# Soft Donations Workshop - Part 2

This application is created as a workshop for the [Angular Corse](https://softuni.bg/trainings/2760/angular-january-2020) @ [Software University](https://softuni.bg)

# Task Requirements

So far, we have created **one** **dynamic** **page** which **lists all causes** and **details** about cause. The next step is to implement **few** **more** **pages** and **route** **between** **them**. You will be provided with **HTML** and **CSS** skeleton for all other additional pages that you must create. There is one catch, the half of these pages require **authentication.** You must create some **properties** or **services** that **fake** this **authentication**. Because we don’t know yet how to manipulate **forms** properly or the real case with the authentication and authorization. We will fix that later when we have the knowledge to do that.

### Logged in navigation bar

When the "**user**" is "**logged in**" he should see the following navigation:

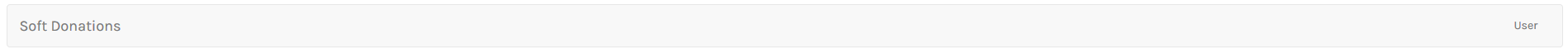
* [**Soft Donations**] **button** should refer to the home page ("**localhost:4200**")
* [**Create cause**] **button** should refer to the create cause page ("**localhost:4200/cause/create**")
* [**Profile**] **button** should refer to the "user profile" ("**localhost:4200/user/profile**")



### Not logged in navigation bar

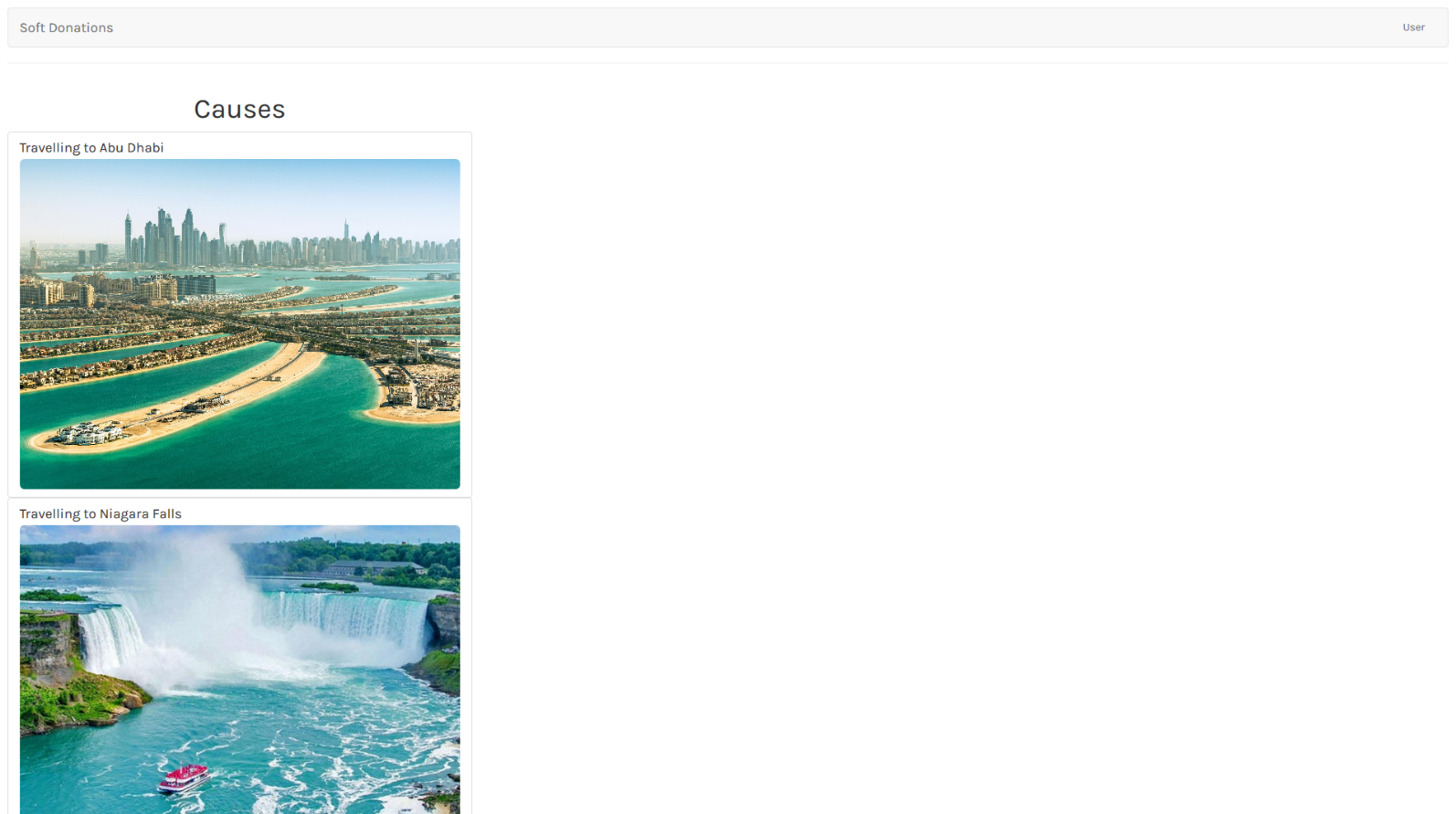
When the "**user**" is "**not logged in**" he should see the following navigation:

* [**Soft Donations**] **button** should refer to the home page ("**localhost:4200**")
* [**User**] **button** should refer to the user page ("**localhost:4200/user**")



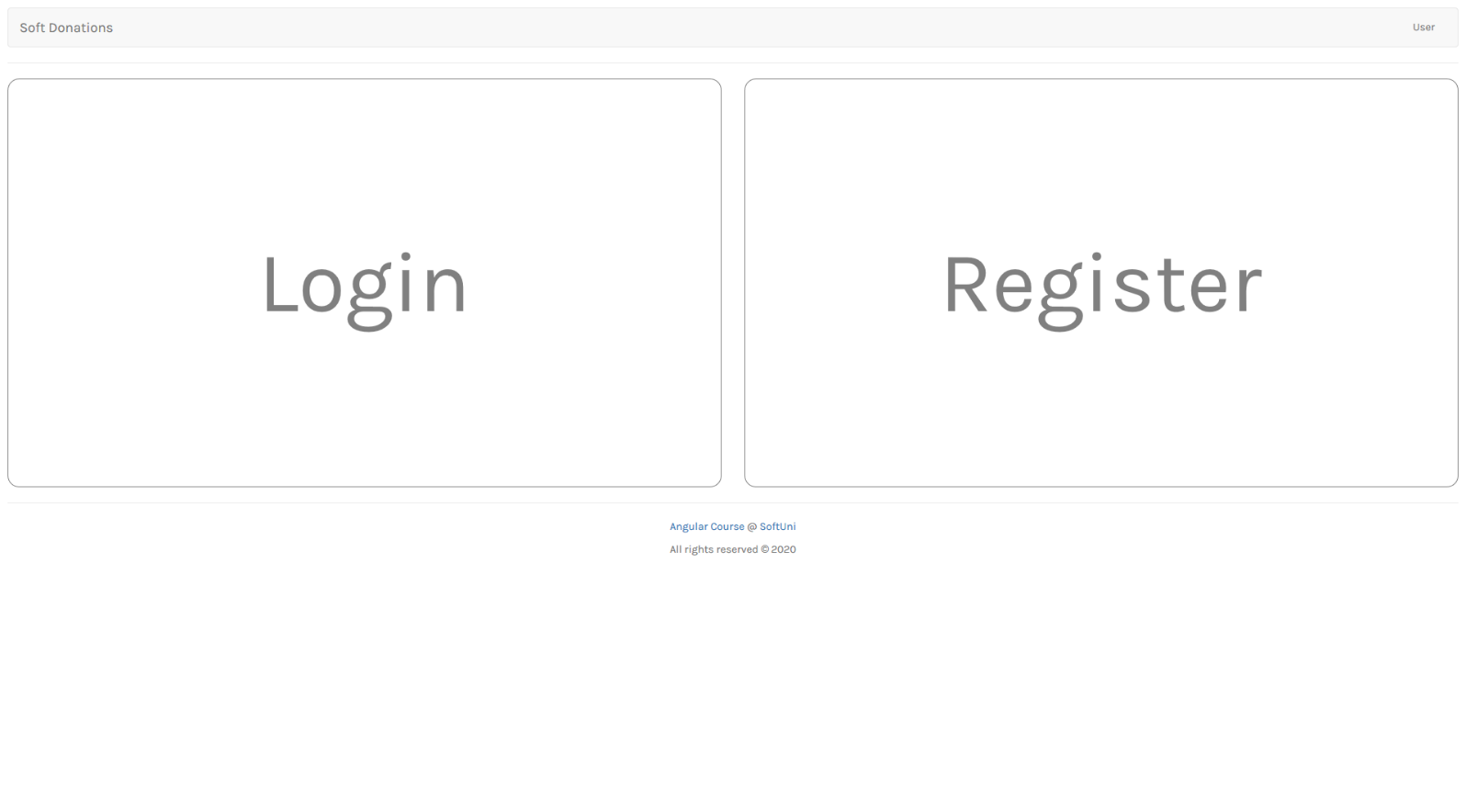
## Home page (Not logged in)

This is the home page ("**localhost:4200**"). This is the page that we have so far. Use it like home page for our application. All "users" can access this page no matter of their authentication. The only difference will be in the navigation bar.



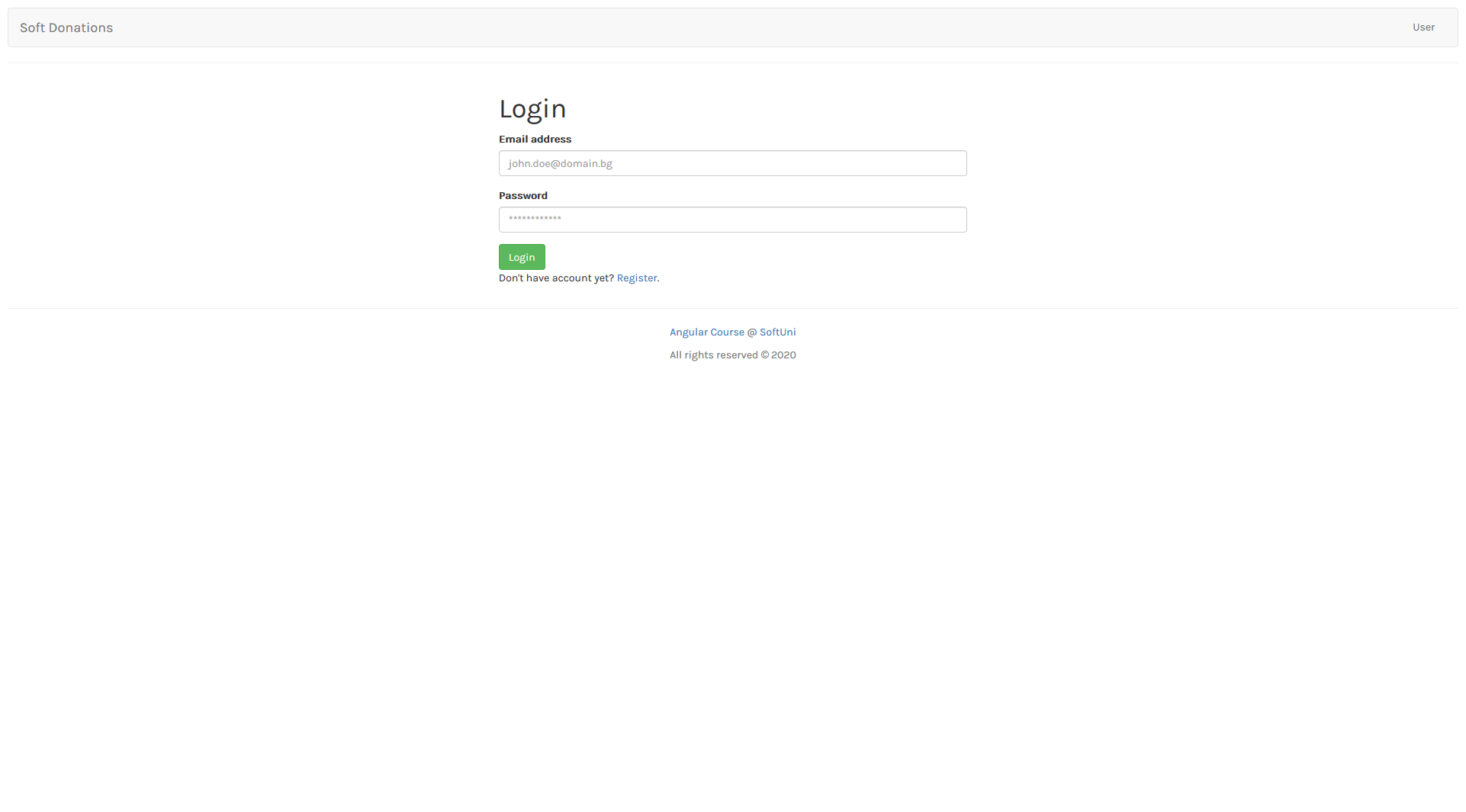
## User (Not logged in)

This is the user page ("**localhost:4200/user** "). Page where the "user" can choose to **login** or **register**. Each of the links should refer to the responding view.



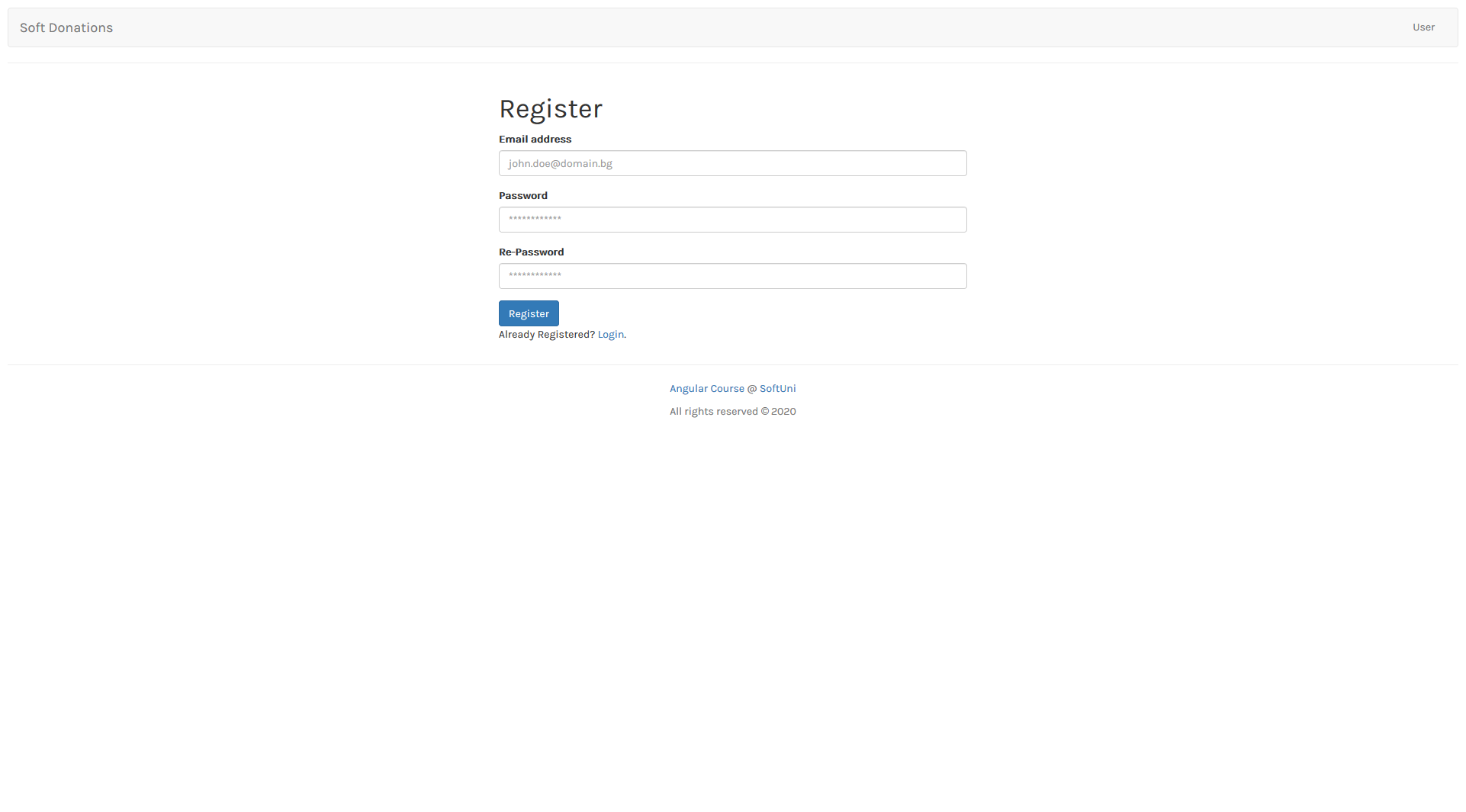
## Login (Not logged in)

This is the login page ("**localhost:4200/user/login**"). This will be the **one of the two points** of your app that the fake **isLoggedIn property** can be changed to **true**. For now, we don’t have database with all registered users. So, make sure **each registered** user is stored in **userService** or some other appropriate place. Each of them can logged in successfully.



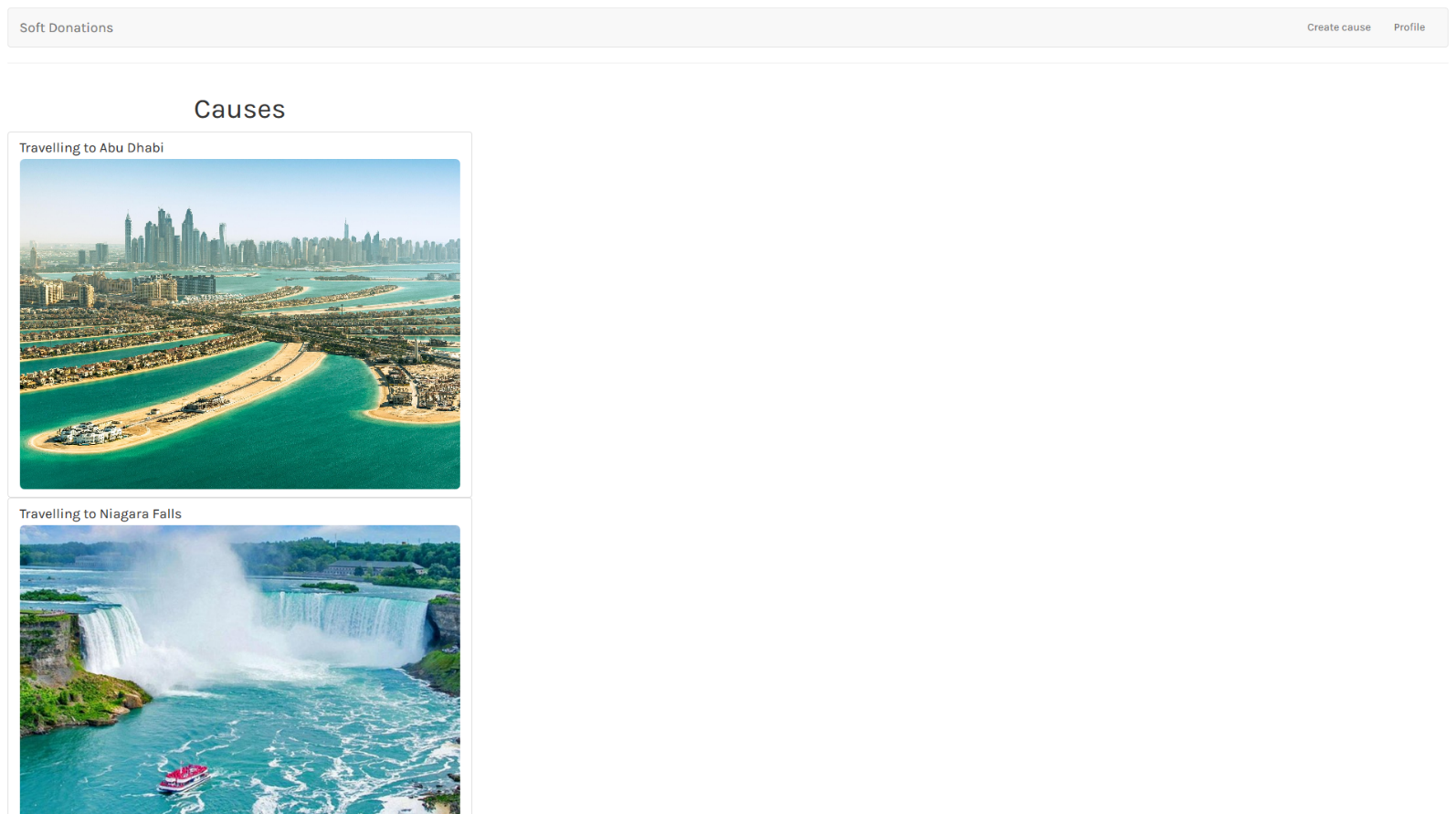
## Register (Not logged in)

This is the register page ("**localhost:4200/user/register**") This will be the **second point** of your app that the fake **isLoggedIn property** can be changed to **true**. For now, we don’t have database with all registered users. So, make sure **each registered** user is stored in **userService** or some other appropriate place. So, each of them can logged in successfully.



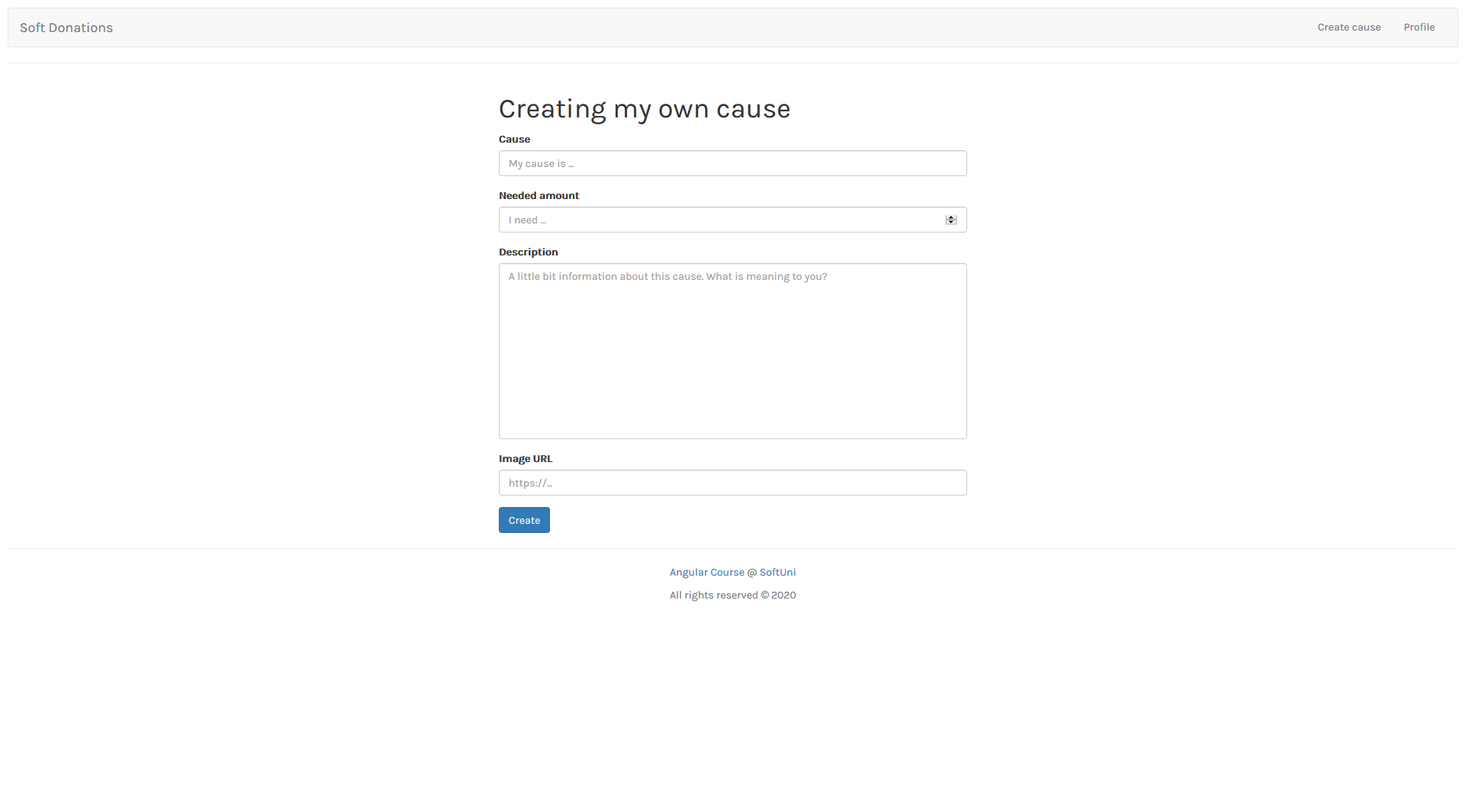
## Home page (Logged in)

This is the home page ("**localhost:4200**"). This is the page that we have so far. Use it like home page for your application. All "users" can access this page no matter of their authentication. The only difference will be in the navigation bar.



## Create cause (logged in)

This is the create cause page ("**localhost:4200/cause/create** ") where each **user** (logged in) can create his own cause. You can try to make **"POST" request** to ("**localhost:4200/causes**") with the given cause information. After successful creation, redirect the current "user" to the home page where all causes is listed.



## Profile (Logged in)

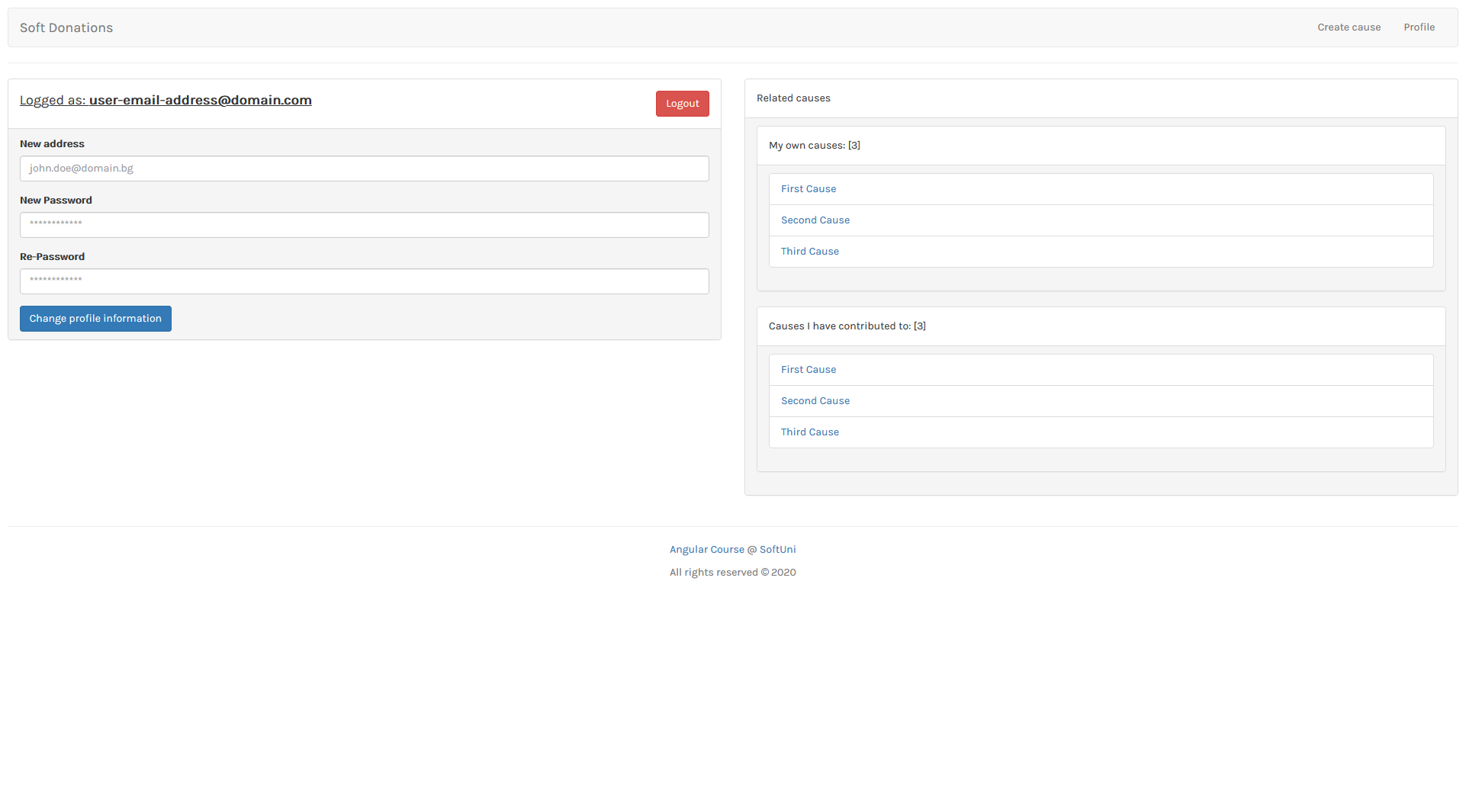
This is the profile page ("**localhost:4200/user/profile**"). This page will show the information about the currently logged in user. For now, the data in this page will be **static**, except if you create more than the **fake** **isLoggedIn** **property**.

[**Logout**] **button** will fake log out. If that happen just **change** the **user** **status** (**isLoggedIn** **property**) to **false**, **redirect** to the home page and **make sure** the user routes of the application are protected (can no longer be accessed).

The form will change the user information, but that will happen next time when you know how to control forms.

**Related** **causes** will show information about all causes that is related to the current user. The first list (my own causes) will be the list of all causes created by the current user and the count of them. The second list (causes I have contributed to) will be the list of all causes that the current user is donated to and the count of them.

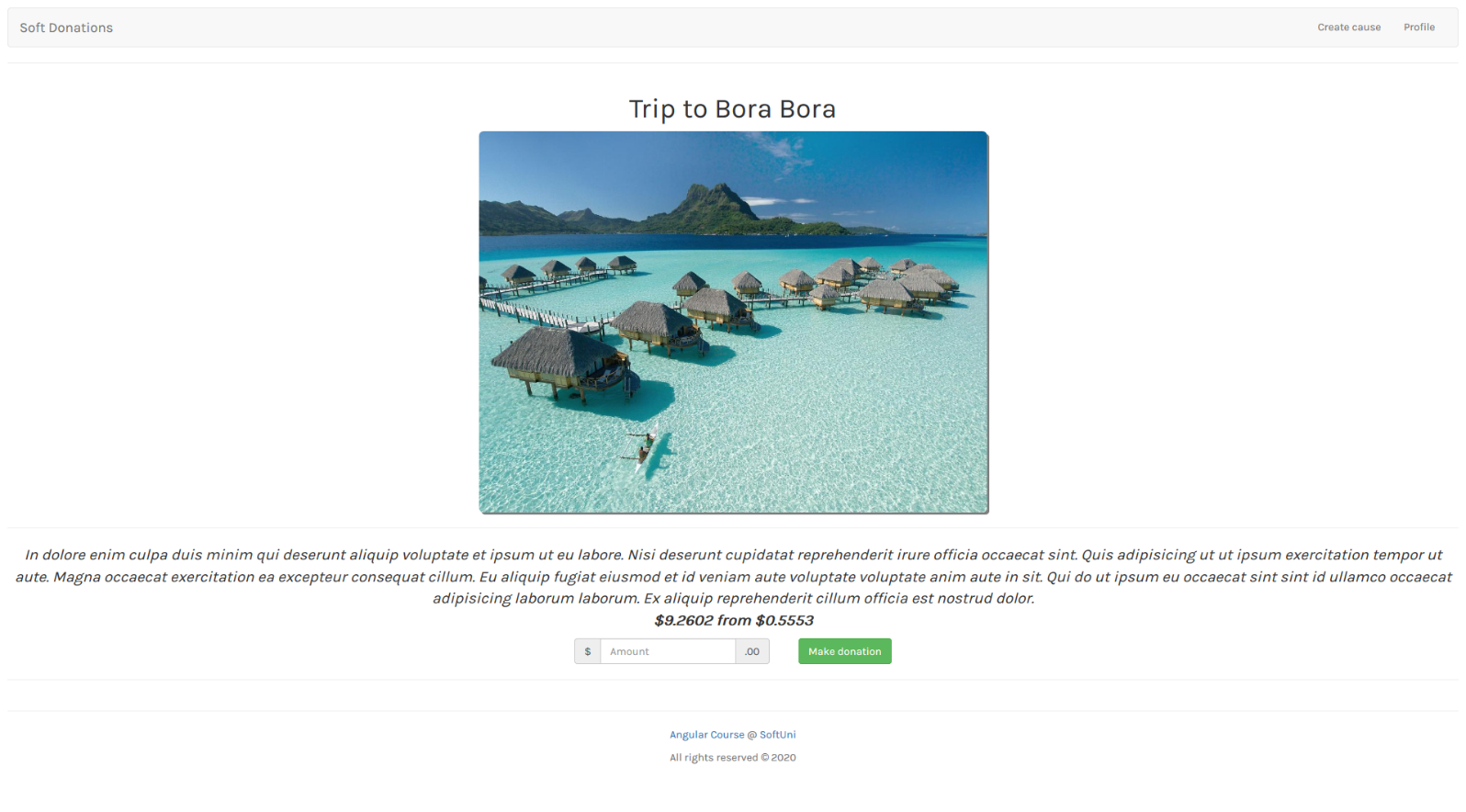
Each cause should be a link to detailed view for the current cause. ("**localhost:4200/cause/:causeId**")



## Details (Logged in)

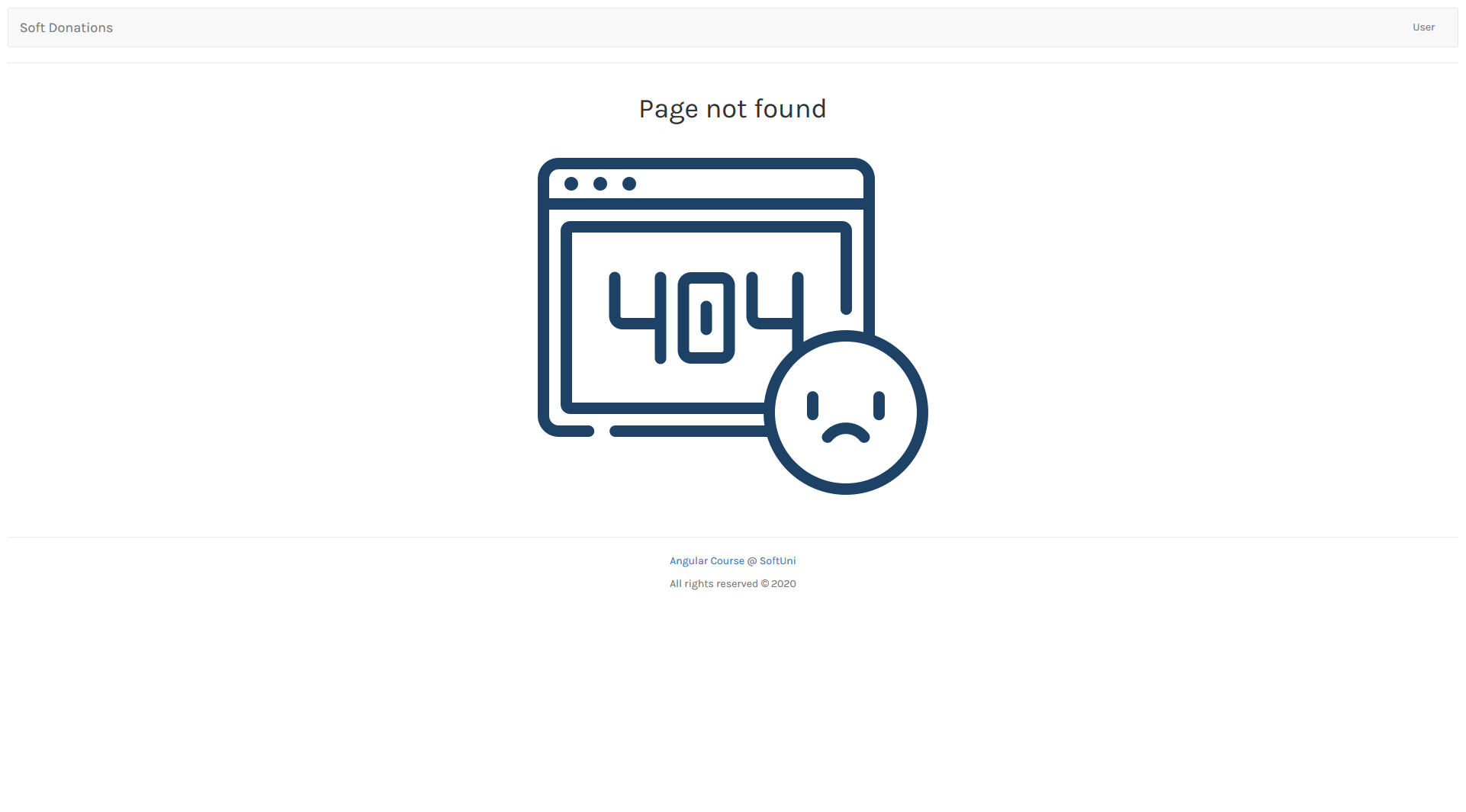
This is the login page ("**localhost:4200/cause/:causeId** ")

This is the page where only the information about current cause is shown. This page can be **accessed** from the links in **profile** **page** (where all related causes are listed).



## Invalid routes

This is the page for all **invalid** **routes** ("**localhost:4200/??????** "). Use it if invalid path or wrong one is accessed.



# Protected routes

Make sure all **user** (logged in) pages are **protected**. That means if your fake "**isLoggedIn**" **property** is **false** the logged in pages **can't be accessed**. Use guards to do that protection.=

***GOOD LUCK* 😊**