Final Sprint

React, HTML, CSS, and JavaScript.

Hello all, and welcome to the final sprint. For this sprint, you can work in teams of up to three people. For team preferences, please let Nicholas know as soon as possible who your preferred teammates are – if you’d prefer to be placed in a random team of three, you can also let us know, and we’ll do that for you as well. If we don’t hear from you, we’ll assume you want to do the sprint individually. **Please let us know by the end of the day on Monday at the latest.** For 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 both Nicholas and Roy in a group 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. Note there is also an individual component of this project described at the end of this document that all students must complete on their own.

# 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)
2. Implement the project in React
3. Use proper Semantic HTML tags where applicable
4. Comment your code for clear representation of its purpose
5. CSS and React code that approximates the design in the Figma mockup.
6. Must use React Router (A minimum of 3 pages.)
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-scoped 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. For this project, we're going to make a simple social media webapp. The design can follow inspiration from any social media app of your choosing as many of them follow the same philosophies (you may find inspiration looking at Facebook, Twitter, Instagram, reddit, etc). The canonical version of this app has at least three screens, described in the paragraphs below.

Firstly, it is going to have a sign-in splash screen that opens when the app first opens. All this screen will do is prompt the user for a name and password to login and upon doing so be taken to their feed. You should include dummy data for an account you can use to login (this can easily be hard coded in and is mostly for testing/validation purposes of the login feature). There should also be a sign-up page which the user can enter their name and password, and then be taken to **their** feed - the feed page should reflect at least one piece of user data for the created account (for example their name/username that they entered in the signup) - you can accomplish this locally with either useState or localStorage – no need to create a real backend. For simplicity’s sake the actual feeds themselves between logging in/signing up aren’t required to be different (Though large brownie points to you if you can pull it off!).  
  
The feed will display all the relevant content to the user. Again, think Facebook, Twitter, or Instagram in which you can easily scroll through content. This content should contain mock data for users, profile pictures, likes, text content and “uploaded” images. Lastly the user should be able to like other posts or make a post themselves which will then be added to the feed.

Again, this is just one project idea, if it doesn't appeal to you, Nicholas and Roy are happy to help your team brainstorm ideas.

As before, a Figma design is required with submission. If you come to Roy and Nicholas early with your Figma designs, we can give you advice and make sure you're on the right track.

# Individual Project

In addition to the main project with 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, a modal, a lightbox, 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 Roy and Nicholas – , 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 actually 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.)