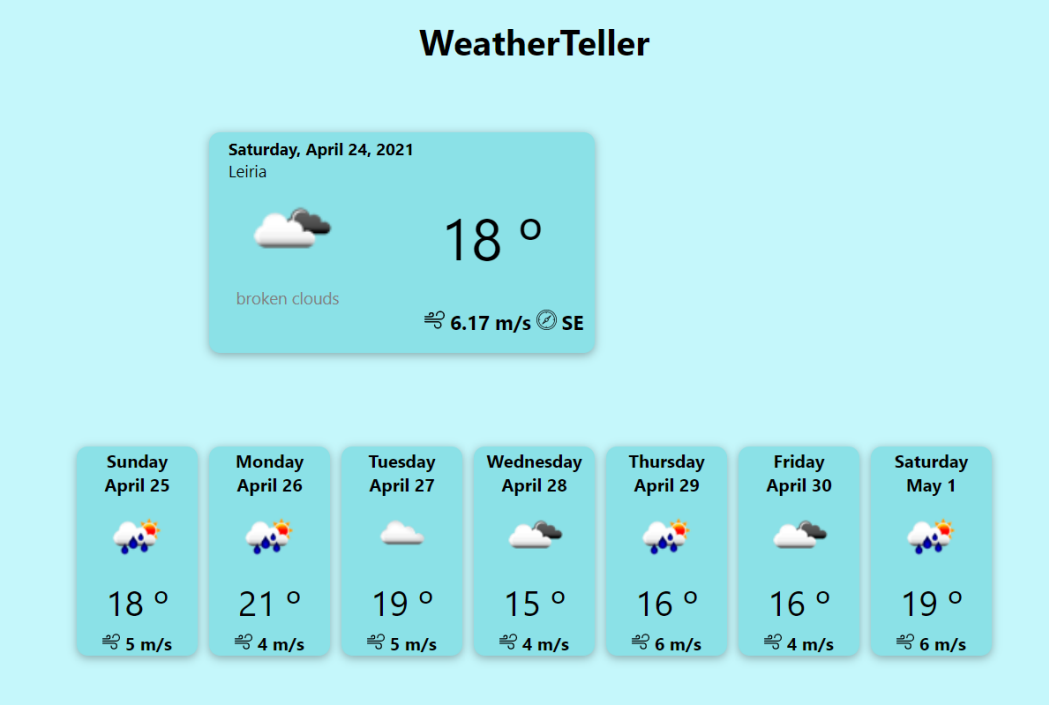
App Documentation

The WeatherTeller

By Gonçalo Sousa

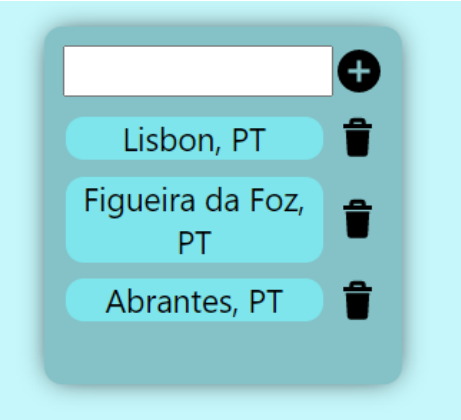
The WeatherTeller

The WeatherTeller is a simple weather application to present Current and Forecast Weather for the next 7 days. The weather is automatically fetched for the user device coordinates – if the user accepts the prompt so the browser gets the coordinates – when he enters the page.



The app also presents a City selection box, in which we can:

- Add new cities, by clicking the “Plus” button – cities must respect format: e.g. Leiria, PT (case insensitive).
    - The app fetches the city coordinates – if they exist – and then fetches the current and forecast weather for that location.
- Get the weather for previously added cities, by clicking any city present in the list. – the cities added are persisted in the browser storage, so the user does not loose them when he refreshes or leaves the app.
- Delete a city previously added, by clicking the “Thrash” button in front of it



The application structure is defined by a Higher Order Component Layout (with header and footer) that contains the main WeatherPage component. This component is the only stateful component and holds most application logic.

There are some auxiliary files that hold and export methods:

- Services: methods that fetch data from the APIs or the user device
- Utils: auxiliary methods to perform some tasks.
- Constants: constant string used throughout the app
- Hooks: custom hooks created to perform some tasks

In addition, the application is equipped with a set of tools to improve the user interface:

- Spinner: When the app is loading (fetching any data or waiting user interaction) a spinner is presented so the user knows there is something being done behind the curtains.
- ErrorModal: When an error occurs in the application, this error should be presented to the user in a modal that overlaps the application. The user can click the ok button or backdrop to dismiss it.
- Tooltips: User is presented with a hint when he hovers the city input box, and the add and delete buttons.

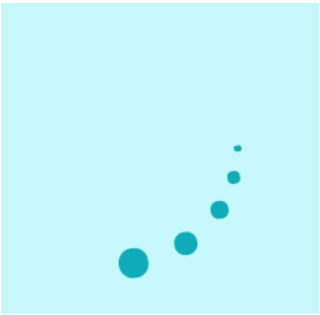


Figure 1 - Spinner

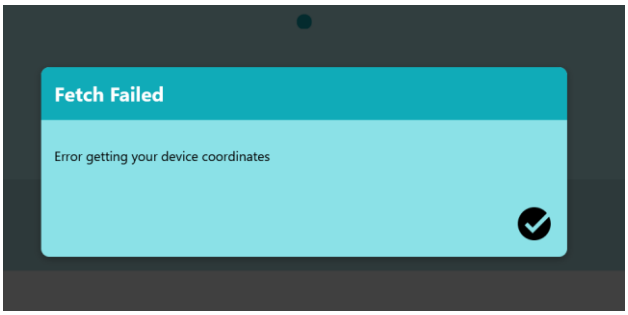


Figure 2 - ErrorModal

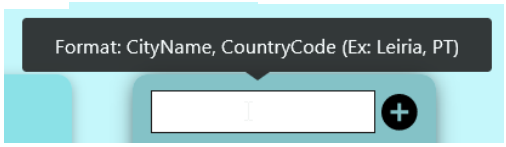


Figure 3- Tooltip

Run Commands

To run the application, we need to open any terminal, command line, or other option and Docker installed on the computer (https://www.docker.com/products/docker-desktop).

Open the terminal, cd (change directory) to the path of the application folder “weather-teller” (e.g. C:\Users\GoncaloSousa\Desktop\WeatherApp\weather-teller)

Use the command ***docker-compose up.***

The app will start running, when it finishes the build open a browser on ***http://localhost:3001/***

To stop the application, simply hit CTRL + C or use command ***docker-compose stop***.

Docker

The WeatherTeller is an app that runs inside a docker container.

Docker offers many other benefits like encapsulation, isolation, portability, and control.

Docker containers are small, and they start instantly.

Docker can be run on any OS-compatible host (Linux or Windows) that has the Docker runtime installed.

Dependencies

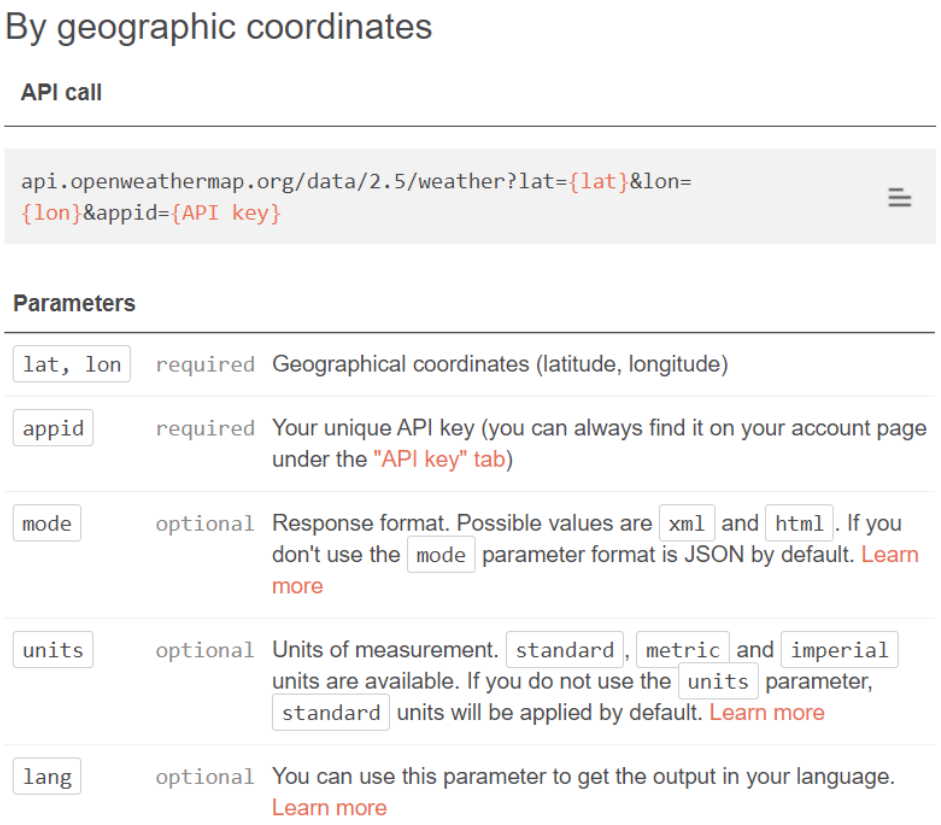
In this application, a few third-party packages were installed: to present some icons (present in the buttons) and the Tooltip package to present the hint on the user interface controls.

These packages where added to the package-json file so docker installs them, so the app runs smoothly.

- "react-tooltip": "^4.2.17",
- "@iconify-icons/gridicons": "^1.1.1",
- "@iconify/react": "^1.1.4"

API endpoints

Current Weather API – Get the current weather given a coordinate pair - https://openweathermap.org/current



By geographic coordinates

API call

```
api.openweathermap.org/data/2.5/weather?lat={lat}&lon=
{lon}&appid={API key}
```

Parameters

`lat, lon`	required	Geographical coordinates (latitude, longitude)
`appid`	required	Your unique API key (you can always find it on your account page under the "API key" tab)
`mode`	optional	Response format. Possible values are `xml` and `html`. If you don't use the `mode` parameter format is JSON by default. Learn more
`units`	optional	Units of measurement. `standard`, `metric` and `imperial` units are available. If you do not use the `units` parameter, `standard` units will be applied by default. Learn more
`lang`	optional	You can use this parameter to get the output in your language. Learn more

Forecast Weather API – Get the forecast weather given a coordinate pair - https://openweathermap.org/api/one-call-api

How to make an API call

API call

```
https://api.openweathermap.org/data/2.5/onecall?lat={lat}&lon={lon}&exclude={part}&appid={API key}
```

Parameters

`lat, lon`	required	Geographical coordinates (latitude, longitude)
`appid`	required	Your unique API key (you can always find it on your account page under the "API key" tab)
`exclude`	optional	By using this parameter you can exclude some parts of the weather data from the API response. It should be a comma-delimited list (without spaces). Available values: • `current` • `minutely` • `hourly` • `daily` • `alerts`
`units`	optional	Units of measurement. `standard`, `metric` and `imperial` units are available. If you do not use the `units` parameter, `standard` units will be applied by default. Learn more
`lang`	optional	You can use the `lang` parameter to get the output in your language. Learn more

Geocoding API – Get the coordinate pair given a city name - https://openweathermap.org/api/geocoding-api

Coordinates by location name

How to make an API call

API call

```
http://api.openweathermap.org/geo/1.0/direct?q={city name},{state code},{country code}&limit={limit}&appid={API key}
```

Parameters

`q`	required	City name, state code (only for the US) and country code divided by comma. Please use ISO 3166 country codes.
`appid`	required	Your unique API key (you can always find it on your account page under the "API key" tab)
`limit`	optional	Number of the locations in the API response (several results can be returned in the API response)

Responsivity

The WeatherTeller application is responsive on small and big screens, changing card and inner elements sizes to a minimum screen size 240 x 400 pixels.