OXFORD BROOKES UNIVERSITY

**School of Engineering, Computing and Mathematics**

COMPT5047 Applied Software Engineering

Week 1 Practical Exercise

1. **Work During Practical Class**

Work with your coursework group to complete the following tasks.

1. Team building exercise (10 Minutes)

* Exchange contact numbers and setup a communication medium (such as a WhatsApp group).
  + Decide how the group will operate. For example, to elect a team leader for your coursework group, or to take turns to be the group leader.
  + Set a weekly group meeting date, time and venue.
  + Record the results of this task in a text file.

1. Set up SE platforms for individual work and group coursework (20 Minutes)

* Each member of the group (include the team leader) to create an empty GitHub repository for practical exercises of individual study.
* The team leader to create an empty GitHub repository for the group’s coursework and add all team members to the repository as contributors.
* The team leader to upload the file of records of the team building exercise to the coursework GitHub repository.
* Each team member to download the file from Coursework GitHub repository to your computer.
* Each team member to upload the file to your individual GitHub repository.

1. Work on the practical exercise case study (20 Minutes, *Continue after the class if unfinished*):

* Each student reads the *Case Study for Practical Classes*, i.e. the *Mobile Shopping System*, independently.
* Each student selects one user type and note down the functional requirements of the selected user type.

*Hint*:

1. Discuss with the group members to ensure each member does a different subsystem.
2. Use the coursework case study specification as an example and a template to complete this task.
3. Consult the practical class tutor if you have any question about the case study.

* The group holds a meeting, and each member reports to the group the functional requirements of the user type.
* The group discusses on what data are required by a subsystem for one user type but must be provided by other subsystems.

1. **Exercises After Practical Class**
2. In the context of the Case Study for Practical Classes, i.e. the *Mobile Shopping System*, apply the Lehman theory of software evolution to answer the following questions.

* What is the type of the software in Lehman’s classification? Why?
* For each type of software uncertainty in Lehman theory, what specific presentations of the uncertainty are most likely to occur in the development and operation of the system?
* Which type of uncertainty will dominant the development and evolution of the system? And why?

1. Read the textbook chapters listed in the *Further Reading* slide of the lecture notes. Apply the knowledge about software lifecycle and process models to answer the following questions.

* For each of the software process models taught in the lecture, what are the advantages and disadvantages to apply to the development of the system?

*Hint*: You should focus on the types of uncertainties associated to the software that may be easy or difficulty and/or costly to prevent and/or to minimise the impact when it occurs.

* Which process models you would use to develop the system? Why?
* With the process model that you selected, what is your risk management plan to prevent each type of uncertainties to occur, and/or to minimise the impact of the risk when it occurs?

1. **Coursework (**To be completed after the practical class**)**
2. For each student to work individually:
   1. Read coursework specification to understand what you are required to do to complete the coursework.

*Hint*:

* There are tasks that you have not learned and unable to do. But, no worries, you will learn how to do these tasks in this module.
* There may be tasks that you have learned in year 1 modules. However, this module required a much deeper understanding of the topic and skill to complete the tasks. This module will further develop your knowledge and skill to meet the requirements.
  1. Read coursework case study description to understand the system to be developed.

*Hint*:

* Although each student will only develop one subsystem, it is important to know the functional requirements of whole system. This is the foundation for collaboration with your teammates.
  1. Read the marking scheme to understand the expected quality on your work and the required level of knowledge and skill to complete the tasks.

1. For the group:
   1. The team leader to prepare a meeting agenda before the meeting, take notes during the meeting, and write the meeting minutes based on the note after the meeting.
   2. At this week’s group meeting,
      * You should decide who is working on which subsystem.
      * Each member to report to the group the functional requirements of the selected subsystem.
      * The group should discuss what are the data required by one subsystem but must be provided by other subsystems.
   3. The meeting agenda and meeting minutes must be uploaded to GitHub immediately after the meeting.