COOKBOOK: YOUR VIRTUAL KITCHEN ASSISTANT

By

U. Mukesh - AP23110010175

Vaishnavi - AP23110010047

Kiran - AP23110010256

Ramya Sri - AP23110010404

SUBMITTED TO:

Himanshu Mishra

SRM UNIVERSITY, AP

(DECEMBER, 2025)

Introduction: -

Cooking is a universal human experience, but navigating the modern digital space, one finds it increasingly hard to access clear, reliable, and well-structured recipes. At a time when innumerable websites and video portals exist, for users, these issues persist: inconsistent formats of recipes, annoying ads, fuzzy instructions, and a lack of standards on saving or modifying preferred dishes. The modern cook would also want to have at their fingertips other useful features that make the cooking process easier and more enjoyable, such as checklists for ingredients, cooking timers, and multilingual support.

The growth of web applications has opened ways to easily streamline how individuals deal with everyday tasks. Among these, digital cooking assistants take precedence. However, most of the existing platforms either focus solely on recipe listing or deliver cluttered and ad-filled experiences that disrupt the user's flow. Very few platforms actually combine search, structured presentation, personalization, and interactive utilities into one smooth interface.

To overcome these limitations, Cookbook has been developed as a user-friendly and feature-rich virtual kitchen assistant. Built using modern frontend technologies, the platform offers a clean and organized recipe exploration experience, powered by structured data. It also includes smart features like voice search, multilingual translation, ingredient checklists, cooking timers, theme switching, and even uploading one's own recipe. Next to browsing and discovery, users can save their favourite recipes, create their own dishes, and maintain their very own digital cookbook.

This project shows how considerate UI/UX design and lightweight, modern technologies can change the usual act of cooking into one that is guided, interactive, and engaging. Cookbook lays the foundation for a digital cooking ecosystem that will scale into a comprehensive meal-planning, nutritional-tracking, community-driven platform in the future.

Scenario-Based Intro: -

Imagine someone stepping into their kitchen, clueless about what to make for dinner. Rather than surfing through various websites, watching long videos, or flipping through some recipe book, they open the Cookbook app. Within a few clicks, they can search for recipes based on ingredients they already have at home, such as tomatoes, pasta, or paneer, and instantly view step-by-step cooking instructions, preparation time, and ingredient lists.

Meanwhile, the user will be able to save all their favourite dishes, create custom recipes, and organize everything into a personal digital cookbook. Features like voice search make it easy to find meals hands-free, while checklists and cooking timers guide them step-by-step through the cooking process. If they prefer another language, the built-in translation tool helps them view recipes in their preferred choice.

Meanwhile, food recipe creators or admins can upload new dishes, update the existing recipes, and manage the content with ease via the platform.

This Cookbook System simulates a real-world cooking assistant and provides an integrated electronic solution for finding, organizing, and making recipes easily.

Target Audience: -

Home Cooks & Food Lovers

- People who want quick, reliable, easy-to-follow recipes and love exploring new dishes.
- Users who need a personal space to save, organize, and revisit their favourite recipes.

Busy Students & Working Professionals

- Individuals searching for quick meal ideas using ingredients they already have.
- People who prefer guided cooking with timers, checklists, and simple step-by-step help.

Content Creators & Recipe Contributors

- Home chefs wanting to upload, preserve, and share their own unique recipes.
- Users helping build a growing community-driven digital cookbook.

Developers, Learners & Tech Enthusiasts

- Students learning modern web tech through real-world projects like Cookbook.
- Developers exploring UI/UX, API integration, and interactive recipe app features.

Project Goals and Objectives: -

The Cookbook platform demonstrates complete frontend fundamentals including:
• Recipe browsing and filtering
• Form validation for adding custom recipes
• API integration for fetching recipe data
• JSON-based data storage and management
• Clean and intuitive UI/UX for seamless cooking flow Authentication

Modern Tech Stack

• Built using React.js for fast and dynamic UI
• Styled with CSS / Tailwind CSS for responsive design
• JSON or API-based recipe data handling
• Optional backend for recipe uploads and user storage
• Fully responsive interface for mobile, tablet, and desktop

Seamless Cooking Workflow

• Users can search recipes by ingredient, cuisine, or name
• View structured details: ingredients, steps, and cook time
• Save & organize favourite recipes
• Create and upload custom recipes
• Use interactive tools like checklists, timers, and voice search

Key Features:-

Interactive Recipe Interface

• Smooth and dynamic UI powered by React.js components.
• Instant updates and real-time rendering of recipe details, filters, and saved items.

Recipe Search & Filtering

• Search recipes by ingredients, cuisine, meal type, or keywords.
• Filter results instantly using React state management.

User-Friendly Recipe Display

• Structured layout showing ingredients, cooking steps, images, and preparation time.
• Clear, distraction-free UI for hassle-free cooking.

Custom Recipe Creation

• Users can upload their own recipes with title, ingredients, and instructions.
• Form validation ensures clean and accurate recipe submissions.

Save & Manage Favorites

• Users can bookmark recipes to build their personal cookbook.
• Favorites stored using JSON, localStorage, or optional backend integration.

Voice Search

• Hands-free searching for recipes while cooking.
• Faster access to dishes without typing.

Multilingual Translation

• Translate recipe steps and ingredients into supported languages.
• Makes the platform accessible to diverse users.

Ingredient Checklist & Cooking Timer

• Step-by-step cooking assistance with real-time timers.
• Checklists help users track ingredients during cooking.

Modern Tech Stack Integration

• React.js for component-based UI and state management.
• CSS / Tailwind CSS for responsive, clean design.
• APIs or JSON files for recipe data management.
• Optional backend for storing user-created recipes.

Responsive Design Testing

• Fully responsive layout working smoothly across mobiles, tablets, and desktops.
• Modern styling ensures consistent experience everywhere.

Developer-Friendly Environment

• Built using VS Code + React DevTools + npm ecosystem.
• JSON Server + React is perfect for fast prototyping and learning full-stack workflows.

PRE-REQUISITES: -

Before running the **Cookbook** Recipe Application, the following tools, software, and technical knowledge are required. Since your project uses **React.js** for the frontend, **Fetch API** for external data retrieval, and **JSON Server** for storing custom recipes, the setup remains lightweight and beginner-friendly without requiring SQL databases or heavy backend frameworks.

Code Editor (VS Code Recommended)

A professional code editor is needed to write and manage your project files.
• **Visual Studio Code** is ideal because it supports JSX, modern JavaScript, React components, and JSON formatting.
• Useful extensions include **Prettier**, **ESLint**, **React Snippets**, and **Live Server** for quick testing.

Web Browser (Chrome / Firefox)

A modern browser is required to run and debug your React-based Cookbook application.
• **Google Chrome** is preferred due to strong DevTools and the **React Developer Tools** extension.
• Firefox/Edge can be used for cross-browser checking and UI testing.

Node.js & npm (Required for React)

React applications depend on Node.js to run the development environment.
• Node.js provides the JavaScript runtime for executing React scripts.
• npm installs required packages such as:

- React
- Fetch API
- JSON Server

You **cannot** run a React project without Node.js and npm.

React.js Framework Knowledge

Since your Cookbook app is built with React, developers should understand:
• Component creation (Functional Components + JSX)
• Props and State
• Hooks like **useState** and **useEffect**
• Routing with **React Router** (if used for recipe pages)
• Event handling and dynamic rendering
• Fetch API integration inside useEffect

This knowledge ensures smooth building of interactive recipe pages and state-driven UI updates.

Fetch API (TheMealDB) Knowledge

Your project fetches live recipe data using the **TheMealDB API**, so developers must know:
• How to make GET requests using fetch()
• Handling JSON responses
• Displaying API data dynamically (images, ingredients, steps)
• Searching meals by name or ingredient using API endpoints
• Error handling (empty results, network issues)

JSON Server (Local Mini-Backend for Custom Recipes)

For storing user-generated recipes locally, JSON Server is used. Developers should know:
• Starting a JSON Server using npx json-server --watch db.json
• Performing CRUD operations using Fetch API

- GET → load all custom recipes

- POST → add new recipes
- PUT/PATCH → update recipes
- DELETE → remove recipes
  • JSON structure for recipe storage
  • How JSON acts as a mini-database for the project

Local Development Server (React Scripts)

React provides an automatic development server using:

npm start

No Apache, PHP, or MySQL is required.
If backend simulation is needed, you may use:
• **Json-server** (recommended)
• Optional: custom Node.js API
• Or static JSON files for simple use

Basic HTML, CSS, and JavaScript Skills

Even though React handles most of the UI, developers must know:
• Writing HTML-like structure using JSX
• Styling using CSS, modules, or Tailwind
• JavaScript ES6+ features such as:

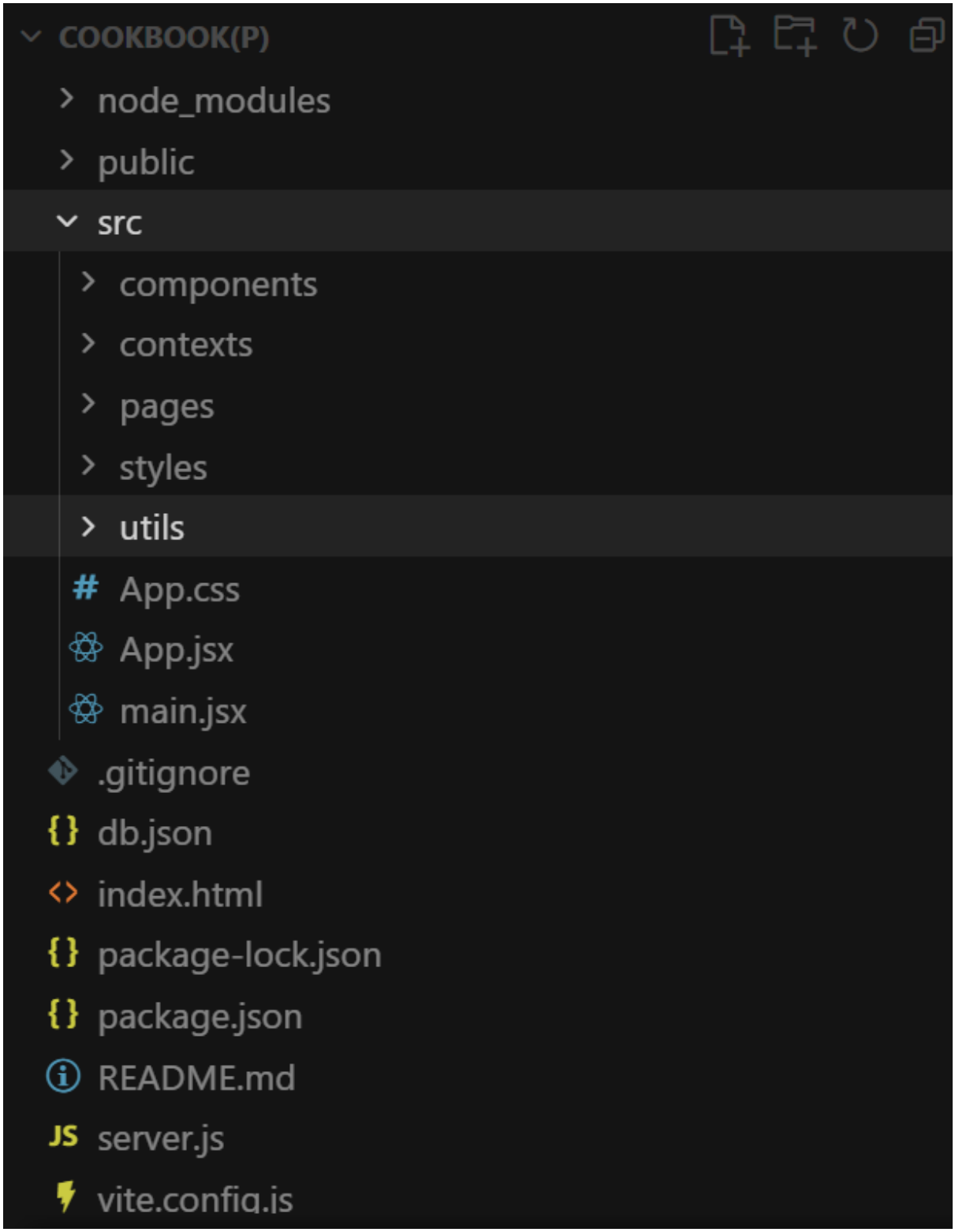
- Arrow functions
- Promises
- async/await
- Array methods (map, filter, reduce)

These skills are essential for managing recipe lists, search functionality, form validation, and UI updates.

System Requirements

To run the Cookbook system smoothly:
• Minimum **4GB RAM** (8GB recommended)
• Dual-core processor
• 5GB free disk space
• Stable internet connection for installing npm packages and fetching recipes from TheMealDB API

Project structure:



The project contains:

• **Frontend (React.js)**

Contains all UI elements and screens.

- **Components:** Navbar, Footer, Recipe Cards, Search Bar, Voice Search, Timer, Checklist, Theme Toggle, etc.
- **Pages:** Home, About, Community, MyCookBook, UploadRecipe, EditRecipe, Signup, Login.
- **Contexts:** Global states like language and theme.
- **Styles:** Global CSS and theme files.
- **Utils:** API handlers, localStorage utilities.

• **Backend (JSON Server)**

- **db.json** stores user-created and community recipes.
- JSON Server handles CRUD operations for recipes without needing MySQL or PHP.

• **Fetch API (TheMealDB)**

- Uses **Fetch API** to load thousands of recipes from TheMealDB.
- Supports search by dish, ingredient, or category.

• **Assets (Media Files)**

- Includes images, icons, and graphics used in UI components.

• **Configuration Files**

- **package.json:** Project dependencies and scripts.
- **vite.config.js:** Vite build configuration.
- **db.json:** Local database.
- **. gitignore:** Ignored files.
- **index.html:** Root file for React.
- **server.js:** JSON Server setup.

PROJECT FLOW: -

Project demo:

Before starting to work on this project, let's see the demo.

Demonstration:

https://drive.google.com/file/d/1hzNLxNGcZU-7w15evFJ2N2IEI7ItsBEr/view?usp=drive_link

Code Explanation video:

https://drive.google.com/file/d/1uNS7Dea5my1eeI6EKSurbUFEHE323McS/view?usp=drive_link

App.js component

```jsx
import React, { useEffect, useState } from 'react';
import { BrowserRouter as Router, Routes, Route, useNavigate } from 'react-router-dom';
import { initTheme } from './utils/theme';
import { TranslationProvider } from './contexts/TranslationContext';
import Navbar from './components/Navbar';
import Footer from './components/Footer';
import Home from './pages/Home';
import MyCookbook from './pages/MyCookbook';
import UploadRecipe from './pages/UploadRecipe';
import Community from './pages/Community';
import EditRecipe from './pages/EditRecipe';
import About from './pages/About';
import SignIn from './pages/SignIn';
import SignUp from './pages/SignUp';
import RecipeDetails from './components/RecipeDetails';
import './App.css';

const AppContent = () => {
  const navigate = useNavigate();
  const [searchQuery, setSearchQuery] = useState('');

  useEffect(() => {
    initTheme();
  }, []);

  const handleVoiceSearch = (query) => {
    setSearchQuery(query);
    navigate('/');
    // Trigger search after navigation
    setTimeout(() => {
      const event = new CustomEvent('voiceSearch', { detail: query });
      window.dispatchEvent(event);
    }, 100);
  };
```

```
  return (
    <div className="app">
      <Navbar onVoiceSearch={handleVoiceSearch} />
      <main className="main-content">
        <Routes>
          <Route path="/" element={<Home />} />
          <Route path="/my-cookbook" element={<MyCookbook />} />
          <Route path="/upload" element={<UploadRecipe />} />
          <Route path="/community" element={<Community />} />
          <Route path="/community/edit/:id" element={<EditRecipe />} />
          <Route path="/about" element={<About />} />
          <Route path="/signin" element={<SignIn />} />
          <Route path="/signup" element={<SignUp />} />
          <Route path="/recipe/:id" element={<RecipeDetails />} />
        </Routes>
      </main>
      <Footer />
    </div>
  );
};

const App = () => {
  return (
    <TranslationProvider>
      <Router>
        <AppContent />
      </Router>
    </TranslationProvider>
  );
};

export default App;
```

Code Description: -

Imports React: Required to create components and use React features like useState and useEffect.

Imports all components & pages:

These are different sections/screens of your CookBook website:

Components:

• Navbar
• Footer
• RecipeDetails

Pages:

• Home
• MyCookbook
• UploadRecipe
• Community
• EditRecipe
• About
• SignIn
• SignUp

Each one is a separate UI section of your app.

Defines AppContent Component :

AppContent handles:
• Page routing
• Voice search
• Theme loading
• Navigation

This is the core functional block of your UI.

useEffect :

Runs only once to apply the website theme:

useEffect(() => {

initTheme();}, []);

Main content area using Routes:

Each <Route /> loads a different page based on the URL.

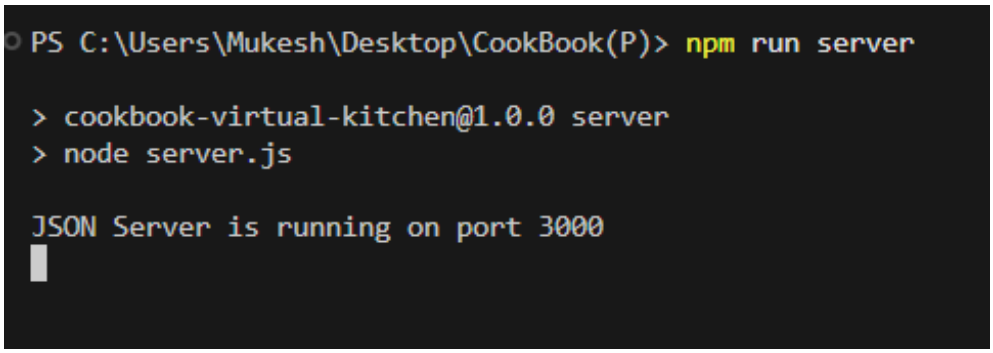
• "/" → Home
• "/my-cookbook" → My saved recipes

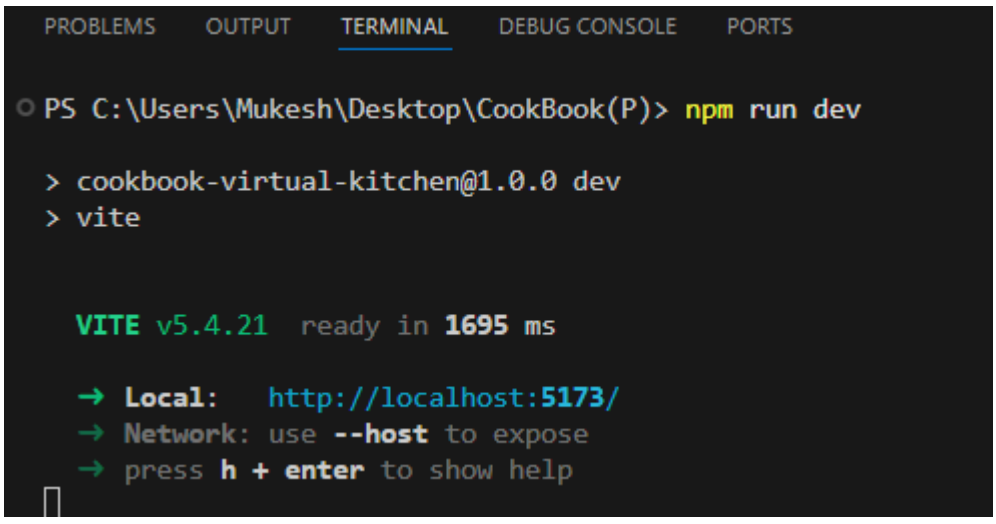
- "/upload" → Upload new recipe
- "/community" → Community recipes
- "/community/edit/:id" → Edit recipe with dynamic ID
- "/about" → About Page
- "/signin" → Login page
- "/signup" → Signup page
- "/recipe/:id" → Individual recipe details

This is how your entire website switches pages.

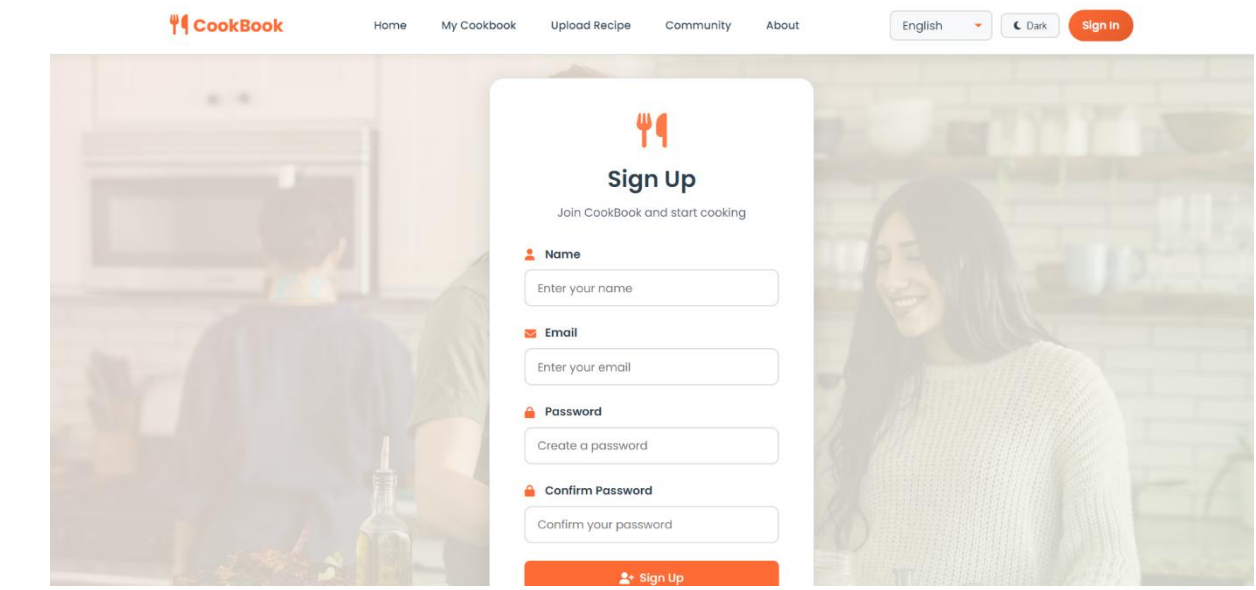
Project Execution:

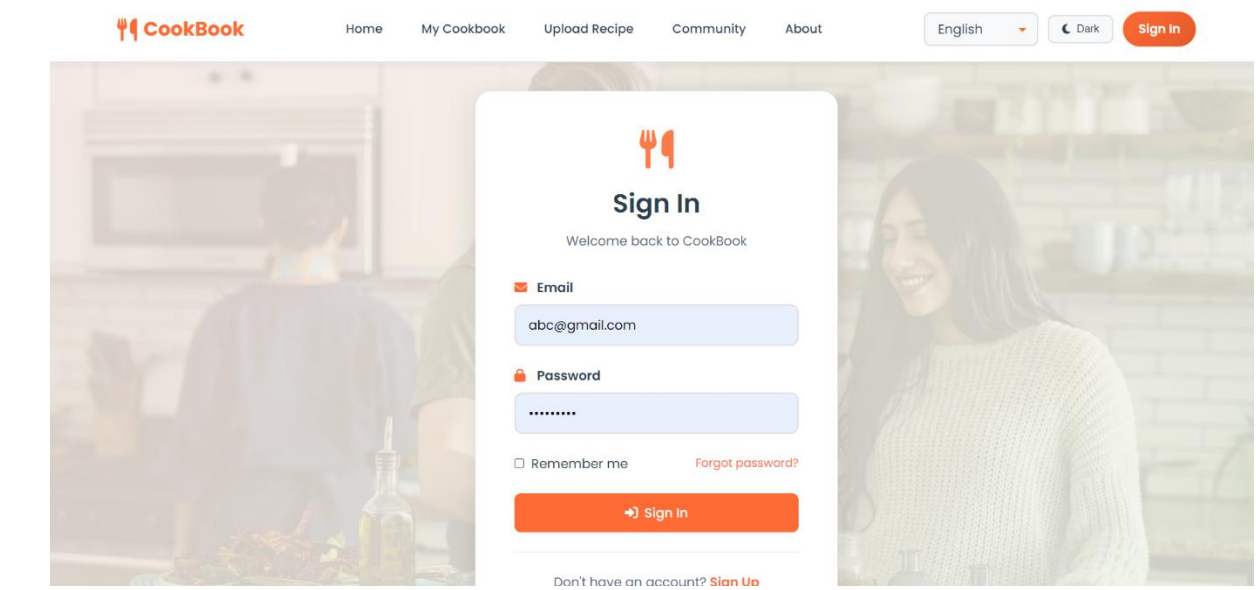
After completing the code, run your CookBook JSON server using “npm run server” and start the frontend using “npm run dev”. These commands will launch your JSON Server API and your Vite React application.

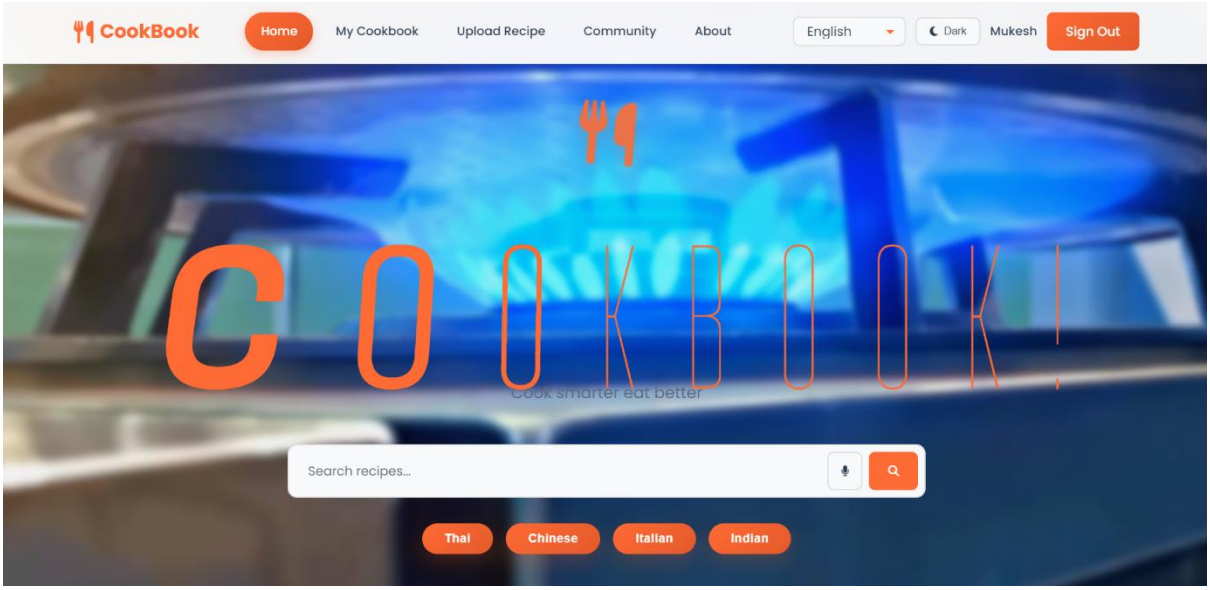


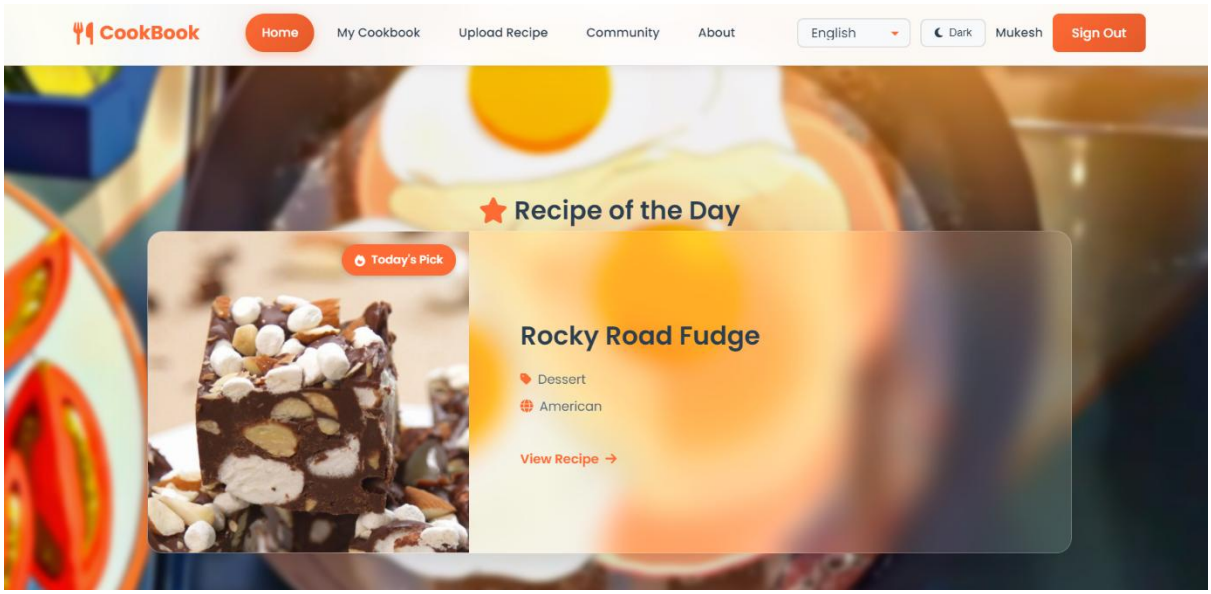


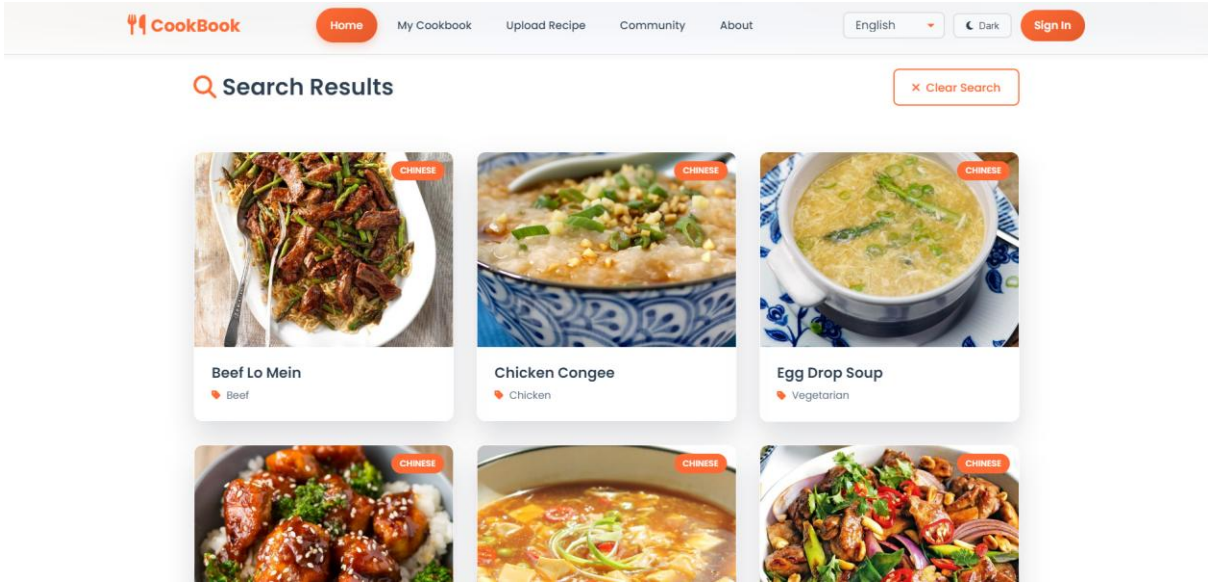
- **Output**

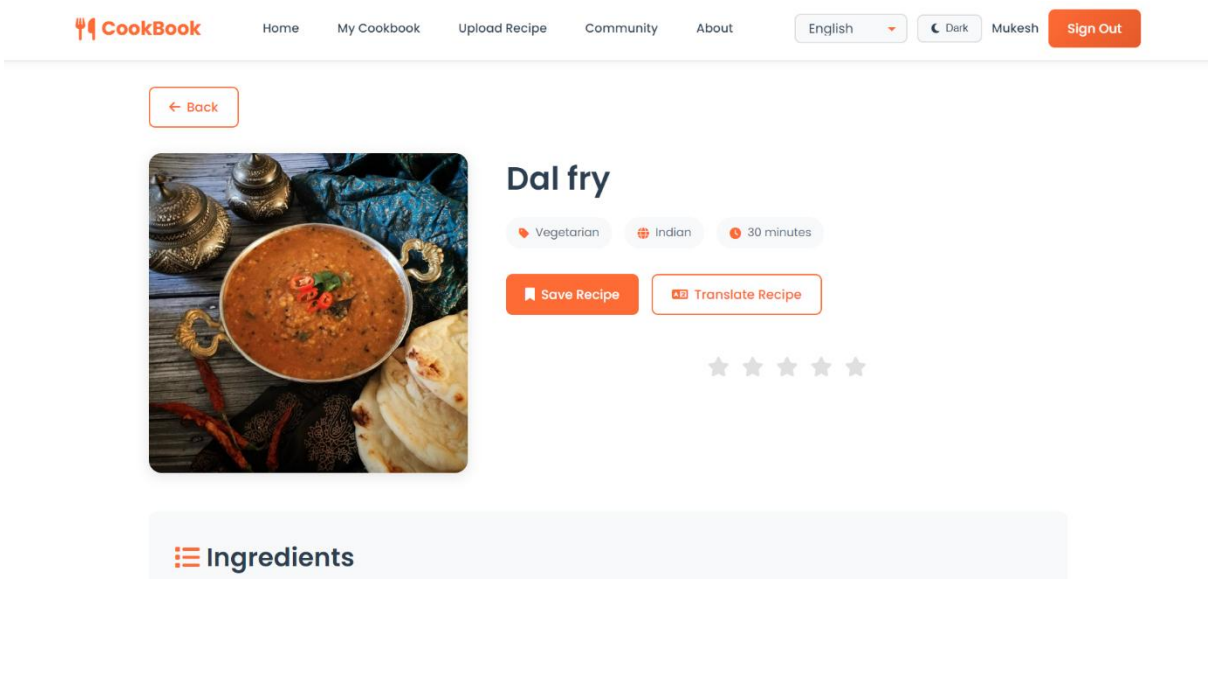

CookBook
Home
My Cookbook
Upload Recipe
Community
About
English
Dark
Sign In
Sign Up
Join CookBook and start cooking
Name
Enter your name
Email
Enter your email
Password
Create a password
Confirm Password
Confirm your password
Sign Up

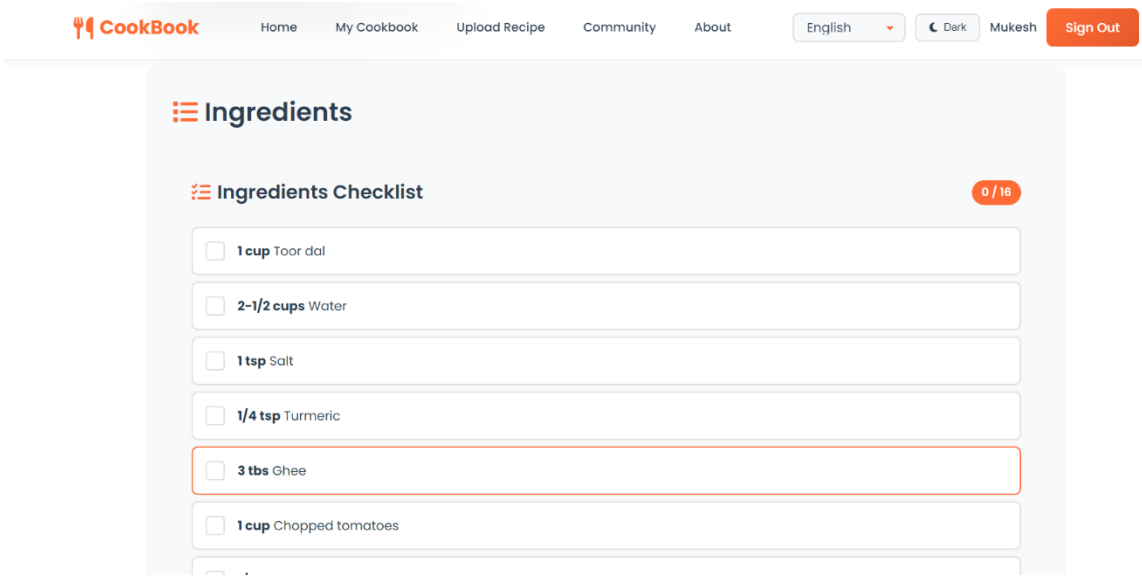

CookBook
Home
My Cookbook
Upload Recipe
Community
About
English
Dark
Sign In
Sign In
Welcome back to CookBook
Email
abc@gmail.com
Password
Remember me
Forgot password?
Sign In
Don't have an account? Sign Up

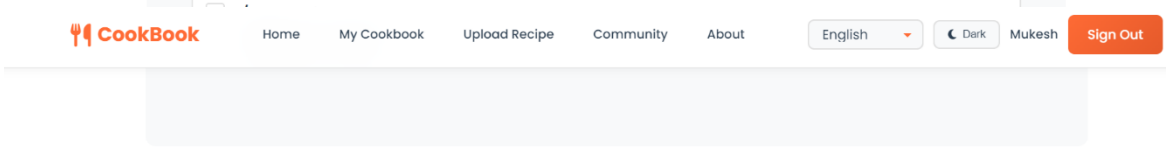

CookBook
Home
My Cookbook
Upload Recipe
Community
About
English
Dark
Mukesh
Sign Out
COOKBOOK!
Cook smarter eat better
Search recipes...
Thai
Chinese
Italian
Indian

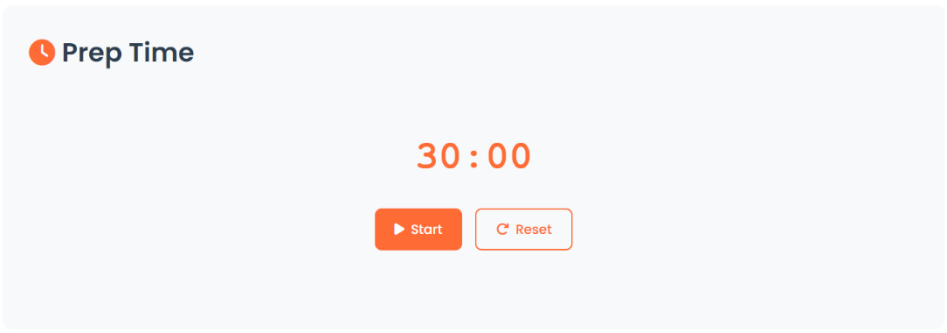

CookBook
Home
My Cookbook
Upload Recipe
Community
About
English
Dark
Mukesh
Sign Out
Recipe of the Day
Today's Pick
Rocky Road Fudge
Dessert
American
View Recipe


CookBook
Home
My Cookbook
Upload Recipe
Community
About
English
Dark
Sign In
Search Results
Clear Search
CHINESE
Beef Lo Mein
Beef
Chicken Congee
Chicken
Egg Drop Soup
Vegetarian


CookBook
Home
My Cookbook
Upload Recipe
Community
About
English
Dark
Mukesh
Sign Out
Back
Dal fry
Vegetarian
Indian
30 minutes
Save Recipe
Translate Recipe
Ingredients

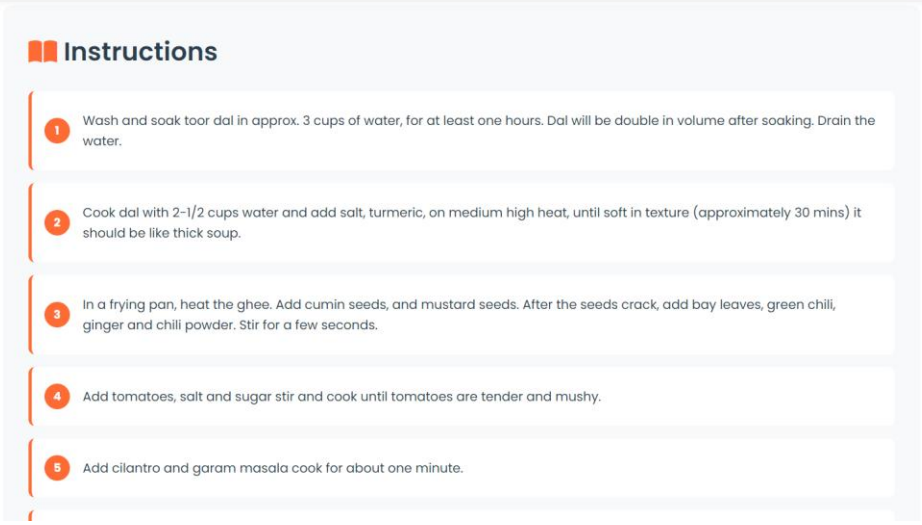

CookBook
Home
My Cookbook
Upload Recipe
Community
About
English
Dark
Mukesh
Sign Out
Ingredients
Ingredients Checklist
0 / 16
1 cup Toor dal
2-1/2 cups Water
1 tsp Salt
1/4 tsp Turmeric
3 tbs Ghee
1 cup Chopped tomatoes

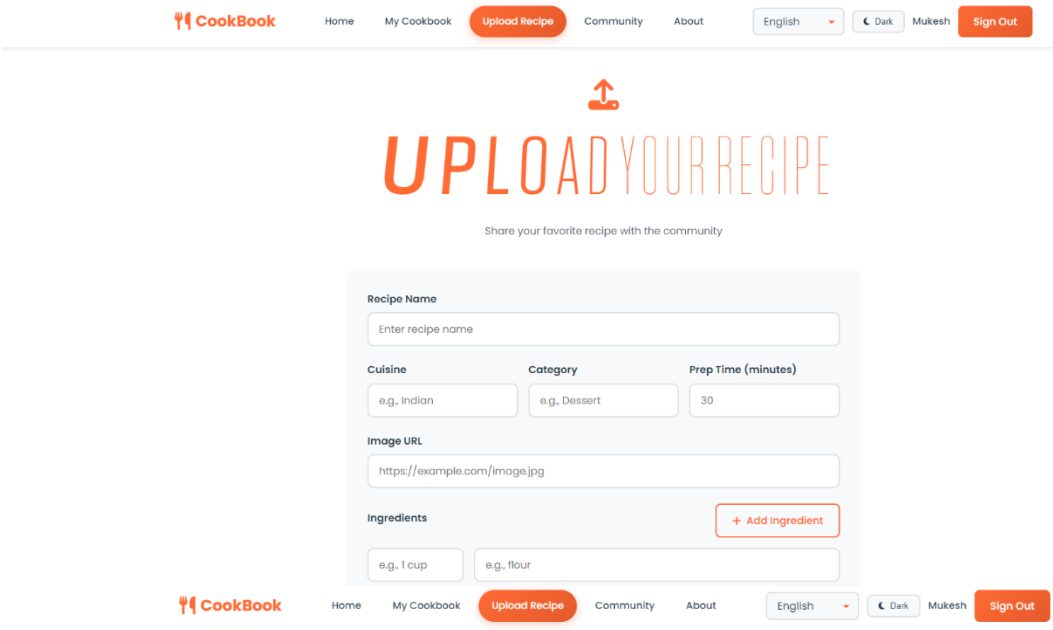

CookBook
Home
My Cookbook
Upload Recipe
Community
About
English
Dark
Mukesh
Sign Out

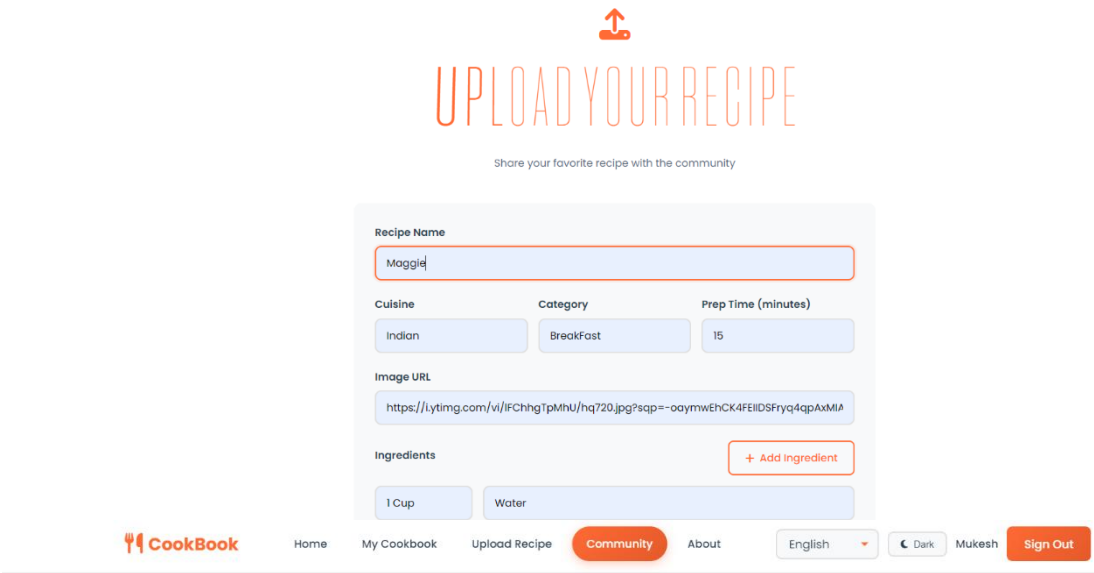

Prep Time
30 : 00
Start
Reset

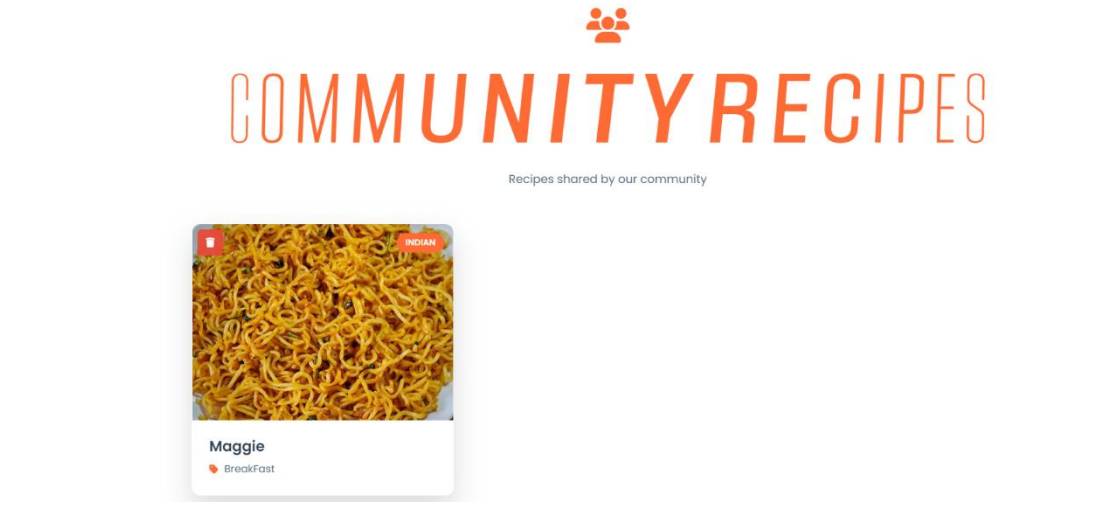

Instructions

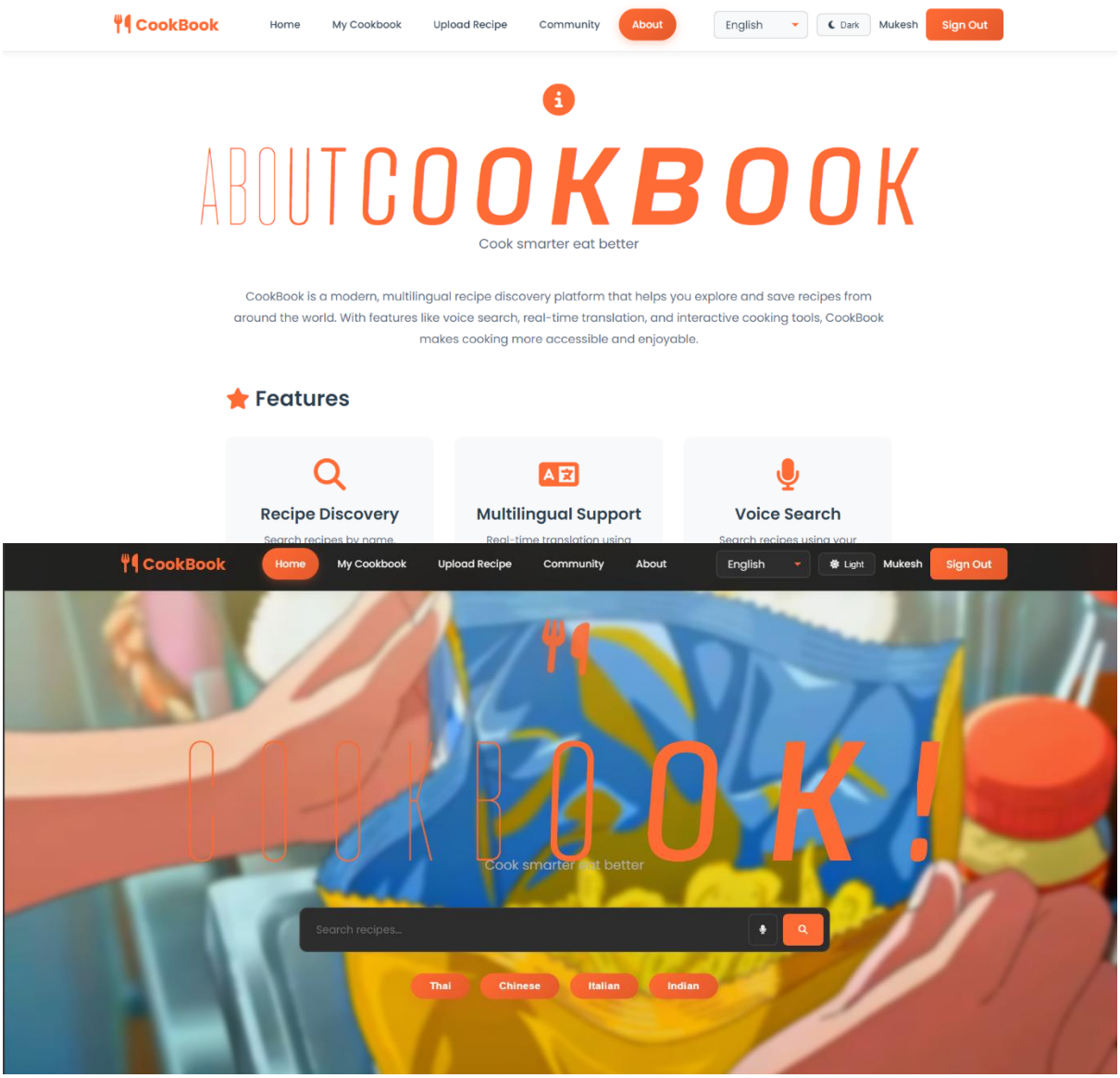

CookBook
Home
My Cookbook
Upload Recipe
Community
About
English
Dark
Mukesh
Sign Out


Instructions
1
Wash and soak toor dal in approx. 3 cups of water, for at least one hours. Dal will be double in volume after soaking. Drain the water.
2
Cook dal with 2-1/2 cups water and add salt, turmeric, on medium high heat, until soft in texture (approximately 30 mins) it should be like thick soup.
3
In a frying pan, heat the ghee. Add cumin seeds, and mustard seeds. After the seeds crack, add bay leaves, green chili, ginger and chili powder. Stir for a few seconds.
4
Add tomatoes, salt and sugar stir and cook until tomatoes are tender and mushy.
5
Add cilantro and garam masala cook for about one minute.


CookBook
Home
My Cookbook
Upload Recipe
Community
About
English
Dark
Mukesh
Sign Out
UPLOAD YOUR RECIPE
Share your favorite recipe with the community
Recipe Name
Enter recipe name
Cuisine
Category
Prep Time (minutes)
e.g., Indian
e.g., Dessert
30
Image URL
https://example.com/image.jpg
Ingredients
+ Add Ingredient
e.g., 1 cup
e.g., flour


CookBook
Home
My Cookbook
Upload Recipe
Community
About
English
Dark
Mukesh
Sign Out
UPLOAD YOUR RECIPE
Share your favorite recipe with the community
Recipe Name
Maggie
Cuisine
Category
Prep Time (minutes)
Indian
BreakFast
15
Image URL
https://i.ytimg.com/vi/IFChhgTpMhU/hq720.jpg?sqp=-oaymwEhCK4FEIIDSFryq4qpAxMI
Ingredients
+ Add Ingredient
1 Cup
Water


CookBook
Home
My Cookbook
Upload Recipe
Community
About
English
Dark
Mukesh
Sign Out
COMMUNITY RECIPES
Recipes shared by our community
INDIAN
Maggie
BreakFast



GitHub Link:

https://github.com/Mukesh258/CookBook

Project Demo link:

https://drive.google.com/drive/folders/12FI2FtWujsgsg405dhL3v1NbCXkkSzgE?usp=drive_link