Project 02 – Website with Search Engine

Software Engineering Bootcamp – Preethi Sridhar

Some ideas I have:

**1. A website with football stats from each team – you can search the team up and look for the stats, players stats, if the game is happening right now, etc.** – I like this one!

2. Search movies – find summaries and their ratings

3. Weather website – search cities and obtain the weather.

To make a website that allows a user to:

* Enter and submit and keyword
* Make requests to the API
* Recieves a response
* Displays images from the response
* Uses custom CSS grid to display the images
* Works well on both desktop and mobile devices

Find an API for this sports website!

PLANNING

1. As a user, I want to search for an NFL player so I can see their stats

2. As a user, I want to search for an NFL team so I can view their performance

3. As a user, I want to see a clean and responsive grid so that it looks good on both desktop and mobile devices.

WIREFRAMES

Will be included on Sunday 9.7.25

APPLICATION PROGRAMMING INTERFACE/API chosen

**Workflow Requirements**

This project requires a bit more work with APIs than was directly covered in the material.  You are encouraged to watch the following video that walks through the process of using jQuery/Ajax to request data from and API, and display images on screen.

[Helpful, relevant video walk-through](https://www.youtube.com/watch?v=qevTMc4kVxM)

The following requirements are related to how you go about building your project

The planning phase should be completed prior to beginning the development phase and/or touching any code.

Wireframes can be done on paper or using any number of widely available applications. A free one that may be useful is [draw.io](https://www.draw.io/).

**Planning phase**

1) Create at least 3 user stories  
2) Create wireframes for desktop and mobile views

**Development phase**

3) Create a GitHub repository on Github.com (before you start coding)  
4) Clone it to your local machine (before you start coding)  
5) You will need to [obtain an API key through Giphy](https://developers.giphy.com/)  
6) Review the [documentation for the "Search Endpoint"](https://developers.giphy.com/docs/)  
7) Make frequent commits throughout your development that are descriptive, such as "adds todos reducer" (throughout development/coding process)

**Technical Requirements**

The following requirements are related to what your code should contain

1) Your site only needs to contain one HTML page, but there should still be multiple links in your menu (even if they don't link to other pages)  
2) There should be an input field (with a type of search) & a submit button  
3) A user should be able to type in a search phrase, click submit, and your site should query the Giphy API based on the search expression that your users enter  
4) Iterate through the returned data, and for each returned object in the array, find an image in the returned JSON and append that image to the screen  
5) Your project should contain three files: ***index.html***, ***style.scss***, ***style.css*** and ***main.js*** and your project directory structure should look like picture below

**Design**

6) Your styling should all be done in your style.scss file (SASS), and you should use a CSS Preprocessor to watch that file for changes and output it to your ***style.css*** file. (Optional – for extra points. A solution with a simple CSS file is also accepted)

**Site-wide**  
7) Choose a Google Font that you've never heard of before and use that to style your site title  
[Here is a helpful video!](https://www.youtube.com/watch?v=9ksLij2oMe4)  
  
**Desktop**

8) Use flex to ensure your site's name and nav bar appear aligned to the sides of the header (like in the screenshot below)  
9) Use flex; to ensure your input field and submit button appear side by side  
10) Create your own custom grid classes in css, that make use of the flex property, and leverage these in your code so that the images appear in rows and columns  
  
Mobile (320px and below)

Use a @media query to ensure that:

11) Your site title and navbar stack vertically  
12) Your navigation items stack vertically  
13) Your images stack vertically in a single column