

■ neon-notion – Full Code & Directory Structure

■ Backend Structure

```
backend/ ■■■ index.js # Main Express server ■■■ database/ ■■■ mongoose.js # MongoDB connection (hardcoded URI) ■■■ schema.js # Page schema (title, content, tags, tasks, timestamp)
```

■ Frontend Structure

```
frontend/ ■■■ public/ ■ ■■■ index.html ■■■ src/ ■■■ App.js # Main React component with sidebar/editor ■■■ App.css # Full custom aesthetic styling ■■■ index.js # ReactDOM.render()
```

■ backend/index.js

```
const express = require("express"); const mongoose = require("./database/mongoose");
const Page = require("./database/schema"); const cors = require("cors"); const app = express();
app.use(express.json()); app.use(cors()); // CRUD Routes
app.get("/pages", async (req, res) => { const pages = await Page.find(); res.json(pages); });
app.post("/pages", async (req, res) => { const page = new Page(req.body); await
page.save(); res.json(page); });
app.put("/pages/:id", async (req, res) => { const page =
await Page.findByIdAndUpdate(req.params.id, req.body, { new: true });
res.json(page); });
app.delete("/pages/:id", async (req, res) => { await
Page.findByIdAndDelete(req.params.id); res.json({ success: true }); });
app.listen(5000, () => console.log("Server started on port 5000"));
```

■ backend/database/mongoose.js

```
const mongoose = require("mongoose");
mongoose.connect("mongodb://localhost:27017/neon-notion", { useNewUrlParser: true,
useUnifiedTopology: true, });
module.exports = mongoose;
```

■ backend/database/schema.js

```
const mongoose = require("mongoose");
const pageSchema = new mongoose.Schema({ title:
String, content: String, tags: [String], tasks: [{ text: String, completed: Boolean }],
timestamp: String, });
module.exports = mongoose.model("Page", pageSchema);
```

■ frontend/src/App.js (Overview)

```
import React, { useState, useEffect } from "react"; import axios from "axios"; import  
"./App.css"; function App() { const [pages, setPages] = useState([]); const  
[selectedPage, setSelectedPage] = useState(null); useEffect(() => {  
  axios.get("http://localhost:5000/pages").then((res) => setPages(res.data)); }, []); //  
  Render sidebar and editor... } export default App;
```

■ What's Included

- ✓ Full backend (Express + MongoDB)
- ✓ Full frontend (React app)
- ✓ .NEON sidebar branding
- ✓ Modern custom CSS theme
- ✓ README.md and project structure
- Project Name: neon-notion
- Author: Aditya Singh
- License: MIT