

PROTOCOL01 | Horizon 2025, IIITV-ICD

Title: Real-Time Collaborative Code Editor

Team Name : Protoc01

TEAM INNOVATORS

Members	Student ID
Kshitiz Gangwar	202311045
Aryan Panday	202311018
Naman Gautam	202311055
Manini Rajbhar	202311049
Santanu	202311076

OBJECTIVE AND SOLUTION OVERVIEW

Objective:

Developers often face challenges in real-time collaboration on code, leading to inefficiencies in team-based projects.

Existing editors may lack seamless version control, offline editing, and real-time sync capabilities.

Solution Overview:

- A **web-based** code editor that allows multiple users to **edit** the same code file simultaneously.
- Features **real-time synchronization**, **built-in version control**, and **offline editing support** with automatic sync once reconnected

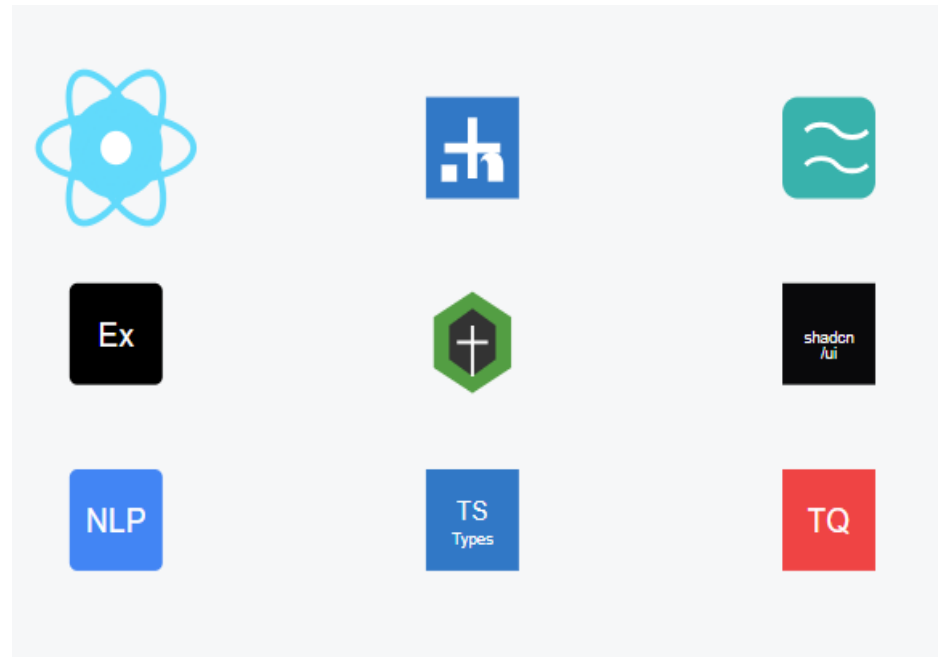
KEY FEATURES & TECHNOLOGIES

KEY FEATURES

Real-time multi-user collaboration
Conflict resolution & version control
Offline mode with automatic sync
Syntax highlighting & code formatting
Secure authentication & access control

Technologies Used:

- **Frontend:** React.js, Tailwind CSS
- **Backend:** Node.js, Express.js
- **Database:** MongoDB (store projects, users, edits)
- **Collaboration Engine:** WebSockets, Operational Transform (OT) / CRDT
- **Authentication:** JWT (JSON Web Tokens), OAuth



Demo/Workflow:

Demo/Workflow:

User logs in → Dashboard loads

Opens a project → Joins live session

Multiple users edit code → Changes sync in real-time

Edits are saved → Version history maintained

Offline edits auto-sync once online

CHALLENGES AND LEARNINGS

Challenges Faced:

- Accurately extracting relevant skills from unstructured resume text
- Ensuring type consistency between frontend forms and backend data storage
- Creating an algorithm that balances technical matching with soft skills
- Building an intuitive UI that simplifies complex matching data

Key Learnings:

- The importance of unified type definitions between frontend and backend
- Value of user-centric design in complex technical applications
- Benefits of iterative development and testing with real resume data
- Balancing automation with human oversight in recruitment processes

🏠

🔗

🔗

No codespaces available

🔑

👤

🔔

Untitled

EXPLORER

📁 Folder 1

📄 file1.js

📄 file2.js

📁 Folder 2

1 // Sample file content

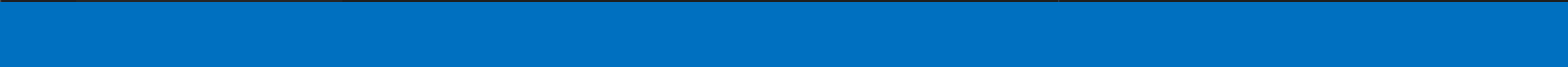
2 console.log("Hello world");

▶

Terminal

Welcome to the terminal! Type your code and press Run.

\$



```
1 // Sample file content
2 console.log("Hello world");
```

Create New Codespace



Codespace Name

Enter codespace name

Logo URL

Enter logo URL

Access Key

Enter access key



Drag and drop files here or click to browse

<> Create Codespace

Connections



John Doe



Jane Smith



Access Key



+

New Codespace

Pull Request

>_

Untitled

EXPLORER

Folder 1

file1.js

file2.js

Folder 2

1

// Sample file content

2

console.log("Hello world");

Notifications

Connection Requests

Alex Johnson

Wants to connect

Accept

Decline

Codespace Requests

Feature: Add Authentication

by Sarah Miller

Review

@SIH Idea submission - Template

10