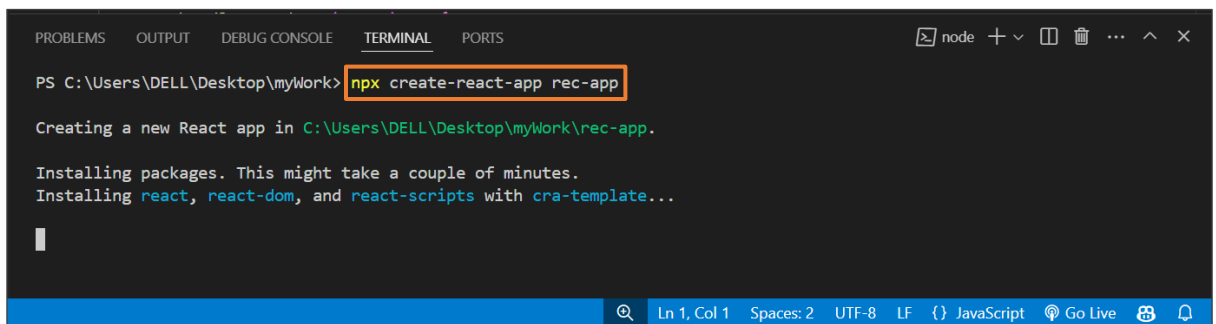


Answer Key

1. Set up the project in Visual Studio Code and GitHub utilizing npx and GitHub Copilot for code generation

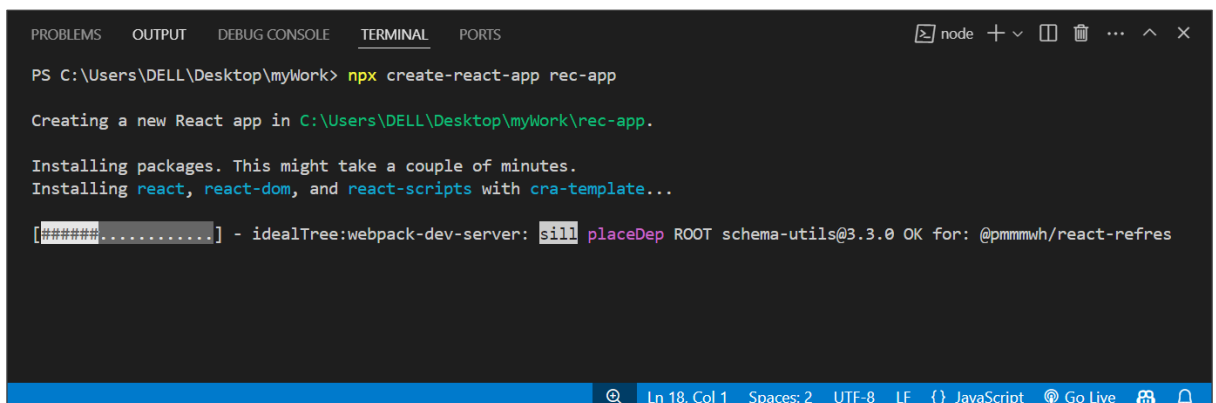
1.1 Set up a new React.js project with the default project structure and configuration using the following command:

npx create-react-app rec-app



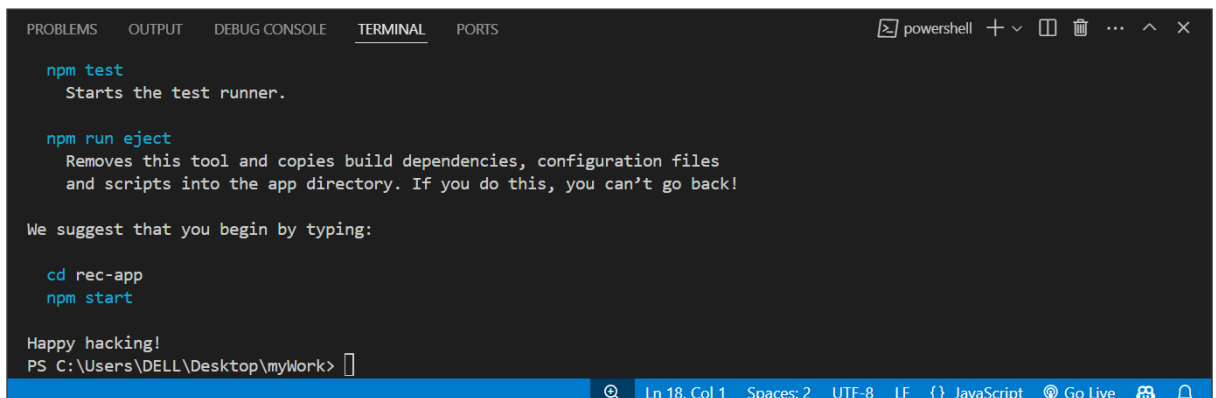
```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\DELL\Desktop\myWork> npx create-react-app rec-app
Creating a new React app in C:\Users\DELL\Desktop\myWork\rec-app.
Installing packages. This might take a couple of minutes.
Installing react, react-dom, and react-scripts with cra-template...
```

npx creates the new project in the ~\Desktop\myWork\rec-app folder as follows:



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\DELL\Desktop\myWork> npx create-react-app rec-app
Creating a new React app in C:\Users\DELL\Desktop\myWork\rec-app.
Installing packages. This might take a couple of minutes.
Installing react, react-dom, and react-scripts with cra-template...
[#####.....] - idealTree:webpack-dev-server: sill placeDep ROOT schema-utils@3.3.0 OK for: @pmmmwh/react-refres
```

The React.js project is successfully created, showing the following after creation:



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
npm test
  Starts the test runner.

npm run eject
  Removes this tool and copies build dependencies, configuration files
  and scripts into the app directory. If you do this, you can't go back!

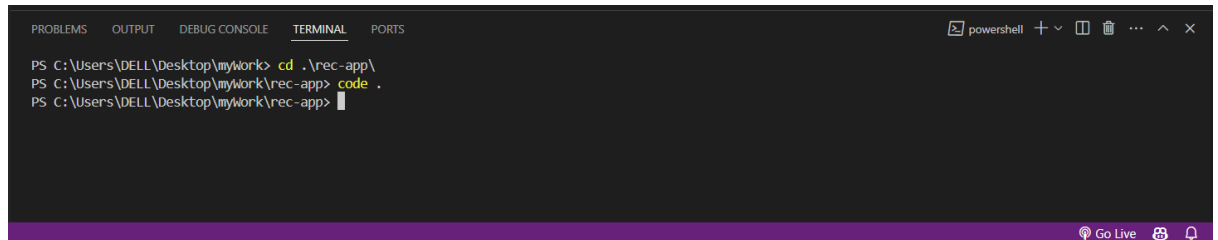
We suggest that you begin by typing:

  cd rec-app
  npm start

Happy hacking!
PS C:\Users\DELL\Desktop\myWork>
```

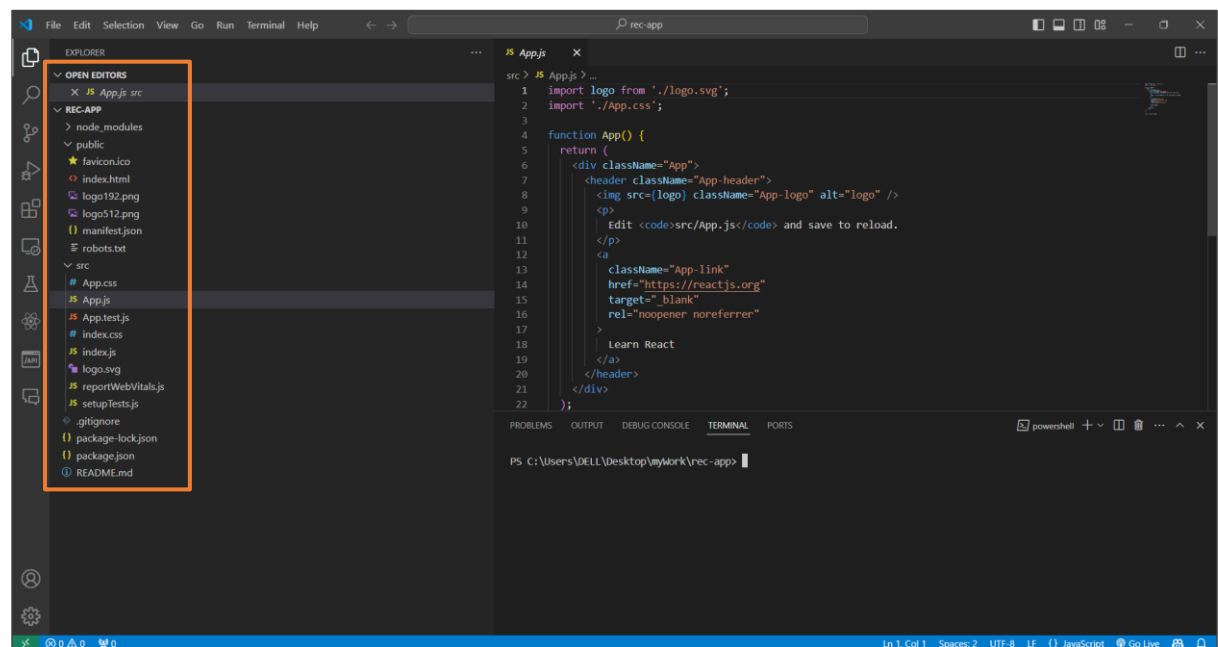
1.2 Execute the following command to navigate to the folder where it is created and to open that project in another instance of Visual Studio:

```
cd .\rec-app\  
code .
```



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS  
PS C:\Users\DELL\Desktop\myWork> cd .\rec-app\  
PS C:\Users\DELL\Desktop\myWork\rec-app> code .  
PS C:\Users\DELL\Desktop\myWork\rec-app> |
```

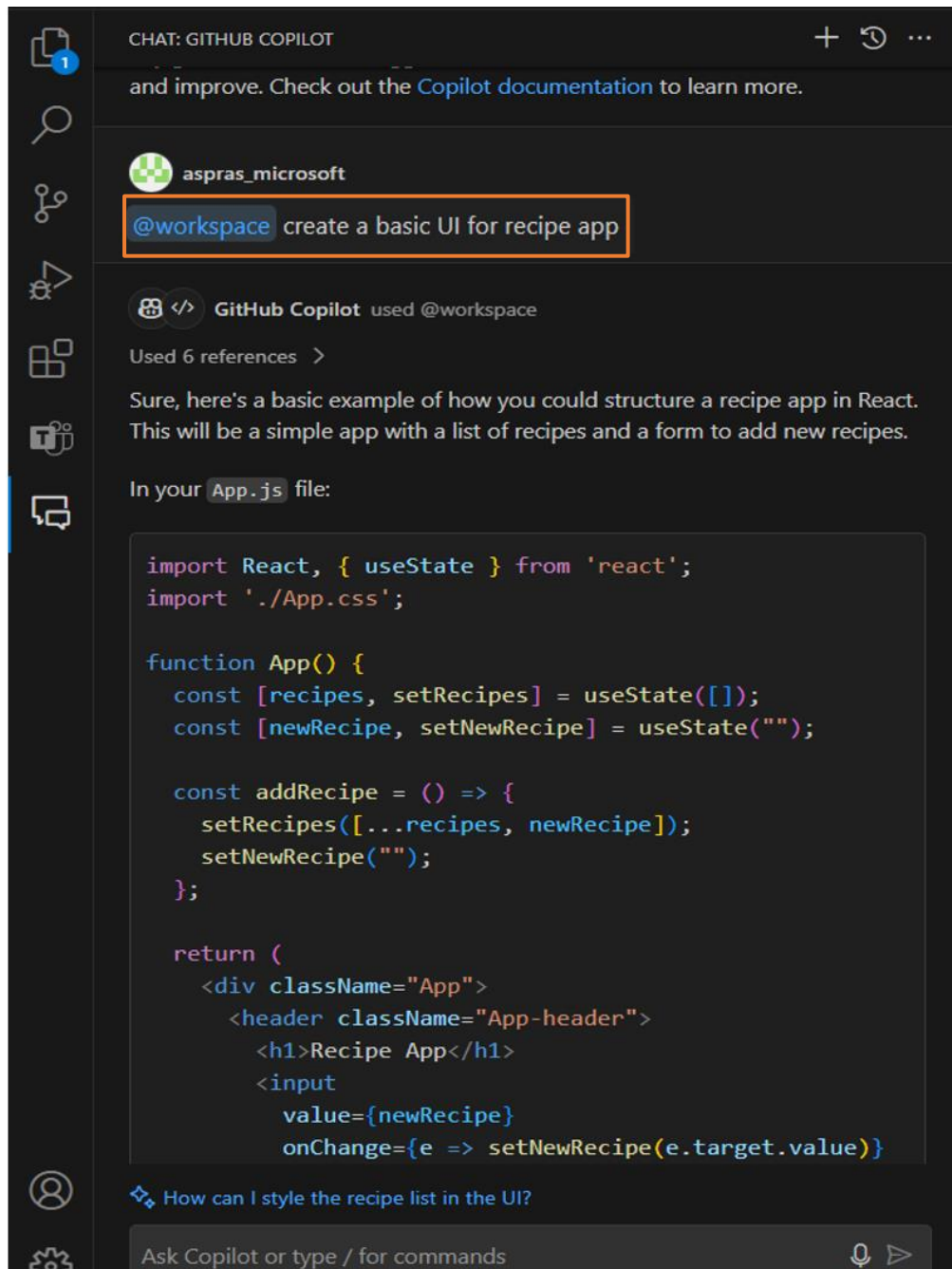
The basic React.js structure appears as follows:



2. Generate a basic user interface for recipe data using GitHub Copilot

2.1 Enhance the project by prompting GitHub Copilot to generate React.js code for a basic UI for the recipe app

@workspace create a basic UI for recipe app

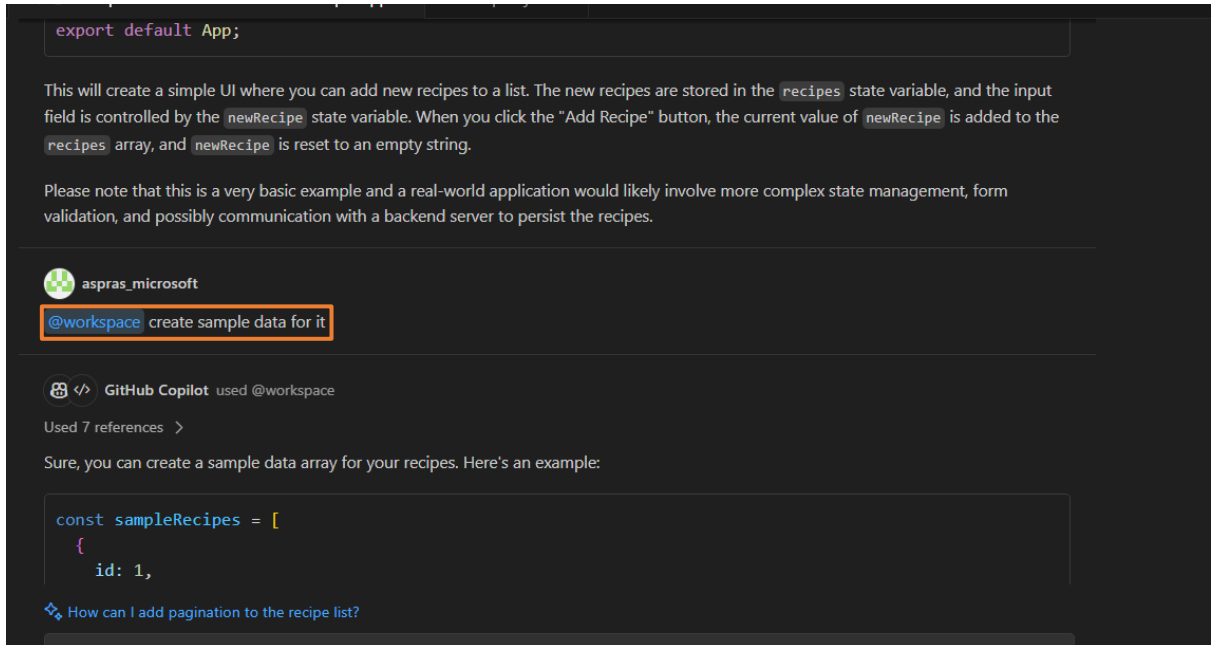


Note: Copy the contents of the generated script and paste them into the existing src\App.js file

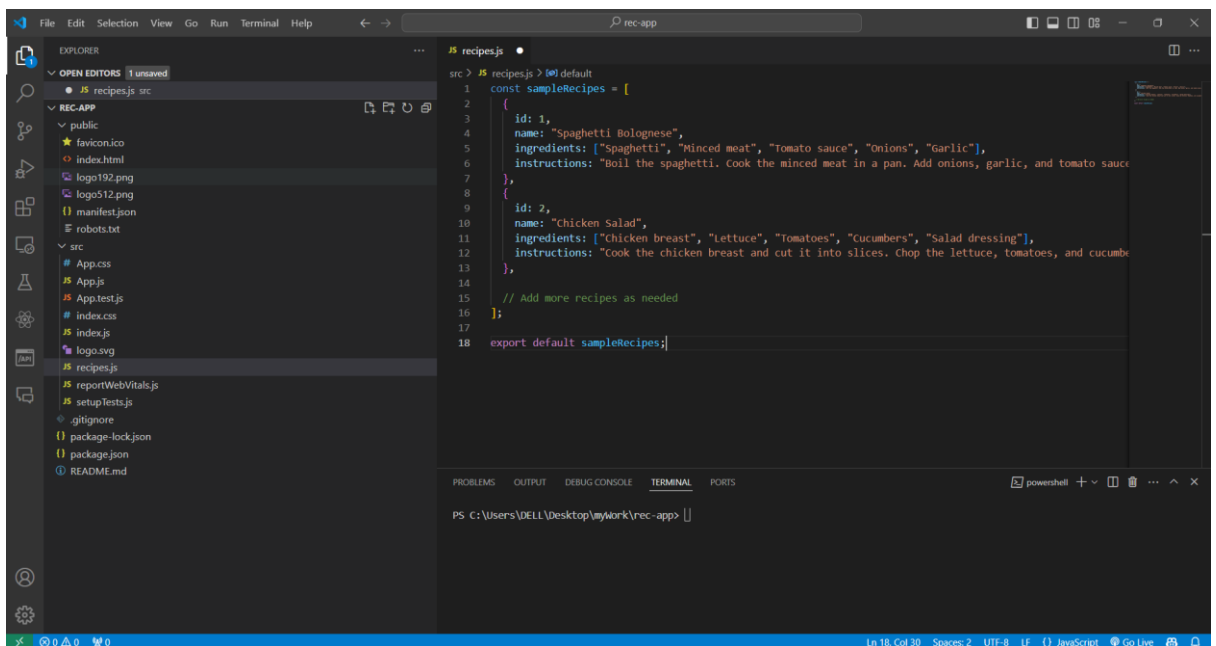
3. Generate sample data using GitHub Copilot

3.1 Prompt GitHub Copilot to create sample data for the existing recipe app

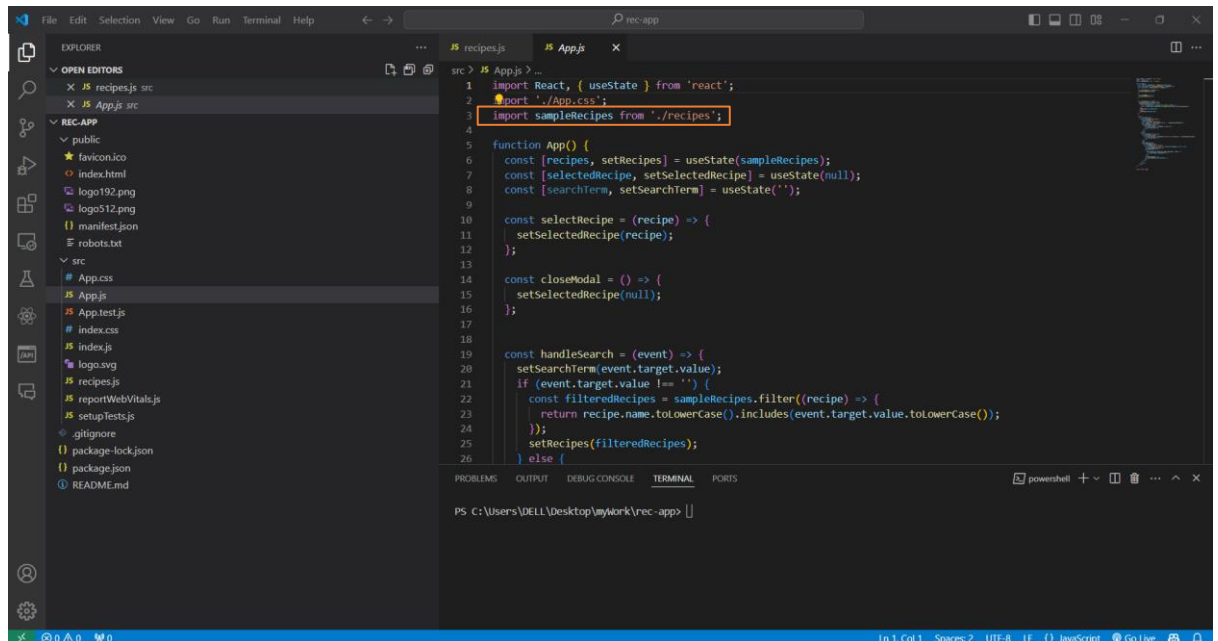
@workspace create sample data for it



This generates the sample data for the recipe application. Create a file named **recipes.js**, copy the sample data generated into this file, and save it in the **rec-app\src** folder



3.2 Open **App.js** and import the newly created **recipe.js** by adding the following script:
import sampleRecipes from './recipes';



4. Generate React code to display the recipes using the user interface

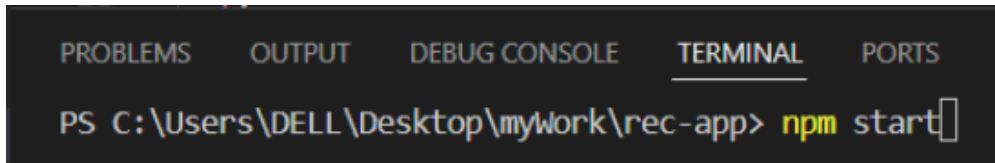
4.1 Prompt GitHub Copilot to generate code to display the recipe data using the user interface
show these recipes in UI



Note: Copy the contents of the generated script and paste them into the existing `src\App.js` file

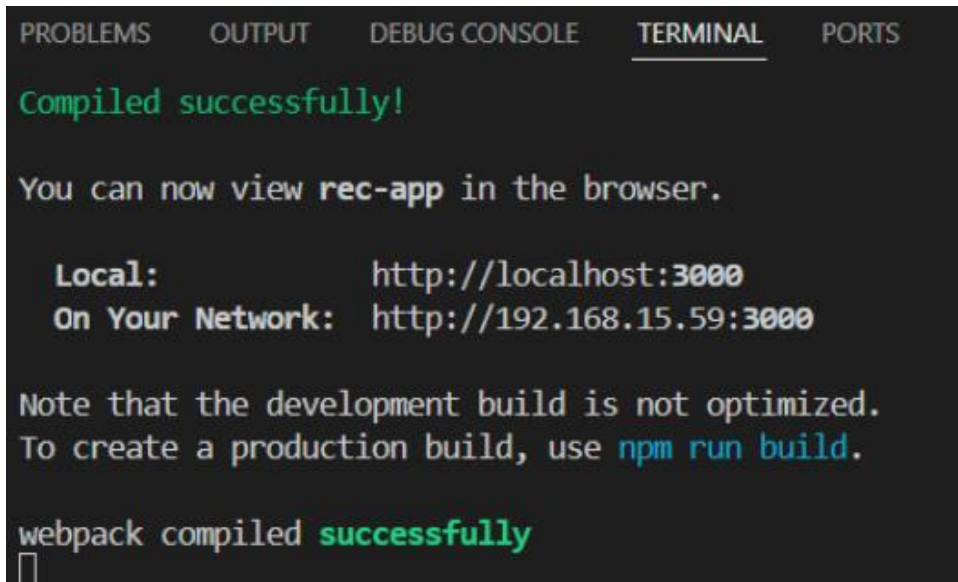
4.2 Run the react project after updating App.js

npm start



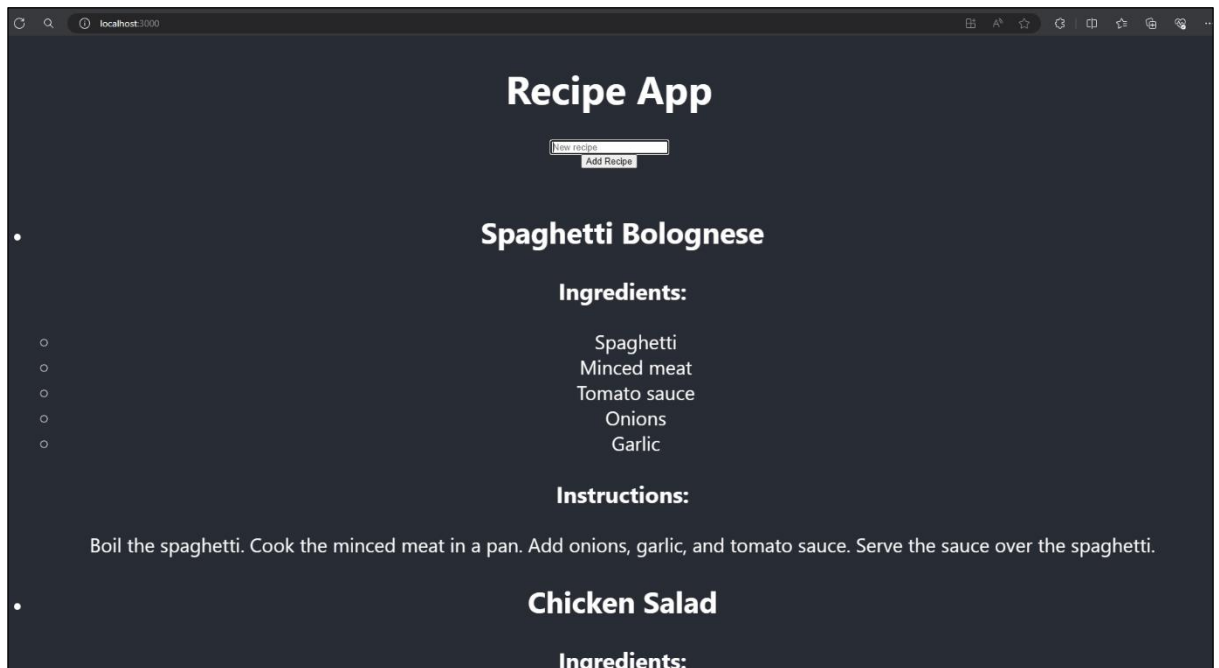
```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  
PS C:\Users\DELL\Desktop\myWork\rec-app> npm start
```

The following message will be displayed after successful execution:



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  
  
Compiled successfully!  
  
You can now view rec-app in the browser.  
  
Local:      http://localhost:3000  
On Your Network: http://192.168.15.59:3000  
  
Note that the development build is not optimized.  
To create a production build, use npm run build.  
  
webpack compiled successfully  
█
```

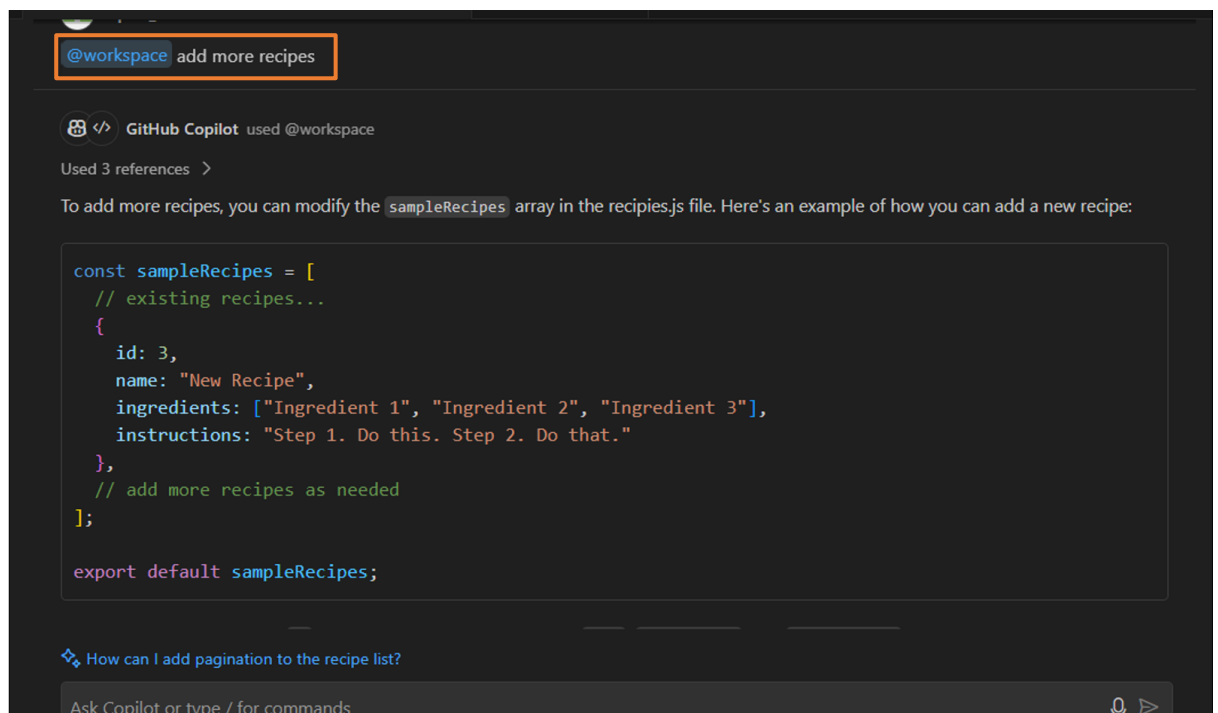
The output will appear as follows:



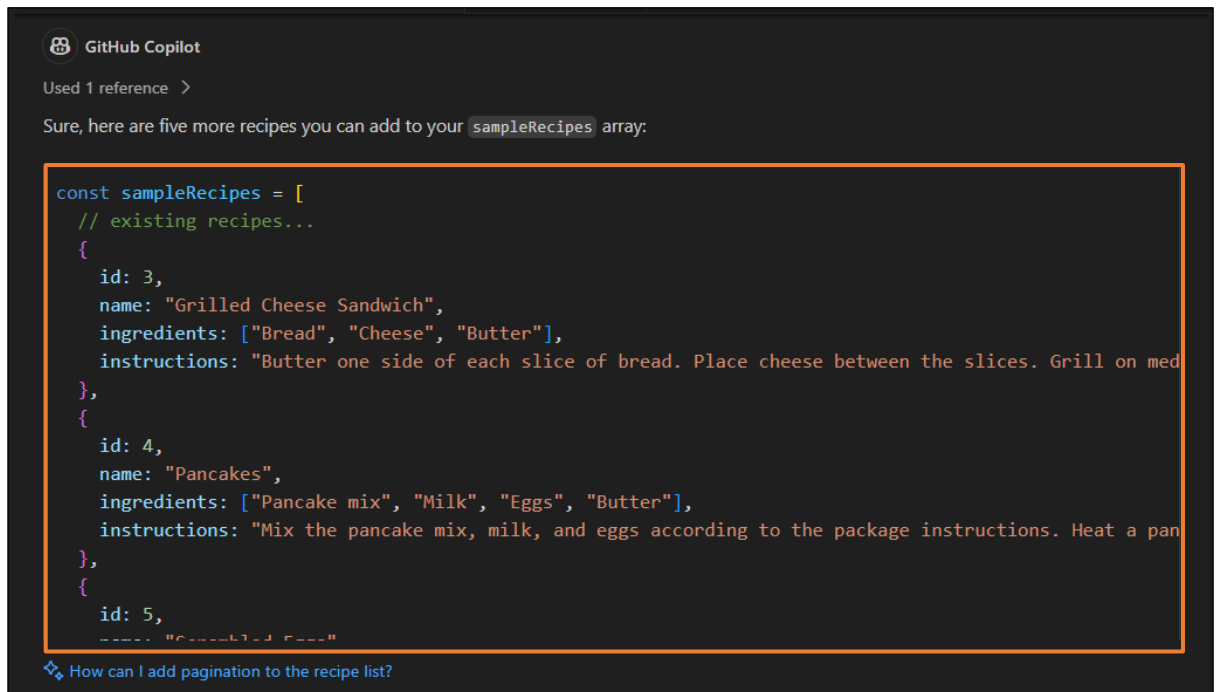
5. Generate a React code to add more recipes to the recipe app

5.1 Improve the recipe app by adding more recipes using GitHub Copilot prompts

@workspace add more recipes



The following React.js code will be generated by GitHub Copilot:



GitHub Copilot

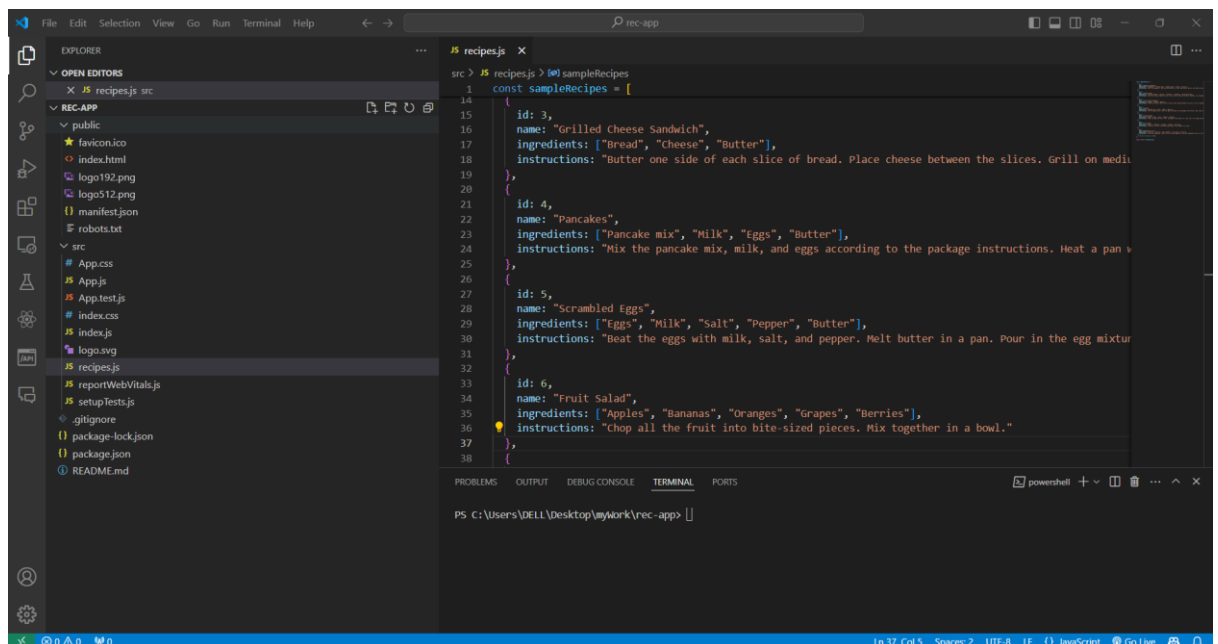
Used 1 reference >

Sure, here are five more recipes you can add to your `sampleRecipes` array:

```
const sampleRecipes = [
  // existing recipes...
  {
    id: 3,
    name: "Grilled Cheese Sandwich",
    ingredients: ["Bread", "Cheese", "Butter"],
    instructions: "Butter one side of each slice of bread. Place cheese between the slices. Grill on med
  },
  {
    id: 4,
    name: "Pancakes",
    ingredients: ["Pancake mix", "Milk", "Eggs", "Butter"],
    instructions: "Mix the pancake mix, milk, and eggs according to the package instructions. Heat a pan
  },
  {
    id: 5,
    name: "Scrambled Eggs",
    ingredients: ["Eggs", "Milk", "Salt", "Pepper", "Butter"],
    instructions: "Beat the eggs with milk, salt, and pepper. Melt butter in a pan. Pour in the egg mixtur
  },
  {
    id: 6,
    name: "Fruit Salad",
    ingredients: ["Apples", "Bananas", "Oranges", "Grapes", "Berries"],
    instructions: "Chop all the fruit into bite-sized pieces. Mix together in a bowl."
  }
]
```

How can I add pagination to the recipe list?

Update the file **recipes.js** by copying all the generated data into it

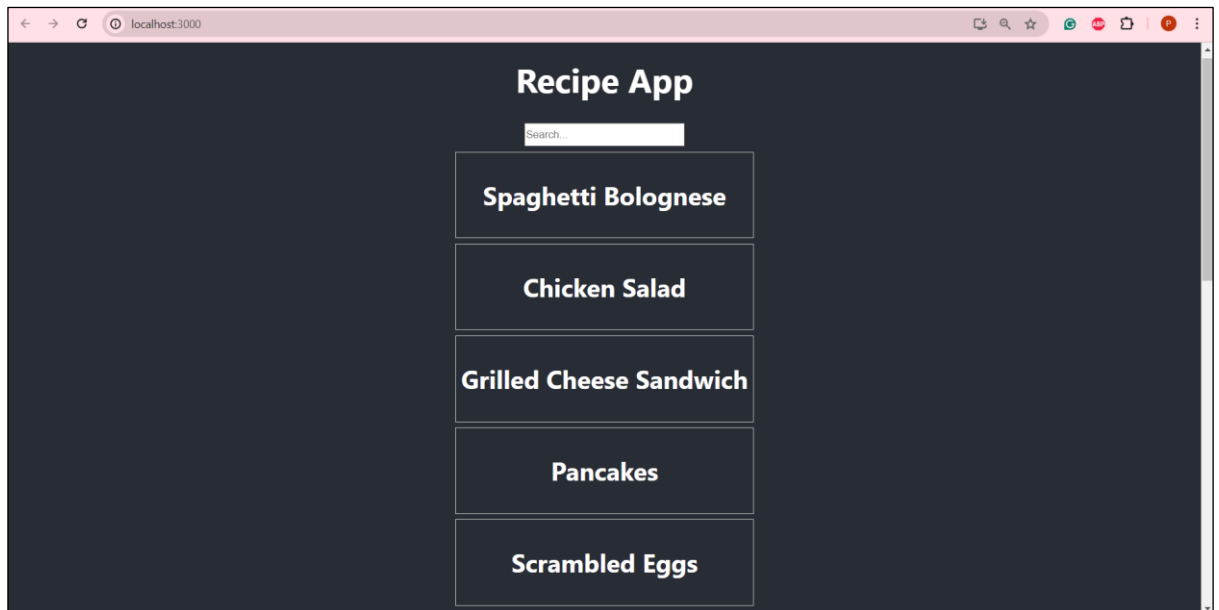


VS Code interface showing the `recipes.js` file being updated with the generated data. The file content is as follows:

```
src > JS recipes.js > 10 sampleRecipes
1 const sampleRecipes = [
14
15 {
16   id: 3,
17   name: "Grilled Cheese Sandwich",
18   ingredients: ["Bread", "Cheese", "Butter"],
19   instructions: "Butter one side of each slice of bread. Place cheese between the slices. Grill on med
20 },
21 {
22   id: 4,
23   name: "Pancakes",
24   ingredients: ["Pancake mix", "Milk", "Eggs", "Butter"],
25   instructions: "Mix the pancake mix, milk, and eggs according to the package instructions. Heat a pan v
26 },
27 {
28   id: 5,
29   name: "Scrambled Eggs",
30   ingredients: ["Eggs", "Milk", "Salt", "Pepper", "Butter"],
31   instructions: "Beat the eggs with milk, salt, and pepper. Melt butter in a pan. Pour in the egg mixtur
32 },
33 {
34   id: 6,
35   name: "Fruit Salad",
36   ingredients: ["Apples", "Bananas", "Oranges", "Grapes", "Berries"],
37   instructions: "Chop all the fruit into bite-sized pieces. Mix together in a bowl."
38 },
39 ]
```

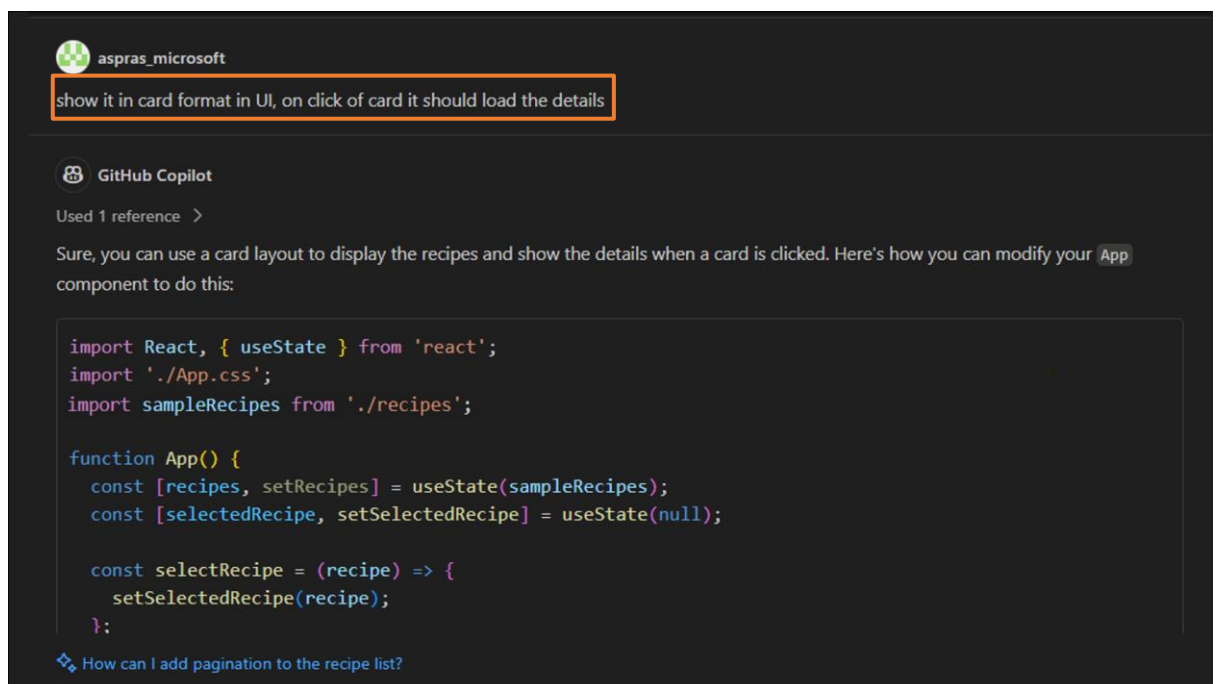
The terminal shows the command: `PS C:\Users\DELL\Desktop\mywork\rec-app>`

Execute the React project using **npm start** after updating the recipe.js file. The output will appear as follows:



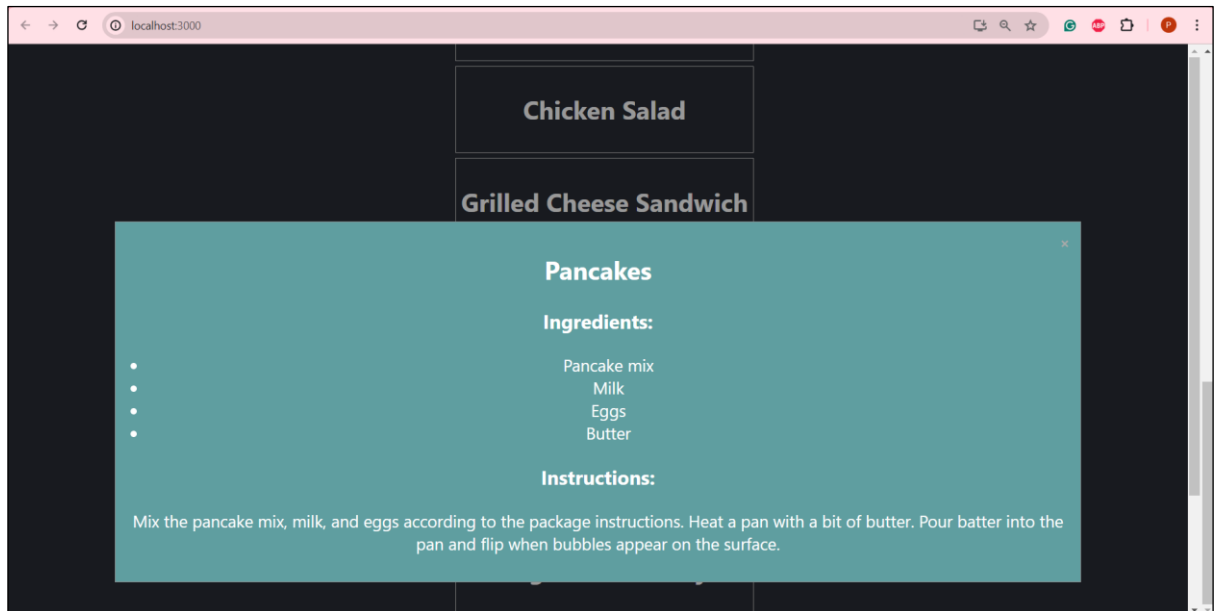
6. Create the Recipe Detail Component to display detailed information about each recipe

6.1 Prompt GitHub Copilot to generate a React code for displaying details of each recipe show it in card format in UI, on click of card it should load the details



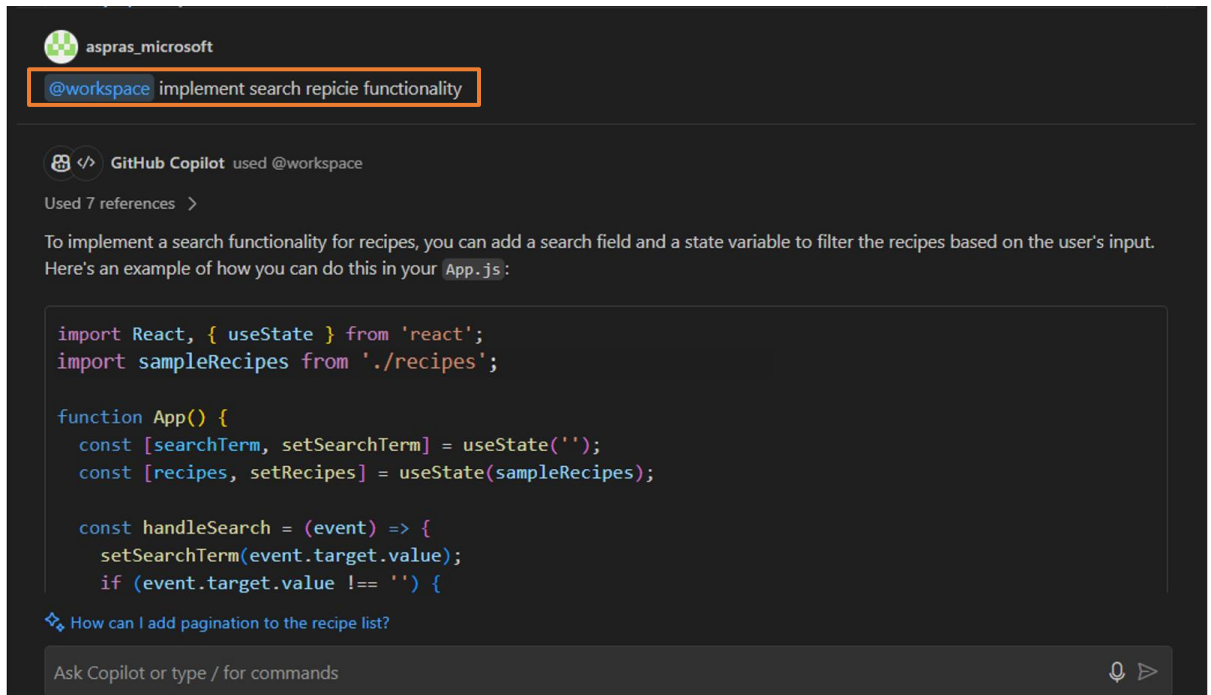
Note: Copy the contents of the generated script and paste them into the existing **src\App.js** file

After updating App.js file, execute the React project using **npm start**. The output will appear as follows:



7. Implement search functionality to enable users to search for specific recipes

7.1 Design a search feature by prompting GitHub Copilot @workspace implement search recipe functionality



Note: Copy the contents of the generated script and paste them into the existing `src\App.js` file

The output will appear as follows:

