

# BackbaseWeatherApp

---

Weather application that include 5 locations namely Johannesburg, Capetown, Windhoek, New York and Amsterdam.

Johannesburg will be loading by default but clicking on the labels at the top will fetch the weather for that City.

At the bottom of the card you can expand the weather card to receive a 4 day weather forecast for the chosen location.

## Components

- **HomeComponent** - As the name says, this is the default page and main page. This component also fetches the data from the `WeatherService`
- **WeatherCardComponent** - A reusable component that receives its data through `@Input()` properties and then displays all the weather data in a user friendly manner.

## Services

- **OpenWeatherService** - A service that will fetch all the weather information from the OpenWeatherApi service and then return it in a `observable` call to the calling method.
- **WeatherIconService** - Small helper service that will return a url to the required `svg` icon based on the current weather conditions.

## Other

- **OpenWeatherInterceptor** - An `Interceptor` that will intercept all http calls being made to the OpenWeatherApi and append it with our `account's ID`.
- **TempPipe** - A custom pipe that will convert the value received from Kelvin to Celcius.
- **weather.model** - File contains all models for our application. We use these models to map all incoming requests from the API to have a type safe way of accessing all the properties.

## Theme

A custom theme is set up using angular Material UI library and can be changed by using the `src/theme.scss` file

## Unit tests

Unit tests is available and can be run by using the command `ng test`, and code coverage can be checked by using `ng test --code-coverage`.

This project was generated with [Angular CLI](#) version 8.3.18.

## Development server

---

Run `ng serve` for a dev server. Navigate to `http://localhost:4200/`. The app will automatically reload if you change any of the source files.

## Code scaffolding

---

Run `ng generate component component-name` to generate a new component. You can also use `ng generate directive|pipe|service|class|guard|interface|enum|module`.

## Build

---

Run `ng build` to build the project. The build artifacts will be stored in the `dist/` directory. Use the `--prod` flag for a production build.

## Running unit tests

---

Run `ng test` to execute the unit tests via [Karma](#).

## Running end-to-end tests

---

Run `ng e2e` to execute the end-to-end tests via [Protractor](#).

## Further help

---

To get more help on the Angular CLI use `ng help` or go check out the [Angular CLI README](#).