

CityUniversity of Seattle



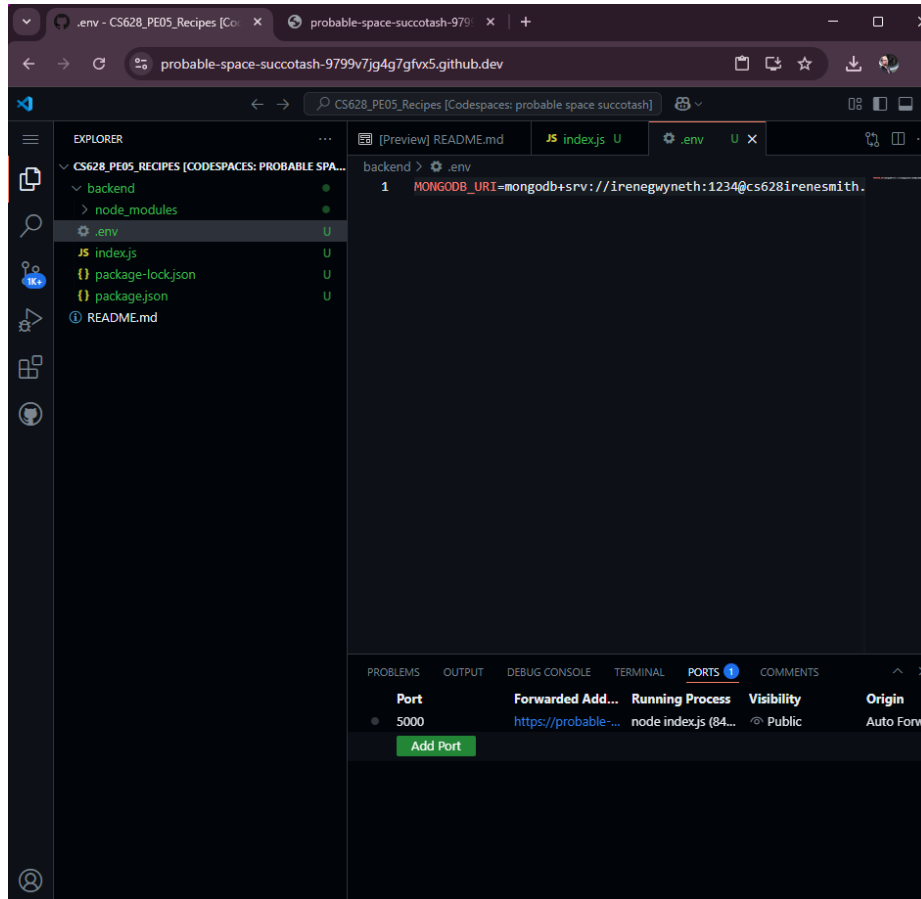
CS628_01_ON: Full Stack Dev-Web Application **Programming Exercise 05_Recipes**

Module 08-09_Irene Smith_Output
WINTER_2024

Hyper Link : https://github.com/irenegwyneth/CS628_PE05_Recipes

Output: Backend Settings

Backend Structure



Connected
to
MongoDB

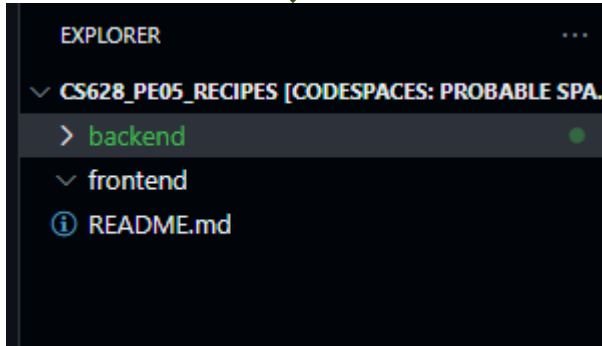
probable-space-succotash-9799v7jg4g7gfvx5-5000.app.github.dev

Recipe Finder API is running

**Next Step:
Frontend**

Output: Frontend Settings

**Next Step:
Create React
App**



```
@irenegwyneth →/workspaces/CS628_PE05_Recipes (main) $ cd ..
@irenegwyneth →/workspaces $ cd /CS628_PE05_Recipes
bash: cd: /CS628_PE05_Recipes: No such file or directory
@irenegwyneth →/workspaces $ cd CS628_PE05_Recipes
@irenegwyneth →/workspaces/CS628_PE05_Recipes (main) $ mkdir frontend
@irenegwyneth →/workspaces/CS628_PE05_Recipes (main) $ cd frontend
@irenegwyneth →/workspaces/CS628_PE05_Recipes/frontend (main) $ npx create-react-
app .
Need to install the following packages:
create-react-app@5.1.0
Ok to proceed? (y) y
```

```
We suggest that you begin by typing:

cd /workspaces/CS628_PE05_Recipes/frontend
npm start

Happy hacking!
@irenegwyneth →/workspaces/CS628_PE05_Recipes/frontend (main) $
```

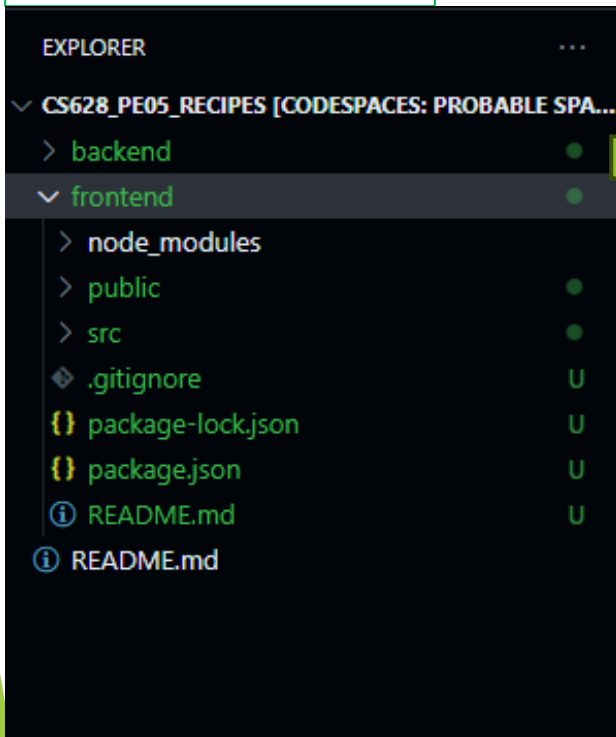
**Next Step:
Adding
dependencies**

```
@irenegwyneth →/workspaces/CS628_PE05_Recipes/frontend (main) $ npm install axios
react-router-dom

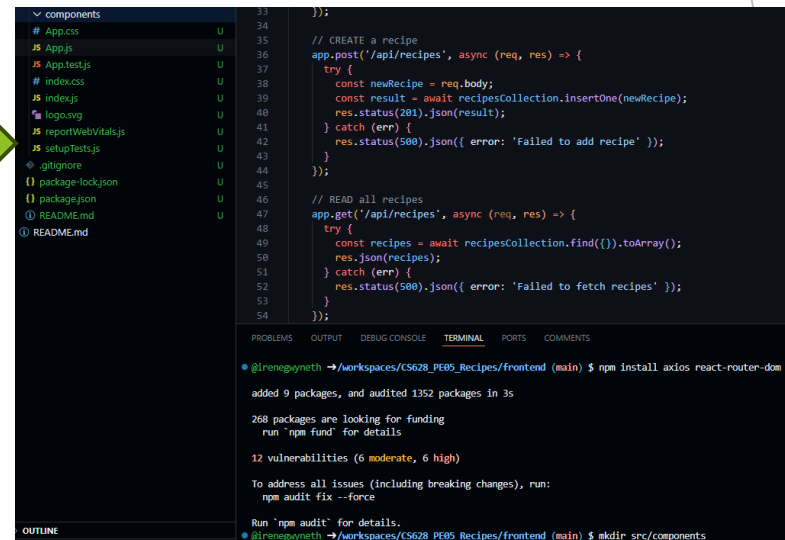
added 9 packages, and audited 1352 packages in 3s

268 packages are looking for funding
run `npm fund` for details
```

Frontend Structure



Next Step: Create Components for Recipe details



Adding .js files



Output: Frontend Settings



Frontend
settings:
index.js



```
1 import React from 'react';
2 import ReactDOM from 'react-dom/client';
3 import { BrowserRouter } from 'react-router-dom';
4 import App from './App';
5
6 const root = ReactDOM.createRoot(document.getElementById('root'));
7 root.render(
8   <BrowserRouter>
9     <App />
10   </BrowserRouter>
11 );
12
```



Frontend
settings:
App.js

```
9 function App() {
10   return (
11     <div>
12       <nav>
13         <h1>Recipe Finder</h1>
14         <Link to="/">Home</Link> | {' '}
15         <Link to="/add-recipe">Add Recipe</Link>
16       </nav>
17
18       <Routes>
19         <Route path="/" element=<RecipeList /> />
20         <Route path="/add-recipe" element=<AddRecipe /> />
21         <Route path="/recipe/:id" element=<RecipeDetails />
22         <Route path="/recipe/:id/edit" element=<EditRecipe
23       </Routes>
24     </div>
25   );
26 }
27
28 export default App;
29
```

Output: Frontend Settings



Frontend
settings:
index.js



Frontend
settings:
App.js

```
EXPLORER
CS628_PE05_RECIPES [CODESPACES: PROBABLE SPA...]
  > backend
  > frontend
  > node_modules
  > public
  > src
    > components
      JS AddRecipe.js
      JS EditRecipe.js
      JS RecipeDetails.js
      JS RecipeList.js
      # App.css
      JS App.js
      JS App.test.js
      # index.css
      JS index.js
      logo.svg
      JS reportWebVitals.js
      JS setupTests.js
      .gitignore
      package-lock.json
      package.json
      README.md
      ...

Source Files
frontend > src > JS index.js > ...
1 import React from 'react';
2 import ReactDOM from 'react-dom/client';
3 import { BrowserRouter } from 'react-router-dom';
4 import App from './App';
5
6 const root = ReactDOM.createRoot(document.getElementById('root'));
7 root.render(
8   <BrowserRouter>
9     <App />
10  </BrowserRouter>
11 );
12
```



```
EXPLORER
CS628_PE05_RECIPES [CODESPACES: PROBABLE SPA...]
  > backend
  > frontend
  > node_modules
  > public
  > src
    > components
      JS AddRecipe.js
      JS EditRecipe.js
      JS RecipeDetails.js
      JS RecipeList.js
      # App.css
      JS App.js
      JS App.test.js
      # index.css
      JS index.js
      logo.svg
      JS reportWebVitals.js
      ...

Source Files
frontend > src > JS App.js > ...
9 function App() {
10   return (
11     <div>
12       <nav>
13         <h1>Recipe Finder</h1>
14         <Link to="/">Home</Link> | {' '}
15         <Link to="/add-recipe">Add Recipe</Link>
16       </nav>
17
18       <Routes>
19         <Route path="/" element={<RecipeList />} />
20         <Route path="/add-recipe" element={<AddRecipe />} />
21         <Route path="/recipe/:id" element={<RecipeDetails />} />
22         <Route path="/recipe/:id/edit" element={<EditRecipe />} />
23       </Routes>
24     </div>
25   );
26 }
27
28 export default App;
```

Recipe Finder

[Home](#) | [Add Recipe](#)

All Recipes

Output: Implementation of each .js files



Frontend settings: AddRecipe.js



```
src/components/AddRecipe.js
1 // src/components/AddRecipe.js
2 import React, { useState } from 'react';
3 import axios from 'axios';
4
5 function AddRecipe() {
6   const [name, setName] = useState('');
7   const [ingredients, setIngredients] = useState('');
8   const [instructions, setInstructions] = useState('');
9
10   const handleSubmit = async (e) => {
11     e.preventDefault();
12     try {
13       // Construct the new recipe object
14       const newRecipe = { name, ingredients, instructions };
15
16       // Send POST request to your backend
17       await axios.post('http://localhost:3000/api/recipes',
18         newRecipe);
19
20       alert('Recipe added successfully!');
21       // Optionally reset fields or navigate away
22       setName('');
23       setIngredients('');
24       setInstructions('');
25       // If we use react-router's navigate(), after success
26     } catch (error) {
27       console.error('Error adding recipe:', error);
28     }
29   };
30
31   return (
32     <div>
33       <h3>Add a New Recipe</h3>
34       <div>
35         <input type="text" value={name} />
36         <input type="text" value={ingredients} />
37         <input type="text" value={instructions} />
38         <button type="submit" value="Add Recipe" />
39       </div>
40     </div>
41   );
42 }
```

Frontend settings: EditRecipe.js



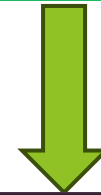
```
src/components/EditRecipe.js
1 // src/components/EditRecipe.js
2 import React, { useState } from 'react';
3 import axios from 'axios';
4
5 function EditRecipe() {
6   const [recipeId, setRecipeId] = useState('');
7   const [name, setName] = useState('');
8   const [ingredients, setIngredients] = useState('');
9   const [instructions, setInstructions] = useState('');
10
11   const handleSubmit = async (e) => {
12     e.preventDefault();
13     try {
14       // Send PUT request to your backend
15       await axios.put(`http://localhost:3000/api/recipes/${recipeId}`,
16         { name, ingredients, instructions });
17
18       alert('Recipe updated successfully!');
19       // Optionally reset fields or navigate away
20       setName('');
21       setIngredients('');
22       setInstructions('');
23     } catch (error) {
24       console.error('Error updating recipe:', error);
25     }
26   };
27
28   return (
29     <div>
30       <h3>Edit Recipe</h3>
31       <div>
32         <input type="text" value={recipeId} />
33         <input type="text" value={name} />
34         <input type="text" value={ingredients} />
35         <input type="text" value={instructions} />
36         <button type="submit" value="Update Recipe" />
37       </div>
38     </div>
39   );
40 }
```

Frontend settings: RecipeDetails.js



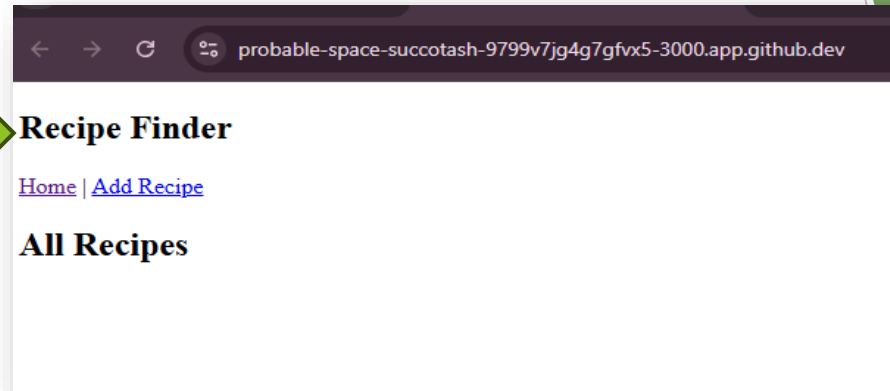
```
src/components/RecipeDetails.js
1 // src/components/RecipeDetails.js
2 import React, { useState } from 'react';
3 import axios from 'axios';
4
5 function RecipeDetails() {
6   const [recipeId, setRecipeId] = useState('');
7   const [recipe, setRecipe] = useState(null);
8
9   const fetchRecipe = async () => {
10     try {
11       const response = await axios.get(`http://localhost:3000/api/recipes/${recipeId}`);
12       setRecipe(response.data);
13     } catch (error) {
14       console.error('Error fetching recipe details:', error);
15     }
16   };
17
18   return (
19     <div>
20       <h3>Recipe Details</h3>
21       <div>
22         <input type="text" value={recipeId} />
23         <button type="button" value="Fetch Recipe" />
24       </div>
25       {recipe ? (
26         <div>
27           <p>Recipe Name: {recipe.name}</p>
28           <p>Ingredients: {recipe.ingredients}</p>
29           <p>Instructions: {recipe.instructions}</p>
30         </div>
31       ) : (
32         <p>Recipe not found.</p>
33       )}
34     </div>
35   );
36 }
```

Frontend settings: RecipeList.js

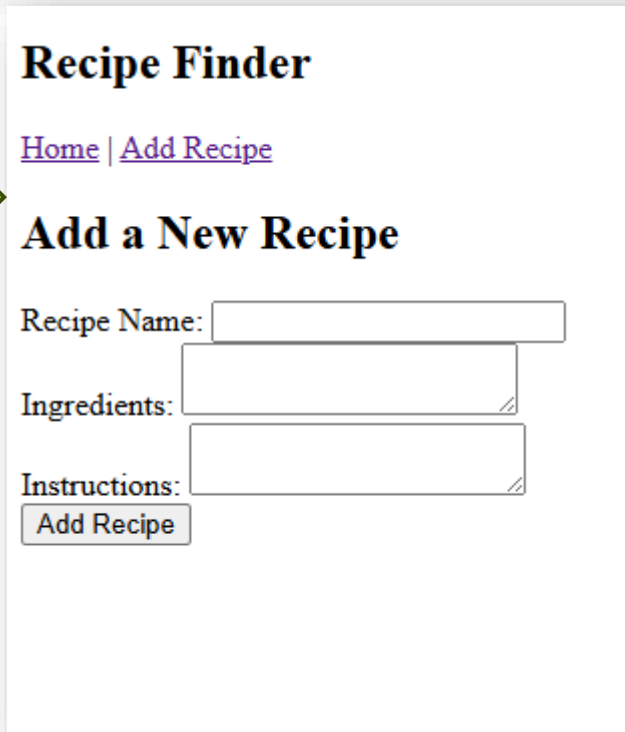


```
src/components/RecipeList.js
1 // src/components/RecipeList.js
2 import React, { useState } from 'react';
3 import axios from 'axios';
4
5 function RecipeList() {
6   const [recipes, setRecipes] = useState([]);
7
8   const fetchRecipes = async () => {
9     try {
10       const response = await axios.get('http://localhost:3000/api/recipes');
11       setRecipes(response.data);
12     } catch (error) {
13       console.error('Error fetching recipe list:', error);
14     }
15   };
16
17   return (
18     <div>
19       <h3>Recipe List</h3>
20       <div>
21         <button type="button" value="Fetch Recipes" />
22       </div>
23       <div>
24         {recipes.map(recipe => (
25           <div>
26             <p>Recipe Name: {recipe.name}</p>
27             <p>Ingredients: {recipe.ingredients}</p>
28             <p>Instructions: {recipe.instructions}</p>
29           </div>
30         ))}
31       </div>
32     </div>
33   );
34 }
```

Next step:
npm start



Recipe
Finder:
Upon
clicking add
new recipe a
form field
will pop-up



Recipe Finder

[Home](#) | [Add Recipe](#)

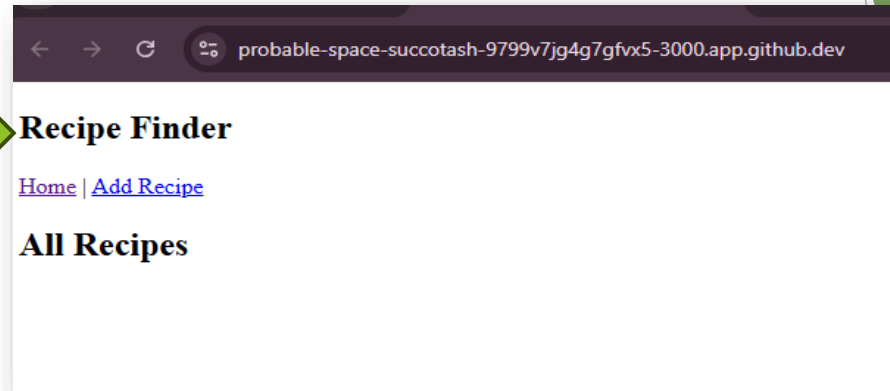
Add a New Recipe

Recipe Name:

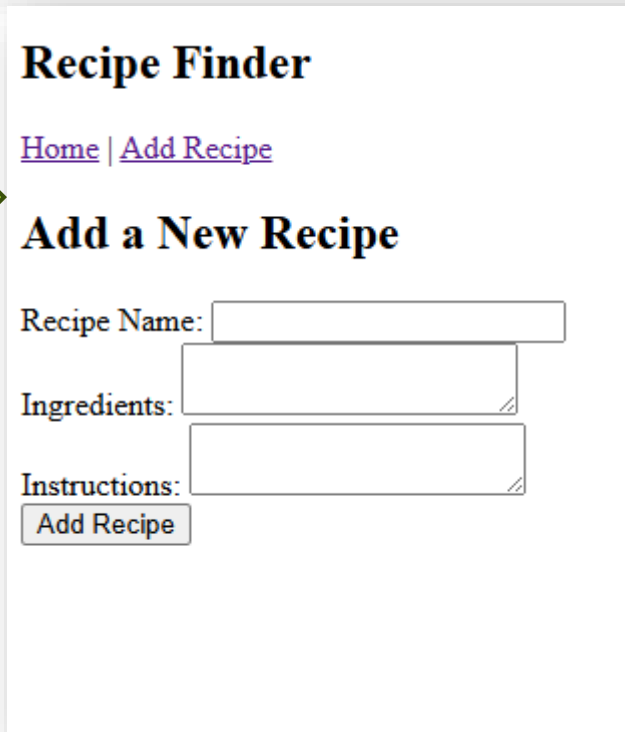
Ingredients:

Instructions:

**Next step:
npm start**



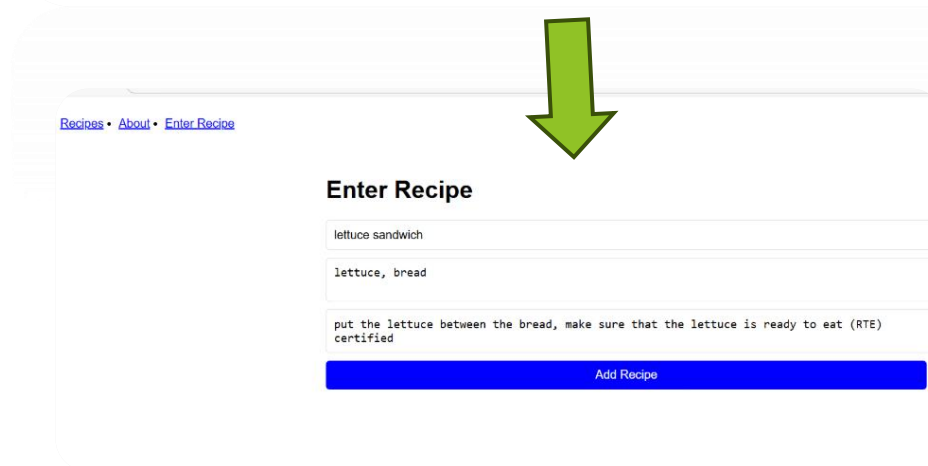
**Recipe
Finder:
Upon
clicking add
new recipe a
form field
will pop-up**



**Recipe
Finder:
try encoding
recipe
sample and
click add
recipe**



The screenshot shows a web browser window with the URL `probable-space-succotash-9799v7jg4g7gfvx5-3000.app.github.dev/add-recipe`. The page title is "Recipe Finder" and it has links for [Home](#) and [Add Recipe](#). The main heading is "Add a New Recipe". The form contains three text input fields: "Recipe Name" with the value "Red Spaghetti", "Ingredients" with the value "Mushrooms or bell peppers", and "Instructions" with the value "Cover and simmer". Below these fields is a blue "Add Recipe" button.



The screenshot shows a web browser window with the URL `probable-space-succotash-9799v7jg4g7gfvx5-3000.app.github.dev/enter-recipe`. The page title is "Enter Recipe" and it has links for [Recipes](#), [About](#), and [Enter Recipe](#). The form contains three text input fields: "Recipe Name" with the value "lettuce sandwich", "Ingredients" with the value "lettuce, bread", and "Instructions" with the value "put the lettuce between the bread, make sure that the lettuce is ready to eat (RTE) certified". Below these fields is a blue "Add Recipe" button.

References for Module 08-9:

Node

- [Node - Getting Started](#)
- [W3Schools - Node.js Tutorial](#)

Express

- [GitHub Codespaces](#)
- [Express - Getting Started](#)
- [Express & MongoDB](#)

Express Router

- [Basic Routing](#)
- [Routing](#)
- [Using the Fetch API](#)

Case Study

- [How to Use MERN Stack: A Complete Guide](#)