

Web Dev 2 - Mid Term - Weather App

Welcome!

This challenge will help you improve your coding skills by building realistic projects.

The Challenge

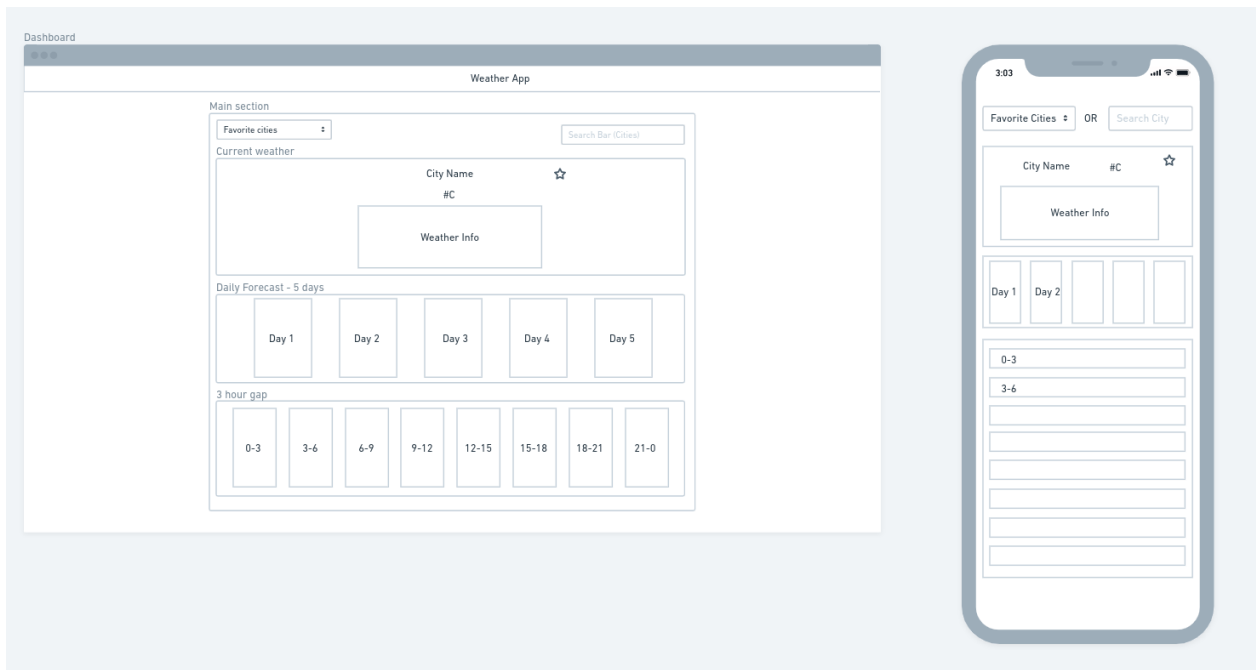
Your challenge is to integrate with the [Open Weather API](#) to pull weather data and display it.

Your users should be able to:

- Search for a city using an input field.
- Save favorite cities by clicking on a star icon.
- See the current weather info based on the selected city.
- See forecast for the next 5 days.
- See weather info for every 3 hours of each day.
- Have good experience using desktop or mobile screen sizes.

Wireframe

The wireframe can be accessed with this [link](#) . You will need to use your best judgment for styles such as font size, font family, padding, margin, color pattern, etc. and decide how you want to display the information on the screen.



Requirements

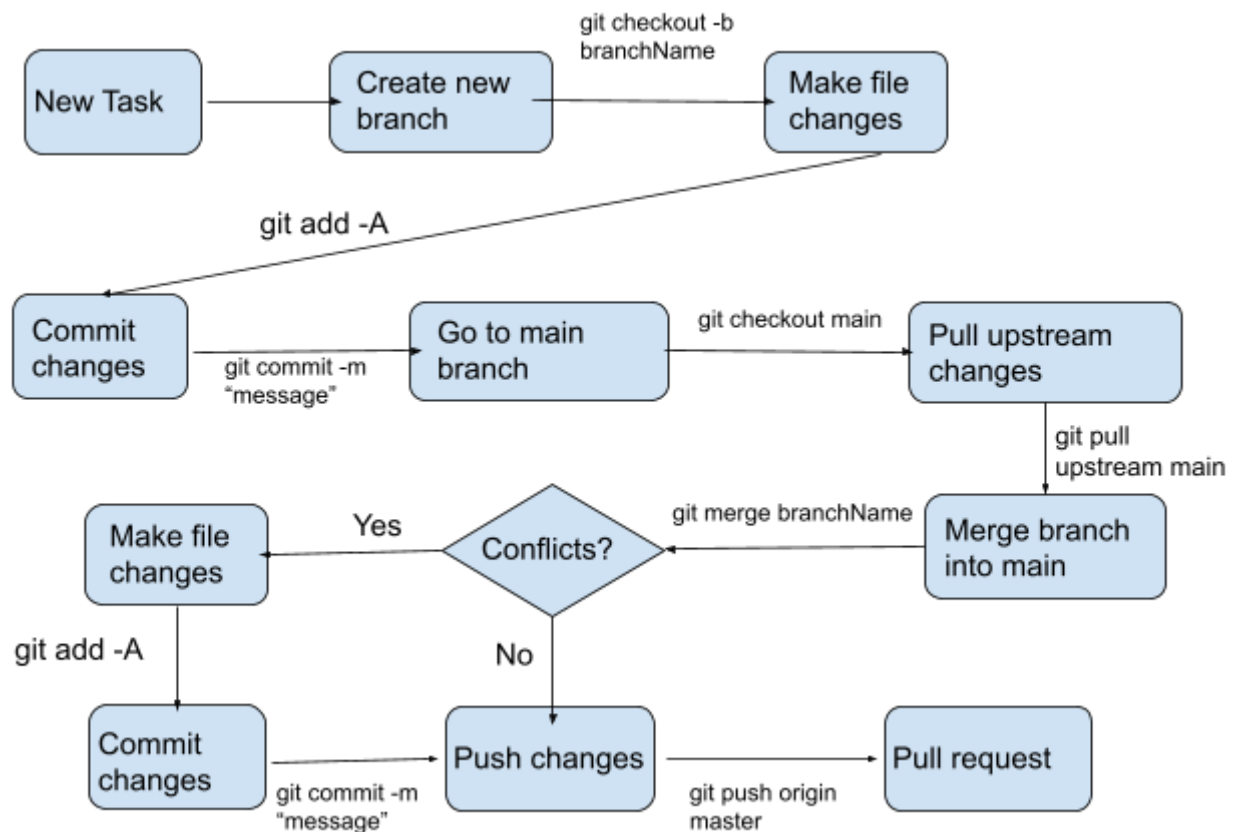
These are the requirements that are mandatory for this project:

1. Follow [wireframe](#) to create the structure of your page for desktop and mobile. It is up to you how to design, include images and icons in the project.
2. Search input should use [Google Places Autocomplete](#) for suggesting cities **only**.
3. The default city should be based on users' location. In case the user blocks the location info, Vancouver should be the default.
4. When the user clicks on the star icon, the city should be included on the dropdown for "Favorite cities".
5. Favorite cities should be stored in [localStorage](#). After the page loads, check if there is something saved there. If yes, append city options in the dropdown.
6. When the user clicks on one of the options for dropdown, the weather info for the selected city should be loaded on the page.
7. Create Fetch API calls to get the [current weather](#) and [forecast for the next 5 days](#).
8. When the user clicks on one of the cards in the "Daily forecast" section, the "3 hour range" section should change the data based on the day clicked.
9. SCSS files should be used for styling files.
10. Media queries should be used for responsive behaviors in the page.
11. Desktop size: 1440px / Mobile: 375px
12. Slides for your presentation are required.

Workflow

- The captain should create the repository and commit the base files for the project (HTML, SCSS and JS files). At this time, font family, color pattern and relative unit should be set and included in the SCSS file.
- After that, the team should clone the captain's repository and split the tasks for each team member. It is recommended to follow the division below:
 - If 3 members:
 - Member 1 : Search input + Favorite cities dropdown + Current weather (most confident with JS)
 - Member 2 : Forecast 5 days
 - Member 3 : 3 hour range
- Focus on the functionalities first.
- After all the features are implemented, work on styling for the desktop version.
- Leave responsiveness as the last task. Remember to use media queries and only make necessary changes on CSS properties. No need to redo the whole design.
- Use the requirements on the section above as a checklist. Make sure you completed all the tasks.
- After the project is completed, create a presentation to show your classmates what you have done, how you split the work, what you learned in the process and what was the most difficult part of the project for you.

Git control (highly recommended to follow)



Hints

Here are a few tips that can help you to avoid some issues during the development of the project:

- Before starting coding, meet with your team and set naming patterns for html/css content and javascript functions to avoid conflicts during merge.
- Divide your code in multiple JavaScript and SCSS files, one for each section of the page and one for functions that could be used in different files. For example, API calls should be called in the main file and the data can be exported to other files (the forecast data will be used in "Daily forecast" and "3 hour range" sections).
- A lot of DOM manipulation will be needed in this project, so review how to use event listeners and how to create/modify/delete content of the page using JavaScript.
- Don't stay stuck for too long in one problem that you cannot solve. First google it, then ask your teammate for help, then ask the instructor for help. Spend 1 hour max to solve the problem or ask for help.
- APIs:
 - Current Weather:
`https://api.openweathermap.org/data/2.5/weather?lat={lat}&lon={lon}&appid={API key}`
 - 5 days:
`https://api.openweathermap.org/data/2.5/forecast/daily?lat={lat}&lon={lon}&cnt={cnt}&appid={API key}`

- 3hr:
`https://api.openweathermap.org/data/2.5/forecast?lat={lat}&lon={lon}&appid={API key}`

Deploy and Submission

Every member of the group should deploy your work in your own GitHub Pages.

Submission must be done in Google Classroom.