Peter Popovics Massage website Project work

Github:	https://github.com/PopovicsPeti/ngmassage.git
Google Firebase host:	ng-massage-4ea7c.firebaseapp.com

I have made a desktop application for a massage company. I have made it in the mobile first approach. I used Angular framework and Bootstrap to build the project work. I embedded a third party booking application (Salonic) into my webpage in order to manage the appointment booking feature. Storing the data and the users I utilized the Google Fire Base while showing the data I installed the Angular Materials.

This work is a presentation so all of the features that it contains can be tried. I have saved the database so time to time I restore the data for further usage.

The webpage is a single page application with rooting service. The public part of the webpage mostly contains static pages that were styled in a unified way.

The services page is an exception because it is using data binding to generate the content.

Those functions which require registration uses the Firebase Auth Rest Api and they implement http requests (POST, GET, DELETE) between the Google Firebase and the project work.

I have established a unique data structure for organizing the data, furthermore I put interceptor and auth guard components into the application.

This application contains three major units.

1. Public components.

These pages can be used without registration. It serves to inform the users about basic information from the company and the workers.

- home page
- services
- booking (Salonic application link)
- contacts
- log in

2. Registered user unit.

This component stores the user's data.

- booking page of the profile

    - The users can delete their profile and they can check their appointments as well. This page has a badge which will be active if the user has at least 10 successful appointments.
- the howl appointment calendar
- medical contract
    - They can fill their medical status in order to make the therapy more useful.
- medical records
    - The therapist can write notes and tips in here to the user.

3. Admin page.

Only admins can reach this section. The application stores all of the users' data here that the admin can check and modify.

The admin can do the following:

- the personal and medical information about the user (check)
- the appointments (check, create, modify, delete)
- the medical records (check, create, modify, delete)
- the user's profile (delete)

My personal email is the "username" of the admin component.

email: p.popovics92@gmail.com

password: Admin-admin

These are the usable fictive users:

ROLE	USERNAME	PASSWORD
admin	p.popovics92@gmail.com	Admin-admin
user	test1@angular.com	12345678
user	test2@angular.com	12345678
user	test3@angular.com	12345678
user	test4@angular.com	12345678
user	test5@angular.com	12345678