# Angular Project Assignment

Your task is to **design** and **implement** a web application using Angular. Use a service like Firebase for your **back-end** or create your own with Node.js and MongoDB or a framework in another language (ASP.NET, Spring, Symfony). It can be a discussion forum, blog system, e-commerce site, online gaming site, social network, or any other web application by your choice.

## Application Structure

The application should have:

* public part (accessible without authentication)
* private part (available for registered users)

### 1.1 Public Part

The public part of your projects should be visible **without authentication**. This public part could be the application start page, the user login and user registration forms, as well as the public data of the users, e.g. the blog posts in a blog system, the public offers in a bid system, the products in an e-commerce system, etc.

### 1.2 Private Part (User Area)

Registered users should have personal area in the web application **accessible after** **successful login**. This area could hold for example the user's profiles management functionality, the user's offers in a bid system, the user's posts in a blog system, the user's photos in a photo sharing system, the user's contacts in a social network, etc.

## General Requirements

Your Web application should use the following technologies, frameworks and development techniques:

* At least **3** **(Three)** different **dynamic pages** that fetch data from the back-end, allowing their content to change with the application state (pages like about, contacts, login, register do not count towards that figure)
* **Specific pages** for listing records and record details
* Communicate to a **remote service** (via REST, sockets, GraphQL, or a similar client-server technique).
* Implement **authentication**
* **Implement all CRUD operations (create, read, update, delete)**
* **At least 5 routers**
* **Meaningful commits to a source control system like GitHub, Bitbucket etc. for at least 3 days.**

**IMPORTANT:** If your project **doesn't cover** these conditions, you will **not** be graded!

## Other Requirements

* Use TypeScript with specific types (avoid the type "any"), interfaces, pipes, observables
* Apply **error handling** and **data validation** to avoid crashes when invalid data is entered.
* Implement route guards for the private AND the public part: guest users shouldn‘t be able to access private pages, logged-in users shouldn‘t be able to see the login/register pages
* Brief **documentation** on the project and project architecture (as .md file)
* Good usability. Good UI and UX. (NGBootstrap, MDL, CSS Grids or another method of your choice)
* The **GitHub** repo must be **public**.

## Project Defense

Each student will have to deliver a **defense** of their work in front of **one** other student, trainer, or mentor. Students will have **only 15 minutes** for the following:

* **Demonstrate** how the application works (very shortly)
* Show the **source code** and explain how it works
* Show any bonus functionalities they have implemented

Please be **strict in timing**! On the **15th** minute you **will be interrupted**! It is good idea to leave **the last 5 minutes for questions** from the trainers.

Be **well prepared** for presenting maximum of your work for minimum time. Open the project assets **beforehand** to save time.

The project defense will be happening **online** through **Discord** (or another platform of this type).

## Bonuses

* Deploy the application in a **cloud environment.**
* Use a **file storage cloud API**, e.g. **Dropbox**, **Google Drive** or other for storing the files.
* Use of features of HTML 5 like Geolocation, SVG, Canvas, etc.
* Use **Angular Animations** somewhere in your application.
* Write **unit tests** for your components.
* Use RxJS powered **state management** for Angular applications, inspired by Redux (ngRx store)
* Anything that is not described in the assignment is a bonus if it has some practical use.

## Assessment Criteria

### General Requirements – 30 %

### Functionality Presentation – 50 %

Adequately and clearly demonstrate the requested functionality. Know your way around the application and quickly demonstrate the code. Evaluation in this section is also based on the **structure** of your code.

### Answering Questions – 20 %

Answer questions about Angular specifics and potential functionality outside the scope of the project.

**Bonuses – Up to 10 %**

Additional functionality or libraries outside the general requirements, with motivated usage.

## Submission Deadline

You **must** submit your project before **23:59** on **23rd March** using a survey that will show up the **16th March**. A presentation **schedule** will be available on the **04th April** and will include only the projects that were **submitted beforehand.** **Non-submitted projects will NOT be evaluated.** You can work on your project before **23:59** on **03rd April.**

## Restrictions

You can use **parts** (some components, routing configurations, form validation etc...) of the **course workshop**, but you are **NOT** allowed to use the **whole workshop** as your project assignment. You are NOT allowed to use the HTML & CSS structure from any SoftUni course.

## Project Challenge

The **three best projects** will win a discount for the next course or module:

* First place – 80% discount voucher
* Second place – 50% discount voucher
* Third place – 30% discount voucher

The ranking of the projects is done **based only on the submitted project** (it does not include the assessment of the theoretical exam). Please make sure your project works when downloaded from the repository and keep it available at the same link up to 3 weeks after the exam.

The voucher could be used for **one course or for one module in the open or the professional program at SoftUni**. It **cannot be divided** into parts or **given to another person**. The voucher is valid for **one year** since the announcement of the winners.