FULL STACK (NEXT.JS & EXPRESS.JS) Project

1. Backend (server.js)_Express.js code-

```
// backend-server.js

const express = require("express");

const cors = require("cors");

const bodyParser = require("body-parser");


const app = express();


// Enable CORS for Next.js frontend (port 3000 in your case)
app.use(
  cors({
    origin: "http://localhost:3000",
    methods: ["GET", "POST"],
    allowedHeaders: ["Content-Type"],
  })
);


// Parse JSON bodies
app.use(bodyParser.json());


// Dummy user
```

```javascript
const USER = { email: "test@example.com", password: "123456" };

// Dummy products
const products = [
  { id: 1, name: "Laptop", price: 75000 },
  { id: 2, name: "Smartphone", price: 40000 },
  { id: 3, name: "Headphones", price: 5000 },
];

// Login API
app.post("/api/login", (req, res) => {
  console.log("Login attempt:", req.body); // Debugging log
  const { email, password } = req.body;
  if (email === USER.email && password === USER.password) {
    return res.json({ success: true });
  }
  res.status(401).json({ success: false, message: "Invalid credentials" });
});

// Products API
app.get("/api/products", (req, res) => {
  res.json(products);
});
```

```
const PORT = 5000;

app.listen(PORT, () =>

  console.log(`    Express API running on http://localhost:${PORT}`)

);
```

2. Frontend(Next.js) code-

2.1Homepage code-

```
// src/app/page.js

export default function HomePage() {

  return (

    <main style={{ textAlign: "center", marginTop: "50px" }}>

      <h1>Welcome to My App    </h1>

      <a href="/login">Go to Login</a>

    </main>

  );

}
```

2.2   LoginPage code-

```
3.   // src/app/login/page.js

    "use client";


    import { useState } from "react";
```

```jsx
export default function LoginPage() {

    const [email, setEmail] = useState("");

    const [password, setPassword] = useState("");

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


    const handleLogin = async (e) => {

      e.preventDefault();


      try {

        const res = await fetch("http://localhost:5000/api/login", {

          method: "POST",

          headers: {

            "Content-Type": "application/json",   //     needed

          },

          body: JSON.stringify({ email, password }),  //    body must be JSON
string

        });


        const data = await res.json();


        if (res.ok) {

          setMessage("    Login successful!");

          window.location.href = "/products"; // redirect to products page

        } else {
```

```
      setMessage("    " + data.message);

    }

  } catch (error) {

    setMessage("    Something went wrong, try again.");

    console.error("Login error:", error);

  }

};


  return (

    <main style={{ display: "flex", justifyContent: "center", alignItems: "center",
height: "100vh" }}>

      <form onSubmit={handleLogin} style={{ display: "flex", flexDirection:
"column", gap: "1rem" }}>

        <h1>Login Page    </h1>

        <input type="email" placeholder="Email" value={email} onChange={(e) =>
setEmail(e.target.value)} required />

        <input type="password" placeholder="Password" value={password}
onChange={(e) => setPassword(e.target.value)} required />

        <button type="submit">Login</button>

        <p>{message}</p>

      </form>

    </main>

  );

}
```

2.3 ProductsDisplay code-

```
//src/products/page.js

"use client";

import { useEffect, useState } from "react";

export default function ProductsPage() {
  const [products, setProducts] = useState([]);

  useEffect(() => {

    fetch("http://localhost:5000/api/products") // backend API

      .then((res) => res.json())

      .then((data) => setProducts(data))
```

```jsx
      .catch((err) => console.error("Error
fetching products:", err));

  }, []);


  return (

    <main style={{ padding: "2rem" }}>

      <h1>    Products</h1>

      {products.length === 0 ? (

        <p>Loading products...</p>

      ) : (

        <ul>

          {products.map((p) => (

            <li key={p.id}>

              {p.name} - ₹{p.price}
```

```
            </li>

          ))}

        </ul>

      )}

    </main>

  );

}
```
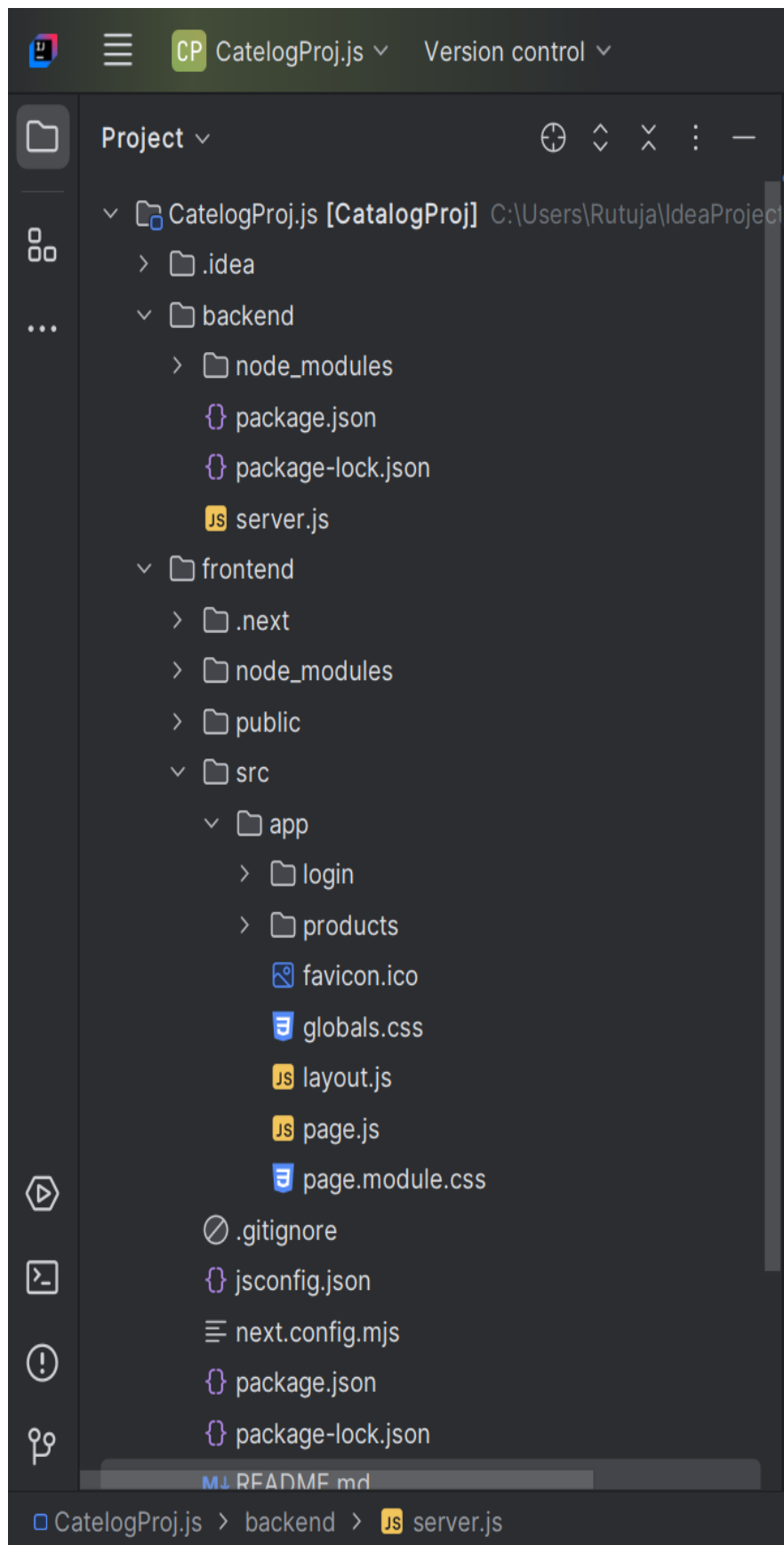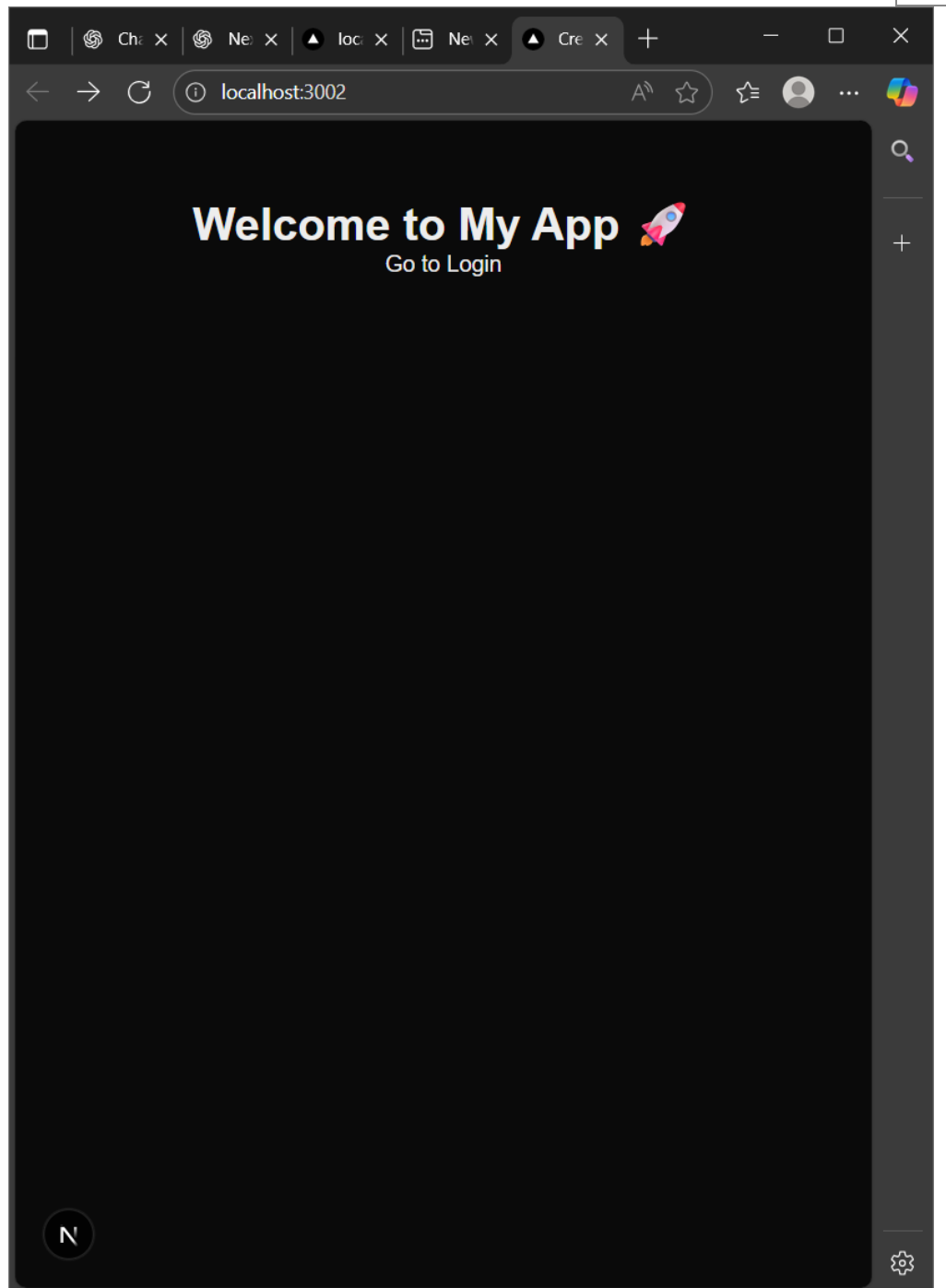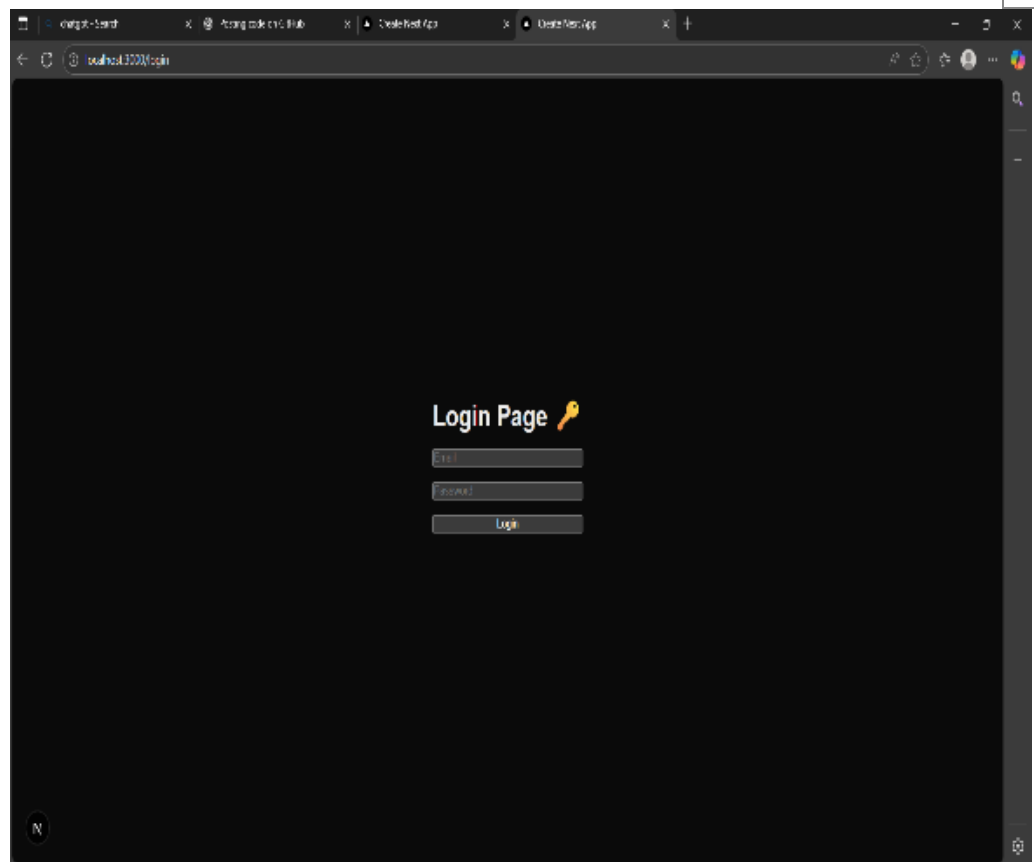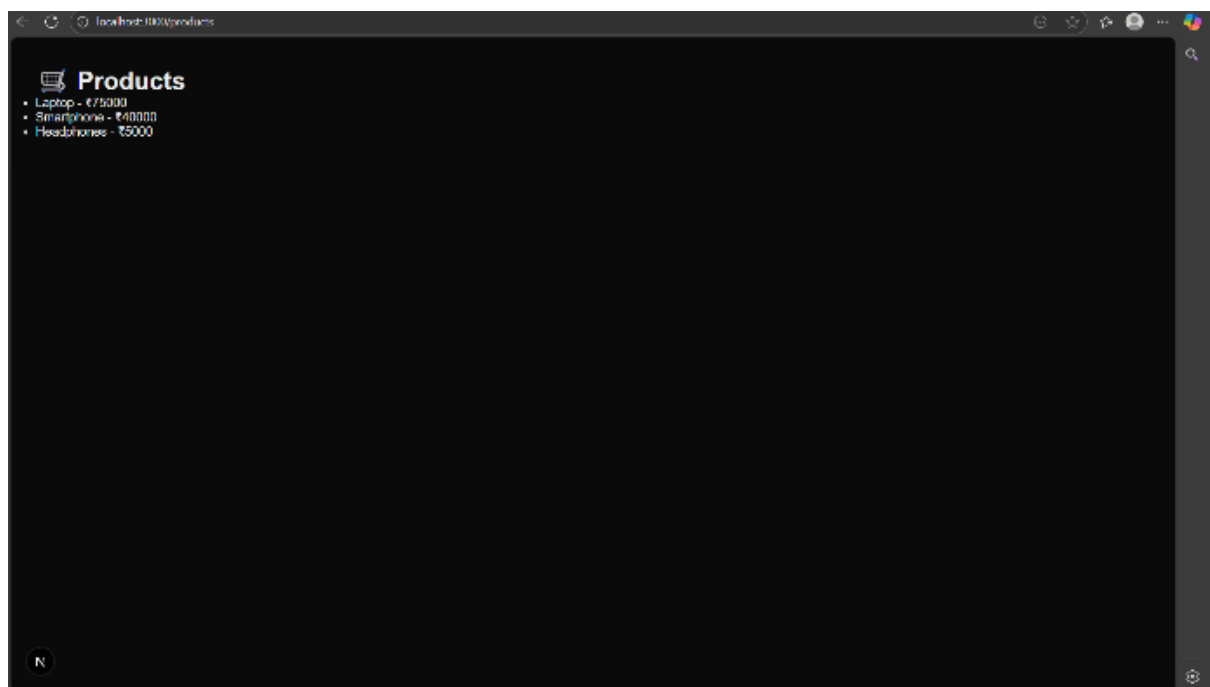
- Project structure

- Output-Homepage

● Output-Login Page

- ## Output-Product List

I executed this project using the IntelliJ coding platform.

   o   <u>Terminal Promts(Intellij)-</u>

1. Backend prompt-

   cd backend
   Node server.js

   //<u>Success when</u> -"   Express API running on http://localhost:5000"

2. Frontend prompt-

   cd frontend
   npm run dev

   //<u>success when</u> -" > frontend@0.1.0 dev
   > next dev

       Next.js 15.4.6
   - Local: http://localhost:3000
   - Network: http://192.168.29.110:3000
     Starting...
     Ready in 3.8s
    Compiling / ...
    Compiled / in 3.7s (537 modules)"

o   <u>To check API manually-</u>

curl -X POST http://localhost:5000/api/login -H "Content-Type: application/json" -d "{\"email\":\"test@example.com\",\"password\": \"123456\"}"

put this command in COMMAND PROMPT to ensure "success" in API checking

<u>Links to see output at browser-</u>

- Homepage Link - http://localhost:3000/
- LoginPage Link - http://localhost:3000/login
- Productpage Link - http://localhost:3000/products

<u>Folder link-</u>

https://1drv.ms/u/c/e874148fcc41f070/EXYJXz0HQi9FswEOn9PJ-u8BguwslO38ycxBaA38tJgw5g?e=NglaRb

THANK YOU !!