



Overview	
Lectures	>
Schedule	>
Exam1	>
Labs	>
Lab 1	
Lab 2	
Lab 3	
Lab 4	
Lab 5	
Lab 6	
Lab 7	
Lab 8	
Lab 9	
Project	>
Syllabus	>
Guides	>
Regrade Requests	🔗

Lab 9: Kick start group project

Overview


In this lab, we will kickstart the group projects.

To receive credit for this lab, you MUST be present during the recitation. Please note that submissions will be due right before your respective lab sessions in the following week. For Example, If your lab is on this Friday 10 AM, the submission deadline will be next Friday 10 AM. There is a "NO LATE SUBMISSIONS" policy for labs.

Learning Objectives

All of these tie in closely with the group project.

- ☐ L01. Learn how to structure an application with a GUI, database, and a middle layer connecting the UI to data, while optionally integrating APIs.
- ☐ L02. Gain practical experience in containerizing applications using Docker to ensure portability and consistency across different environments.
- ☐ L03. Develop skills in applying Agile or hybrid methodologies, such as Scrum and Kanban, to manage project tasks and team collaboration effectively.
- ☐ L04. Learn to design high-level use case diagrams and wireframes to define the application's key features, user interactions, and interface layout.


 PROJECT GUIDE

Please read the [project guide](#) alongside this writeup to get more information on the expectations for this group project

Part A

Complete [Pre-Lab quiz](#) on Canvas before your section's lab time.

Part B

 NEW SUBMISSION GUIDELINES

For all labs starting this one, you will be submitting all your work on the repository you create this lab. Once you have uploaded the deliverables to the repository, you are required to submit a link to the relevant files on Canvas. One submission per group is sufficient. **Missing submissions on Canvas will lead to a 0 for that assignment.**

In today's lab, you will work on starting off your project. Your TA will inform you of the approved project idea. You will be working with your team of 4-6 students on this project for the next 4 weeks. You will submit the project proposal by the end of this lab.

Project Overview

Your application must have a GUI, a database, API data source(optional), a middle layer connecting your UI to your data. The code should be built within Docker containers so they are portable. Make sure to consider these core requirements when designing the architecture of your application. You are required to use the technology stack that you have learned in this course. Refer to the [project guide](#) for detailed expectations. If you would like clarifications, please ask your instructor/TA.

Deliverables

Before starting this discussion, assign a scribe who will start a document and take notes on the items listed below. The team will proofread and submit this document as your deliverable for this week. Please refer to Canvas for the due date.

1. Team Number

You have been assigned a team number, on Canvas, that is represented as your recitation section number plus a one-digit identifier. This team number MUST appear on ALL deliverables/submissions posted by your team to your git repository and to Canvas.

2. Team name

You will collectively choose a name for your team to represent a vision you have for the application you're creating. It can be used in addition to your Team Number for all deliverables and presentations.

3. Team Members

List the First & Last name of team members, their GitHub usernames and their email addresses.

4. Application Name

Pick a befitting name to convey the purpose of your application. To be used in all presentations(weekly to your TA and the final one)

5. Application Description

A short (2-3 paragraphs) overview description of the application. Provide enough information to explain what functionality and value your product will provide to users of your application. This should help you to define the scope of your project.

6. Audience

Define the target audience for your application by identifying the characteristics of an ideal user. Clearly articulate the specific problem your software aims to solve for this user group. Additionally, consider the usability and accessibility of the application, ensuring that interactions are intuitive and convenient for the intended audience.

Understanding their background helps shape features and design.

7. Vision Statement


A simple, one or two sentence statement describing the clear and inspirational desired state resulting from your team's efforts to create your application. Following is a template you might want to use:

For [target customer], **Who** [statement of the need]. **The** [product name] **is a** [product category] **that** [key benefit, compelling reason to buy]. **Unlike** [primary competitive alternative], **our product** [statement of primary differentiation]

Examples

- Amazon:** "To be Earth's most customer-centric company, where customers can find and discover anything they might want to buy online."
- Tesla:** "To create the most compelling car company of the 21st century by driving the world's transition to electric vehicles."
- Facebook:** "People use Facebook to stay connected with friends and family, to discover what's going on in the world, and to share and express what matters to them."
- CU Boulder:** "To be a leader in addressing the humanitarian, social, and technological challenges of the twenty-first century."
- IKEA:** "To create a better everyday life for many people."
- Nordstrom:** "Offer the customer the best possible service, selection,quality, and value."
- Google:** "To provide access to the world's information in one click."

8. Version Control

 INFO

You are required to create a **public** GitHub repository and add all the members of the team to it. Share the link to this repository in this document.

This repository should have the following folders:


- TeamMeetingLogs** - The minutes of the meeting with your TA every week will be recorded in a file here. You will be updating the same file every week with the latest updates. Points to include in the minutes would be:
 - Decisions made
 - Alternative actions/options discussed
 - Follow-up items, including agreed-on roles and responsibilities
- MilestoneSubmissions** - All course-related documents, including this one, will be stored in this folder.
- ProjectSourceCode** - The source code and all relevant project documentation for the application will be stored in this folder.
- ReadMe.md** - Refer to the [project guide](#) for more information on this.
 - For a detailed view of your repository, refer to the [project guide](#)
- .gitignore** - Please remember to create a .gitignore file in your "ProjectSourceCode" folder in the repository

9. Development Methodology

Since this project is scheduled for 4 weeks, we encourage you to use an Agile Methodology. You can also use a hybrid set of practices from different methodologies. For example, you can use Scrum meetings and Kanban boards to track the progress of your project. The important thing is to have a plan and stick to it.

10. Communication Plan

Describe in a sentence or two how your team plans to communicate with each other during the course of the project. You must identify a collaboration tool for team members to utilize for coordination of their work and communication among team members and with the TA. Some tools you could use: MS Teams, Slack, Discord, Google Groups, etc. Please refrain from using IMs for the purposes of this project.

 TIP

You can track communication for project development tasks as comments on the user story issues you create on the GitHub project board. That will help you remain organized.

11. Meeting Plan

- Team Meeting:** Identify the day(s) and time(s), mode, and location your team has agreed upon for regular meetings. "Modality" refers to how you are meeting (face-to-face), group video chat (like Zoom, Google Meet), etc.
- Weekly meeting with TA:** You will also mention the meeting day, time and location (physical or online) for the weekly meeting with your TA here.
 - Please refer to [these instructions](#) for expectations in your weekly meeting with your TAs. Do read the entire section carefully.

12. Use Case Diagram

Create a high-level Use Case Diagram for your application. Your diagram must show at least 6 key features of your application.

This activity should help you identify the end users of the application and the ways they would interact with the system. You can reference the examples covered in class. This will also help you scope out the features of your application. You can refer to [this page](#) for more information on how to create a Use Case Diagram.

13. Wireframes

You will create wireframes for each page of the application.

Wireframes are low-fidelity visual representations of the user interface of your application. They are used to plan the layout of the application and to communicate the flow of the application to the team. You can use any tool of your choice to create the wireframes. Hand sketched wireframes are also acceptable. You can refer to [this](#) for more information on why and how to create wireframes.


Part C

There is no part C for this lab 🤖

Submission Guidelines

Commit and upload your changes

```
[user@host] $ git add .
               git commit -m "Added/Updated all files for lab 7"
               git push
```

 TIP

After you have run the commands given above, please navigate to your GitHub remote repository (on the browser) and check if the changes have been reflected. Once that is done, please submit a link to your repository on Canvas.

You will be graded on the files that were present before the deadline. If the files are missing/not updated by the deadline, you could receive a grade as low as 0. This will not be replaced as any new pushes to the repository after the deadline will be considered as a late submission and we do not accept late submissions. Please do not submit a regrade request for this.

Regrade Requests

Please use this [link](#) to raise a regrade request if you think you didn't receive a correct grade. If you received a lower than expected grade because of missing/not updated files, please do not submit a regrade request as they will not be considered for reevaluation.

Rubric

Criteria	Description	Points
Part A - Lab Quiz	Complete the Pre-Lab Quiz before your lab	20
Team Number	The team number is specified on the document	1
Team Name	The team name is specified on the document	1
Team Member	The team number details are specified on the document	2
Application Name	The application name specified on the document	3
Application Description	There is a sensible description of the application	5
Audience	There is a sensible description of the user base	3
Vision Statement	There is a clear vision for the problem and the target audience that will benefit from such an application	5
Version Control	Repository is set up with the necessary folders and the outline of a readme.md with headings for all the required information. .gitignore file must also be included.	5
Development Methodology	A development methodology is decided upon and the Github project is set up with epics and user stories	15
Communication Plan	The team has agreed upon a communication platform and time to meet	5
Meeting Plan	A meeting time and modality has been decided on with the TA	5
Use Case Diagram	The use case diagram specifies a few use cases	15
Wireframes	Wireframes for UIs are included	15
	Total	100

Previous

« Lab 8

Next

Project »

Overview	
Learning Objectives	
Part A	
Part B	
Project Overview	
Deliverables	
1. Team Number	
2. Team name	
3. Team Members	
4. Application Name	
5. Application Description	
6. Audience	
7. Vision Statement	
Examples	
8. Version Control	
9. Development Methodology	
10. Communication Plan	
11. Meeting Plan	
12. Use Case Diagram	
13. Wireframes	
Part C	
Submission Guidelines	
Rubric	