

CipherStudio Assignment Document

Title: Build CipherStudio - Your Browser-Based React IDE

1. Objective

Your task is to design and implement a browser-based React IDE called CipherStudio, where users can:

1. Create and manage multiple files for a React project.
2. Write React code in the browser.
3. Run and preview their project live inside the browser.
4. Save and reload their projects to continue later.

The goal is to simulate a real online development environment similar to platforms like [NextLeap.js](#) or [CodeSandbox](#), while testing your full-stack design thinking, frontend skills, and integration ability.

2. Features to Implement

Core Features (Required):

- File Management: Users can create, delete, and organize project files.
- Code Editor: A rich code editor to write React components (Monaco Editor or Sandpack).
- Live Preview: Display the React project output in real-time as the code changes.
- Save & Load Projects: Users can save their project state and reload it later from localStorage with the help of projectId.
- UI/UX: Clean and intuitive interface

Optional/Bonus Features:

- Theme switcher (dark/light).
- Rename files/folders
- Login/Register
- Deployment/Hosting
- Autosave feature (toggle)
- Responsive UI: Usable on desktop and tablet screens.

You may use open-source libraries (like [Sandpack](#)) for React code execution. The goal is integration and feature implementation, not building a compiler from scratch.

3. Tech Stack

Layer	Recommended Technology
Frontend	React / Next.js
Code Execution	Sandpack (CodeSandbox)
Editor	Monaco Editor or Sandpack editor
Database	MongoDB (Atlas preferred) and AWS S3 for files
Backend	Node.js / Express.js (for project persistence)
Deployment	Vercel (Frontend) + Render/Railway/Cyclic (Backend)

You may choose alternatives if needed, but ensure they support the required functionality.

4. Deliverables

1. GitHub Repository
 - Frontend code and backend code
 - README.md explaining project structure and features (optional)

2. Live Demo URL (Optional)
 - Vercel deployment for frontend
 - Backend deployed on Render/Railway
 3. Project Demo (Optional)
 - A 2–3 minute video demonstrating the IDE: creating files, writing code, running the project, and saving/loading functionality.
 4. Documentation (Optional)
 - Describe the high-level architecture
 - Explain any major design decisions
-

5. Timeline

- Duration: **3 days**
 - Focus on completing the core MVP first before adding optional features.
 - Prioritize usability, structure, and working preview functionality over exhaustive polish.
-

6. Assessment Criteria

Category	Weight
Core functionality	40%
Code structure & readability	20%
UI/UX clarity	20%
Creativity & additional features	10%
Documentation & demo	10%

7. Reference & Helping Guide

To make it easier for you to get started, we have prepared a detailed helping guide containing:

- Suggested project architecture and layout
- Recommended MongoDB schema for storing projects
- Sample Sandpack integration code snippets
- UI design references

Note: The helping guide is meant to clarify the project requirements and help you complete it faster. You are not required to follow it exactly - feel free to innovate and add features.

https://docs.google.com/document/d/1doDjOHvLLJliqlrXF5td_AgJF2RUdNy_mTUF0gamSFXY/edit?usp=sharing

8. Submission Form

Google Form Link:

https://docs.google.com/forms/d/e/1FAIpQLSccT4cl-g-_87Zg0gu_xXwW6mq_u8-MqOJeSY794-fN_aQaZnA/viewform?usp=header