

To-Do List for myCourseVille
Group #20

Team Members

| | |
|--------------------------|------------|
| Sataporn Kruatejah | 6430368521 |
| Sarita Taveemunputtakarn | 6432181821 |
| Charoonroj Amornpativet | 6531306921 |
| Pupipat Singkhorn | 6532142421 |

2110221 Computer Engineering Essentials
Department of Computer Engineering
Faculty of Engineering Chulalongkorn University

Content

| | |
|-------------------------------|----|
| Introduction | 2 |
| To-Do List | 2 |
| To-Do List for myCourseVille | 3 |
| Instructions for use | 4 |
| Get start | 4 |
| Add your task | 5 |
| Priority of task | 6 |
| Edit/delte task | 6 |
| Type of task | 7 |
| Basic requirements | 8 |
| Single-page Web Application | 8 |
| myCourseVille login | 9 |
| Specific tasks related to MCV | 9 |
| Libraries/frameworks/plugins | 9 |
| Contribution of each member | 9 |
| Challenging requirements | 10 |
| GUI and Responsive | 10 |
| Web API | 11 |
| Implementation Plan | 12 |
| System Overview | 12 |
| Management Overview | 12 |
| Implementation Support | 14 |
| Summary | 15 |

Introduction

To-Do List

A to-do list is a comprehensive list of tasks that need to be completed. While it may seem like a simple concept, creating an effective to-do list can be a challenging task. Without a well-organized and structured system, your to-do list may not be as useful as you initially intended.

The benefit of To-Do lists:

1. Improved productivity: By organizing tasks and prioritizing them, a to-do list helps you stay focused on what needs to be done, which can increase productivity and reduce the likelihood of forgetting important tasks.
2. Reduced stress: When you have a lot of tasks to complete, it's easy to feel overwhelmed. A to-do list can help alleviate this stress by breaking down tasks into smaller, manageable steps.
3. Increased motivation: Crossing items off your to-do list can provide a sense of accomplishment and motivate you to continue working through your tasks.
4. Better time management: A to-do list can help you plan your time more effectively by allocating time to complete specific tasks.
5. Enhanced organization: A to-do list provides a clear overview of what needs to be done, allowing you to prioritize tasks and ensure that nothing falls through the cracks.

| Aa Name | Date | Status | |
|----------------------------------|---------------------------|--------|--|
| Exercise OPEN | May 1, 2023 → May 7, 2023 | Doing | |
| Final Project | May 3, 2023 | Done 🎉 | |
| Final Project: Report Submission | May 4, 2023 | Done 🎉 | |
| Take Fig on a walk | May 8, 2023 | To Do | |
| Final Exam | May 11, 2023 | To Do | |

Figure 1.1: Example of a to-do list from the Notion application.

To-Do List for myCourseVille

Typically, educational websites like myCourseVille provide alerts for assignments, but they do not allow users to add or remove them as per their convenience since they fetch data from different subjects.

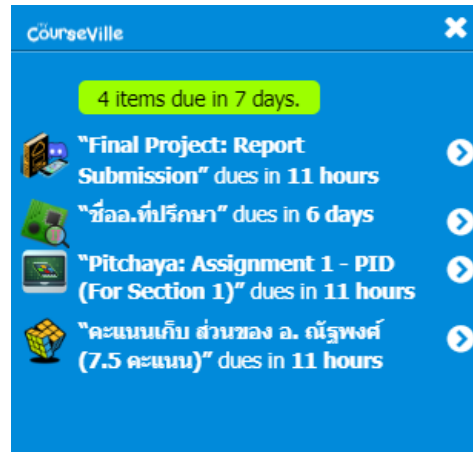


Figure 1.2: Example of alerts for assignments from the myCourseVille website, which includes notifications for upcoming assignment due dates.

Therefore, We have developed a feature that empowers users to add or remove tasks and events as per their requirements, providing them with an organized system to track their responsibilities. By offering the ability to prioritize tasks based on importance and urgency, To-Do List for myCourseVille assists users in effectively managing their time and breaking down complex projects into more manageable tasks. It's important to note that the To-Do List for myCourseVille displays priority based on user input and not MyCourseVille assignments, as it doesn't fetch data from MyCourseVille. Nevertheless, it streamlines scheduling and task completion for users, enhancing their productivity.

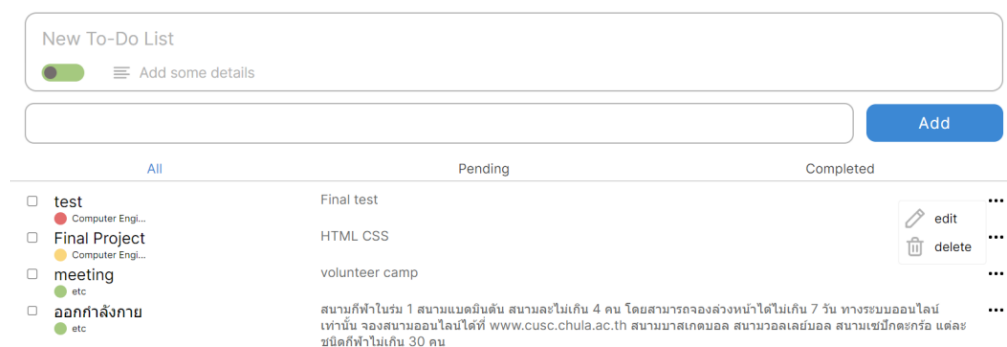


Figure 1.3: An example of our developed website, which includes sorted tasks by priority and add/remove function.

Instructions for use

URL links to your repository on GitHub: https://github.com/saimkrua/CEE_FINAL_PROJ_20

Get start

In order to use our to-do list website we created, please follow this link : 3.221.242.48 . After you enter our website, the first thing you need to do is click the “login” button on the top right of your screen.

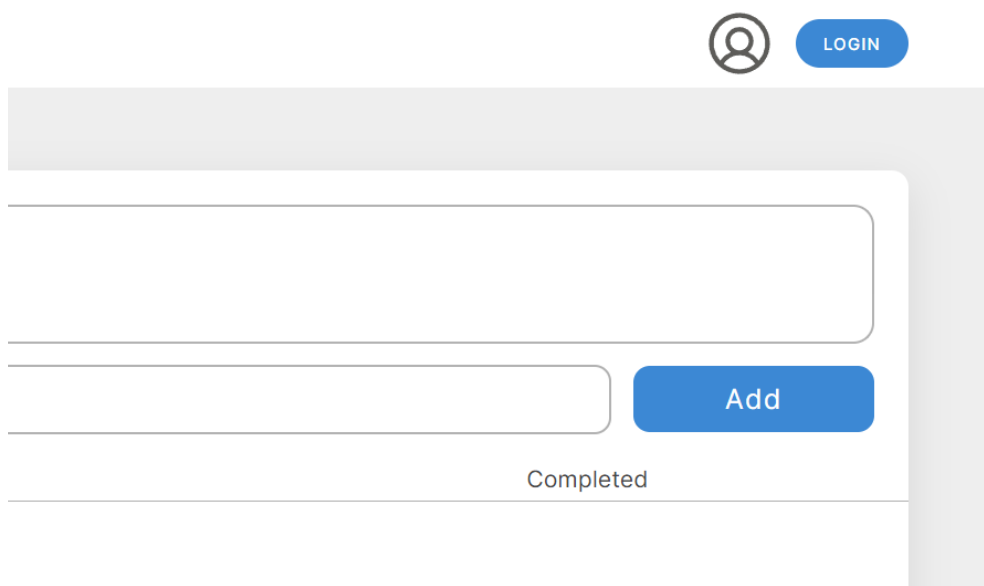


Figure 2.1: Example of top right of the website when you are not logging in.

It will redirect you to the MyCourseVilee login page.

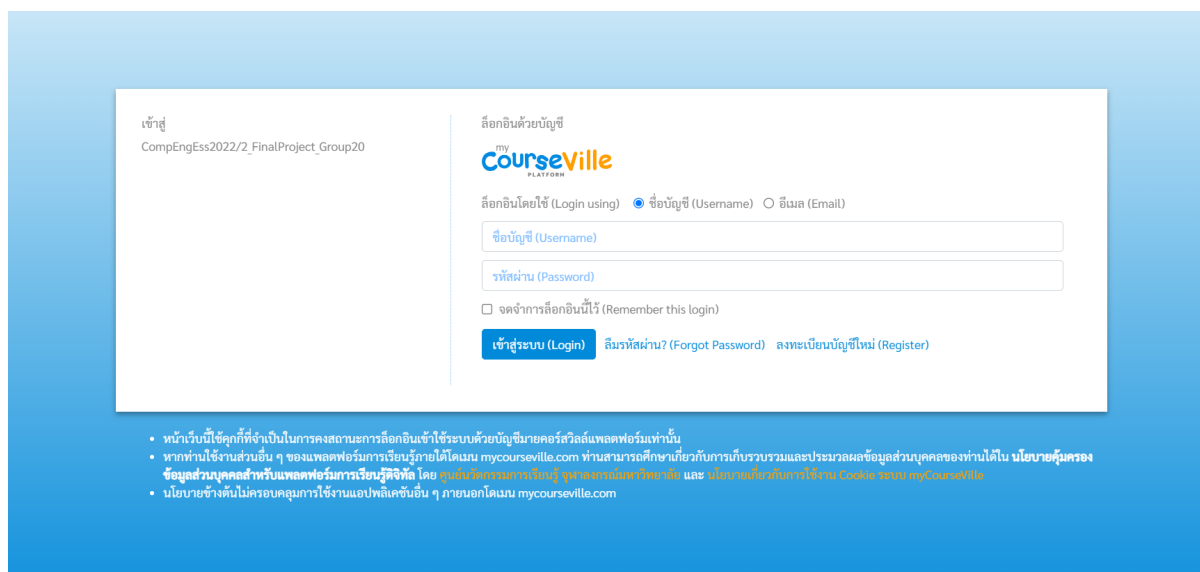


Figure 2.2: MyCourseVilee login page.

After finishing login, you will redirect to the same page with some changes on top right.

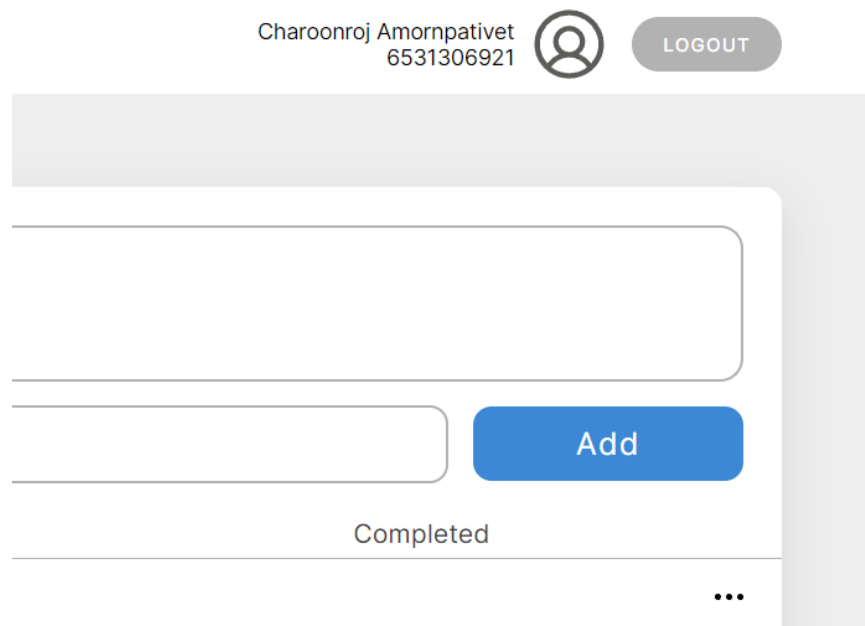


Figure 2.3: Example of top right of the website when you already logged in.

Add your task

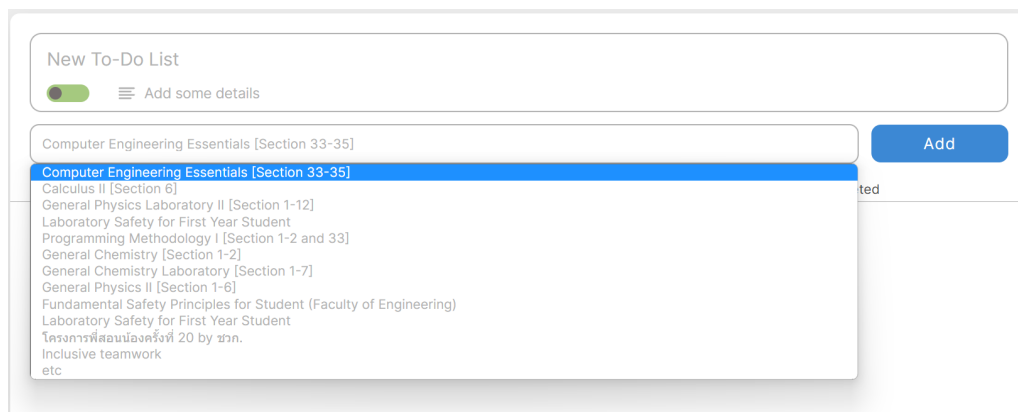


Figure 2.4: Adding task section.

To add a task you need to write your task name, details and select its course. The course you see is all the courses you have and if it is not in selection, you can choose etc and when you have already informed all you can click the add button and then your task will be shown below.

Priority of task



Figure 2.5: Example of how priority works.

Priority is a feature that can help you prioritize your tasks. Start from green to red. You can choose your task priority since before you add your task and if not your task priority will be the same as the previous task you have edited or added. Your to-do list below will show your task by showing high priority tasks before mid and low priority will be at the bottom.

Edit/delete task

All your tasks that you have added can be edited or deleted at any time you want. To access this function you will need to click on “three dot” you see at the right of your desired task.

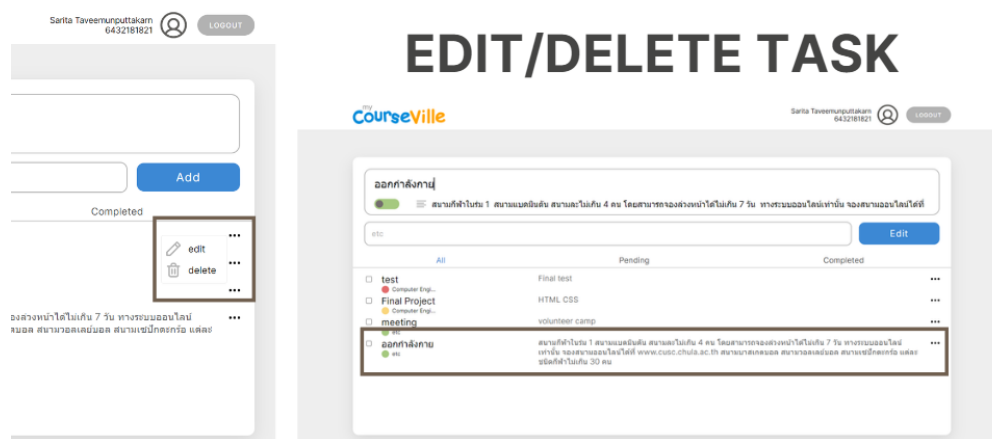


Figure 2.6: Example of how to edit or delete your task.

If you click delete, say goodbye to your task. Your task is gone. But if you click edit all information of that task will show at the same place as when you add your task. You can change everything about your task and when you have already finished, click the edit button and then your task will be updated.

Type of task

Your task also can be separate to 2 types. The default one is “pending”, but if you click square at the left of your task, that task will be crossed and changed to “complete” type. You can also use the filter to see all tasks, only pending tasks or only completed tasks as shown below.

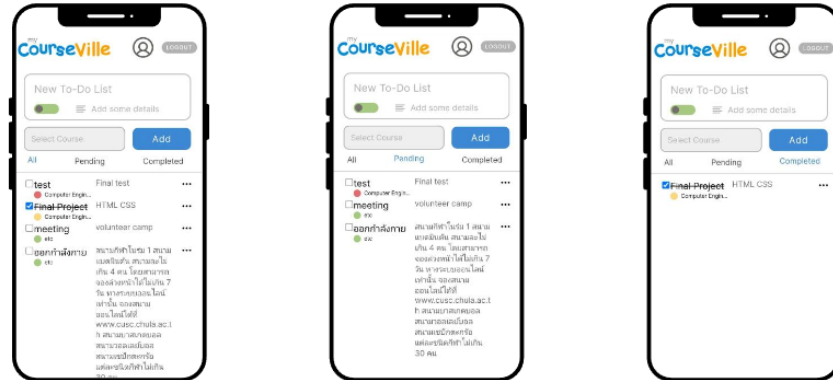


Figure 2.6: Example of task that will be shown in each filter.

Basic requirements

Single-page Web Application

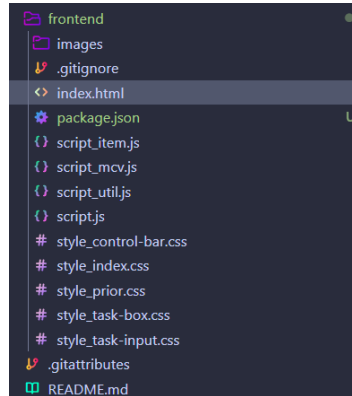


Figure 3.1: Front-end Project Explorer

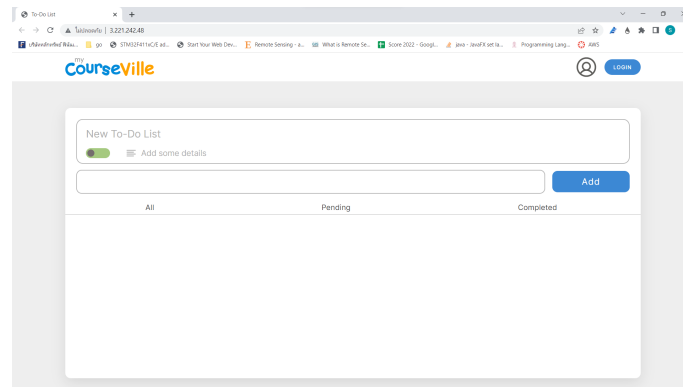


Figure 3.2: The web application when not logging in

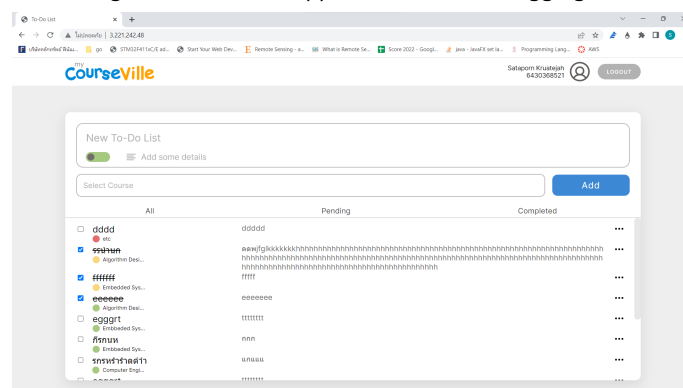


Figure 3.3: The web application when already logging in

Based on Figure 3.1, Shows that the front-end only has one HTML file, which is index.html. Furthermore, The URL remains the same whether the user is logged in or not, as depicted in Figure 3.2 and 3.3.

This implies that the front-end uses a consistent URL for both the logged-out and logged-in states. The differentiation between these states is likely handled within the frontend code, potentially through JavaScript, which dynamically adjusts the content or functionality based on the user's login status.

myCourseVille login

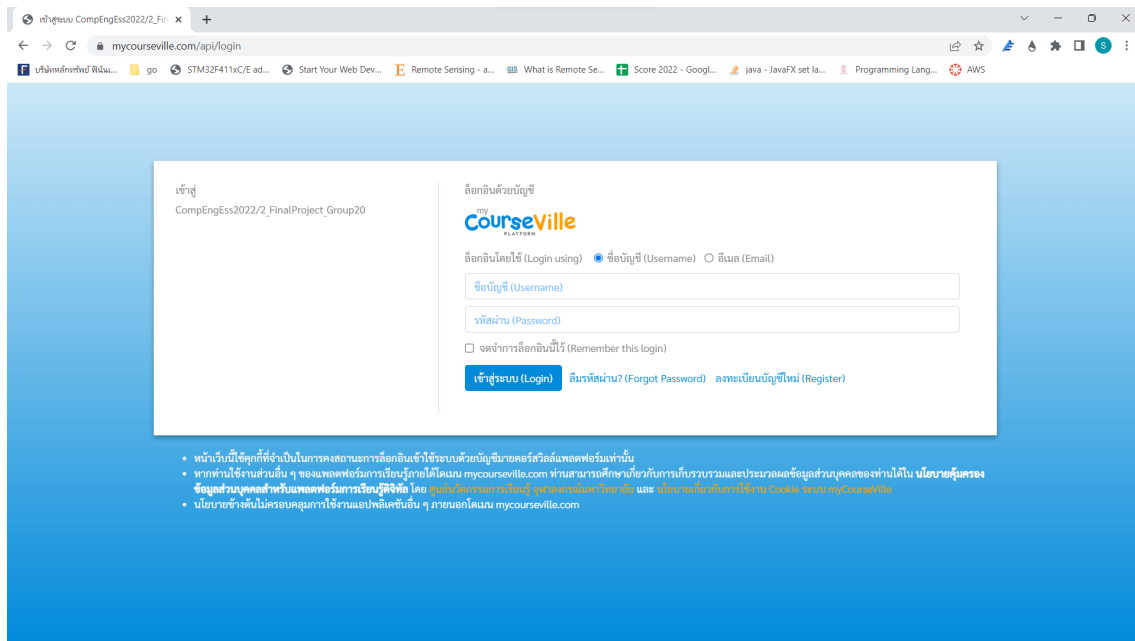


Figure: 3.4 myCourseVille login

Based on Figure 3.4, after clicking the login button (as shown in Figure 3.2), it will redirect to the MyCourseVille login page.

Specific tasks related to MCV

As mentioned in the previous section (Instructions for use), this application has the capability to perform various actions with tasks obtained from MyCourseVille (MCV) users. These actions include adding, editing, deleting, and setting the status of tasks. Additionally, the application has the functionality to sort the tasks on the to-do list based on their priority.

By leveraging these features, users can effectively manage their tasks and prioritize them according to their specific needs. The ability to interact with tasks in such a versatile manner enhances the overall usability and efficiency of the web application.

Libraries/frameworks/plugins

From the URL links provided to the project repository on GitHub in the previous section (Instructions for use), it can be observed that no libraries, frameworks, or plugins are imported.

Contribution of each member

The contribution of the work will be discussed in the implementation section.

Challenging requirements

GUI and Responsive

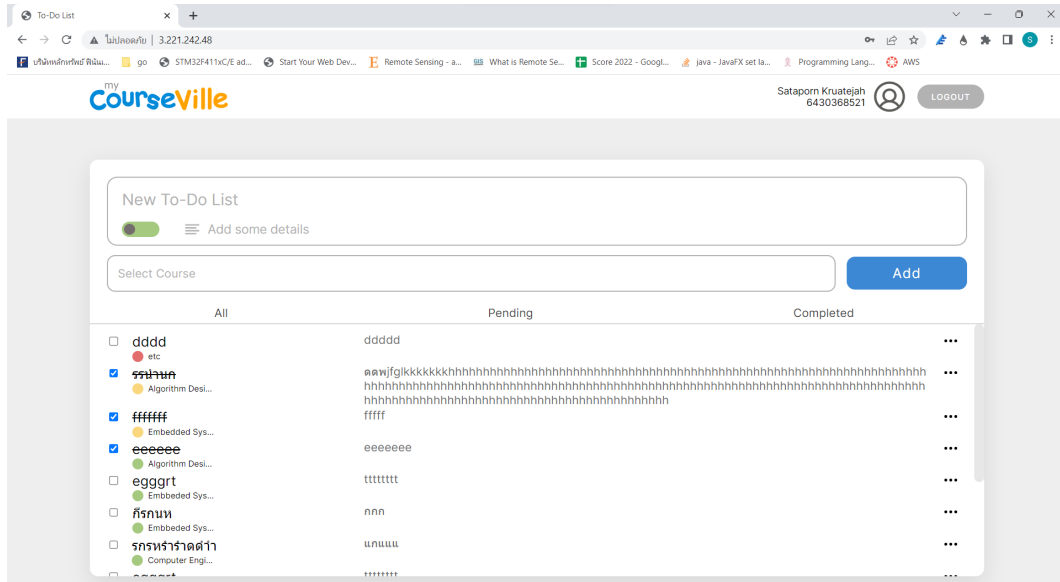


Figure 4.1: The web application for pc

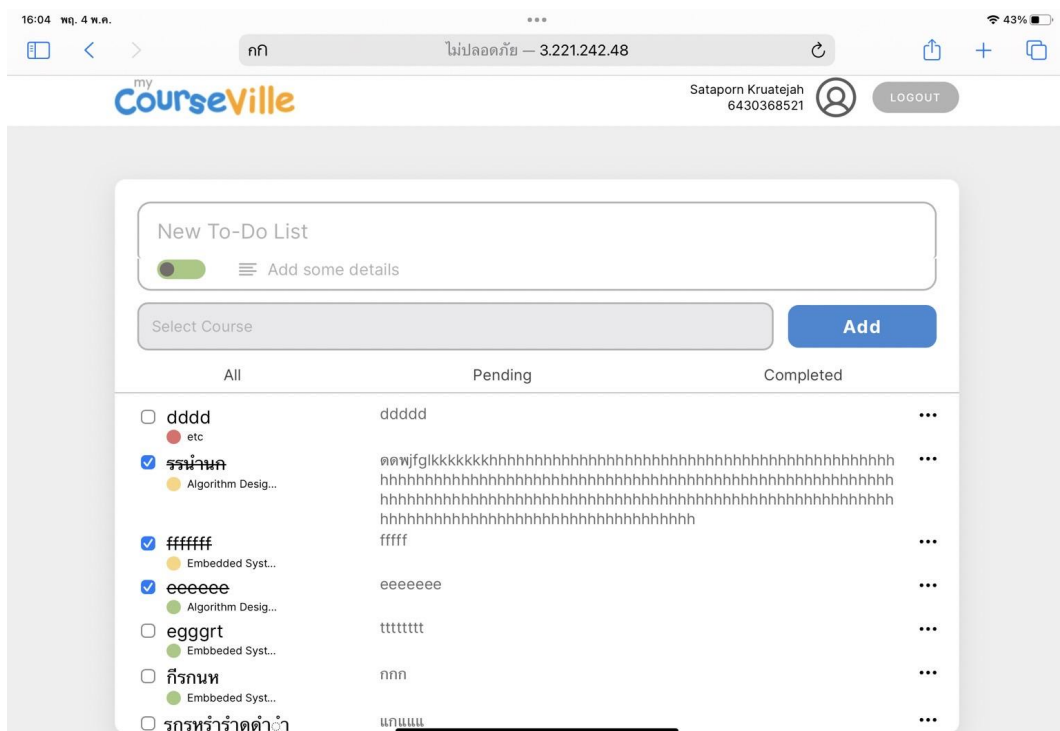


Figure 4.2: The web application for iPad

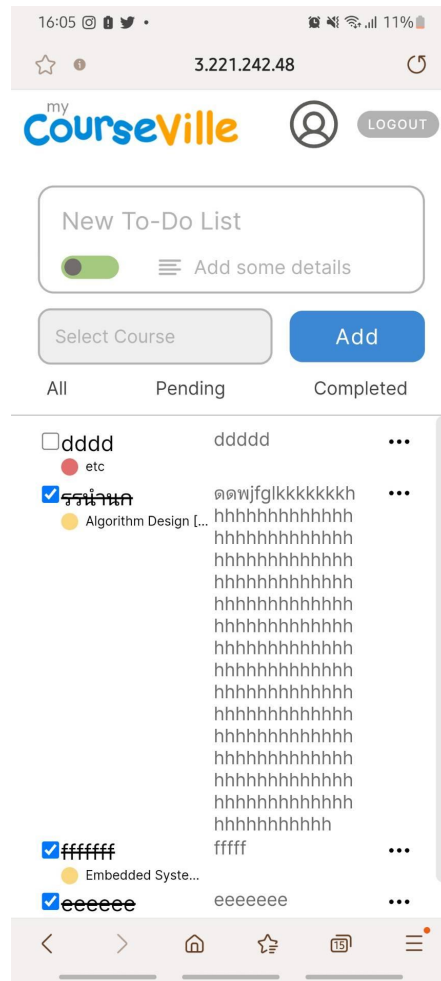


Figure 4.3: The web application for phone

We have put great effort into designing the user interface to embody a clean and modern style. Our objective was to create an interface that is visually pleasing and enhances the overall user experience.

Furthermore, we have ensured that the application seamlessly adapts and interacts with users across various screen sizes and orientations. Whether accessed on a desktop computer, laptop, tablet, or mobile phone, the application effortlessly adjusts to provide an optimal user experience.

Illustrated in Figures 4.1, 4.2, and 4.3, the application exemplifies its consistent and engaging design, maintaining its modern aesthetics across different devices and orientations.

Web API

To implement each feature, we have utilized various web APIs, particularly HTML DOM elements, to enhance the user experience and user interface. These elements play a crucial role in handling events within our web application.

Through the effective utilization of these web APIs, we have been able to enhance the overall functionality and interactivity of our web application.

Implementation Plan

System Overview

System Description

In order to create a to-do list single-page web application according to the purpose of the application, we need to have the following features:

1. Login and Logout using myCourseVille login
2. Add Task
3. Arrange in order of priority
4. Edit/Delete Task
5. Categorize types of tasks

Assumptions and Constraints

We have a total of 13 days, starting from April 19th to May 2nd, to work on the project. The deployment deadline is before May 3rd, which is the day of the presentation.

Management Overview

Description of Implementation

To implement a web app, divide the work into 6 phases.

Phase 1: Draft 1

Phase 2: Draft 2

Phase 3: Draft 3

Phase 4: Responsive Design

Phase 5: Test and Deploy

Phase 6: Presentation

In phases 1, 2, and 3, which involve building the web app, there are three main parts: design, frontend, and backend. In phase 4, make the web app responsive to different devices. In phase 5, test the web app we have built. Once it passes the test, we will deploy it. In phase 6, the final phase, the presentation.

Major Tasks

1. Meeting and Planning
2. Setup
3. Design
4. Frontend
 - 4.1. Structure
 - 4.2. Styling
5. Backend
 - 5.1. Function
 - 5.2. Database
6. Test
7. Deploy
8. Presentation Slide
9. Presentation

Roles and Responsibilities

Design (*2 people, from frontend and backend*)

- Sataporn Kruatejah
- Pupipat Singkhorn

responsibility: Designing web page layout and selecting colors.

Frontend (*2 people*)

- Sarita Taveemunputtakarn
- Sataporn Kruatejah
- Pupipat Singkhorn

responsibility: Creating the layout and styling of a web page.

Backend (*2 people*)

- Sataporn Kruatejah
- Charoonroj Amornpativet

responsibility: Setup, Deploy, Build function, manage and connect databases.

Presentation (*all people*)

- Sarita Taveemunputtakarn
- Sataporn Kruatejah
- Pupipat Singkhorn
- Charoonroj Amornpativet

responsibility: Make presentation slides and presents.

Implementation Schedule

Phase 1: Day 1 -Day 3

Phase 2: Day 4 - Day 9

Phase 3: Day 10 - Day 12

Phase 4: Day 12

Phase 5: Day 12

Phase 6: Day 13

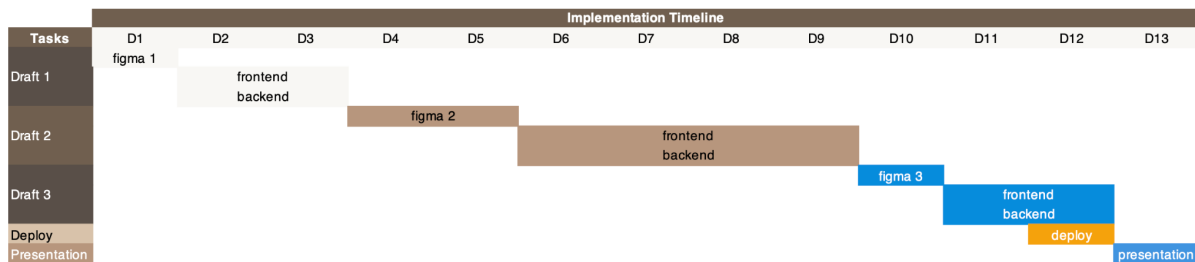


Figure 5.1 Implementation Timeline

Implementation Support

Hardware

- Computer: Individual

Software

Design

- myCourseVille
- Figma

Code

- Visual Studio Code
- HTML
- CSS
- JavaScript
- Node.js
- AWS
- Amazon DynamoDB

Communication

- GitHub
- Discord

Presentation

- Canva
- Zoom

Summary

A to-do list is a comprehensive inventory of tasks that require completion. It serves to help individuals manage their time, increase productivity, and prevent them from overlooking important tasks or activities.

The to-do list for myCourseVille enables users to add or modify their tasks on the myCourseVille platform. Additionally, it allows users to prioritize their tasks based on their level of importance, and the to-do list for myCourseVille sorts the tasks according to the user-defined priorities, enhancing the efficient execution of tasks. The to-do list for myCourseVille also categorizes tasks according to their status, such as all tasks, pending tasks, and completed tasks. This website exhibits an aesthetically pleasing design and user-friendly interface, enabling access to the platform on various devices, including mobile phones, laptops, tablets, and computers.

The project was developed as a single-page web application, using the API from myCourseVille without using any libraries other than those used in the in-class activity. The project is divided into four distinct parts: design, front-end, back-end, and presentation. To ensure optimal collaboration and information sharing among team members, the team chose to leverage Github for efficient file sharing and Discord for streamlined communication.