Angular Assignment: **Hotel Management System**

Requirement:

Technologies: Angular 2+, Typescript, HTML/CSS, Primeng.

Framework: NodeJS, Angular CLI

IDE: Visual Studio Code

> Prepare README.md file, how to set up the project, and run/test.

* README.md file is mandatory for the execution and running process.

HTML/CSS Screens:

1. Login page

2. Home Page

3 Customer module Service

4. Booking module Service

5. Room details

**Login screen**

> When a user tries to log in, check whether the user is available or not. If the user is available allow them to log in.

> If the user does not exist, alert them as "Invalid Credentials".

> Use Authguard to prevent unauthorized access to the service.

> Responsive view design.

> Angular Authguard.

> Custom Directives and pipes.

**Home Page**:

1. Top Header needs to display on all pages.

2. Display the menu

* Home:
  + It should be highlighted when the user navigates from the login page.

Some image banners and static text should be displayed on the home page.

* + Chart – It should give the details about the booking status of the last 7 Days (how many bookings have been done each day).
    - * + Services: (Customer module and Booking module). From the menu user will choose a module, then redirect to a particular list page.
        + Room Details: When a user clicks on the Room details menu, it should display Room images with Room details (bed, ac, fan, Wi-Fi, etc)

**Customer Module Service**:

When the user clicks on the customer module from the menu the user should navigate to the customer module service list screen.

* List all the existing customers as paginated in the table. By default, display 10 records on each page.

List Page Columns

1. Name (username)

2. Customer Id (Auto-generated value)

3. First Name

4. Last Name

5. Email-Id

6. Address

7. Aadhar number

8. Mobile number

9. Created Date Time - use custom directives to convert the Date Time to a logged-in user-selected time zone and locale.

10. Modified Date Time

11. Created username

12. Modified Username

* Top right of the table, show the "+ Add" button. When the user clicks add button, open the popup, and ask for the username (Mandatory).
* In the popup, display the save and cancel button, when the user clicks save redirects to a detail page, and cancel closes the popup.
* Introduce filter and sorting also.

**Customer Add/Edit/View Page**:

* Use HTML forms and directives.
* Inside panel displays the form fields as listed below.

Fields:

1. Username --> Text Box (Allow alphanumeric characters), Max Length: 20, Mandatory

2. First Name --> Text Box (Allow alphanumeric characters and space), Max Length: 30, Mandatory

3. Last Name --> Text Box (Allow alphanumeric characters and space), Max Length: 30, Optional

4. Date of Birth --> date field

5. Mobile number --> Country Code (+91), Text box (Allow Numeric), Max Length: 10, Mandatory

6. Address --> Text Box, Max Length: 100, Mandatory

7. Email-Id --> Email Id validation check and Max Length: 100. Mandatory

8. Aadhar no -->Textbox, mandatory

9. Upload Aadhaar Image --> Support PNG, and JPG formats. Mandatory

10. Upload User Image --> Support PNG, and JPG formats. Optional

11. Created Date Time --> Use custom directives to convert the Date Time to a logged-in user-selected time zone and locale.

1. Modified date time (for edit and view mode only)– should update automatically once the record is updated.
2. Created Username (Should update automatically once the record is added)
3. Modified username (should update automatically once the record is modified)

validations:

* Alert the user when trying to add duplicate records. Unique Field is the username.
* Uploaded image size less than 2MB.
* Should not allow saving without attaching the Aadhar image

**Booking Module Service**:

When the user clicks on the booking module from the menu, the user should navigate to the booking module service list screen.

1. List all the existing bookings as paginated in the table. By default, display 10 records on each page.

List Page Columns

* Name (username)
* Booking Id (Auto-generated)
* Booking Status
* Customer Id
* First name
* Last name
* No of People (adult child)
* Phone
* Room details
* Price details (auto-generated based on the room selected)
* Created Date Time - use custom directives to convert the Date Time to a logged-in user-selected time zone and locale.
* Modified date time
* Created username
* Modified username

2. Top right of the table, show the "+ Add" button. when the user clicks add button redirect to a detail page.

3. Introduce filter and sorting also.

**Booking Add/Edit/View Page**:

a. Use HTML forms and directives.

b. Inside panel displays the form fields as listed below.

Fields:

1. Booking Id (Auto-generated)
2. Customer Id
3. First name
4. Last name
5. No of people, mandatory
6. Phone
7. Room Details -autocomplete dropdown will display Room names
8. Price details (auto-generated based on the room selected).
9. Aadhaar No, mandatory
10. Booking Status (Non-editable field)
11. Created Date Time - use custom directives to convert the Date Time to a logged-in user-selected time zone and locale.
12. Modified date time (for edit and view mode only)– should update automatically once the record is updated.
13. Created Username (Should update automatically once the record is added)
14. Modified username (should update automatically once the record is modified)

**Room details:(view screen)**

Use the list of static data of rooms.

Allow users to view the room availability with price and room amenities. When the user clicks on room details from the home screen

**Admin Screen – List screen and view mode.**

1. Admin will be able to see all the bookings in the list screen and the admin has permission to confirm and cancel the booking. Once the booking is confirmed or rejected, the status should also be updated on the booking and admin screens.

Action buttons in detail screen:

- Confirm booking (status – confirmed)

- Reject booking (status – rejected)

**Note:** No need to use the backend. You can use JSON server for CRUD operation and JSON storage.