

W09-P1:initial setup, change component name by add

The screenshot shows the Visual Studio Code interface with the file `index.js` open in the editor. The code imports React and ReactDOM, and defines an `App` component. A blue circle highlights the word `App`. In the browser window, the URL `localhost:3000` is shown, and the text "app component" is visible, also highlighted with a blue circle.

```
import React from "react";
import ReactDOM from "react-dom";
import "./index.css";
import App from "./App_57";
import { AppProvider } from "./context_57";
ReactDOM.render(
  <React.StrictMode>
    <AppProvider>
      <App />
    </AppProvider>
  </React.StrictMode>,
  document.getElementById("root")
)

// If you want your app to work offline and load faster
// unregister() to register() below. Note this comes with some performance overhead.
// Learn more about service workers: https://bit.ly/CRA-PWA
```

W09-P2:<about_02> <Error_02> <Navbar_02>

The screenshot shows the Visual Studio Code interface with the file `About_57.js` open in the editor. The code defines a `About` component that returns an `h1` title and a paragraph of placeholder text. A blue circle highlights the word `About`. In the browser window, the URL `localhost:3000/about` is shown, and the text "About Page" is visible, also highlighted with a blue circle.

```
import React from "react";

const About = () => {
  return (
    <section className="section about-section">
      <h1 className="section-title">About Page</h1>
      <p>Lorem ipsum dolor, sit amet consectetur adipisicing elit. Quisquam voluptatibus libero voluptas at. Facere excepturi iure hic vitae veniam, quasi et, incidunt harum in, dolorem ipsum repellat. Commodi, voluptatibus neque placeat architecto dolorem asperiores eos, repellat aut ratione neque incidunt</p>
    </section>
  );
};

export default About;
```

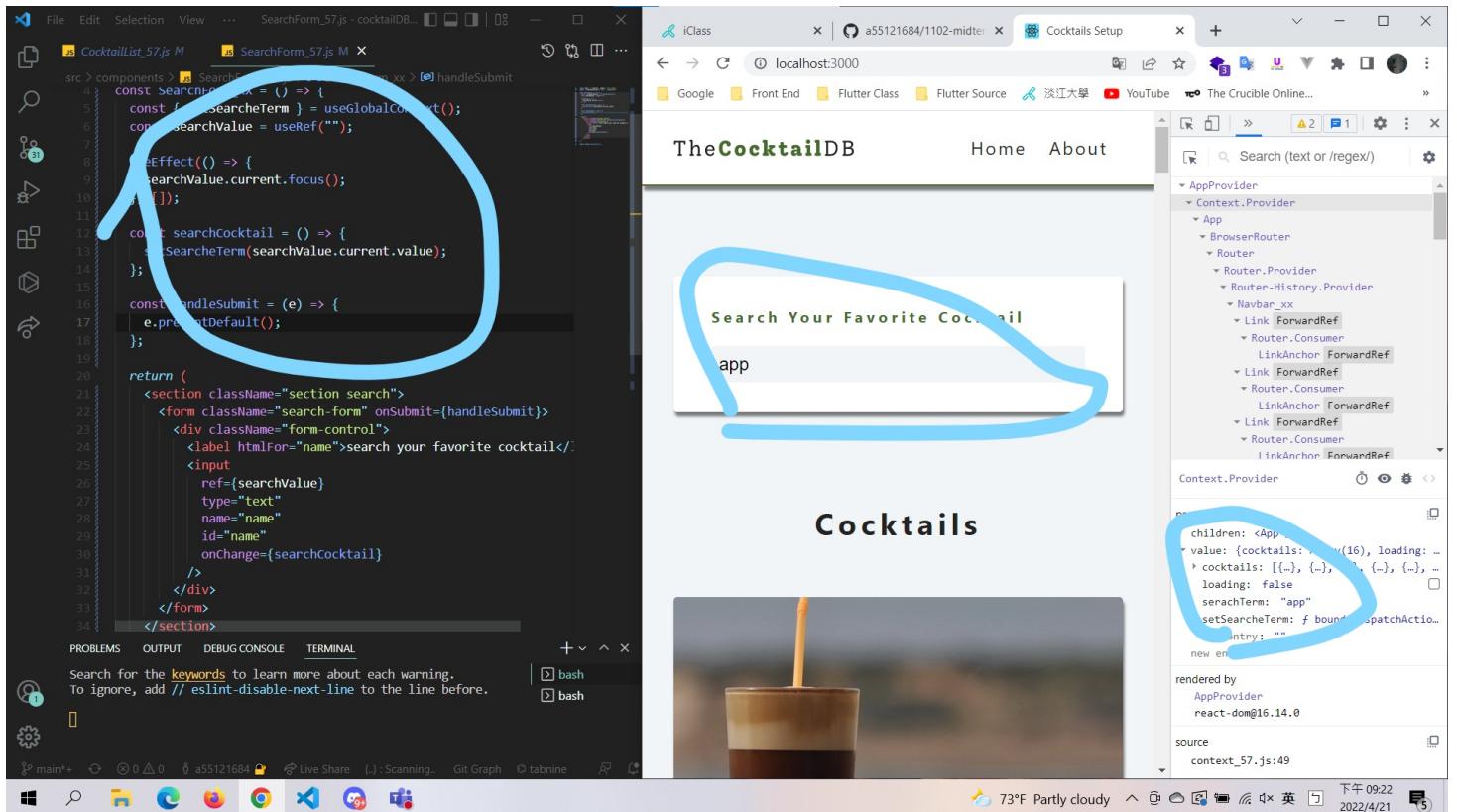
W09-p3:finish context.js and fetch data shown in DevTools

The screenshot shows the development environment for a React application. On the left, the Visual Studio Code interface displays the file structure of the 'cocktailDB_XX' project, including components like 'CocktailList.js' and 'SearchForm.js'. The 'context_57.js' file is open in the editor, showing code for fetching cocktail data from an API and updating the state. A blue circle highlights the fetch call and state update logic. On the right, a browser window shows the 'The CocktailDB' website. The search form has been populated with the term 'blue'. The results page displays two cocktails: 'Bluebird' (Cocktail glass, Alcoholic) and '501 Blue' (Collins Glass, Alcoholic). The browser's DevTools sidebar is visible, showing the component tree and the network tab with requests to the API endpoint.

w09-P4 Show Cocktails with your own searchTerm

This screenshot illustrates the implementation of a custom search feature. The 'context_57.js' file in Visual Studio Code contains code for creating an AppContext provider. A blue circle highlights the section where a custom searchTerm is passed to the provider. The browser window on the right shows the results of a search for 'blue'. It displays two cocktails: 'Bluebird' (Cocktail glass, Alcoholic) and '501 Blue' (Collins Glass, Alcoholic). Below the results, there are two images of cocktails: one blue drink in a martini glass and another blue drink in a tall glass with a lime wedge. The browser's status bar indicates the date and time as April 21, 2022, at 09:00.

W09-P5: Show Cocktails using searchTerm.



w09-P6 All log message in Github

