UP⁴²

UP42 front-end coding challenge

We are excited that you made it to the coding challenge stage, and we are looking forward to reviewing your solution.

For this challenge, we want you to build a simple application capable of fetching and visualizing remote data.
The goal is to create a weather application displaying the weather for a single day.

The data

We will use *OpenWeatherMap 5 days/3 hours forecastas* as the remote data source for this challenge.
More information on the API endpoint can be found on the [OpenWeatherMap documentation](#).

Please use the following endpoint to retrieve the data:

```
https://samples.openweathermap.org/data/2.5/forecast?q=M%C3%BCnchen,DE&appid=b6907d289e10d714a6e88b30761fae22
```

Note: There is no need to create an account.

The design

Our designers have been creating a prototype that you should use as a reference. Please try to keep as close as possible to the design specifications.

- Please use the following prototype to implement the look and feel: [Figma prototype](#).
- The font family used is Roboto regular (400) and bold (700), which you can download from [Google Fonts](#)
- You can download the sun icon in SVG format from this [GitHub gist](#)
- You can access more information on spacing and sizing used in the prototype by creating an account on [Figma](#) and opening [the prototype source](#) *(optional)*

Details

- Your solution should be presented inside a single page application for the web
- We should be able to run your solution from any modern browser
- Ideally, your solution works on small and large screens
- Consider using the README to document your project, thoughts, and conscious decisions

Requirements and submission guidelines

Please make sure to meet the following requirements for your submission:

- You can use libraries, frameworks and starter projects as you see fit (ideally React or Vue.js)
- Code and commit messages should be treated as you would on a real-world task
- Please take some time to think about code quality and testing, and demonstrate your approach to these aspects
- Provide a README with instructions on how to set up, run and test the application
- When submitting, please provide a link to an online Git repository or a ZIP file which includes the local `.git` folder
- Try not to include the word *"UP42"* in the title, the description, or anywhere in your codebase

We estimate that this challenge should take a maximum of 4 hours to complete, but this is not a hard limit. Feel free to take the time you need to ensure that it is reflective of your skills.

Challenge assessment

We assess the challenge according to the following points:

- Clean code and general code structure (e.g. code and style linting, clear file structure)
- Code quality (e.g. separation of concerns, clearly defined UI components, helpers/utilities, etc.)
- Data management (global state, error state and loading state)
- Test coverage
- Use of git and `readme.md`
- CORS solution implementation
- Semantic markup and scalable use of CSS (e.g. scoped scss with BEM, css modules, CSS-in-JS, etc.)
- Design implementation

We hope that you will find this challenge enjoyable & we are looking forward to your solution!

– The Engineering Team at *UP42*