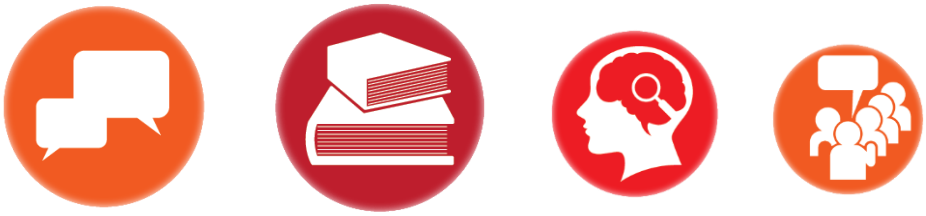
6. At least THREE (3) **activity diagram**. *Hint: Drawing more than one diagram may be a good idea should the first diagram be incorrect.*

7. Draw at least THREE (3) **state chart diagram**. *Hint: Drawing more than one diagram may be a good idea should the first diagram be incorrect.*

8. Discuss your **design rationale** consisting of
    a. Strengths and weakness of the final design (including any known issues with the design as it is) (minimum 2 paragraphs)
    b. Alternative design ideas that were considered (classes, structures, patterns, etc…) and why they were eventually rejected? (minimum 2 paragraphs)

9. A group analysis (to be completed last), consisting of
    a. A brief discussion about the strengths and weaknesses of the project team and how/if these were managed effectively (minimum 2 paragraphs)
    b. An evaluation of lessons learnt about teamwork – what would you do differently (and the same) given the opportunity (minimum 2 paragraphs)

After this assignment, you will learn the following TGC skills:



3.0 Marking Scheme

Marking Criteria	Marks (5 marks for each criterion below)				
	Excellent (5)	Good (4)	Average (3)	Poor (2)	Very Poor (1/0)
[1 (a)] Modified Use Case Diagram					
[2] Class Diagram					
[8 (a)] Rationale – Strength(s) and Weakness(es)					
[8 (b)] Rationale – Alternative Design					
[9 (a)] Discussion – Strength(s) and Weakness(es) of Project Team					
[9 (b)] Evaluation – What would you do differently given another opportunity?					
SUB-TOTAL (T1) = 6 * 5 marks = 30 marks (max)					
Criteria (As per requirements in Section 2.0)	**Marks (10 marks for each criterion below)**				
	Excellent (10/9)	**Good (8/7)**	**Average/Poor (6/5)**	**Very Poor (4/3/2/1/0)**	
[1 (b)] Use Case Descriptions					
[3] User Interface Mock Up					
[4] Test Plan					
[5] Sequence Diagram					
[6] Activity Diagram					
[7] State Chart Diagram					
SUB-TOTAL (T2) = 6 * 10 marks = 60 marks (max)					
T3					
[10] Presentation - Language and Delivery • grammatical accuracy • fluency • pronunciation • intonation • volume and pace					
[11] Expression and Body Language Delivery • confidence and enthusiasm • positive non-verbal body language (eye contact, poise and gestures) • connection with audience					
[12] Video – contents, storyboard and creativity					
[13] Prototype (alpha version is required, but preferably beta version)					
[14] Others – participation in competition(s).					
Total Marks (T1 + T2+T3) = ***Will be scaled down to 30%***					

4.0 Documentation & Submission Instructions

1. The documents for each milestone should be neatly organized with table of contents included.
2. Please ensure that your diagrams **must be drawn using Visual Paradigm** and appear clearly on print.
3. You are to submit a softcopy of your assignment (one per group) to the lecturer at the latest by 5pm due date via digital dropbox at TIMeS.
4. You are required to present your idea and software prototype at this stage. Date of your presentation will be mutually discussed and agreed between you and your lecturer.

Important:

a) Late submissions will be penalized as per school policy.
b) Each student is expected to contribute significantly in all deliverables as the assignment is a joint effort. In the event where a student's contribution is grossly unequal, marks shall be deducted and awarded to a group member who has done the work of his teammate (if applicable).

Use Case Descriptions

Use Case Name	Basic Flow @ Happy Path	Alternate Flow @ Alternate Path	Exception Flow @ Exception Pathway

Use Case Specifications

Name:

Description:

Author(s):

Actor(s):

Location(s):

Status:

Priority:

Assumption(s):

Precondition(s):

Postcondition(s):

Primary (Happy) Path:

Alternate Pathway(s):

-

Exception Pathway(s):

-

Testing

Note that *at least* one of your test cases should be a negative test, meaning that you should be planning a test that will create an error for the system to handle (e.g. Logging in with the wrong password). Several test cases can belong to the one use case, testing different scenarios/outcomes, or else can be across five different use cases.
The template you can follow for the test cases is below. You may use your own template if you wish, but these core elements should be in them, and they must all be consistent.

Template/example for one test case:

Test Case
Test case ID, eg. "TC01", and/or a name for the test case, eg. "Student logging in with correct credentials".

Use-case
Use case ID, eg. "UC01" and/or the use case name, eg. "Login".

Description
One line description of what the test is for; eg." Student logs in successfully supplying username and password."

Procedure
*Steps to follow in order to carry out the test – these steps must be*specific*, as opposed to the use case descriptions that are generic; eg.*
"Follow UC01 normal course with username "TestUser" and password "Test'".

Expected result
Outcome you would expect to see; ie. if it is a negative test, you would expect to see some kind of error message, etc… otherwise, what should happen next; eg. User should be logged in as 'TestUser'. Check that username appears on top.

A sample test table is provided below;

Use Case	**Test Case**	**Description**	**Procedure**	**Expected Results**
UC01: Login	TC01-1: Successful login	Student logs in successfully supplying username and password	Follow UC01 normal flow with username "TestUser" and password "Test'"	User should be logged in as 'TestUser'. Check that username appears on top
	TC01-2:			
	TC01-3:			

ALL THE BEST!