**Event Management App**

**DESCRIPTION:**

Build an Angular app where the admin can view and manage employees.

**Scenario:**

I am working as a web developer and your company has decided to launch a new app called Event Management. The administration team of the organization can view and add their resources such as employees and keep their data saved for future references.

The backend development has been outsourced as APIs and the frontend will be developed in-house by using Angular as a client-side framework. During the sprint planning, you agree to lead this project and develop an app that will let the admin find employees using APIs outsourced by backend engineers.

The tasks that need to be performed by you are:

* Build the application using Angular
* Create a temporary database server for CRUD operation using a JSON local server
* Validate all employee management forms using Form Validation
* Render the application as a Single Page Application

**Tools Required:**

* Angular
* Bootstrap

**The Following Requirements Should Be Met:**

* Admin login page where admin can change the password after logging in if he wants to.
* Admin can view a master list of employee details.
* Admin can create, remove, update, or delete employee details.

**Refer the following steps for JSON server creation:**

1) Execute the command given below:  
npm install -g json-server  
  
2) Create a file with the name db.json and add the code given below:  
{  
    "employees": [  
      {  
        "id": 1,  
        "first\_name": "Sebastian",  
        "last\_name": "Eschweiler",  
        "email": "[sebastian@codingthesmartway.com](mailto:sebastian@codingthesmartway.com)"  
      },  
      {  
        "id": 2,  
        "first\_name": "Steve",  
        "last\_name": "Palmer",  
        "email": "[steve@codingthesmartway.com](mailto:steve@codingthesmartway.com)"  
      },  
      {  
        "id": 3,  
        "first\_name": "Ann",  
        "last\_name": "Smith",  
        "email": "[ann@codingthesmartway.com](mailto:ann@codingthesmartway.com)"  
      }  
    ]  
  }  
   
3) Add the code given below in the script section of your package.json file in the root folder of your Angular application  
  
    "json:server": "json-server --watch db.json"  
  
Note: Before executing your Angular Application, run the command given below to start your json server:  
npm run json-server

**Angular:**

Angular is a development platform, built on [TypeScript](https://www.typescriptlang.org/). As a platform, Angular includes:

* Angular is a platform and framework for building single-page client applications using HTML and TypeScript.
* A component-based framework for building scalable web applications.
* A collection of well-integrated libraries that cover a wide variety of features, including routing, forms management, client-server communication, and more.
* A suite of developer tools to help you develop, build, test, and update your code

**Algorithm:**

**Step -1:**First I created a components called dashboard component,signup, and login component.

Using the commands to create the components

**“ng g c dashboard “🡪dashboard**

**“ng g c login “🡪login**

**“ng g c signup “🡪signup**

**Step- 2:** Create a api service for performing the curd operation:

**“ng g s api”** by using this command.

**Step -3:**Add the appropriate module in that components.

**Step- 4:** we can create db json called db.json for storing the values into the db:

For installing and run use the following command:

Installing the command is **”npm install -g json-server”.**

To run server the command is **”json-server --watch db.json”.**

**Create a database db.json:**

{

  "posts": [

    { "id": 1, "title": "json-server", "author": "typicode" }

  ],

  "comments": [

    { "id": 1, "body": "some comment", "postId": 1 }

  ],

  "profile": { "name": "typicode" }

}

**Step -5:** After adding the appropriate modules run the server for webpage

“**ng serve**” Command is used .

It will get the localhost port number is [**http://localhost:4200**](http://localhost:4200)

**Flowchart:**

login

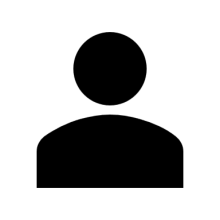
Login

success

login

Dashboard

For curd operation

****

signup

login

signup