AngularMaterialDashboard

> This project is only useful if you want to use the **Angular Material** framework.

This project was generated with Angular CLI version 6.0.8.

What does this project include?

- Login GUI
- Login logic implementation
- Responsive side nav
- Navigation links with icons
- Toolbar
- Home page
- Customizable theme colors

Customize colors

In theme.scss you can set the three colors of Angular Material theme:

- primary
- accent
- warn

You have to use Angular Material colors. The initial configuration is:

```scss
$app-primary: mat-palette($mat-red);
$app-accent: mat-palette($mat-blue-gray, A200);
$app-warn: mat-palette($mat-red);
```

You can find a color list here:

https://material.io/tools/color/#!/?view.left=0&view.right=0

Navigation links

You can add your own navigation links. There is an array of links in `src/app/utils/main-nav-links.ts`.

You can add your own link by specifying:

- Name: name to display in navigation.
- Link: link to redirect.
- Icon: this is optional. You can find an icon list in the following link: https://material.io/tools/icons/?style=baseline

Login logic

To implement your own login logic you have to edit the `login()` and `logout()` methods in `src/app/services/auth/auth.service.ts`.

```
public login(email: string, password: string): Promise<any> {
    this.userIsLogged = true;
    return Promise.resolve(true);
}

public logout(): Promise<any> {
    this.userIsLogged = false;
    return Promise.resolve(true);
}
```

Important

You can implement your own logic to login but you must set the variable `this.userIsLogged` with the correspondent value and return the promise with the same value.

If you don't set the variable or don't return the promise, the login won't work.

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.