Step 1: Set Up the Backend with Node.js and Express

First, create a new directory for your project and navigate into it:

```
mkdir todo-app
cd todo-app
```

Initialize a new Node.js project:

```
npm init -y
```

Install the required dependencies:

```
npm install express cors body-parser
```

Create a file named `server.js` in the root of your project directory and set up the server:

server.js:

```
const express = require('express');
const bodyParser = require('body-parser');
const cors = require('cors');

const app = express();
app.use(cors());
app.use(bodyParser.json());

let todos = [];

// Get all todos
app.get('/api/todos', (req, res) => {
  res.json(todos);
});

// Add a new todo
app.post('/api/todos', (req, res) => {
  const newTodo = { id: Date.now(), text: req.body.text, completed: false };
  todos.push(newTodo);
  res.status(201).json(newTodo);
});

// Update a todo
app.put('/api/todos/:id', (req, res) => {
  const id = parseInt(req.params.id);
  const todo = todos.find(todo => todo.id === id);
  if (todo) {
    todo.text = req.body.text !== undefined ? req.body.text : todo.text;
    todo.completed = req.body.completed !== undefined ? req.body.completed : todo.completed;
    res.json(todo);
  } else {
    res.status(404).json({ message: 'Todo not found' });
  }
});

// Delete a todo
app.delete('/api/todos/:id', (req, res) => {
  const id = parseInt(req.params.id);
  todos = todos.filter(todo => todo.id !== id);
  res.status(204).end();
});

const PORT = process.env.PORT || 3001;
app.listen(PORT, () => {
  console.log(`Server running on port ${PORT}`);
});
```

Start the server by running:

```
node server.js
```

Step 2: Set Up the Frontend with React

Open another terminal and navigate to the project directory. Use Create React App to set up the frontend:

```
npx create-react-app client

cd client
```

Install Axios for making HTTP requests:

```
npm install axios
```

Step 3: Create the React Components

Replace the contents of `src/App.js` with the following code:

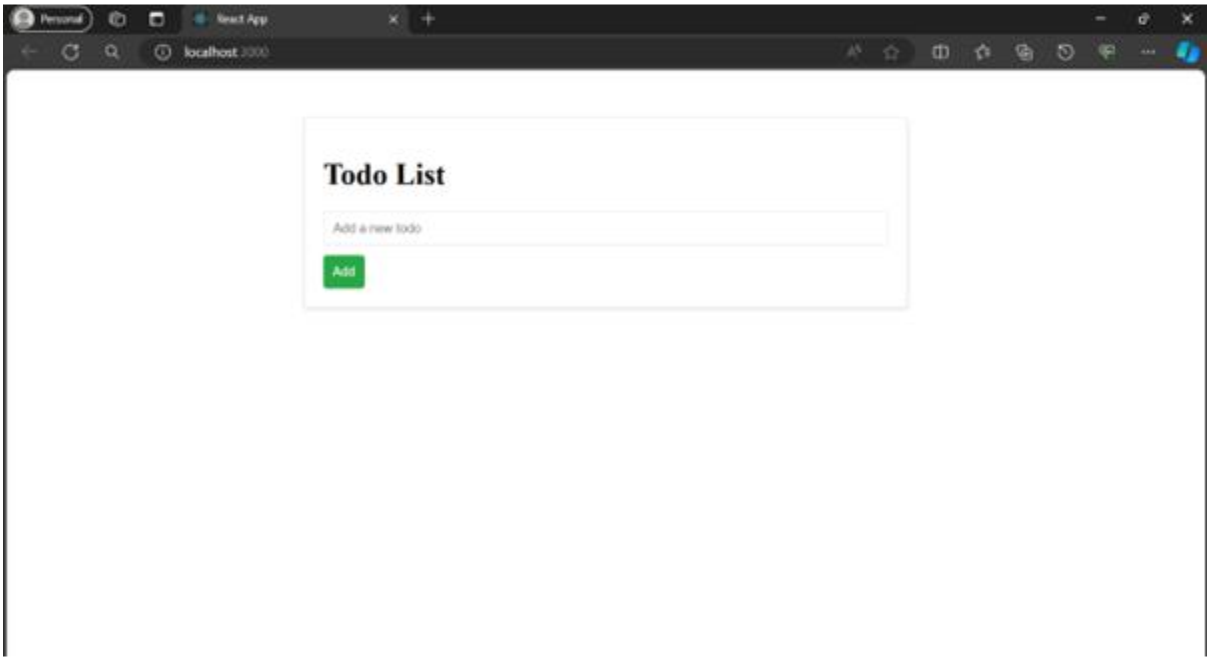
src/App.js:

```
import React, { useState, useEffect } from 'react';

import axios from 'axios';

import './App.css';


function App() {

  const [todos, setTodos] = useState([]);

  const [newTodo, setNewTodo] = useState('');


  useEffect(() => {

    axios.get('http://localhost:3001/api/todos')

      .then(response => {

        setTodos(response.data);

      })

      .catch(error => {

        console.error('There was an error fetching the todos!', error);

      });

  }, []);


  const handleAddTodo = (e) => {

    e.preventDefault();

    axios.post('http://localhost:3001/api/todos', { text: newTodo })
  };

      .then(response => {
        setTodos([...todos, response.data]);
        setNewTodo('');
      })
      .catch(error => {
        console.error('There was an error adding the todo!', error);
      });
  };

  const handleToggleTodo = (id) => {
    const todo = todos.find(todo => todo.id === id);
    axios.put(`http://localhost:3001/api/todos/${id}`, { completed: !todo.completed })
      .then(response => {
        setTodos(todos.map(todo => todo.id === id ? response.data : todo));
      })
      .catch(error => {
        console.error('There was an error updating the todo!', error);
      });
  };

  const handleDeleteTodo = (id) => {
    axios.delete(`http://localhost:3001/api/todos/${id}`)
      .then(() => {
        setTodos(todos.filter(todo => todo.id !== id));
      })
      .catch(error => {
        console.error('There was an error deleting the todo!', error);
      });
  };

  return (
      <div className="app">
        <h1>Todo List</h1>
        <form onSubmit={handleAddTodo}>
          <input
            type="text"
            value={newTodo}
            onChange={(e) => setNewTodo(e.target.value)}
            placeholder="Add a new todo"
          />
          <button type="submit">Add</button>
        </form>
        <ul>
          {todos.map(todo => (
            <li key={todo.id} className={todo.completed ? 'completed' : ''}>
              <span onClick={() => handleToggleTodo(todo.id)}>{todo.text}</span>
              <button onClick={() => handleDeleteTodo(todo.id)}>Delete</button>
            </li>
          ))}
        </ul>
      </div>
    );
  }


  export default App;
```

Step 4: Run the Application

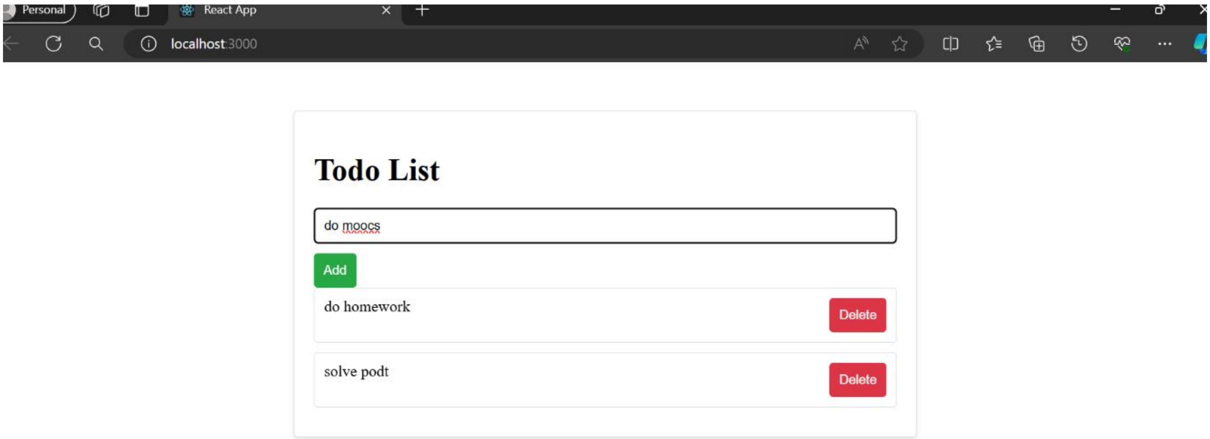
1. Ensure your backend server is running by executing `node server.js` in the project root directory.
2. Start the React application:

```
sh
cd client

npm start
```



Todo List

Add a new todo



Todo List

do moocs

do homework

solve podt