

**Report of**

**Frontend Engineering**

**CS186**

**Project Name: Recipe Finder**

**Submitted By: Submitted To:**

Ankit Mishra

2110991966

Group- 19

## Department of Computer Science & Engineering Chitkara University Institute of Engineering & Technology, Rajpura, Punjab.

**Recipe Finder**

**Introduction:** The Recipe Finder Frontend Project is an exciting web application designed to help users discover, explore, and organize a wide variety of recipes. This project aims to make your cooking experience more enjoyable and convenient.

In this project, the frontend component is the user-facing interface of the Recipe Finder application.

The frontend is typically built using web technologies, such as HTML, CSS, and JavaScript, along with various web development frameworks and libraries.

**Objective:** The primary objective of the Recipe Finder frontend is to create a user-friendly and engaging interface that facilitates the efficient and enjoyable exploration of recipes.

It aims to make the process of meal planning and cooking more convenient and enjoyable while fostering a sense of community and exploration in the world of food.

**Technologies Used in this Project:** The Recipe Finder Frontend Project is typically built using a modern technology which may include

**HTML/CSS:** To create the structure and style of the web pages.

**JavaScript:** To add interactive and dynamic features to the application.

**Frontend Frameworks:** Such as React, API fetch to streamline development and enhance user experience.

**Responsive Design:** To ensure the application functions well on a range of devices.

## Flow Chart

**Start**

## 

## User arrives at Recipe Finder Frontend

## User can choose to:

## - Search for Recipes

## - Explore Recipe Collections

## If the user selects "Search for Recipes":

## User enters search criteria (e.g., keywords, ingredients, cuisine)

## Display search results with options to:

## - View a Recipe

## - Save a Recipe to Favorites

## If the user selects "View a Recipe":

## Display the selected recipe with details:

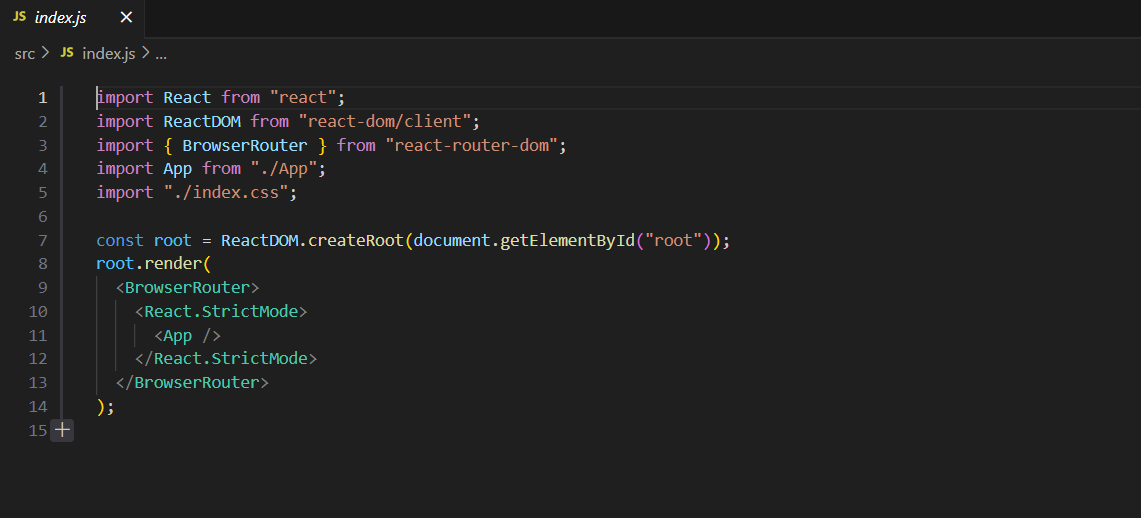
## - Ingredients

## - Preparation steps

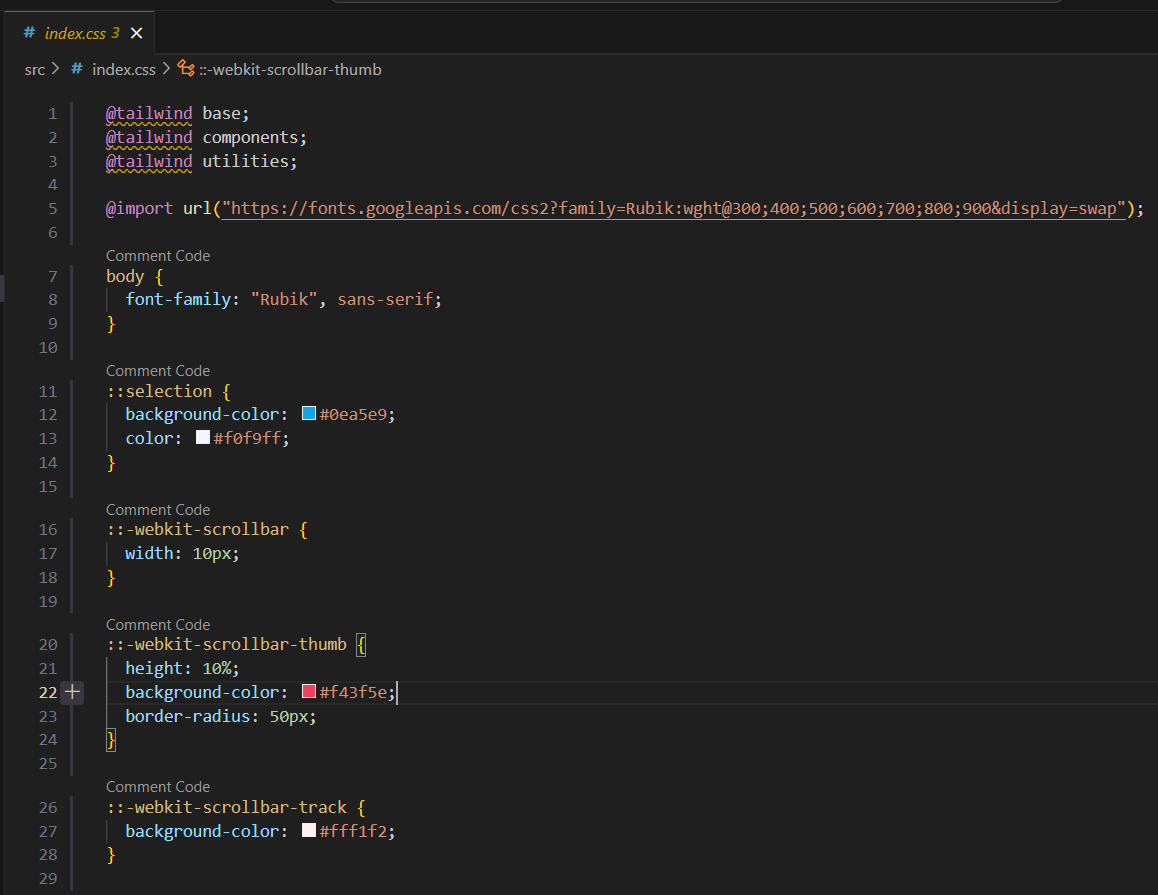
**- Cooking time**

## If the user selects "Save to Favorites" or "Add to Shopping List," the frontend updates user data.

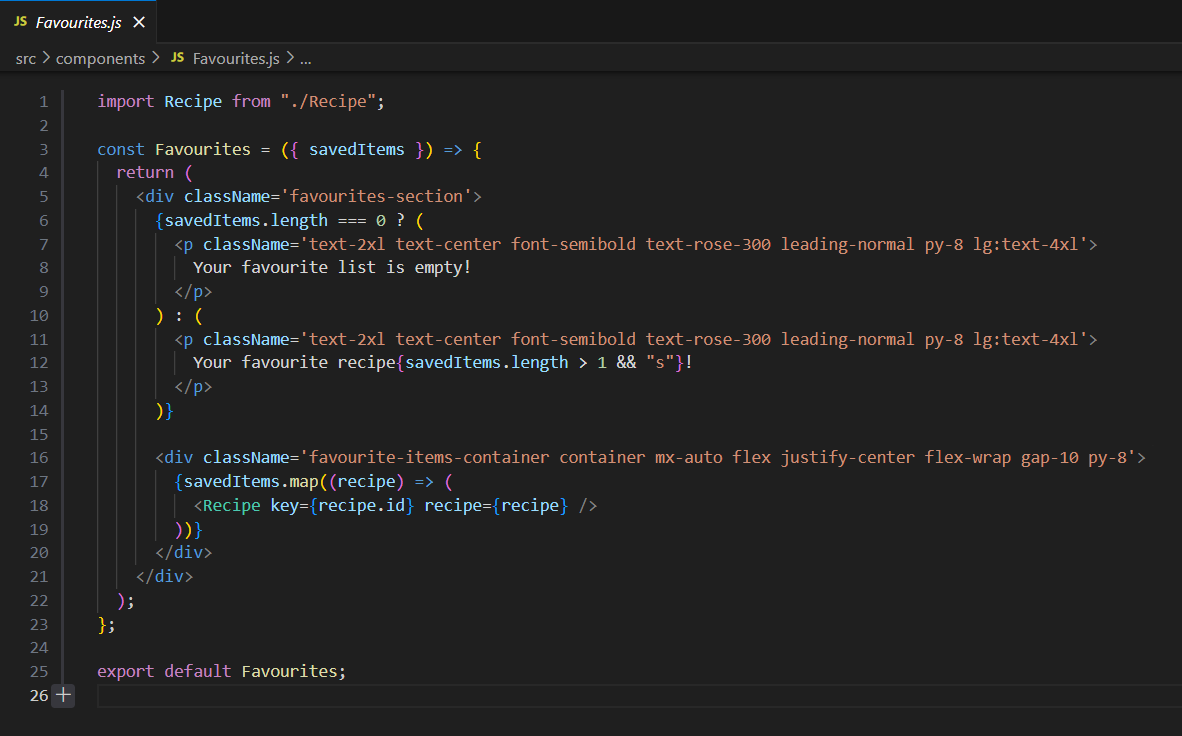
**End**



The entire code is setting up your React application, rendering it inside the HTML element with the ID "root," and wrapping it in a **BrowserRouter** for routing and a **React.StrictMode** for development mode debugging. When you run your application, the content of the **App** component will be displayed within the "root" element, and routing will be managed using **BrowserRouter**. The styles defined in the **index.css** file will be applied to the components in your application.



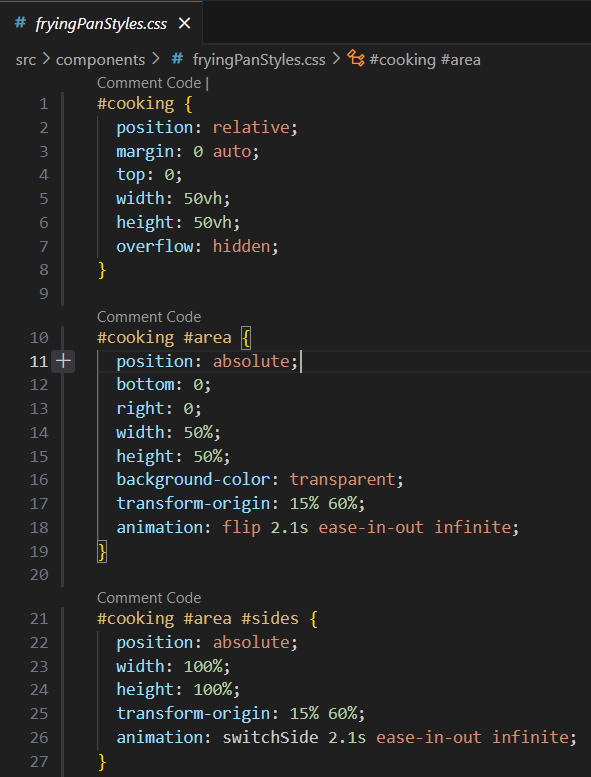
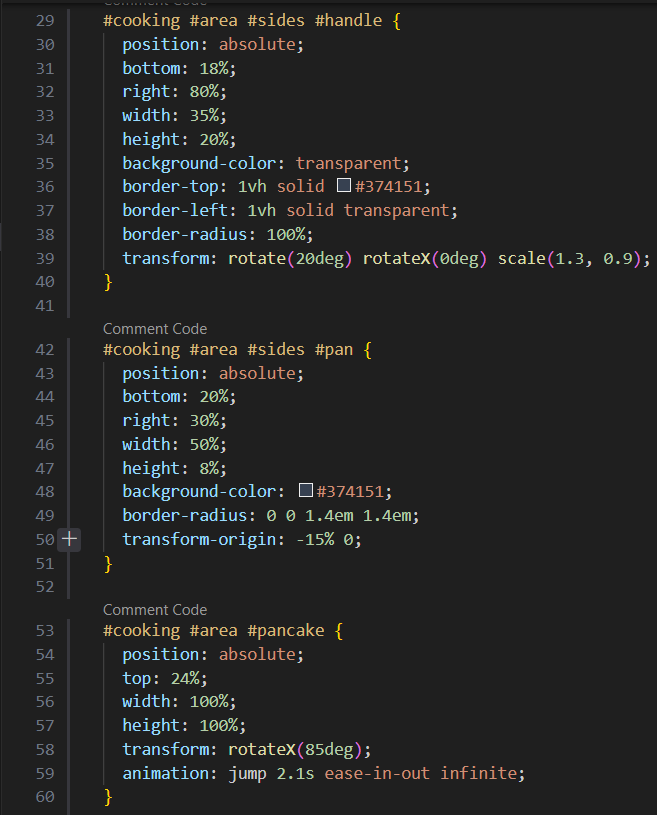
This code customizes your project's base styles, font selection, and scrollbar appearance to make it visually appealing and in line with your design preferences. It's important to note that the scrollbar styles (**::-webkit-scrollbar**) are specific to WebKit-based browsers and may not be visible in all browsers.

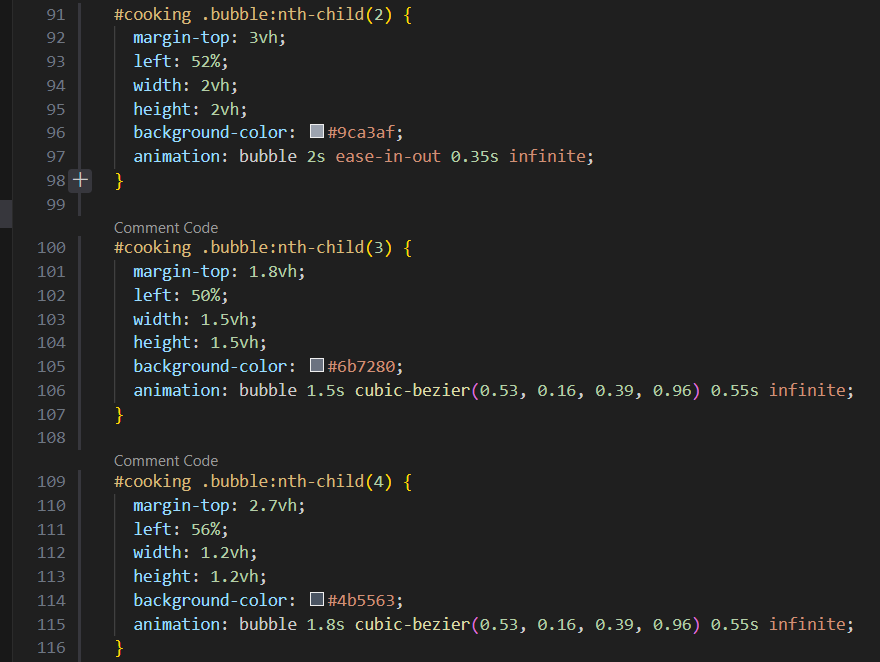


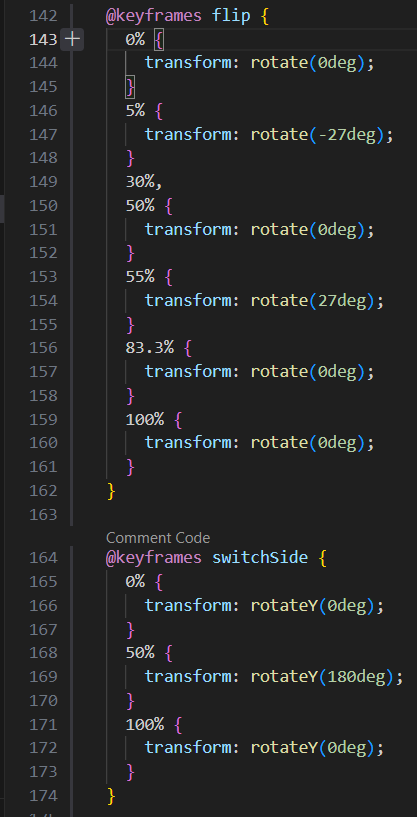
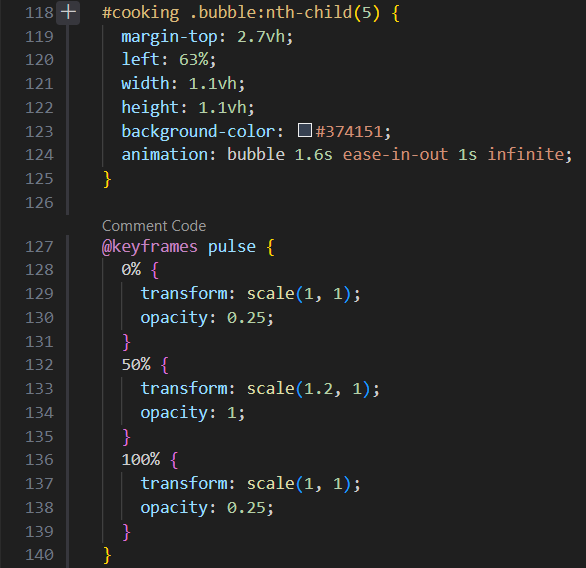
The **Favourites** component is responsible for rendering a section of your application that displays favorite recipes. It checks if there are saved items, adjusts the displayed message accordingly, and maps over the saved recipes to render each one using the **Recipe** component. This code is a part of a larger React application and can be used to create a user-friendly favorites section for a recipe-related website or app.

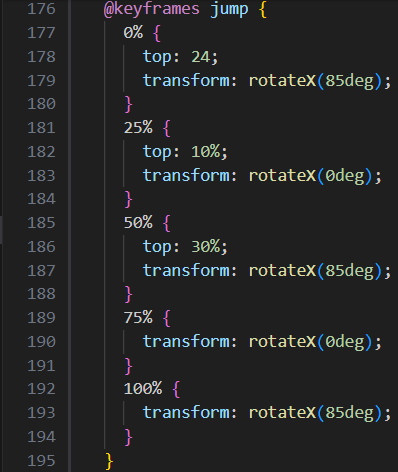
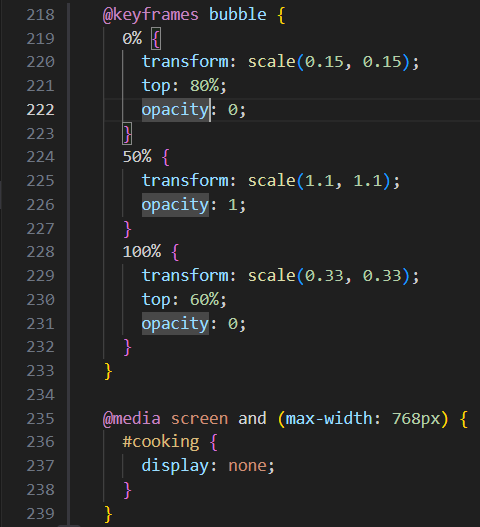


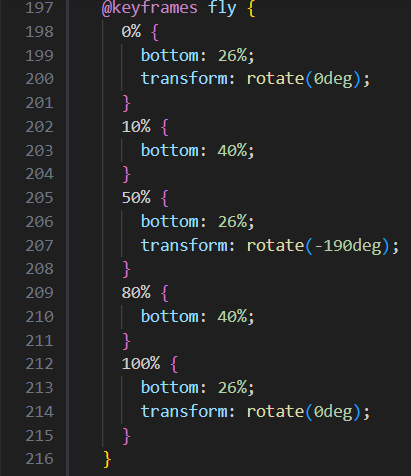
The main purpose of this component seems to be for visual representation or illustration. It's not interactive but could be used in a larger application to create a cooking-related visual element, possibly in the context of a cooking or recipe website or app. The specific styling is defined in the associated "fryingPanStyles.css" file, which determines the appearance of the frying pan and its elements.



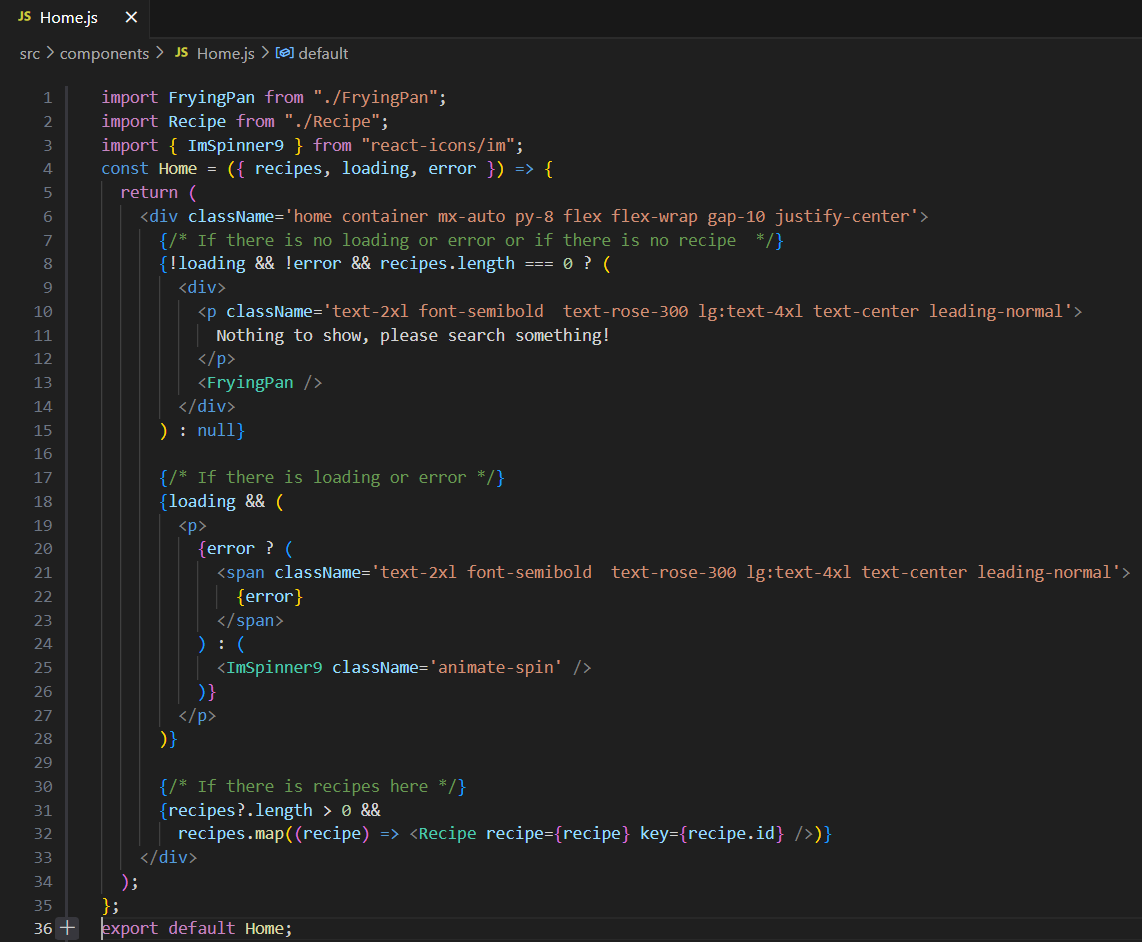




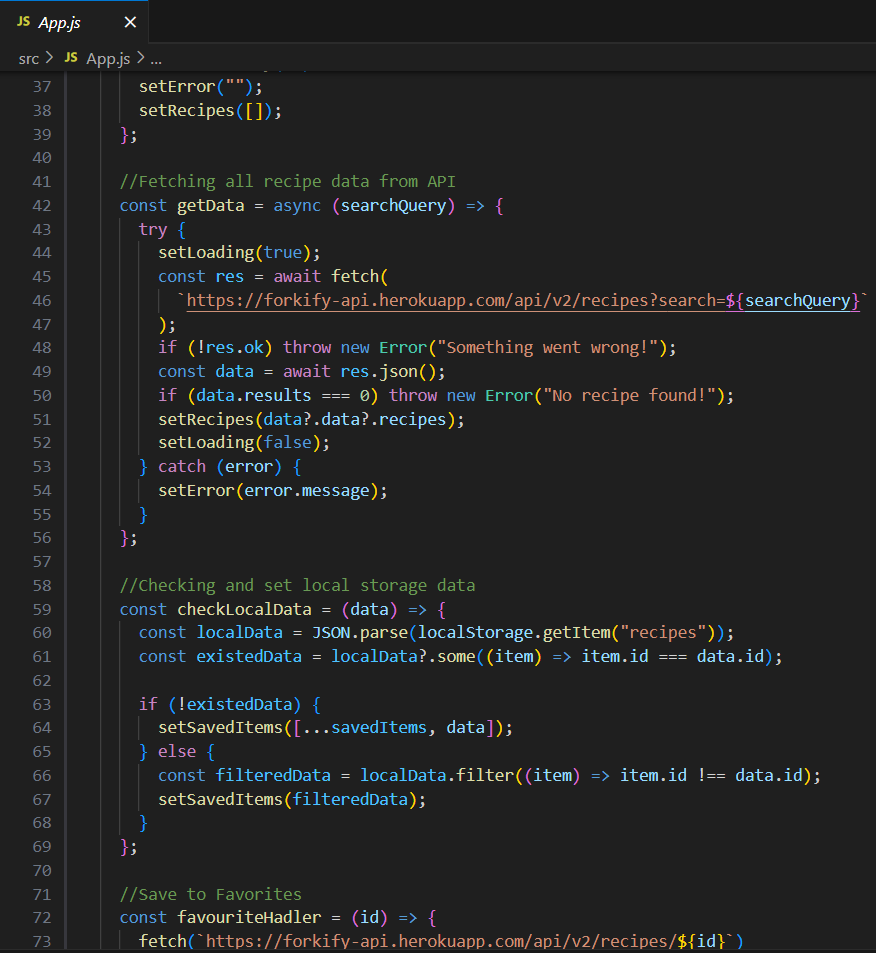
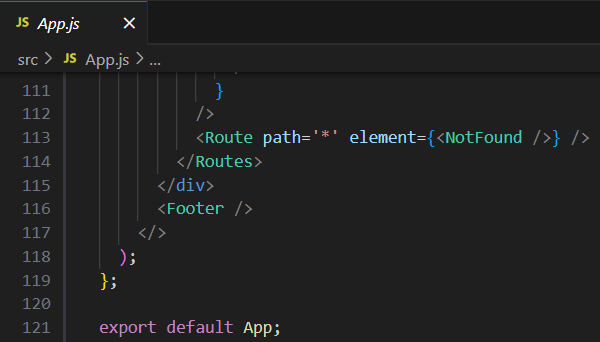


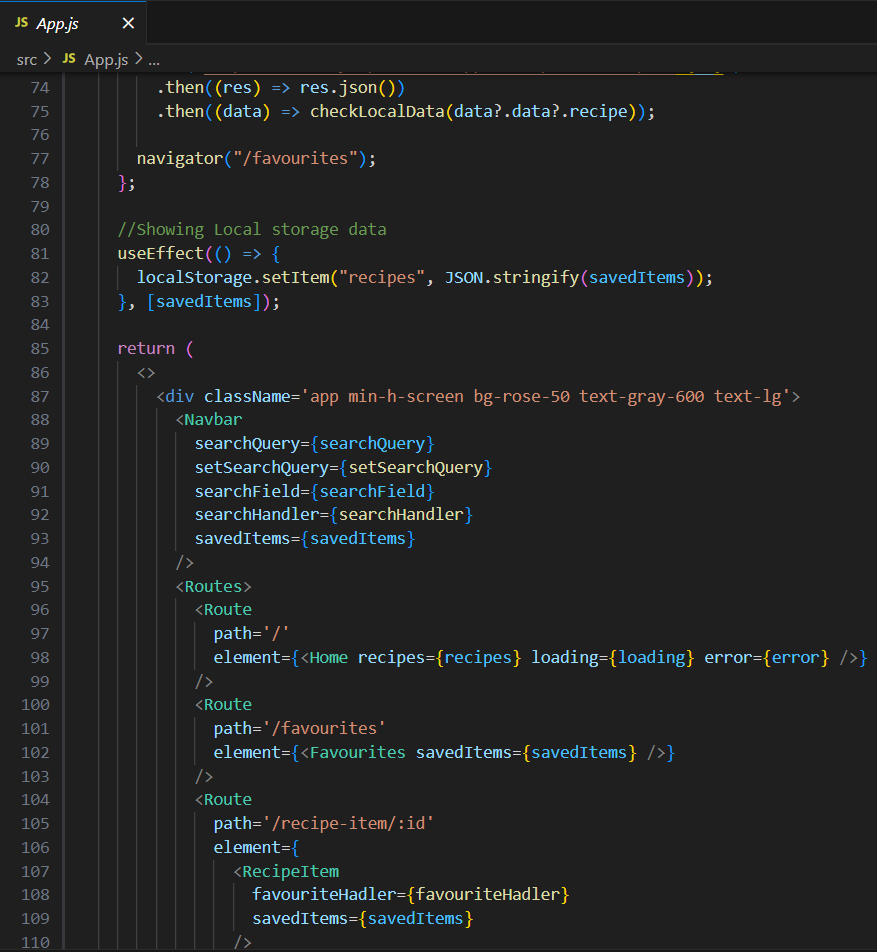


The provided CSS code defines the styling and animations for a visual representation of a cooking scene, including a frying pan, pancake, and bubbles. The animations use keyframes to control the movement and appearance of these elements. The responsive design ensures that the cooking scene is hidden on smaller screens.



The **Home** component is responsible for rendering the main content of the home page. It conditionally displays different content based on the loading status, errors, and the presence of recipe data. It encourages users to search for something if there are no recipes and provides loading and error feedback during data retrieval. It may also include a visual representation of a frying pan using the **FryingPan** component.





This code sets up a React application for searching, displaying, and saving recipes. It uses React Router for navigation, manages application state with hooks, and communicates with an external API to fetch recipe data. The application provides a user interface for searching recipes, viewing individual recipes, and saving favorite recipes to local storage.

After all that Final Presentation:

