

**Team Members:** Michael Davis, Lily Krohn, Chris Lopez, Kyle Marek, Zachary Mohling, Griffin Stout Advisor: Dr.Goce Trajcevski

# **Problem Statement**

The modern software development process calls for a growing dependence on remote development and collaboration. However, development normally occurs on separate computers that all need to have the same system environments and access rights. This puts strain on a team that can't physically work together, and their systems that need to be increasingly more powerful and connected.

# Solution

- Creating a collaborative setting where configuration and storage of projects are offloaded to the cloud.
- Offering joint, real-time editing of source files and GIT integration as well as project and team management, • Allowing easy collaboration for work-from-home and

geographically remote developers

# Intended Users

- Any frontend JS developer or team which would prefer to keep project files on a consistent development environment
- Intended use is for when individuals or teams who want to work on the same code at one time

• WIDE frees the user of having to install software or have multiple people working together on one computer.

# **Design Requirements**

## **Functional Requirements**

- Collaborative editing
- Built-in Git UI for version control
- Autosaving and source history
- Create/Import projects
- Authentication of users
  - for login, projects, teams
- Execution output and
- interaction
- Compile and serve
- artifacts

### Non-Functional Requirements

- Low text collaboration latency
- High scalability
- Quick bug and feature deployments
- High availability

## Constraints / Environment

### Constraints

### Environment

- Time main constraint
- Learning curves of
- new technologies
- Budget \$500
- Users can operate and run WIDE on their web browser of choice.

## **Design Approach**

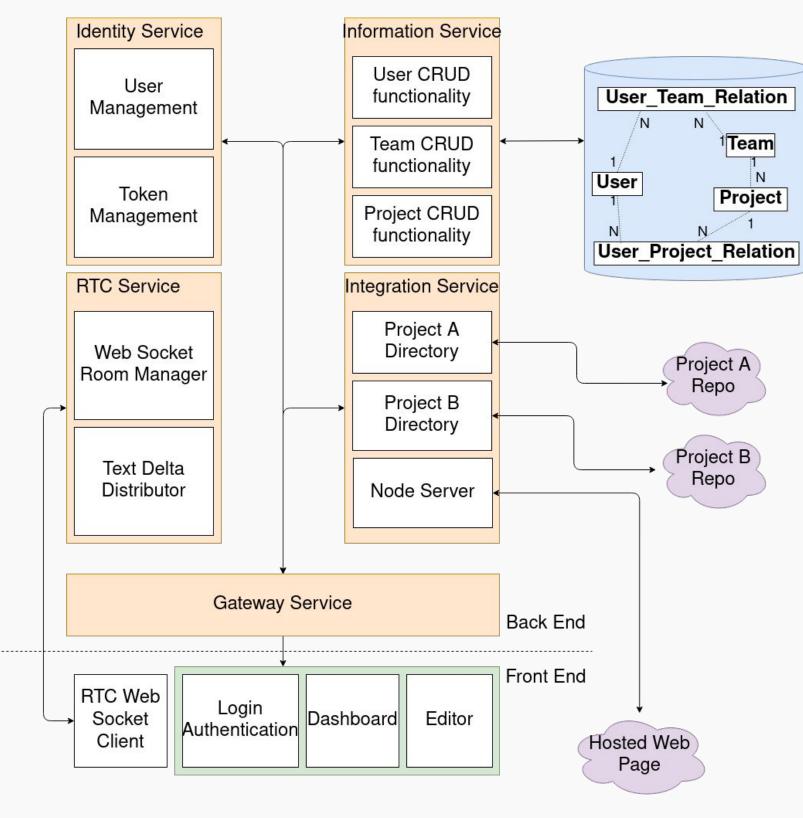
### **Frontend Components**

- Authentication / Login
  - Email validation, error handling
  - Confirmation email
  - Activation tokens and cookies
- Projects & Teams Page
  - Teams create / add / remove users from teams
  - Projects create / view projects **RTC Service**
  - Project description
- Monaco Editor & File Structure
  - Autocomplete & syntax highlighting
  - Fetches files from server and displays file tree

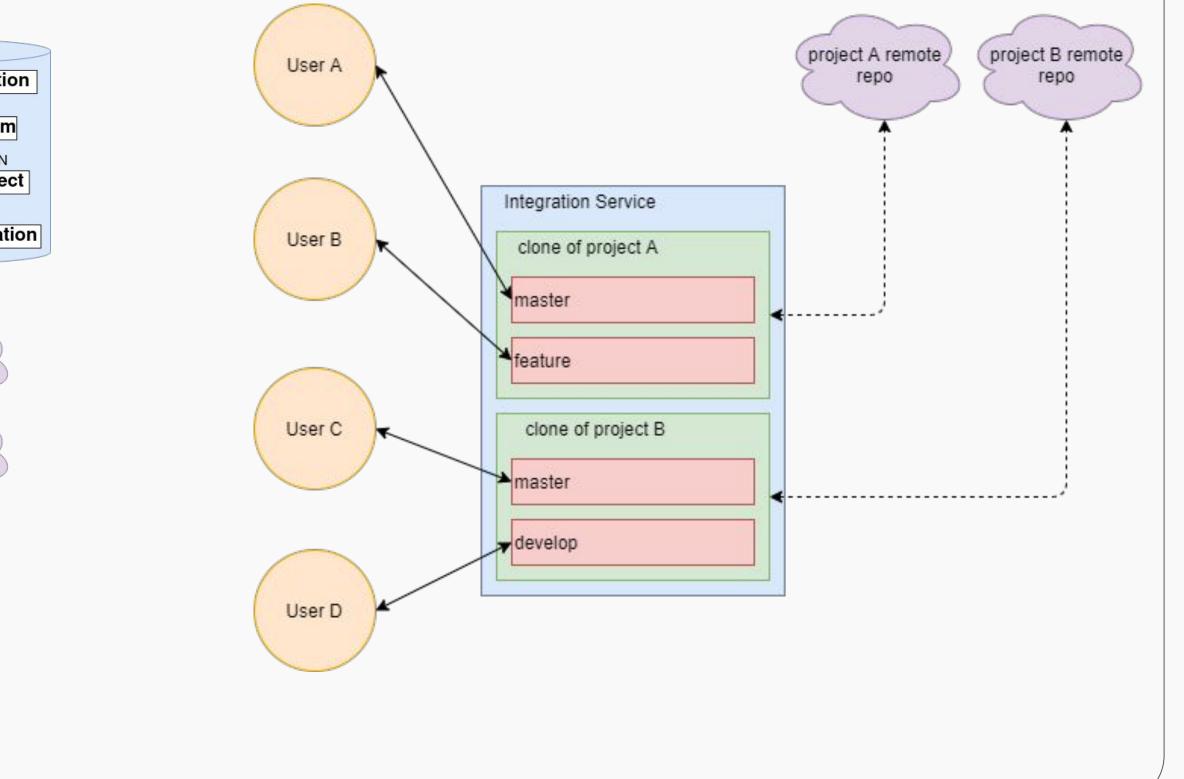
- **Backend Components**
- Identity Service
  - User & Session management
  - User authentication
- Information Service
  - Interface with the DB
  - Provide dashboard entries
  - DB entity management
- - Real-time communication
  - Cursor tracking, file saving
  - Integration Service
    - Host project files
    - Provide project structure
    - Handle git- requests



#### **Block Diagram**



#### **Integration Service Diagram**





### Frontend

**TypeScript / React** 

- Strong typing system avoids increasing runtime error opportunities
- Included composite and utility types Monaco Editor

### Backend

**Kubernetes** 

- Autonomous container orchestration
- Easy deployment & fault tolerance Docker
- Provides a reproducible production environment
- Efficient alternative to virtual machines

Golang

- Standard library with built-in concurrency primitives
- Statically compiled into a single, versionable binary

		Sign out					<b>Eventeed</b> Testing	
email Address *		<pre>&gt; SchedulingAlgorithm &gt; .git &gt; .git &gt; scheduling &gt; public &gt; public &gt; src App.js Container, Grid, S Typography, S src Algorithms.js Algorithms.js Algos.js Algos.</pre>					<ul> <li>Frontend Testing</li> <li>React-testing-library &amp; Jest</li> <li>TDD</li> <li>Backend Testing</li> </ul>	
Pa	assword *	■ setupTests.js       19       const         20       for         21       m         22       },         ■ READ ME.md       23         ■ debug.log       25         ■ package-lock ison       26	<pre>useStyles = makeStyles((theme) =&gt; ({ mControl: { iniWidth: 200, illInput: { idth: "60px", liumInput: { idth: "120px",</pre>				<ul><li>Postman</li><li>Manual Testing of</li></ul>	
	Don't have an account? Sign up	29 spa 30 b 31 }, 32 })); 33 34 funct 35 cor 36	<pre>spacer: {     height: theme.spacing(2),     }, })); function App() {     const classes = useStyles();     const [algosRan, setAlgosRan] = useState(false);</pre>					services in Go testing
000	Copyright © thewebide.com 2020.	Current Branch: $\checkmark$ $\textcircled{1}$ $\textcircled{1}$ $\textcircled{1}$ $\overbrace{0}$	<pre>ist [RMSfailed, setRMSfailed] = useState(false); ist [EDFfailed, setEDFfailed] = useState(false); ist [LLFfailed, setLLFfailed] = useState(false); ist [DMSfailed, setDMSfailed] = useState(false); ist [RMSpreemptions, setRMSpreemptions] = useState(0); SIGN OUT</pre>	000	WIDE	I	• TDD	SIGN OUT
								Oursed Bu
*	Teams		Owned By Anyone  NEW TEAM	*	Projects			Owned By Anyone   NEW PROJECT
1	Frontend Team This is the frontend team for WIDE.	Repo Link Test			<b>Kyle's Website Portfolio</b> This is Kyle's static website portfolio that he made for his resume.	Scheduling Algorithms This project is for CPRE 458 and it is a javascript project. It's purpose is to take in task sets and show the schedules that are		<b>Monaco Editor</b> The Monaco Editor is the code editor that powers VS Code. A good page describing the code editor's features is here. It is licensed
	Members of this team: Zachary Mohling (zmohling@gmail.com) Griffin Stout (gdstout27@gmail.com) Kyle Marek (kylemarektest@gmail.com) displayName (email@email.com) 	Members of this team: displayName (email@email.com)			Team: Frontend Project	produced by RMS, DMS, EL	S, EDF, and LLF under the MIT License and supports Classic	
					·			