

# **COOKBOOK: YOUR VIRTUAL KITCHEN ASSISTANT**

## **SRGSK Cookbook (React Application)**

### **TEAM DETAILS**

**Team ID: SWTID1741152458**

**Team Size: 5**

**Team Leader: SWETHA V-**  
venkatdhanam1972@gmail.com

### **Team members**

- 1.SWATHI V- swathivswathi66@gmail.com
- 2.GEETHAPRIYA S- geethuma0529@gmail.com
- 3.RIYA TIWARI - riyatiwari2325@gmail.com
- 4.KEERTHIKA V -  
keerthi4123@gmail.com

### **Introduction:**

SRGSK Cookbook is a revolutionary web application designed to change the way you discover, organize, and create recipes. It caters to both novice and professional chefs, offering a user-friendly interface, robust features, and a vast collection of inspiring recipes.

### **Description:**

Welcome to the forefront of culinary exploration with SRGSK Cookbook! Our cutting-edge web application is meticulously crafted to transcend the boundaries of culinary experiences, catering to the tastes of both passionate cooking enthusiasts, and seasoned professional chefs. With an emphasis on an intuitive user interface and a robust feature set, SRGSK Cookbook is poised to revolutionize the entire recipe discovery, organization, and creation process. Designed with a commitment to user-friendly aesthetics, SRGSK Cookbook immerses users in an unparalleled culinary adventure. Navigate seamlessly through a vast expanse of culinary inspiration with features such as dynamic search effortlessly.

From those taking their first steps in the kitchen to seasoned professionals, SRGSK Cookbook embraces a diverse audience, nurturing a dynamic community united by a shared passion for the art of cooking. Our vision is to reshape how users interact with recipes, presenting a platform that not only sparks inspiration but also fosters collaboration and sharing within the vibrant culinary community

## **Scenario based introduction:**

Sarah rummaged through the fridge, the fluorescent light casting an unappetizing glow on the wilting lettuce and forgotten container of yogurt. Dinnertime with her teenage son, Ethan, was fast approaching, and her usual creative

spark was missing. "What are we even going to eat?" Ethan groaned from the doorway, his phone glued to his ear.

Suddenly, a memory surfaced. Her friend, Maya, had been raving about a new recipe platform called SRGSK Cookbook **Technical Architecture:**

The user experience starts with the SRGSK Cookbook web application's UI, likely built with a framework like

React or Vue.js for a smooth, single-page experience. This UI interacts with an API client specifically

designed for SRGSK Cookbook. This client handles communication with the backend, but with a twist: it

leverages Rapid API, a platform providing access to various external APIs.



## **Project Goals and Objectives:**

The primary goal of SRGSK Cookbook is to provide a user-friendly platform that caters to individuals

passionate about cooking, baking, and exploring new culinary horizons. Our objectives include:

- **User-Friendly Experience:** Create an interface that is easy to navigate, ensuring users

can effortlessly discover, save, and share their favourite recipes.

- **Comprehensive Recipe Management:** Offer robust features for organizing and

managing recipes, including advanced search options.

- **Technology Stack:** Leverage modern web development technologies, including React.js,

to ensure an efficient, and enjoyable user experience.

## **Features of SRGSK Cookbook:**

✓ **Recipes from the MealsDB API:** Access a vast library of international recipes spanning diverse cuisines and dietary needs.

✓ **Visual recipe browsing:** Explore recipe categories and discover new dishes through curated image galleries.

## **PRE-REQUISITES:**

1. **Node.js and npm:**

<https://nodejs.org/en/download/>

## **Installation instructions:**

<https://nodejs.org/en/download/package-manager/> React.js:

\* Create a new React app

\* Navigate to the project directory

## 2. Running the React App

Start the development server:

**npm start**

This command launches the development server, and you can access your React app at <http://localhost:3000> in your web browser.

3. **HTML, CSS, and JavaScript:** Basic knowledge of HTML for creating the structure of

your app, CSS for styling, and JavaScript for client-side interactivity is essential.

4. **Development Environment:** Recommended editors include Visual Studio Code, Sublime Text, or WebStorm.

- **Visual Studio Code:** Download from <https://code.visualstudio.com/download>
- **Sublime Text:** Download from <https://www.sublimetext.com/download>
- **WebStorm:** Download from <https://www.jetbrains.com/webstorm/download>

Navigate to the project directory and install

dependencies: `cd news-app-react npm install`

Start the development server:

`npm start`

## **Project Milestone:**

### **Milestone 1: Project Setup & Configuration**

Install necessary tools including React.js, React Router DOM, Axios, and Tailwind CSS.

Set up the basic project structure.

Open the project folder to install necessary tools. In this project, we use:

- React.js
- React Router Dom
- React Icons
- Bootstrap/Tailwind CSS
- Axios

For further reference, use the following resources

<https://react.dev/learn/installation> <https://react-bootstrap-v4.netlify.app/gettingstarted/introduction/>  
<https://axios-http.com/docs/intro>  
<https://reactrouter.com/en/main/start/tutorial>

### **Milestone 2: Development Phase**

Implement routing for seamless navigation.

Develop essential components like the Navbar, Hero section, and popular categories.

Fetch trending recepies using an API and display it on the homepage.

Create category pages to display different foods based on selected topics.

## Fetching dishes via api:

### 1. Fetching all the available categories

Here, with the API request to Rapid API, we fetch all the available categories.

This code snippet demonstrates how to fetch data from an API and manage it within a React component. It leverages two key functionalities: state management and side effects.

```
const {id} = useParams();

const [items, setItems] = React.useState([])

useEffect(() => {
  fetchItems(id)
}, [window.location.href])

const fetchItems = async (idd) => {
  await axios.get(`https://www.themealdb.com/api/json/v1/1/filter.php?c=${idd}`)
    .then(response => {
      setItems(response.data.meals)
      console.log(response.data.meals)
    })
    .catch(error => console.error(error));
}
```

### 2. Fetching Recipe details

With the recipe id, we fetch the details of a certain recipe.

- It leverages the useState hook to establish a state variable named recipe (which is initially empty). This variable acts as a container to hold the fetched recipe data.

- The `useEffect` hook comes into play to execute a side effect, in this instance, fetching data from an API endpoint. The hook takes a callback function (`fetchRecipe` in this case) and an optional dependency array. The callback function is invoked after the component renders and whenever the dependencies in the array change. Here, the dependency array is left empty `[]`, signifying that the data fetching should occur only once after the component mounts.

```
const [id] = useParams();

const [recipe, setRecipe] = React.useState()

useEffect(() => {
  fetchRecipe()
}, [])

const fetchRecipe = async () => {
  await axios.get(`https://www.themealdb.com/api/json/v1/1/lookup.php?i=${id}`)
    .then(response => {
      setRecipe(response.data.meals[0])
      console.log(response.data.meals[0])
    })
    .catch(error => console.error(error));
}
```

## Project Execution

Once the development is complete, start the application using: `npm start`

This will launch the app at `http://localhost:3000`, displaying key components:



## **User interface Snips;**

- Hero Component
- Popular Categories
- Trending dishes
- News letter
- Categories dishes page
- Recipe page