Real-Time Code Collaboration - Project Documentation

Project Overview

This project is a **Real-Time Code Collaboration** tool built using Java and the Maven build system. It enables multiple users to collaborate on coding tasks in real time, with features such as user authentication, session management, and a shared coding environment.

Table of Contents

Technologies Used

- **Java 8+**
- **Maven** for dependency management
- **Spring Framework** for application configuration and controllers
- **JSP** with **JSTL** for the front-end
- **WebSocket** for real-time communication
- **MySQL** for data persistence
- **Tomcat 9+** as the servlet container
- **JUnit 5** for testing

Project Structure

src

├── main

│ ├── java

```
│   │   └── com
│   │       └── codecollab
│   │           ├── app
│   │           ├── config
│   │           ├── controllers
│   │           ├── dao
│   │           ├── model
│   │           └── utils
│   └── resources
│       └── application.properties
├── webapp
│   ├── createSession.jsp
│   ├── index.jsp
│   ├── joinSession.jsp
│   ├── user-form.jsp
│   ├── user-profile.jsp
│   └── workspace.jsp
└── test
    └── com
        └── codecollab
            └── dao
                └── UserDAOTest.java
```

Key Directories and Files

- **Controllers:** Handles HTTP requests and responses.
- **DAO:** Handles database operations.
- **Model:** Represents the business data.
- **JSP Pages:** Front-end views for different functionalities.

Features

1. **User Authentication:**
    - Sign-up and login functionality.
    - Session management.
2. **Real-Time Collaboration:**
    - Multiple users can edit the same code file in real time.
    - Changes are reflected immediately using WebSocket.
3. **Project and User Management:**
    - Create and manage coding projects.
    - User profiles with project history.
4. **Code Editor:**
    - Integrated code editor with syntax highlighting.
    - Save and retrieve code files.

Setup Instructions

Prerequisites

1. Install **Java 8+**
2. Install **Maven**
3. Install **MySQL**

Steps

1. Clone the repository:

git clone https://github.com/your-repo/real-time-code-collaboration.git

2. Navigate to the project directory:

cd real-time-code-collaboration

3. Configure the database:
    - Create a MySQL database:

CREATE DATABASE code_collab;

    - Update application.properties with your database credentials:

- spring.datasource.url=jdbc:mysql://localhost:3306/code_collab
- spring.datasource.username=root

spring.datasource.password=your_password

4. Install dependencies and build the project:

mvn clean install

5. Deploy the project:
    - Place the generated WAR file in the **Tomcat** webapps folder.
    - Start the Tomcat server.
6. Access the application: Open your browser and navigate to http://localhost:8080/real-time-code-collaboration.

Usage

1. **Register and Login**:
    - Access the application and create an account.
    - Login with your credentials.
2. **Create a Session**:
    - Navigate to the "Create Session" page.
    - Enter session details and invite collaborators.
3. **Join a Session**:
    - Use the session link or code to join an existing collaboration session.
4. **Collaborate in Real Time**:
    - Edit code with collaborators and see changes reflected immediately.

Future Enhancements

1. **Advanced Code Editor Features:**
    - Auto-completion, error detection, and code formatting.
2. **Version Control Integration:**
    - Integration with Git for version control.

3. **Enhanced Security:**
    - Add OAuth-based login (e.g., Google, GitHub).
4. **Scalability:**
    - Implement a microservices architecture for larger-scale collaboration.
5. **Mobile Support:**
    - Develop a mobile-friendly interface or app.