22-08-2020

APPLICATION TO DISPLAY F1 WORLD CHAMPIONS LIST

1. USE CASE

Using http://ergast.com/mrd/ to create a single page application that presents a list that shows the F1 world champions starting from 2005 until 2015. Clicking on an item shows the list of the winners for every race for the selected year. We also request to highlight the row when the winner has been the world champion in the same season.

2. API USED

- API for fetching data of selected year races :
  (Parameter used: year)
  http://ergast.com/api/f1/[year].json

- API for fetching data of top five selected race winners of selected season (parameter used: year, circuitid)(constraints: limit 5) :
  http://ergast.com/api/f1/[year]/circuits/[circuitId]/results.json?limit=5

- API for fetching data of selected season world champion:
  (Parameter used: year)
  http://ergast.com/api/f1/[year]/driverStandings/1.json

3. FRAMEWORK, LIBRARIES AND IDE USED

- Angular 10.0.6
- Typescript
- SCSS
- HTML5
- RxJS
- Jasmine and Karma
- VS Code

4. INSTALLATIONS

All the information regarding installations, running and testing the application is available in the file README.md present in the root directory of application.

5. FEATURES & FUNCTIONALITIES

- Angular latest version is used to develop SPA , scalable, component based architecture and cross platform compatibility.
- Developed the application keeping in mind clean code, unit test coverage, scalability, maintainability, enforcement of all constraints, CSS.
- SCSS is used for styling UI. Application is highly responsive.
- Added ease in/out transitions in select year drop down list while hovering in list.
- UX design and color combination is inspired from Mobiquity Portal.
- Developed a lightweight application with least usage of extra plugins.
- RxJS is used for reactive programming using observables to compose asynchronous code.
- Stored race winners data inside scope of application, in order to avoid repeated API calls.
- HTTP Interceptor is used to get formatted data.
- Interface is used in models to define data type, so that data exactly matches the properties set forth in the interface. It promotes strong typing in angular application.
- Data conversion service is used for converting data into more clean, feasible, accessible and reusable format.
- Proper Error handling in case of client and server side errors.
- Up to 80% Unit test code coverage.

6. SCOPE OF IMPROVEMENTS

- More transitions and animation could have been added in order to develop UI more attractive.
- Loader element on API calls could have been added. As of now APIs are fetching data really fast, there is not much need of it but it is still an improvement.
- As data is stored for race winners for every season to avoid repetitive calls, same could have been implemented for season's race list and world champion.

7. APPLICATION SCREENSHOT