# CSE470: Software Engineering, Project Outline

## Project Purpose:

The purpose of the project is to provide hands-on training to the students on how to develop a software from scratch.

## Key Definitions and Examples:

**Requirement:** an action/capability that is expected of a software/product to fulfil
**Feature:** a set of logical actions that need to be performed in order to fulfil a part of a requirement

**Example of a requirement**

1. Customers can add products to their cart
2. Customers can track their delivery status

**Example of features**

- For requirement (1) in the examples, the features which need to be implemented are:
    - The system should be able to display products to customers
    - The system should be able to book products and mark them as added to the cart of a customer if the product is in stock
- For requirement (2) in the examples, the features which need to be implemented are:
    - The system should be able to show all undelivered orders from the customer
    - The system should be able to store data regarding the status of the order
    - The system should be able to store data regarding the products in the order
    - The system should be able to show the status of the products and the order to the customer

**Tech stack:** The language and the environment which is used to develop and run an application

**Example of tech stacks**

- **LAMP** Stack (Linux OS, Apache web server, MySQL database, PHP language)
- **MERN** Stack (MongoDB database, Express JS backend, React JS frontend, Node JS environment)

# Project Constraints

- Students may develop either a web application, android application, iOS application or a desktop application.
- Students may use any tech stack/programming language/scripting to develop their application
- The application developed **MUST** follow the MVC architecture
- Students may use existing frameworks (Laravel, Flask, Express, Angular, Vue) to help with the development as long as it implements the MVC architecture. (Django is not permitted since it implements MVVT instead of MVC architecture)
- Students may use libraries/packages to help with the development of their project as long as it does not implement any major feature of the project. A library/package implementing any major feature is considered illegal and the student using the library will be penalized. Consulting the respective faculty member before using a library/package is recommended.
- Students must develop the project individually and from scratch. They may not take help from any source, copy code from any source or use any tutorials which directly shows how to implement the project that they have chosen.
- Students may not use a code base which they/others may have worked before.
- Students must select a project that they want to implement which is of lower-moderate complexity
    a. The project must have at least five (5) requirements
    b. Each requirement must have at least four (4) features
    c. Login/registration will not be considered as features or requirement for the purpose of this task
- Students must define the requirements and the features of their project while submitting their project idea

# Project Process Details

- Students will form a group of 4 members to develop the whole project
- We will follow Agile methodology to implement the project whereas scrum must be followed by the team.
- There will be 4 sprints to complete the whole Software implementation and each sprint will be managed by one sprint lead from the group.
- The sprint durations will be of 2 weeks each, after the end of a sprint the sprint lead will be changed to another member of the group
- The sprint lead will have to collect the information from each of the group members about their project progress and submit it at the end of the sprint
- The information that the sprint lead needs to collect and/or provide are:
    - The tasks that each member wanted to complete during the sprint (this information has to be collected at the beginning of the sprint)

    - The tasks that each member has completed (this information has to be collected at the end of the sprint)
    - Score each team member according to the progress that they have made in completing their project
    - Identify the team members who made the least progress
    - Identify and report the problems faced by each team member
- Each Group must use github to maintain their project code. There will be one repository per group and all the members should work there as a collaborator by creating branches.
- Each student must push their code to the project repo at the end of each sprint regardless of if there are incomplete features or errors or bugs
- Student must complete their projects within four (4) sprints
- After the sprints, each student must attend a 15 minutes viva where
    - They will show their project demo in 5 mins
    - They will be asked questions about their project code individually for 10 mins
- All viva will be recorded and stored for review purposes if possible
- The viva may be taken offline or online and the dates and times will be assigned by the section faculty
- Students **MUST** create the github repository for their project and add the section faculty as a collaborator to the repository during the first sprint
- Students **MUST** submit their project idea, requirements and feature before the beginning of the first sprint
- A section faculty may change the project, the requirements and/or the features if the section faculty determines that the project is not complex enough, or too complex or for purposes of removing potential duplication