THE OPEN UNIVERSITY OF SRI LANKA
DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING
BACHELOR OF TECHNOLOGY
EEX4347 Software Engineering Concepts

Answers should be clear, and readable. Use A4 paper. Unclear, unreadable, copied and direct reproduction from the textbook will not gain any points for the answers. Answers should be uploaded into the respective Drop boxes in the LMS Course EEX4347 - Software Engineering Concepts.

[IMPORTANT] All drop boxes will automatically close at the end of the submission date. Any submissions in any other form (e.g. emails, submission of hardcopies etc.) will not be accepted after the due date unless there are extenuating circumstances beyond the control of the student. In such a situation, the student should contact the course coordinator BEFORE the due date and make alternative arrangements. No appeals will be entertained after the due date.

==========

TMA 1

The aim of this course is to enable you to gain knowledge and skills in Structured Software Engineering methods as well as Object Oriented Software Engineering methods. Therefore, each assignment of this project work has two parts, i.e. the structured and the object oriented part. You have to do both parts.

Develop a proposal using the given SRS Template. A template (an example on how to write a project proposal) is given in Moodle class. You MUST use the template.

Problem Description

One of the challenges faced by the students at the Open University of Sri Lanka is the isolation and lack of peer support in Open and Distance Learning (ODL). To address this issue, the university proposes a Virtual Study Groups and Social Learning Platform. This platform aims to bridge the gap between students and foster a sense of community, despite the physical distance inherent in ODL. By leveraging modern communication and collaboration tools, the platform will provide students with opportunities for interaction, collaborative learning, and peer support. Key features include the creation and management of virtual study groups, collaborative tools such as shared whiteboards and document co-editing, and discussion forums and channels for topic-based academic discussions. Additionally, direct messaging will enable personalized communication, and peer mentoring programs will match experienced students with newcomers for academic and emotional support.

The platform will also offer interactive online classes and workshops through video conferencing tools like Zoom or Microsoft Teams, with breakout rooms for small group discussions and activities. Social media integration with platforms like Facebook or LinkedIn will create private groups for courses, facilitating informal student connections and event announcements. Gamified learning experiences, including interactive quizzes and games through tools like Kahoot! or Quizlet, will encourage friendly competition and social interaction. Leaderboards and badges will foster a sense of achievement and motivation.

The system requirements definition activity is intended to discover the requirements of the system as a whole. This process involves consultation with all stakeholders including customers and end users. You may interview your peers or other students of the OUSL or any other stakeholder of the system to identify the requirement.

The requirement definition phase usually concentrates on deriving;

- Functional and Non-functional requirements
- System properties
- System scope

Create the Software Requirement Specification (SRS) using the template given in Moodle.

You should complete the following tasks for TMA 1.

Tasks to be completed by

(a)	Draw a comprehensive Use case diagram to illustrate the main Business processes	25 Marks
(b)	Upload your Use Case diagram to the discussion forum in the LMS by Wednesday, 23 July 2024	4 Marks
(c)	Mark the submissions by two other students out of a maximum 20 Marks. Give at least 3 constructive feedback comments to each of the students. This activity must be completed by 5 August 2024.	6 Marks
(d)	Write functional requirements (minimum 15)	15 Marks
(e)	Write non-functional requirements (minimum 5)	5 Marks
(f)	Define the system scope	5 Marks
(g)	The finalised SRS (using the template given in the Course in in the LMS)taking into account comments by your peers should be submitted to the Drop Box for TMA1 by **10 August 2024.**	

========== **END of TMA1** ==========

TMA 2

Investigation into the Impact of Large Language Models (LLM)s and Generative Pre-trained Transformers (GPT) in the Discipline of Software Engineering

Artificial Intelligence (AI) has significantly transformed various aspects of our lives, and the introduction of Generative Pre-trained Transformers (GPT) has opened up new possibilities in utilizing AI in everyday applications. One such area is software generation, where platforms like ChatGPT, Google Bard, GIT CoPilot, and Amazon CodeWhisperer allow for automatic code generation based on problem descriptions. There is a growing debate about whether AI tools will eventually replace software engineers, thus, leading to a reduced demand for their skills and expertise.

Your taks is to do a detailed investigation into the statement "AI tools and apps will soon replace Software Engineers" and write a report on it. Your report should contain:

(a) Currently available tools (platforms, APIs, Interfaces, Browser Plugins etc.) with references (include at least 10 tools and a brief paragraph describing main features of each)
(b) Arguments presented by leading experts in support of the statement with references
(c) Arguments presented by leading experts against the statement with references
(d) Evaluation of your findings
(e) Conclusions

You may use any available tool including AI tools such as ChatGPT, Google Bard and Microsoft Bing during the investigation. If you use any such tools, your prompts (questions you asked from the AI tools) should be included in an annexure.

Your report should be less than 5 A4 pages with 1" margins and in a font size not less than 11 points (excluding any annexures and a cover page). Report should be submitted to the TMA 2 Dropbox in the LMS by due date. Report should be submitted in an editable format such as Microsoft Word, Open Office, Wordpad, and Notepad. Documents submitted as Image files or PDF files will not be marked.

The final report should be written by you in your own style.

Any report with material found to be Copied from any available source including AI generated text will score 0 marks.

========== **END of TMA2** ==========

Mini Project

Mini project consists in developing the entire system for which the SRS was prepared in TMA1. You may use either Java Language or Python to develop the system.

You should submit a final report on the project as well as the source code of your programs by due date using the Mini Project Drop Box in the LMS. The source code files should be put in a folder and the zipped folder should be uploaded into the Dropbox marked .

Mini Project evaluation will consist of evaluation of the submitted project report, a Viva-Voce Examination (Oral examination) and the demonstration of the completed program. Mini-project demonstration and viva voce will be about 15 minutes for a student. The student should present the project and demonstrate the system within a maximum of 10 minutes. 5 minutes will be allocated for Questions from the evaluation panel.

Student must bring a copy of the final report (hardcopy or softcopy) and the source code (as a soft copy only), and the program in an executable format when he/she is attending the viva voce.

Marks for the Mini Project are allocated as follows:

Mini Project Report	50%
Working Program and Viva voce	50%

[IMPORTANT] *To become eligible to sit for the final examination, you should score 40% or more for the mini project.*

For the mini project, attending a viva-voce examination is compulsory. If you do not attend the presentation, 0 marks will be allocated as the for mini project mark. This will make you ineligible to sit for the final examination of this subject.

The date and time for mini project demonstration will be notified in due course.

Format for the Mini Project Report (50% of total project marks)

1. Table of Contents
2. Introduction
3. Requirement Analysis (10%) - need use case diagram, requirement elicitation methodology
4. High-level design (10%) – class diagram, collaboration diagram, activity diagram
5. Data Modelling (10%) – Relational database, normalization

6. Detail design (08%) – screen flow diagram, any other necessary diagrams
7. Testing Plan and Results (08%)
8. Conclusion or Discussion
9. References
10. Appendix – for any other relevant details such as screen layouts, report formats

Please modularise your program and write enough comments to make it readable. 5% of the marks allocated for the program will be given for program readability and comments.

Note : Hard Copy of the source code is NOT required.

========== **END of Mini Project** ==========