



**School of Computer Science**

**COMP20050: Software Engineering  
Project-II**

**Week1 Laboratory Practical**

<b>Module Coordinator</b>	<b>Ravi Reddy Manumachu</b>
<b>Module TA</b>	<b>Maryam Gillani</b>

In the lab sessions this week, you will create your group, a GitHub repository for the project, and start the architectural sprint. Note you must complete the group and GitHub repository creation by the end of this week.

## Group Creation

Self-select your group at My Class > My Groups on Brightspace.

Send your group details to the TA.

After the group selection expiry date this week, any class members who have not self-selected will be automatically and randomly allocated to a group.

## GitHub Repository

Git and GitHub will be the source code repo and version control for this project.

Setup Git on your computer. Setup your GitHub account online.

Setup a GitHub repo for the project. **THIS REPO MUST BE PRIVATE.**

The repo name should include your Group numbers allocated on Brightspace.

Share the repo with your team members.

Give repo access to the TA (user name: gillanimaryam) and the Module Coordinator (user name: ravimanumachu).

## Trello Kanban Boards

Explore **Trello** that provides task boards to track your project features (software development tasks).

## Software Architectural Design

You will start the software architectural design of HexOust board game.

You will research and gather information related to designing **software architectural design**.

While performing the design activity, you will seek answers to the following questions.

- **What is software architectural design?**
- **What are different software architecture styles and characteristics?**
- **What are the software architecture patterns best suited for this project?**
- **What are the best tools for software architectural design?**

You are allowed to use any modelling tool (like **UML**) to show the external, interaction, structural, and behavioral perspectives of your software system for HexOust.

Once you have the high-level design completed, then you can start thinking about **low-level design and implementation**, which includes interfaces, classes, data structures, and algorithms.

Your low-level design will be the basis for your **project plan** comprising four sets of features (sprints of two week duration) to complete the software implementation.