



# Recipe App – Setup & Testing Guide

## Contents

Links .....	1
Testing API.....	3
Using Postman .....	3
Sample Register Request.....	4
Folder structure .....	5
Screen Shorts .....	7

## Links

**GitHub Link –**

<https://github.com/Manusha122/RecipeApp.git>

**Netlify -**

<https://recipe-app-manu.netlify.app>

## **1** Clone the Project

**Open a terminal and run:**

```
git clone https://github.com/Manusha122/RecipeApp.git
```

```
cd recipe-app
```

## **2** Backend Setup

**Go inside backend folder:**

```
cd backend
```

**Install dependencies:**

```
npm install
```

(uses **nodemon** – if missing, install with `npm install -g nodemon`)

**Start backend:**

```
npm run dev
```

## **3** Frontend Setup

**Open new terminal → go inside frontend folder:**

```
cd frontend
```

**Install dependencies:**

```
npm install
```

**Start frontend:**

```
npm run dev
```

# Testing API

Import RecipeApp.postman\_collection.json into Postman and test endpoints.

## Run

- Backend: <http://localhost:5000>
- Frontend: <http://localhost:5173>

## Testing the Backend APIs

### Using Postman

We already created a Postman collection:

 RecipeApp.postman\_collection.json

1. Import into Postman.

2. Test APIs:

- POST /api/auth/register → create new user.
- POST /api/auth/login → login and get JWT.
- GET /api/recipes/categories → fetch recipe categories.
- POST /api/recipes/favorites → add favorite (requires JWT).
- GET /api/recipes/favorites → get user favorites.

## Sample Register Request

**POST http://localhost:5000/api/auth/register**

**Content-Type: application/json**

```
{  
  "name": "John Doe",  
  "email": "john@example.com",  
  "password": "123456"
```

}

## c. Sample Login Response

```
{  
  "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6...",  
  "user": {  
    "_id": "66d9df12b...",  
    "name": "John Doe",  
    "email": "john@example.com"  
  }  
}
```

# Folder structure

recipe-app/

  └── backend/

    └── package.json

    └── .env       # create from sample below

    └── server.js

    └── src/

      └── config/db.js

      └── middleware/auth.js

      └── models/Favorite.js

      └── models/User.js

      └── controllers/authController.js

      └── controllers/recipeController.js

      └── controllers/favoriteController.js

      └── routes/

        └── authRoutes.js

        └── recipeRoutes.js

        └── favoriteRoutes.js

  └── frontend/

    └── package.json

    └── index.html

    └── src/

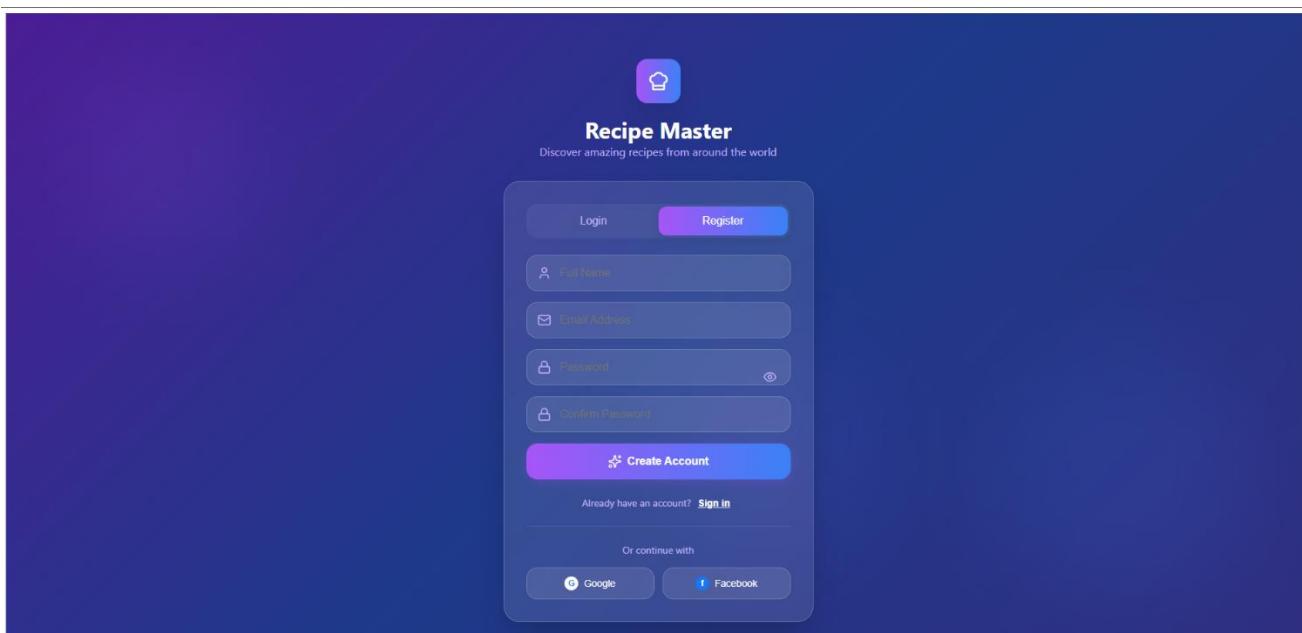
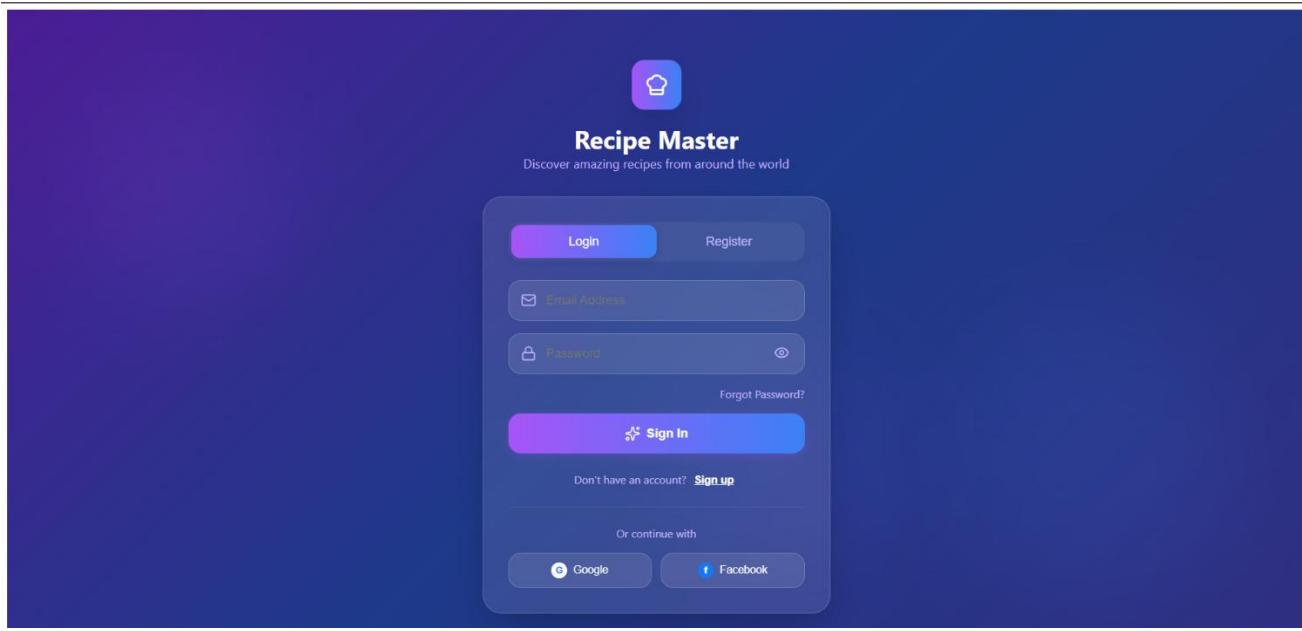
      └── main.jsx

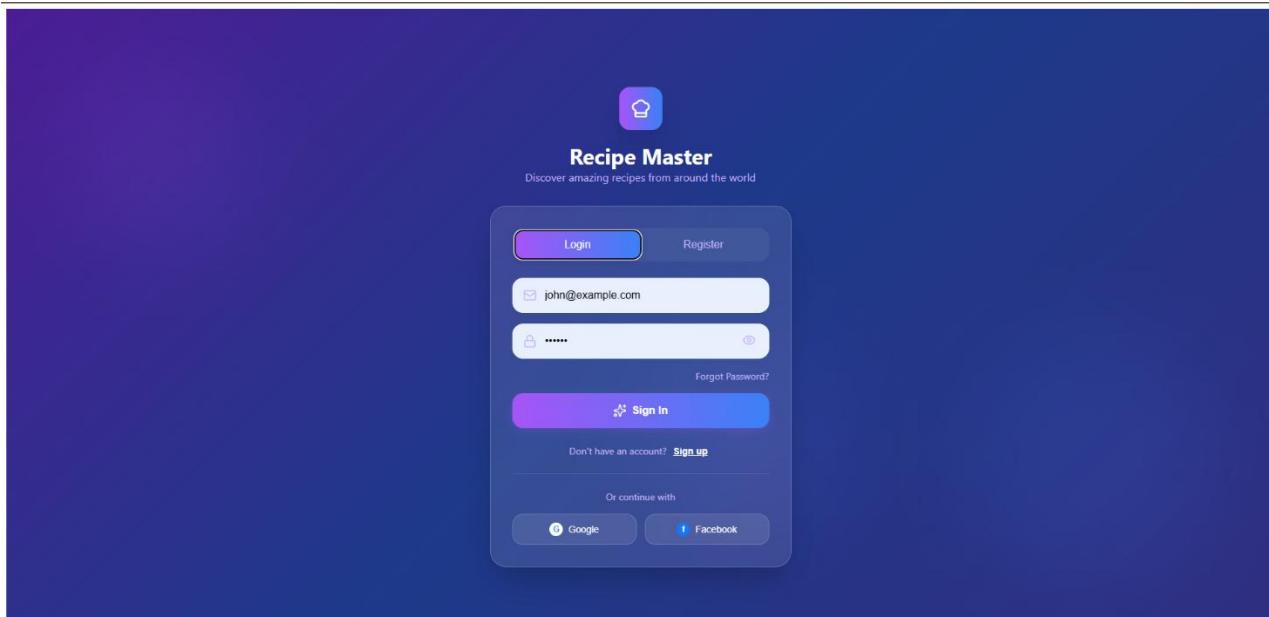
      └── App.jsx

      └── styles.css

```
api.js
context/AuthContext.jsx
components/
|   ProtectedRoute.jsx
|   CategoryPills.jsx
|   RecipeCard.jsx
|   RecipeModal.jsx
pages/
|   Login.jsx
|   Dashboard.jsx
|   Favorites.jsx
```

# Screen Shorts





## Recipe Discovery

Explore culinary delights from around the world

**Categories**

- Beef
- Chicken
- Dessert
- Lamb
- Miscellaneous

Search recipes (e.g., Italian, Pasta, Chicken...)

### Beef Recipes (48)

Beef and Mustard Pie      Beef and Oyster pie      Beef Asado

[Details](#) [Details](#) [Details](#)

## Recipe Discovery

Explore culinary delights from around the world

**Categories**

- Beef
- Chicken
- Dessert
- Lamb
- Miscellaneous

Search recipes (e.g., Italian, Pasta, Chicken...)

### Chicken Recipes (36)

15-minute chicken & halloumi burgers      Ayam Percik      Brown Stew Chicken

[Details](#) [Details](#) [Details](#)

## Recipe Discovery

Explore culinary delights from around the world

### Categories

Beef    Chicken  
Dessert    Lamb  
Miscellaneous

Search recipes (e.g., Italian, Pasta, Chicken...)

### Dessert Recipes (65)



Apam balik

Details



Apple & Blackberry Crumble

Details



Apple Frangipane Tart

Details



Banana Pancakes

Details



Pancakes

Details



Yellow Cake

Details

## Recipe Discovery

Explore culinary delights from around the world

### Categories

Beef    Chicken  
Dessert    Lamb  
Miscellaneous

Search recipes (e.g., Italian, Pasta, Chicken...)

### Lamb Recipes (15)



Kapsalon

Details



Keleya Zaara

Details



Lamb and Lemon Souvlaki

Details



Lamb Stew

Details



Biryani

Details



Rice Pilaf

Details

## Recipe Discovery

Explore culinary delights from around the world

### Categories

Beef    Chicken  
Dessert    Lamb  
Miscellaneous

Search recipes (e.g., Italian, Pasta, Chicken...)

### Miscellaneous Recipes (12)



Bean & Sausage Hotpot

Details



Callaloo Jamaican Style

Details



Chakchouka

Details



Roast Chicken

Details



Spicy Lentil Stew

Details



Omelette

Details

# Recipe Discovery

Explore culinary delights from around the world

▼ Favorites

→ Logout

## Categories

Beef    Chicken  
Dessert    Lamb  
Miscellaneous

Search recipes (e.g., Italian, Pasta, Chicken...)

### Miscellaneous Recipes (12)



Bean & Sausage Hotpot



Callaloo Jamaican Style



Chakchouka



# Recipe Discovery

Explore culinary delights from around the world

▼ Favorites

→ Logout



Loading delicious recipe...

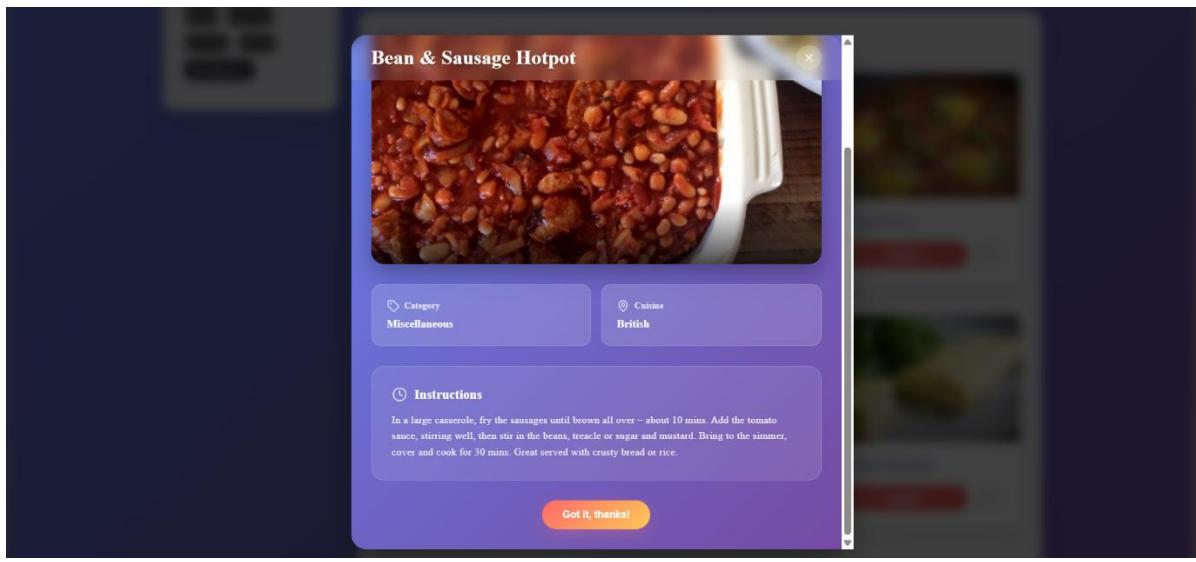
### Bean & Sausage Hotpot



Category  
Miscellaneous

Cuisine  
British

#### Instructions



## My Favorites

[← Back to Dashboard](#)

Bean & Sausage Hotpot [Remove](#)

Beef and Mustard Pie [Remove](#)

## Recipe Discovery

Explore culinary delights from around the world

[▼ Favorites](#) [→ Logout](#)

Categories

- Beef
- Chicken
- Dessert
- Lamb
- Miscellaneous

Search: pie

Search Results (5)

Beef and Mustard Pie [Details](#)

Beef and Oyster pie [Details](#)

Minced Beef Pie [Details](#)

MongoDB Compass - localhost:27017/recipe\_app.favorites

Connections Edit View Collection Help

**Compass**

My Queries

**CONNECTIONS (1)**

- localhost:27017
  - admin
  - config
  - emojiDB
  - employeeDB
  - local
  - recipe\_app
    - favorites
    - users

Welcome users stories favorites recipe\_app +

localhost:27017 > recipe\_app > favorites

Documents 0 Aggregations Schema Indexes 3 Validation

Type a query: { field: 'value' } or [Generate query](#) [Explain](#) [Reset](#) [Find](#) [Options](#)

[ADD DATA](#) [EXPORT DATA](#) [UPDATE](#) [DELETE](#)

25 0 of 0

This collection has no data  
It only takes a few seconds to import data from a JSON or CSV file.

[Import data](#)

MongoDB Compass - localhost:27017/recipe\_app.favorites

Connections Edit View Collection Help

**Compass**

My Queries

**CONNECTIONS (1)**

- localhost:27017
  - admin
  - config
  - emojiDB
  - employeeDB
  - local
  - recipe\_app
    - favorites
    - users

Welcome users stories favorites recipe\_app +

localhost:27017 > recipe\_app > favorites

Documents 0 Aggregations Schema Indexes 3 Validation

Type a query: { field: 'value' } or [Generate query](#) [Explain](#) [Reset](#) [Find](#) [Options](#)

[ADD DATA](#) [EXPORT DATA](#) [UPDATE](#) [DELETE](#)

25 1-2 of 2

```
_id: ObjectId('68a9ac42861d882ad6ab0c476')
user: ObjectId('68a981a315c9656eeb08de0e0')
mealId: "52374"
name: "Beef and Mustard Pie"
thumb: "https://www.themeladb.com/images/media/meals/sytuqu1511553755.jpg"
createdAt: 2025-08-23T11:05:18.855+00:00
updatedAt: 2025-08-23T11:05:18.855+00:00
__v: 0
```

```
_id: ObjectId('68a9ac42861d882ad6ab0c4b4')
user: ObjectId('68a981a315c9656eeb08de0e0')
mealId: "52384"
name: "Bean & Sausage Hotpot"
thumb: "https://www.themeladb.com/images/media/meals/vxuuyrx1511302687.jpg"
createdAt: 2025-08-23T11:55:46.680+00:00
updatedAt: 2025-08-23T11:55:46.680+00:00
__v: 0
```

MongoDB Compass - localhost:27017/recipe\_app.users

Connections Edit View Collection Help

**Compass**

My Queries

**CONNECTIONS (1)**

- localhost:27017
  - admin
  - config
  - emojiDB
  - employeeDB
  - local
  - recipe\_app
    - favorites
    - users

Welcome users stories favorites recipe\_app +

localhost:27017 > recipe\_app > users

Documents 0 Aggregations Schema Indexes 1 Validation

Type a query: { field: 'value' } or [Generate query](#) [Explain](#) [Reset](#) [Find](#) [Options](#)

[ADD DATA](#) [EXPORT DATA](#) [UPDATE](#) [DELETE](#)

25 1-4 of 4

```
_id: ObjectId('68a981a315c9656eeb08de0e0')
name: "John Doe"
email: "john@example.com"
password: "$2a$10$1Suq31Xe9bfDvrWmBaxuoRewrutBjkuTck7hhDQFxZhjkFb47G"
createdAt: 2025-08-23T08:53:15.843+00:00
updatedAt: 2025-08-23T08:53:15.843+00:00
__v: 0
```

```
_id: ObjectId('68a98bc015c9656eeb08de17')
name: "Manisha"
email: "manu@example.com"
password: "$2a$10$1Suq31Xe9bfDvrWmBaxuoRewrutBjkuTck7hhDQFxZhjkFb47G"
createdAt: 2025-08-23T09:37:04.559+00:00
updatedAt: 2025-08-23T09:37:04.559+00:00
__v: 0
```

```
_id: ObjectId('68a9901f5f736128119c3c5f')
name: "Samon"
email: "samon@example.com"
password: "$2a$10$0Kn2wZdlAC0vW6mLrAgJ.q7RsQNGyKdjBcnLzbxEtyVL4U.c506u"
createdAt: 2025-08-23T09:55:43.916+00:00
updatedAt: 2025-08-23T09:55:43.916+00:00
__v: 0
```

The screenshot shows the Postman application interface. In the top navigation bar, 'Home' and 'Workspaces' are selected. The main workspace is titled 'My Workspace'. A collection named 'New Collection' is currently selected. A request card is displayed for a POST operation to 'http://localhost:5000/api/auth/register'. The 'Body' tab is active, showing a raw JSON payload:

```
1 {  
2   "name": "John Doe",  
3   "email": "john@example.com",  
4   "password": "123456"  
5 }  
6
```

The response status is '200 OK' with a response time of 810 ms and a size of 1.31 KB. The response body is shown in JSON format:

```
1 {  
2   "user": {  
3     "id": "68a981a315c9656eeb88de00",  
4     "name": "John Doe",  
5     "email": "john@example.com"  
6   }  
7 }
```

The screenshot shows the Postman application interface. In the top navigation bar, 'Home' and 'Workspaces' are selected. The main workspace is titled 'My Workspace'. A collection named 'New Collection' is currently selected. A request card is displayed for a POST operation to 'http://localhost:5000/api/auth/login'. The 'Body' tab is active, showing a raw JSON payload:

```
1 {  
2   "email": "john@example.com",  
3   "password": "123456"  
4 }  
5
```

The response status is '200 OK' with a response time of 110 ms and a size of 1.31 KB. The response body is shown in JSON format:

```
1 {  
2   "user": {  
3     "id": "68a981a315c9656eeb88de00",  
4     "name": "John Doe",  
5     "email": "john@example.com"  
6   }  
7 }
```