Assignment Category: Assignment-09_category_daisy

🌿 GreenNest – Indoor Plant Care & Store

[Assignment Video](#)

🌱 Project Theme

GreenNest is an elegant single-page web application built for plant lovers who want to nurture and decorate their homes with healthy indoor plants. The platform allows users to explore plant care guides, buy plants, and book expert consultations — ensuring a greener and healthier living space during any season.

🎯 Project Goals

1. Build a responsive, visually appealing plant care and store platform.

2. Implement secure Firebase authentication (Signup, Login, Google Sign-In, Forgot Password).

3. Fetch plant data from JSON for product/service listings.

4. Create a protected route for service details and profile management.

5. Maintain SPA functionality with a minimalist and calming design.

🧩 Core Features & Functional Requirements

1. Layout Structure

Navbar

- Logo: *GreenNest*

- Navigation Links: `Home`, `Plants`, `My Profile`

- Conditional rendering:

    - Logged in → show user avatar + dropdown with `displayName` and `Logout`.

    - Logged out → show `Login` and `Register` buttons.

Footer

- Quick Links (About, Contact, Privacy Policy)

- Social Media Icons (Instagram, Facebook, Pinterest)

- "© 2025 GreenNest. All rights reserved."

Others Requirements

- Navbar and Footer remain visible on all routes.

- No crash or reload errors on route navigation.

---

2. JSON Data Setup

Create a `plants.json` file containing **at least 6 indoor plant objects**.
 Each should have:

- `plantId`, `plantName`, `category`, `price`, `rating`

- `description`, `image`, `availableStock`, `careLevel`, `providerName`

Example:

```
[
 {
  "plantId": 1,
  "plantName": "Snake Plant",
  "category": "Air Purifier",
```

```
  "price": 18,
  "rating": 4.8,
  "availableStock": 10,
  "careLevel": "Easy",
  "description": "A hardy plant that purifies indoor air and thrives in low light.",
  "image": "https://i.postimg.cc/example1.png",
  "providerName": "UrbanGreen Studio"
 }
]
```

3. Home Page Sections

- **Hero Section:**
  Use **Swiper.js** or **Framer Motion** for a smooth, nature-inspired slider featuring plants and care slogans.

- **Top Rated Indoor Plants:**
  Display cards fetched from JSON with image, name, price, rating, and "View Details" button.

- **Plant Care Tips Section:**
  Static or fake JSON data about watering, sunlight, and fertilizing tips.

- **Meet Our Green Experts:**
  Show 3–4 plant care experts with images, names, and specialization.

- **Extra Section (Optional):**
  "Eco Decor Ideas" — showcasing how to style your space with plants.

4. Plant Details Page (Protected Route)

Accessible only when logged in.
 If unauthenticated → redirect to Login, then back to details after login.

Show complete JSON info with:

- Large image, name, description, price, rating, stock

- "Book Consultation" form with fields:

- Name

- Email

- "Book Now" button
  On submit → show success toast and clear form.

---

5. Authentication

1. Login Page:

Show a Login page with a form, so that the user can Log in this application.
- Show a Title for Login.
- Form with the following fields
  - Email,
  - Password,
  - Forget Password,
  - Login button

If the user logs in successfully then
- navigate him to his desired Route or Home page.
- If not, show him an error with a toast/error message anywhere in the form.

There will be some other options like
- Show the user a Link for Signup so that he can go to the Signup page.
- Show users a Social Login Button (Google only) . on Clicking it:
  - user authenticates with Google
  - Navigate him to his desired Route or Home page. **(Don't Miss it)**

2. Signup Page:

Create a signup page with a form, so that the user can signup himself in this application.
- Show a Title for signup and a Form with the following fields
  - (Name, Email, Photo-URL, Password & Register Button)

If the user sign up successfully then

- navigate him to the Home page.
- If not, show him an error with a toast/error message anywhere in the form.

Implement password validation

For password validation, you need to follow the below criteria. Show a password error in the form, and don't Register for an invalid password

- Must have an Uppercase letter in the password
- Must have a Lowercase letter in the password
- Length must be at least 6 character

There will be some other options like

- Show the user a Link for Login so that he can go to the Login page.
- Show users a Social Login Button (Google only) . on Clicking it
  - User authenticates with Google
  - Navigate the user to the Home page. (Don't Miss It)

💡 Don't implement email verification method as it will inconvenience the examiner. If you want, you can add these after receiving the assignment result.

---

6. My Profile Page

- Show user info:

  - Name

  - Email

  - Photo

- Include "Update Profile" button → updates `displayName` and `photoURL` using `updateProfile()` method.

🧪 Challenges

1. Implement `updateProfile()` for real-time user updates.

2. Functional "Forgot Password" with Gmail redirect.

3. Add password toggle (show/hide) on both Login & Signup pages.

4. Add an extra homepage section with creative value (e.g., "Plant of the Week").

What to Submit

Your Github Repo Link :
Your Live Link :