

# Final Sprint – Spring 2025

React, HTML, CSS, and JavaScript.

Hello all, and welcome to the final sprint. Please note following about the final sprint:

There are **two parts** of this final sprint

- A. First a group project (that is ideally done by a team of three people)
- B. An individual project that each student must submit independently.  
*(Refer to page 5 of this document)* – Although this individual project will not be graded, it is still mandatory. You may submit it along with the group project or at a later time, as it will not impact your final grade.

For team preferences, please enter your team information on the excel sheet provided [By Vanessa] **ASAP (if not done yet)**.

For the group part of this sprint, you have two options: firstly, you can work on a project of your own design if you'd like, or alternatively you can implement the project that we've designed for you.

If you'd like to work on your own idea, please send a written proposal for the project to Noman in a private message. We will review your proposal to help you scope the project and make sure that you don't take on too much given the time that we have available.

Replicating popular apps can be a great idea, especially if one catches your interest or aligns well with your team's preferences

## Project Requirements

1. Have a Figma design done up for your project following the design principles that Levin taught you in UI/UX. (Generally speaking – a reasonable user experience, with some thought given to the aesthetics of the page) – need to go to Levin once again!

2. Implement the project in React
3. Use proper Semantic HTML tags where applicable
4. Write proper comments in your code for clear representation of its purpose
5. CSS and React code that approximates the design in the Figma mockup.
6. Must use:
  - a. React Router (A minimum of 4 components/pages)
  - b. Units test for a minimum of 3 components
  - c. A mock Json server that serves the endpoints for UI testing
  - d. A context file to serve the context.
7. Upload the finished project to GitHub.

Beyond that, the world is your oyster.

Be creative and do what you think would be fun and interesting.

If your team can't think of a project, I have a website here with a few project ideas.

<https://www.freecodecamp.org/news/5-react-projects-you-need-in-your-portfolio/>

Some of those projects are a little over-sscoped for us, so if you want to choose one from that site, let me know what features you intend to implement ahead of time so I can help you from getting in over your head.

Replications of popular apps are always good as well if one strikes the fancy of you and your team.

## Default Project

If you'd prefer not to think of your own project, no worries, we have one designed for you here. [Some of you have already chosen similar projects in Sprint 1, so make sure to present this in a more advanced and refined form]

**[The following should serve as a reference for minimum requirements should you opt to work on your own idea]**

Develop an **E-Commerce Web Application** using **React**. The application provides users with essential e-commerce functionalities such as product listing, viewing product details, adding items to a shopping cart, and checking out. It leverages a mock server or fake API for simulating backend interactions to ensure a seamless development and testing process.

## Features

### 1. Product Listing Page:

- Displays a grid or list of products fetched from a mock API.
- Each product includes an image, name, price, and a "View Details" or "Add to Cart" button.

### 2. Product Details Page:

- Provides detailed information about a selected product, including:
  - Larger product image
  - Description
  - Price
  - Available quantity
- Option to add the product to the cart.

### 3. Shopping Cart:

- Displays a list of all products added to the cart.
- Shows the total price of the items in the cart.
- Allows users to remove products or update their quantities.

### 4. Shopping Cart Context:

- To hold the context and values

### 5. Checkout Page:

- Presents a summary of the cart items.
- Option to proceed with payment or simulate the checkout process.

- Simple confirmation message upon checkout.

## Mock API

A mock server using **JSON Server** is implemented to handle data interactions. This fake API simulates the following endpoints:

- GET /products - Fetches the list of products.
- GET /products/:id - Fetches details for a specific product.
- POST /cart - Adds a product to the cart.
- GET /cart - Retrieves items in the cart.
- DELETE /cart/:id - Removes a product from the cart.

In addition to above files, you would like to have a tests folder somewhere in your submission that has at least 3 unit tests for any of the given components.

## Submission:

1. Submit only the github link to your project, with following two extra objects:
  - a. Video: include a “ONE MINUTE” video, just showing the demo run of your project – no need show or explain the code. (I will do it from github 😊 )
  - b. For this group project, a brief outline of the contributions made by each group member. Don’t include micro information, just a brief representation of everyone’s effort.

## Individual Project – (may or may not be a React project)

In addition to the main project, which is (optionally) group-based, all students **must** make a portfolio website for themselves that showcases your projects and skills. This website should, at a minimum, feature some dynamic JavaScript (for example, DOM manipulation, or even something more involved, like an interactive widget - one student made a little game that was embedded in their site once to fulfil this requirement – pretty much anything will work, if you're not sure, ask Vanessa – , feature good use of semantic HTML tags, and should approximate a Figma design that must also be submitted. Most students take this portfolio as a base and grow it over time throughout the rest of their studies, so try to make it as stylish as you can, it can act as sort of a living resume for you. Many people will submit just a GitHub profile when they're applying for jobs, but with a portfolio page, you can showcase the code on your GitHub profile – you can take pictures or videos that show it running in optimal, controlled conditions, and you can tell a story behind each project, painting your work in the best possible light. It's really a curated experience of your work for employers. If you want some examples of portfolio websites, see:

<https://fourandthree.com/> (Levin's Website)

<https://www.awwwards.com/websites/portfolio/> (A collection of examples.)

## Submission:

1. Submit your individual github link to your project, with following:
  - a. Video: include a “ONE MINUTE” video, just showing the demo run of your project – no need show or explain the code. (I will do it from github 😊 )
  - b. For the group project, a brief outline of the contributions made by each group member. Don’t include micro information, just a brief representation of everyone’s effort