02 - Postcode and Google Maps application

Build an application that uses the Postcode API and Google maps. The application allows people to type in a postcode, after which detailed information of that postcode (street, city, province) is displayed. A Google Map map will then open showing the relevant location including a marker.

The application has the following requirements:

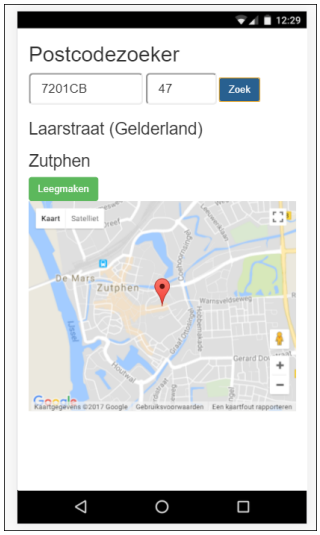
- The app uses Angular as the front-end framework.
- The architecture of the app must comply with the rules of the Angular Style Guide (https://angular.io/guide/styleguide).
- Make sure you have an attractive design. You may use a CSS framework such as Bootstrap, but this is not required. You may also use additional libraries such as Angular Material or PrimeNG. This is also not mandatory.
- The app must be usable on mobile devices (responsive).
- The app contains two search boxes (postcode and house number) and a *Search* button.
    - If the visitor enters a zip code and clicks on *Search*, the corresponding address, city and province will be displayed.
    - Below the address a Google Map of the location is shown.
- *There is an Empty* button that erases the data and input fields, and hides the Google Map.
- Make sure the app is scalable. Use the Single Responsibility principle for components. Use services to communicate with the backend.

WORKING METHOD:

- Use the Angular CLI to start the project. If necessary, add additional libraries or Angular Modules.
- Sketch out on paper what the layout will look like. Consider desktop and mobile usage (via Chrome DevTools).
- Create the component with two text boxes and a button. The button calls up a function such as `getAddress (postal code, number)`, which reads the values from the text boxes and passes them on to a service.
- Create a service to access the postcode API. The general address is
    - http://api.postcodedata.nl/v1/postcode/?postcode=<postcode>&streetnumber=<number>&ref=domainname.nl&type=json
    - Dynamically process the given postcode and house number.
    - Return the result to the component, which displays the found data in the user interface.
    - If no address was found, display a message in the user interface.

- Request a Google Maps API key and use it in the application. Guidelines for this can be found at https://developers.google.com/maps/ and https://developers.google.com/maps/documentation/javascript/get-api-key?hl=en#key .
- The return result of the Postcode API also contains a latitude/longitude value. Use this to initialize an element `<div id="map"> on the` page. An example of this can be found at the bottom of https://developers.google.com/maps/.
    - For example, process this code in a function `initMap()` that you call after the result from the service has been processed.

For example, the app may look like this:



OPTIONAL

- Use the Angular Google Maps add-on, available at https://angular-maps.com/guides/getting-started/.
- Instead of creating your own Google Map, you can `use` the components `< agm-map> and <` agm-marker> .
    - Look for yourself how this goes. In any case, you will need an API key for this as well.
- Finally, also optional: generate a distribution-build of your application with `ng build -prod [--aot]` and publish it on the internet. You can for example use free hosting on Github Pages. Examples of this are described on:
    - https://github.com/angular/angular-cli/wiki/stories-github-pages
    - https://alligator.io/angular/deploying-angular-app-github-pages/
    - or use the `angular-cli-ghpages` package directly: https://github.com/angular-buch/angular-cli-ghpages
- Send the URL to the teacher for assessment ;-)

Good luck!
Peter Kassenaar, info@kassenaar.com
