# ITCS371 Introduction to Software Engineering
# Faculty of Information and Communication Technology

## Project Phase 0, Semester 1/2023

## Score: 3 points

## Submission Date:

Sunday 3rd September 2023 @ 23:55 via GitHub
(Fourth week of the class)

### Overview

The project is designed to simulate the software development process. Groups, representing software development companies, will gather requirements from their external customers, in this instance, the instructors. Each group will be assigned one of the project topics. The term project is segmented into *four phases*, as described below:

*Phase 0:* This phase emphasizes requirement elicitation. The outcome will be a list of requirement statements. Submissions for this phase are due around the fourth week of the class.

*Phase 1:* In this phase, the emphasis is on requirement analysis using use case modeling and context diagrams. The targeted outputs are the use case diagram, use case narrative, and context diagram (DFD Level 0). Groups are required to submit all expected deliverables and present their work in week 8 (before the midterm exam).

*Phase 2:* Here, detailed Data Flow diagrams (DFD) are expected. Subsequently, the group will go deeper into the detailed analysis of the provided requirements.

*Phase 3:* This final phase involves designing the user interface and developing test cases. Groups must submit all expected deliverables and present their findings in week fifteen, which is the last week of the class.

*Bonus:* Earn an extra 10% on your project score by integrating advanced technologies like AI, VR, AR, IoT, or blockchain. To get the bonus, make sure you clearly show how you're using these technologies in both your requirement analysis and software design. Please make sure that you choose the technology that is suitable for your project, and you can explain the benefits of using them.

**Team members:** 5 persons

### Project Phase 0: Requirements Elicitation and Management

For this phase, you'll be given a topic and a preliminary requirement description through a video recording. It's essential to watch the video to fully understand the project's domain and concept. After viewing, you'll need to:

1. Develop a list of both functional and non-functional requirements. Each requirement statement should have an identifier (e.g., R1, R2, R3) for easy traceability.
2. Seek additional information as necessary to complete the requirements.
3. If there are uncertainties, don't hesitate to contact your 'customer' (i.e., the instructors) to clarify any requirements.

### The expected outputs

| Item | Minimum expectation |
|---|---|
| Introduction of the work including overall description of the project, customer, and the business | 500 words |
| Functional requirements | 20 requirements |
| Non-functional requirements | 10 requirements |
| Identifying actors | At least 2 -- based on requirements |
| **Bonus**: Include additional requirement statements that involve the integration or utilization of cutting-edge technologies. | |

### Note

- Detailed instructions for each deliverable will be provided during the class sessions that cover the related content.

### Overall Evaluation Criteria

Your submissions will be assessed based on the following factors:

- **Analysis Quality**:
  - Completeness and accuracy. For instance, ensure your list of functional/non-functional requirements is exhaustive, and all actors are identified.
- **Deliverable Quality**:
  - Readability and presentation, including correct spelling and grammar.

**Submission Instructions using GitHub**

1. Platform Choice: GitHub will serve as our main platform for the project, catering to submissions and feedback exchange.
2. Repository Assignment: Instructors will create and assign a GitHub repository to the leader of each group.
3. Team Access: It's the leader's responsibility to invite all group members to the assigned GitHub repository.
4. Communication: Use GitHub issues for queries or to communicate with instructors (@morakotch for Aj. Morakot and @cragkhit for Aj. Chaiyong). While you can utilize this feature, remember that instructors may also provide verbal feedback during class sessions or designated office hours.

**GitHub Submission Guidelines**

1. **Commit Deadline**: Only the commits made before the deadline will be considered for grading. Should there be any commits after the deadline, the project will revert to the last valid commit prior to the deadline for evaluation.
2. **File Identification:** Ensure the files in your repository clearly indicate their relevance to the project. Adopt descriptive filenames such as 'ProjectDescription' or 'UseCaseDiagram'.
3. **Folder Structure & Deliverables:** All your submissions should be placed in a folder labeled **"phase-0"** in your GitHub repository. Any work outside this repository will not be assessed. As an example, if you address an issue but don't integrate it into the repository, it won't be considered for grading.
4. **File Viewability:** It's vital that every deliverable is easily viewable on the GitHub website. While you're free to use any tool for diagram creation, ensure that the uploaded version to GitHub is in a universally viewable format.

**Your score will be deducted if you do as follow:**

- *** Only the work that is submitted before the deadline is graded. Having no work submitted before that means a ZERO score ***
- Failure to follow the instructions will make you lose points.
- Any issues/problems/obstacles in progressing your project must be informed before each deadline.
- All instructors will consider any issues/problems outside the rules stated above. The decision and the penalty from both instructors are final.