**Real-Time Chat Application Project Plan**

**Project Description**

The objective of this project is to develop a Real-Time Chat Application that enables users to engage in text conversations. This application will support features such as user registration and login, contact lists, and real-time message exchange. Emphasis will be placed on creating an intuitive user interface, ensuring messages are delivered and received with minimal latency, and maintaining data integrity and security.

**System Specifications**

**Hardware Requirements:**

A Laptop, MacBook or any other Linux, windows PC with minimum 8GB RAM

**Software Requirements:**

Python 3.8 or newer/ Flask or Django framework.

SQL Server as the database.

HTML5, CSS3, and JavaScript

WebSocket for real-time communication (\*)

Git for version control.

Visual Studio Code or similar IDE.

**8-Week Project Plan**

**Week 1: Project Setup and Initial Planning**

* Formation of group
* Idea generation

**Week 2: Project Setup and Initial Planning**

* Project planning in group / Brainstorming
* GitHub source control set up.
* Submitting project plan and detail for review

**Week 3: Requirements and System Specification**

* Define detailed functional and non-functional requirements. Document specific scenarios covering user interactions and system behavior.
* Design the system architecture, including database schemas and system workflows. Tools and technologies will be finalized in this phase.

**Week 4: Backend Development - User Authentication**

* Start actual coding by implementing user registration and login functionalities.
* Set up the database models for users and authentication logic.

**Week 5: Backend Development - Messaging Logic**

* Develop the core messaging logic, including APIs for sending, receiving messages, and managing contact lists.

**Week 6: Frontend Development**

* Design and implement the user interface for the chat application using HTML, CSS, and JavaScript.
* Integrate the frontend with the backend /APIs.

**Week 7: Testing and Quality Assurance**

* Conduct thorough testing, including unit tests, integration tests, and user acceptance testing.
* Identify and fix any bugs or issues.

**Week 8: Deployment and Documentation**

* Wrap up and prepare documentation/ User guides.
* Project submission

**Developer Responsibilities**

**James**

Project Leader

Front End

HTML/CSS/JS Web Page

Web Socket Connection to Server

**Goma**

Backend

Database Design

Server setup

**Mike**

Testing

Finalize test plans

Take lead on merging branches on GitHub

Backend

**Jazer**

Technical writing

Rough drafts of weekly assignments (everyone else will contribute as required)

Backend