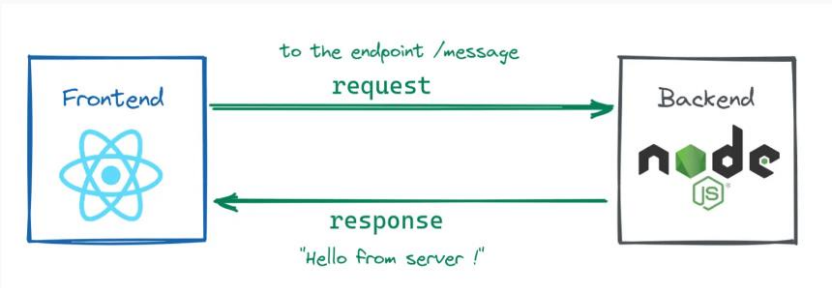
Ex-no:9 M. Rohith
06-05-2024 3122 21 5001 085

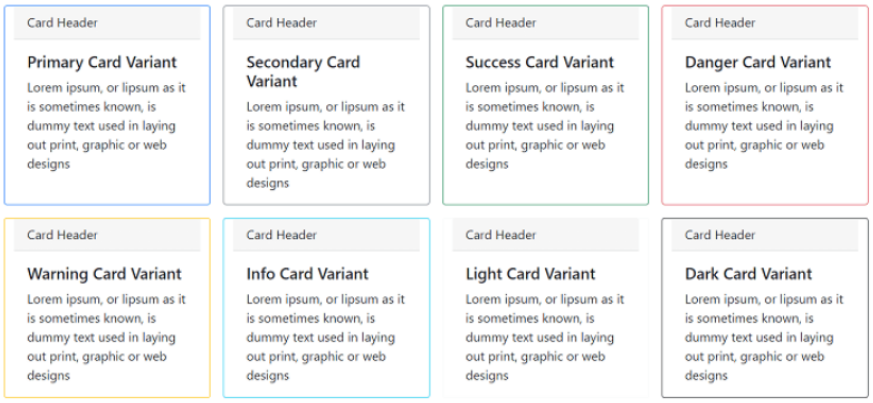
UCS 2611 Internet Programming Lab

Exercise 9. Two Tier Architecture Application using React and NodeJs

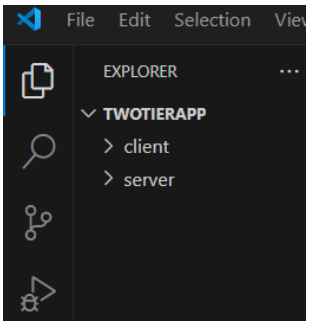
1. Develop two tier architecture application using React and NodeJs - Retrieve the information that store in a node and display it on the browser using two tier architectures.



Task: Store student details (such as name, rollno, contact-info) as array of objects in a node. Retrieve the details of each student from node and display on browser in the form of the cards as shown below image. In the place of Card Header and Card Content - display student roll and student name, contact-info.



Folder Structure:



1. **Create an empty folder and name as "twotierapp"**
2. **In this folder, create an empty folder and name as "server"**
    a. **Building the server using nodejs- backend:**
        i. Installing the dependencies
            1. Open a new terminal and move to server folder using – ***cd* server**
            2. Initialize a Node.js project in it- **npm init -y**
            3. Let's install the required dependencies for our application.

M. Rohith
3122 21 5001 085

npm install express cors

4. For development purposes, we will install nodemon as a development dependency.
   npm install nodemon --save-dev
5. create a new file and name it as “Server.js”- write backend logic

Sample Code: (Server.js)

```
const express = require("express");
const cors = require("cors");
const app = express();

app.use(cors());
app.use(express.json());

app.get("/message", (req, res) => {
  res.json({ message: "Hello from server!" });
});

app.listen(8000, () => {
  console.log(`Server is running on port 8000.`);
});
```

6. **Run the code in terminal**

```
PS D:\reactjs\twotierapp\server> npm start

> server@1.0.0 start
> node server.js

Server is running on port 8000.
```

3. **Building the client-side application using React (front-end)**
   - First, we will create a React application. Open your terminal and run the following command to create a React application.
   - **npx create-react-app client**
   - Go to App.Js and write client-side logic for fetching the data from node and to render on the browser.

Sample code: (App.js)

M. Rohith
3122 21 5001 085

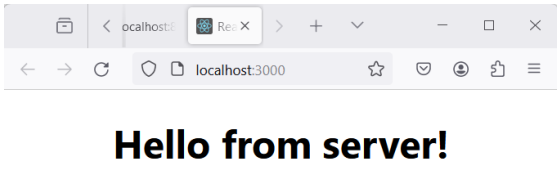
```
import React, { useState, useEffect } from "react";
import "./App.css";

function App() {
  const [message, setMessage] = useState("");

  useEffect(() => {
    fetch("http://localhost:8000/message")
      .then((res) => res.json())
      .then((data) => setMessage(data.message));
  }, []);

  return (
    <div className="App">
      <h1>{message}</h1>
    </div>
  );
}

export default App
```

Run the App.js in terminal: npm start

Output For Above Sample Code:

Ex-no:9
06-05-2024

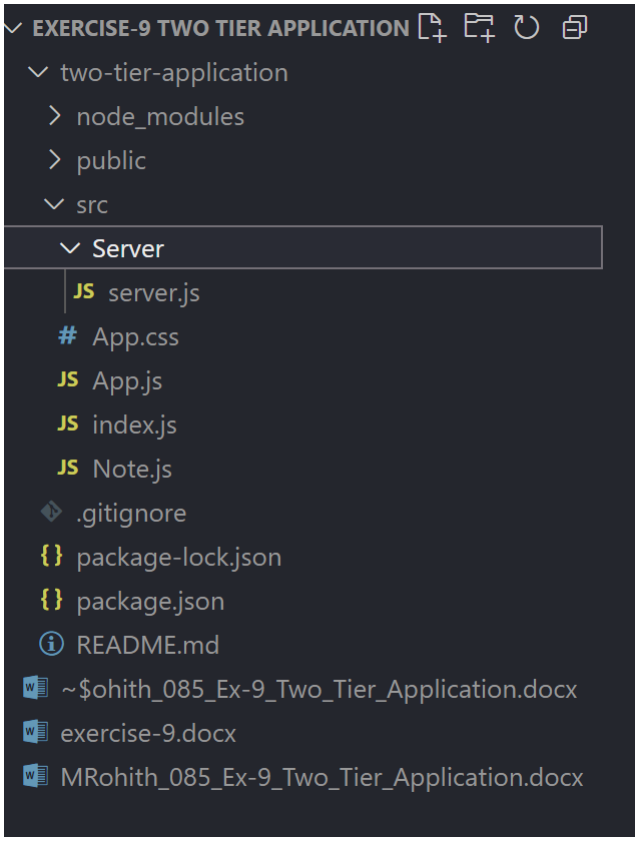
M. Rohith
3122 21 5001 085



Best Practices to be followed:

1. Design before coding
2. Incremental coding
3. Usage of proper naming convention
4. Usage of Comments to the code
5. Indentation of code

Hierarchy:



Ex-no:9 M. Rohith
06-05-2024 3122 21 5001 085

Program code:

server.js

```
const express = require("express");
const cors = require("cors");
const app = express();

app.use(cors());
app.use(express.json());

const data = [
 {
  name: "Alice",
  rollno: "A123",
  phone: "555-123-4567",
  email: "alice@example.com"
 },
 {
  name: "Bob",
  rollno: "B456",
  phone: "555-987-6543",
  email: "bob@example.com"
 },
 {
  name: "Charlie",
  rollno: "C789",
  phone: "555-555-5555",
  email: "charlie@example.com"
 },
 {
  name: "David",
  rollno: "D012",
```

```
  phone: "555-111-2222",
  email: "david@example.com"
 },
 {
  name: "Eva",
  rollno: "E345",
  phone: "555-333-4444",
  email: "eva@example.com"
 },
 {
  name: "Frank",
  rollno: "F678",
  phone: "555-666-7777",
  email: "frank@example.com"
 },
 {
  name: "Grace",
  rollno: "G901",
  phone: "555-888-9999",
  email: "grace@example.com"
 },
 {
  name: "Hannah",
  rollno: "H234",
  phone: "555-444-5555",
  email: "hannah@example.com"
 },
 {
  name: "Isaac",
  rollno: "I567",
  phone: "555-777-8888",
```

```
  email: "isaac@example.com"
 },
 {
  name: "Julia",
  rollno: "J890",
  phone: "555-222-3333",
  email: "julia@example.com"
 }
];

app.get("/message", (req, res) => {
 res.json(data);
});

app.listen(8000, () => {
 console.log("Server is running on port 8000.");
});
```

index.js

```
import React from 'react';
import ReactDOM from 'react-dom/client';
import App from './App';

const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(
 <React.StrictMode>
  <App />
 </React.StrictMode>
);
```

App.js

```
import React, { useState, useEffect } from "react";
import "./App.css";
import Note from "./Note";

function App() {
 const [message, setMessage] = useState([]);

 useEffect(function effect() {
  fetch("http://localhost:8000/message")
   .then((res) => res.json())
   .then((data) => setMessage(data));
 }, []);

 console.log(message);

 return (
  <div className="App">
   <div className="Header">
    <h1>Student Data</h1>
   </div>
   <div className="Note--master">
    {message.map((msg, i) => {
     return (
      <div>
      <Note
       name={msg.name}
       roll={msg.rollno}
       email={msg.email}
       phone={msg.phone}
       key={i}
      />
```

```
          <br/><br/>
          </div>
        );
       })}
     </div>
   </div>
 );}
export default App;
```

App.css

```
.Header{
  text-align: center;
  border:5px solid orange;
  background-color: orange;
  color:white;
}
.Note--master{
  text-align: center;
  border:5px solid orange;
  width: 700px;
  /* height:200px; */
  padding:40px;
  margin:10px;
  display:flex;
  flex-wrap: wrap;
}
```

Note.js

```
import React from "react";
import "./App.css"
```

```
export default function Note({name,roll,phone,email}){
  return(
    <div className="Note">
      <p>{name}</p>
      <p>{roll}</p>
      <p>{phone}</p>
      <p>{email}</p>
    </div>
  )
}
```

Output:

```
PS C:\Rohith\Backup\Desktop\SEM 6\Internet Programming Lab\Exercise-9 Two tier application\two-tier-application\src\Serv
er> node server.js
Server is running on port 8000.
```

```
PS C:\Rohith\Backup\Desktop\SEM 6\Internet Programming Lab\Exercise-9 Two tier application\two-tier-application\src> npm
 start
```

```
Compiled successfully!

You can now view two-tier-application in the browser.

  Local:            http://localhost:3000
  On Your Network:  http://192.168.56.1:3000

Note that the development build is not optimized.
To create a production build, use npm run build.

webpack compiled successfully
```

Ex-no:9
06-05-2024

M. Rohith
3122 21 5001 085

Student Data

Alice	Bob	Charlie	David	Eva
A123	B456	C789	D012	E345
555-123-4567	555-987-6543	555-555-5555	555-111-2222	555-333-4444
alice@example.com	bob@example.com	charlie@example.com	david@example.com	eva@example.com

Frank	Grace	Hannah	Isaac	Julia
F678	G901	H234	I567	J890
555-666-7777	555-888-9999	555-444-5555	555-777-8888	555-222-3333
frank@example.com	grace@example.com	hannah@example.com	isaac@example.com	julia@example.com