

OVERVIEW

write a simple, responsive, web app in angular/react that shows the weather of some city. The user should be able to search for a city and save it to favorites (locally, a server is not required).

We expect to see 2 pages in this app. (weather page and favorites page).

We recommend you use a 3rd party UI library (material, prime, bootstrap, etc.)

API

The API that you'll use for this app is [AccuWeather API](#). Please sign up and create a new app to get an API key.

You will use 3 endpoints: [location autocomplete](#) for the search field, [get current weather](#) and [5-day daily forecast](#). Please read those API docs.

NOTE: this API is limited to 50 requests per day. our recommendation is to save the responses and use them locally during development.

GOALS

1. Show us your coding skills - We want to see your strengths, don't be afraid to go wild, add features and customize the design.
2. Readability - Your code should be readable and self-explanatory with minimum comments. Remove any unused code, console logs, and files. Use logical project structure and code separation.

HOW TO SUBMIT

1. Push your code to your git repository and make it public (we recommend GitHub to use GitHub Pages with ease), **name the repo as firstName-lastName-dateOfStart (John-Doe-01-12-2018)**.
2. Deploy the compiled app to your server or GitHub pages/Heroku/etc - **MUST**

Send us the git repository link and a link to the deployed app (2 links)

SPECS

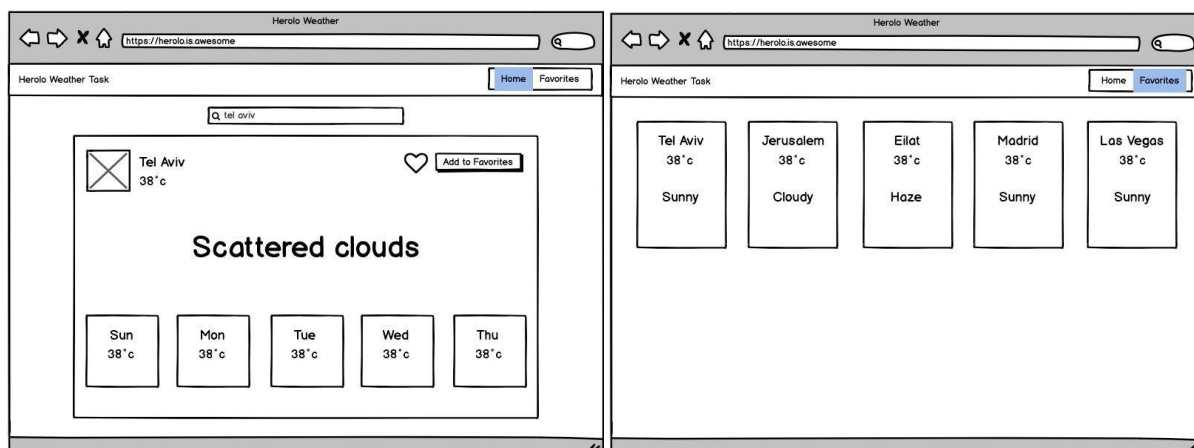
1. Create a header with navigation icons/links/buttons for main and favorites screen.
2. The main screen (weather details) will be composed of a search field to search a location's weather by city name. And below it, the current weather and a 5-day forecast of the searched location. A location should have an indication if it's already saved in favorites, and a button to add/remove from favorites (it can be the same button).
3. Display Tel Aviv weather by default.
4. Favorites screen will be composed of a list of favorite locations. Each location should have an ID, name, and current weather. Clicking on a favorite will navigate to the mainscreen showing the details of that location.
5. Searching should be done in English letters only
6. State management is a must!
7. Responsive design is a must! (flexbox/grid will give you extra points 😊).
8. Error handling is a must! (Can be done with toast, modal).

BONUSES

1. Set the default location by using the [Geolocation API](#). you will need another API endpoint for this: [get location key by lat/lon](#).
2. Add dark/light theme support (add toggle button in the header).
3. Add Celsius/Fahrenheit toggle button.
4. Add animations (with good taste).

EXAMPLE

We've created a UI mockup, so you'll have a basic idea of what we expect to see.



Good luck,