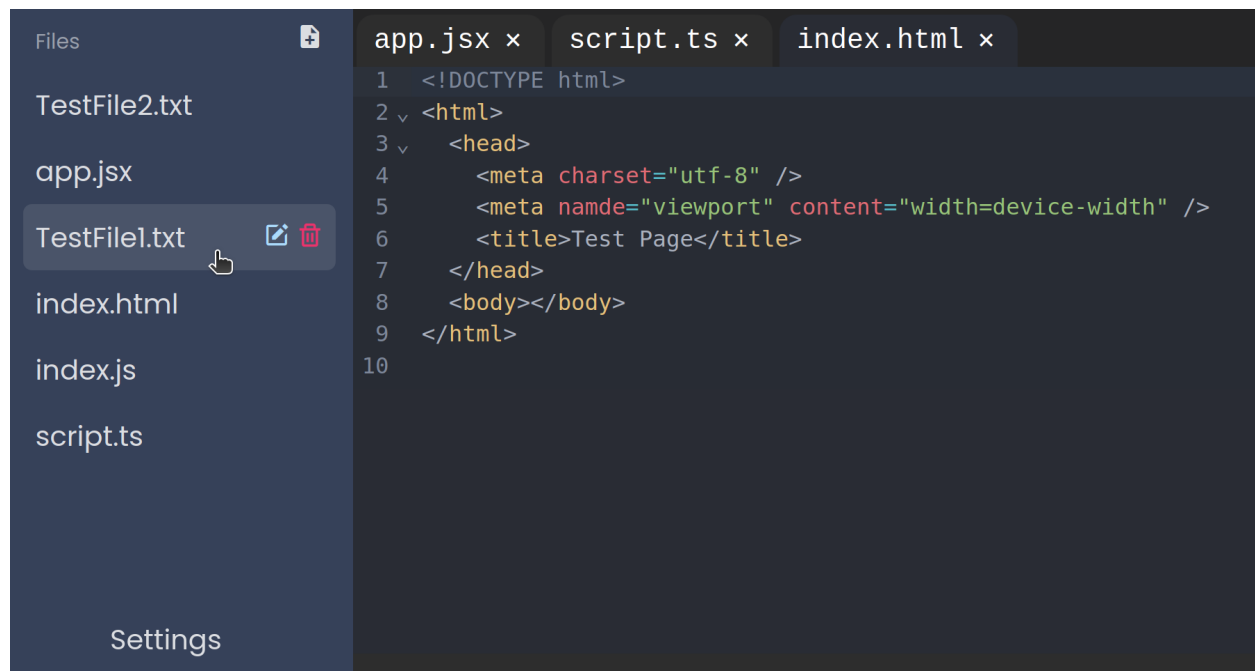


VZCode

<https://github.com/vizhub-core/vzcode>

Description

VZCode is a state-of-the-art multiplayer code editing platform designed to provide a real-time collaborative development experience. It provides a browser-based environment where users can simultaneously edit, allowing for instant collaboration on projects. With features such as file listing, real-time collaboration, and syntax highlighting, VZCode is a formidable solution for teams looking to streamline their development workflows.



Stack

- Front End: [React](#), [Vite](#), [CodeMirror 6](#)
- Back End: [NodeJS](#), [Express](#), [ShareDB](#), [JSON1 Operational Transform](#)

Goals

The project aims to:

- Offer a feasible alternative to VSCode + Live Share for frontend development
- Support instant feedback through hot reloading
- Facilitate easy project-specific IDE embedding
- Enhance user experience with advanced features
- Keep improving based on feedback
- Serve as a core for [VizHub's](#) next-gen editor

Milestones

September

- Team onboarding and development environment setup
- Robust error handling and presentation of syntax errors
- Usability improvements for renaming files
- Set up live demo site where anyone can try VZCode

October

- Intelligent Autocompletions (using TypeScript language server)
- AI-Assisted Coding leveraging OpenAI
- Add interactive widgets from [codemirror-interact](#)

November

- Usability improvements for working with directories (rename, moving files)
- Make VZCode more like VSCode using existing CodeMirror extensions
- Presence improvements (colored cursors, selections, usernames)
- Usability testing & dogfooding VZCode for VZCode development

December

- Documentation and Demo Videos
- Final presentation and write-up
- Final code commits

Team

• Curran Kelleher	curran@vizhub.com	curran
• Jack Hankin	Hankij@rpi.edu	Darmin-Geek
• Dominick Iadevaia	iadevd@rpi.edu	DomaDude123
• Evan Whymark	whymae@rpi.edu	EvWhymark
• Amanda Ruan	ruana2@rpi.edu	aruan20
• Ngawang Ghongwatsang	ghongn@rpi.edu	nghongwatsang
• Michael Krar	krarm@rpi.edu	michaelkrar
• Jordan Lewenthal	lewenj@rpi.edu	jlewenthal2004

Individual Milestones

September

- (Ngawang) Robust error handling and presentation of syntax errors
- (Mike) Usability improvements for renaming files

October

- (Dominick & Evan) Intelligent Autocompletions (using TypeScript language server)
- (Jordan & Ngawang) AI-Assisted Coding leveraging OpenAI
- (Jack) Add interactive widgets from [codemirror-interact](#)

November

- (Amanda & Evan) Usability improvements for working with directories (rename, moving files)
- (Jordan) Make VZCode more like VSCode using existing CodeMirror extensions
- (Evan & Amanda) Presence improvements (colored cursors, selections, usernames)

December

- (everyone) Documentation and Demo Videos
- (everyone) Final presentation and write-up
- (everyone) Final code commits

Specific Individual Goals

Jack Hankin

- Beginning of semester
 - Interactive widget that allows the user to drag to change a Vec2's input values
 - Interactive widget that is a color picker that can change rgb, hsl, and hex colors in code
 - Interactive widget that allows the user to change a rotation by dragging
 - Learn more about React
- Middle of the semester
 - Interactive widget that allows cycling through different options from a union of strings type
- End of the semester
 - Interactive widget that allows cycling through different files of the same extension within a directory

Ngawang Ghongwatsang

- Beginning of semester
 - Become comfortable with JS/React
 - Become comfortable with Git/ Github
- Mid semester
 - Robust error handling and presentation of syntax errors
- End of semester
 - AI-Assisted Coding leveraging OpenAI

Dominick Iadevaia

- Beginning of semester
 - Become comfortable with JS/React, HTML and CSS.
 - Become comfortable with Git/ Github
 - Intelligent autocompletions using Typescript
 - Renaming directories
- Mid semester
 - Deleting directories
 - Presence: Usernames
- End semester
 - Presence: Follow user
 - Drag and drop files between directories

Amanda Ruan

- Beginning of semester
 - Become familiar with repository
 - Refresh on JavaScript/React
 - Presence: Show Selection
- Mid semester
 - Presence improvements (colored cursors, selections, usernames)
 - Usability improvements for working with directories (rename, moving files)
- End of semester
 - Continue improving usability and presence

Evan Whymark

- Beginning of semester
 - Become familiar with react/css/repository
 - Improve understanding of git/version control
 - Usability improvements
- Mid semester
 - Intelligent autocompletions using typescript
- End of semester
 - Presence improvements (Colored cursor, selections, usernames, similar to google docs)

Michael Krar

- Beginning of semester
 - Become familiar with codebase/contribution practices
 - Usability/UX improvements (better file renaming/creation UI, Shrink editor when lines are long, CTRL-click to see definition/references)
- Mid semester
 - Project CI/CD Improvements (.yaml github actions/workflows)
 - Usability improvements(active file in url)
- End of semester
 - Git/Github Integration (version control tab, merge conflict editor)
 - Live demo site(github pages? Netlify?)

Jordan Lewenthal

- Beginning of semester
 - Make VZCode more like VSCode using existing CodeMirror extensions
 - Jump to Definition

- Mid Semester
 - Deep Linking
 - AI-Assisted Coding leveraging OpenAI
- End of Semester
 - Continue working on AI-assisted coding
 - Usability and bug fixes

Spaces

- Github: <https://github.com/vizhub-core/vzcode> | [Kanban Board](#)
- Discord Channel within RCOS

Contribution Guidelines

- All team members will be added to the GitHub repository, no need to fork to PR
- Please self-assign the issue when you pick it up, to avoid duplicated effort
- Please also move the issue to the "In Progress" Kanban column
- Create a draft PR as early as possible to surface work in progress
- Run Prettier on the code before committing
- Students will review each other's PRs
- Curran will do final review, cleanup, and merge
- Use the "Closes #241" syntax to auto-close issues when PRs get merged
- Ideally 1:1 mapping between issues and pull requests
- If you want to work on something with no issue, create the issue first