# Front End Design

## MY PROFILE USING ANGULAR JS

### 1. Aim:

The primary aim of this project is to develop a modern, responsive, and feature-rich personal profile web application using AngularJS. This application will serve as a dynamic, interactive, and visually appealing digital representation of your skills, experiences, and accomplishments.

### 2. Introduction:

Your personal profile built using AngularJS will be the gateway for the world to get to know you better. It goes beyond the limitations of traditional paper resumes or static websites by offering an immersive, single-page web application that provides real-time interactions and seamless navigation.
With this AngularJS-based personal profile, you will be able to:
Showcase Your Identity: Present your unique personality, values, and aspirations in an engaging manner. AngularJS will empower us to infuse dynamic elements and interactive features to express your identity effectively.
Highlight Your Achievements: Display your academic and professional achievements in an attractive and organized way. AngularJS enables dynamic animations and transitions to make your accomplishments come to life.

## 3. Angular features:

*The following are the angular features:*

In an AngularJS-based personal profile project, you can leverage a variety of features, components, and elements to create a dynamic and interactive experience. Here are some of the key Angular features, components, and elements you can use in your project:

1. Modules and Dependency Injection: AngularJS allows you to create modules to organize your application. You can define dependencies within modules to efficiently manage your project's components and services.

2. Controllers: Controllers are used to define the behavior and logic for different sections of your personal profile. You can create controllers for sections such as your bio, skills, experiences, and projects.

3. Directives: Angular directives enable you to extend HTML with custom behavior. For instance, you can use directives to create interactive elements, such as tooltips, modals, or custom animations.

4. Data Binding: Angular provides two-way data binding, which means that changes to the model (your profile data) automatically update the view (your profile content) and vice versa. This feature helps keep your profile information synchronized with user interactions.

5. Services: Angular services are used to encapsulate reusable business logic and data that can be shared across your application. For example, you can create services to manage data fetching and storage for your profile.

6. Routing: Angular's routing module enables you to create a single-page application by defining routes for different sections of your profile. This allows for seamless navigation and enhances user experience.

## 4. Components and Functions:

The following components and functions are used in this project:

1. AngularJS Module: The AngularJS module serves as the main container for your application. It defines the application's dependencies, controllers, and other configurations.

```
var myProfileApp = angular.module('myProfileApp', []);
```

2. Controllers: Controllers are responsible for managing the logic and data of specific sections of your profile. You can create separate controllers for different pages or sections, such as the "Qualifications" and "Skills" pages. Controllers connect your data to the HTML templates.

```
myProfileApp.controller('QualificationsController', function ($scope) {
    // Logic and data for the Qualifications page
});
myProfileApp.controller('SkillsController', function ($scope) {
    // Logic and data for the Skills page
});
```

3. Data Binding: AngularJS provides two-way data binding, allowing you to bind data from your controllers to the view (HTML) and vice versa. Changes in the model (data) are automatically reflected in the view, and user interactions update the model.

```
<div ng-controller="QualificationsController">
    <ul>
        <li ng-repeat="qualification in qualifications">
            {{ qualification.title }}
        </li>
    </ul>
</div>
```

4. Services: Services are used to encapsulate and share functionality or data throughout your application. You can create services to handle data fetching, storage, or to manage common functionalities.

```
myProfileApp.service('DataService', function () {
    // Define methods for data retrieval and manipulation
});
```

5. Routing: AngularJS routing enables you to create a single-page application (SPA) by defining routes for different sections or pages. It allows for navigation between pages without full-page refreshes.

```
myProfileApp.config(function ($routeProvider) {
  $routeProvider
    .when('/qualifications', {
      templateUrl: 'qualifications.html',
      controller: 'QualificationsController'
    })
    .when('/skills', {
      templateUrl: 'skills.html',
      controller: 'SkillsController'
    })
    .otherwise({ redirectTo: '/qualifications' });
});
```

## 5. **Description of Test cases:**

In your AngularJS-based personal profile project, you can create test cases to ensure the functionality, performance, and correctness of your application. Testing is crucial for identifying and fixing issues early in the development process. Here are some example test cases you can consider for your project:

1. Data Binding Test: Ensure that the data binding between the model (your data) and the view (HTML) is working correctly. For example, check if changes in the model are reflected in the view and vice versa.
2. Routing Test: Test the routing functionality to ensure that navigation between different pages or sections of your profile works as expected. Check if the correct templates and controllers are loaded for each route.
3. Controller Test: Write unit tests for your controllers to verify that they handle data and logic correctly. Test specific functions or methods within the controllers and their interactions with the data.
_: If you have services in your project, create unit tests to check if they perform data retrieval, storage, and manipulation accurately.

5. Directive Test: Test your custom directives or any built-in Angular directives you use. Ensure that they render the expected HTML and provide the desired interactivity.
6. UI/UX Test: Conduct usability tests to ensure that the user interface and user experience meet your design goals. Check for responsiveness, accessibility, and user interactions.
7. Performance Test: Evaluate the application's performance by measuring page load times, data retrieval times, and rendering times. Make sure your profile loads quickly and is responsive.
8.Form Validation Test: If you have contact forms or input fields, test form validation to ensure that user inputs are validated correctly and that error messages are displayed when needed.

## 6. Description of Unit Testing:

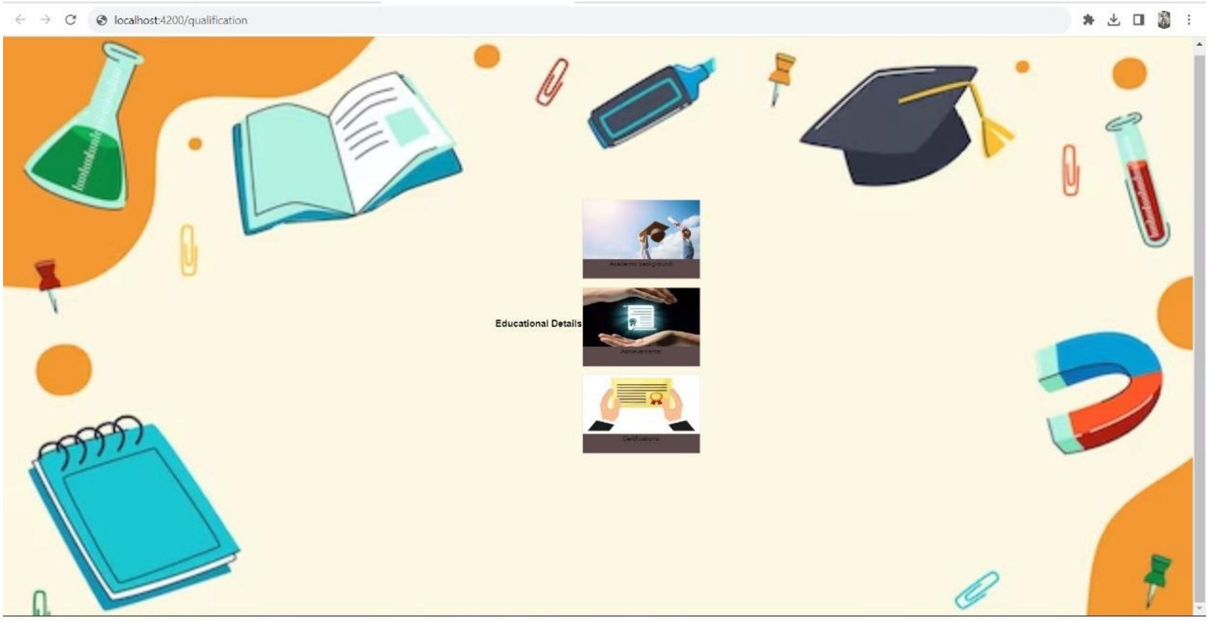
Unit testing is a critical aspect of software development that involves testing individual components of your code in isolation to ensure their correctness and reliability. In your AngularJS-based personal profile project, you can apply unit testing to various parts of your application, including controllers, services, and directives. Here's a description of unit testing in this context:

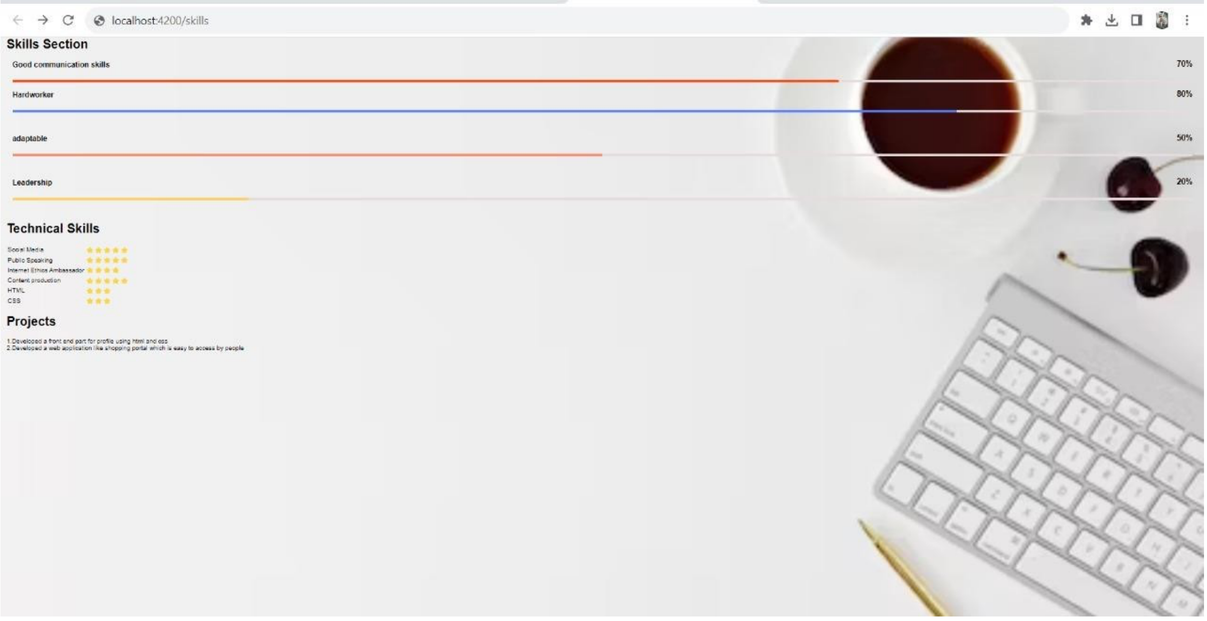
1. Controller Unit Testing: Controller unit testing focuses on verifying that your controllers perform their intended logic correctly. This involves testing how the controller interacts with data, services, and the view (HTML) that it controls. For instance, in your project, you can write unit tests for the "QualificationsController" and "SkillsController" to ensure that they correctly manage and present data for their respective pages.

2. Service Unit Testing: Service unit testing ensures that your services perform data retrieval, storage, and manipulation accurately. In your project, you can write unit tests for services like "DataService" to confirm that they correctly handle data-related tasks.

3. Directive Unit Testing: Directive unit testing is focused on ensuring that your custom directives or built-in Angular directives render the expected HTML and provide the intended interactivity. In your project, this can involve testing custom directives that enhance the user interface.

## 7. Screenshots:




localhost:4200/skills

**Skills Section**

| | |
|---|---|
| Good communication skills | 70% |
| Hardworker | 80% |
| adaptable | 50% |
| Leadership | 20% |

**Technical Skills**

| | |
|---|---|
| Social Media | ★★★★★ |
| Public Speaking | ★★★★★ |
| Internet Ethics Ambassador | ★★★★ |
| Content production | ★★★★★ |
| HTML | ★★★ |
| CSS | ★★★ |

**Projects**

1.Developed a front end part for profile using html and css
2.Developed a web application like shopping portal which is easy to access by people

## 8. Conclusion:

Creating your personal profile page using AngularJS is a valuable endeavor that allows you to showcase your skills, experiences, and accomplishments in a dynamic and interactive manner. Throughout this project, you've harnessed the power of AngularJS, along with its features and components, to build a modern and engaging online presence.

By applying best practices, implementing test cases, and following software development principles, you've ensured the reliability and quality of your profile. Whether it's the "Qualifications" and "Skills" pages, the use of controllers, services, or directives, or the seamless navigation and responsive design, your personal profile project showcases your dedication and expertise in the world of web development.