Project Overview

This web-based Daily Planner application allows users to manage a schedule of daily tasks. Users can add new tasks, view details, edit existing tasks, and delete them. The tasks are displayed in order based on their scheduled time.

The application is built with HTML, CSS, JavaScript, and jQuery, and demonstrates full CRUD functionality: Create, Read, Update, and Delete. All task data is created within an external `.js` file and dynamically manipulated through jQuery

How It Works

First users can create a task by inserting a title, time, and description into the input fields and clicking “Add Task”. This task is then added to the array and rendered to the page.

```
function handleAddTask() {

  if (!title || !time || !description) return;

  tasks.push({
    id: taskIdCounter++,
    title,
    time,
    description
  });

  renderTasks();
```

Next, all of the tasks are displayed in a list. This list is sorted by time using Date. Each task has a view button associated with it which allows you to see its description.

```
function toggleDetails() {
  $(this).closest('li').find('.details').slideToggle();
}
```

For this to work I had to put a placeholder date so that it can properly sort all of the tasks. This is why I have the date as 2025/01/01. Because of this I can sort the tasks by time when tasks get added or updated.

```
function sortTasksByTime() {
  tasks.sort((a, b) => {
    const dateA = new Date(`2025/01/01 ${a.time}`);
    const dateB = new Date(`2025/01/01 ${b.time}`);
    return dateA - dateB;
  });
}
```

Users can also click the “Edit” button which will display the input fields below the task. After editing the save button can be clicked. This updates the task and renders the list.

```
function handleSaveEdit() {


  task.title = newTitle;
  task.time = newTime;
  task.description = newDesc;


  renderTasks();
```

Finally, users can click the Delete button and the task will be deleted from the array, refreshing the list.

```
function handleDeleteTask() {
  const id = parseInt($(this).closest('li').data('id'));
  tasks = tasks.filter(task => task.id !== id);
  renderTasks();
}
```

Repository Link
https://github.com/AidanL9895/FinalCrudProject

Pages Link

https://aidanl9895.github.io/FinalCrudProject/