A Mini Project Synopsis on

Chat Application

S.E. - I.T Engineering

Submitted By

Raj Nikam	**21104032**
Atharva Makode	**21104013**
Yash Kasar	**21104115**
Hitesh Dubey	**21104120**

Under The Guidance Of

Prof. Jayshree Jha

DEPARTMENT OF INFORMATION TECHNOLOGY

A.P.SHAH INSTITUTE OF TECHNOLOGY
G.B. Road, Kasarvadavali, Thane (W), Mumbai-400615 UNIVERSITY OF MUMBAI

Academic year: 2022-23

CERTIFICATE

This to certify that the Mini Project report on Learning System Management has been submitted by Raj Nikam(21104032) Atharva Makode(21104013) and Yash Kasar(21104047), Hitesh Dubey(21104061) who are a Bonafede students of A. P. Shah Institute of Technology, Thane, Mumbai, as a partial fulfilment of the requirement for the degree in **Information Technology**, during the academic year **2022-2023** in the satisfactory manner as per the curriculum laid down by University of Mumbai.

Prof. Jayshree Jha
Guide

Dr. Kiran Deshpande	Dr.UttamD.Kolekar
Head Department of Information Technology	Principal

External Examiner(s)
1.
2.

Place: A.P. Shah Institute of Technology, Thane Date:

ACKNOWLEDGEMENT

This project would not have come to fruition without the invaluable help of our guide **Prof. Jayshree Jha**. Expressing gratitude towards our HoD, **Dr. Kiran Deshpande**, and the Department of Information Technology for providing us with the opportunity as well as the support required to pursue this project. We would also like to thank our teacher **Ms. Shital Agarwal** who gave us her valuable suggestions and ideas when we were in need of them.
We would also like to thank our peers for their helpful suggestions.

TABLE OF CONTENTS

1. Introduction

Real-time chat projects are becoming increasingly popular in today's digital world. They enable users to communicate with each other in real-time, regardless of their location. Python is a popular programming language used for building such projects due to its simplicity, versatility, and ease of use.

In this project, we will be building a real-time chat project using Python. The project will use a client-server model, where clients will connect to the server to exchange messages. The server will act as a mediator, relaying messages between clients.

This chat project will allow users to register, log in, and join chat rooms. Once inside a chat room, users will be able to send and receive messages in real-time. We will use various Python libraries such as Flask, Socket IO, and SQL Alchemy to build this project.

Overall, this project will provide a solid understanding of real-time communication, networking, and web development in Python.

1.1 Purpose

The purpose of a real-time app in Python is to enable users to communicate with each other in real-time. Real-time communication is an essential part of many projects, such as chat rooms, online gaming, and collaborative work environments.

Real-time apps in Python enable users to exchange information immediately, regardless of their location. This is achieved by using web sockets, which allow the server to push data to clients in real-time. This means that the data is not delayed by the typical request response cycle used by traditional web projects.

Real-time apps can be used for a variety of purposes, such as:

Instant messaging and chat rooms: Real-time chat projects allow users to communicate with each other in real-time. This is useful for online communities, customer support, and team collaboration.

Overall, real-time apps in Python can be used for a wide range of projects that require fast and reliable communication.

1.2 Objectives

The objective of our chat project can be defined as follows:

- Enable real-time communication: We enable users to communicate with each other in real-time, regardless of their location. This project would provide a seamless experience, with minimal delay or lag.

- User authentication and authorization: This project will allow users to register and log in securely and provide appropriate access control mechanisms to ensure that users can only access the chat rooms they are authorized to access.

- Chat room management: We allow users to create, join, and leave chat rooms. Our project would also provide appropriate mechanisms to manage chat rooms, such as restricting access or removing users.

- Message exchange: Our chat project enables users to send and receive messages in real-time and to ensure that messages are delivered reliably and in the correct order.

- Scalability and reliability: Our project is designed to handle a large number of users and message and is scalable and reliable.

- Overall, the objective of our real-time chat project in Python is to provide a seamless and reliable communication platform for users, with appropriate security and access control mechanisms.

1.3 Scope

- User Interface: Our chat project is simple and intuitive, allowing users to register, log in, and join chat rooms easily. The interface would also allow users to send and receive messages, and to manage their chat rooms.

- Server-Side: The server-side of our chat project is designed to handle a large number of users and messages, while maintaining reliability and performance. It provides the necessary communication protocols to enable real-time messaging between clients.

- Database: The project is using a database to store user information, chat room details, and message history. This will enable users to easily retrieve their chat history and allow the project to maintain state and provide notifications when new messages are received.

- Testing and Deployment: We would be thoroughly tested to ensure that it is reliable and performs as expected. It should also be designed for easy deployment to enable the project to be easily deployed to production environments.

- Overall, the scope of our real-time chat project includes user interface design, serverside development, security mechanisms, database design, testing, and deployment.

2. Problem Definition

- The need for real-time communication has increased significantly in recent years due to the rise of remote work and the need for digital collaboration. However, traditional communication methods such as email and instant messaging projects do not provide the real-time experience that users require.

- We aim to address this problem by providing a platform for users to communicate with each other in real-time, regardless of their location. This project would provide a reliable and fast communication platform that enables users to send and receive messages instantly.

- The challenge of building a real-time chat project in Python involves designing a scalable and reliable server-side architecture that can handle a large number of users and messages so, we have designed with appropriate security mechanisms to protect user data and prevent unauthorized access to chat rooms.

- Furthermore, our project will provide an intuitive user interface that allows users to easily register, log in, and join chat rooms. The user interface would also provide appropriate mechanisms to manage chat rooms and exchange messages in real-time.

- Overall, the problem definition for our project is to provide a reliable, fast, and secure communication platform that meets the needs of users in a digital world.

3.Proposed system

A proposed system for a real-time chat project in Python can be defined as follows:

- User Registration and Authentication: Users will be able to create accounts with the chat project, providing basic details such as name, email address, and password. User authentication will be implemented to ensure that only registered users can access the chat rooms.

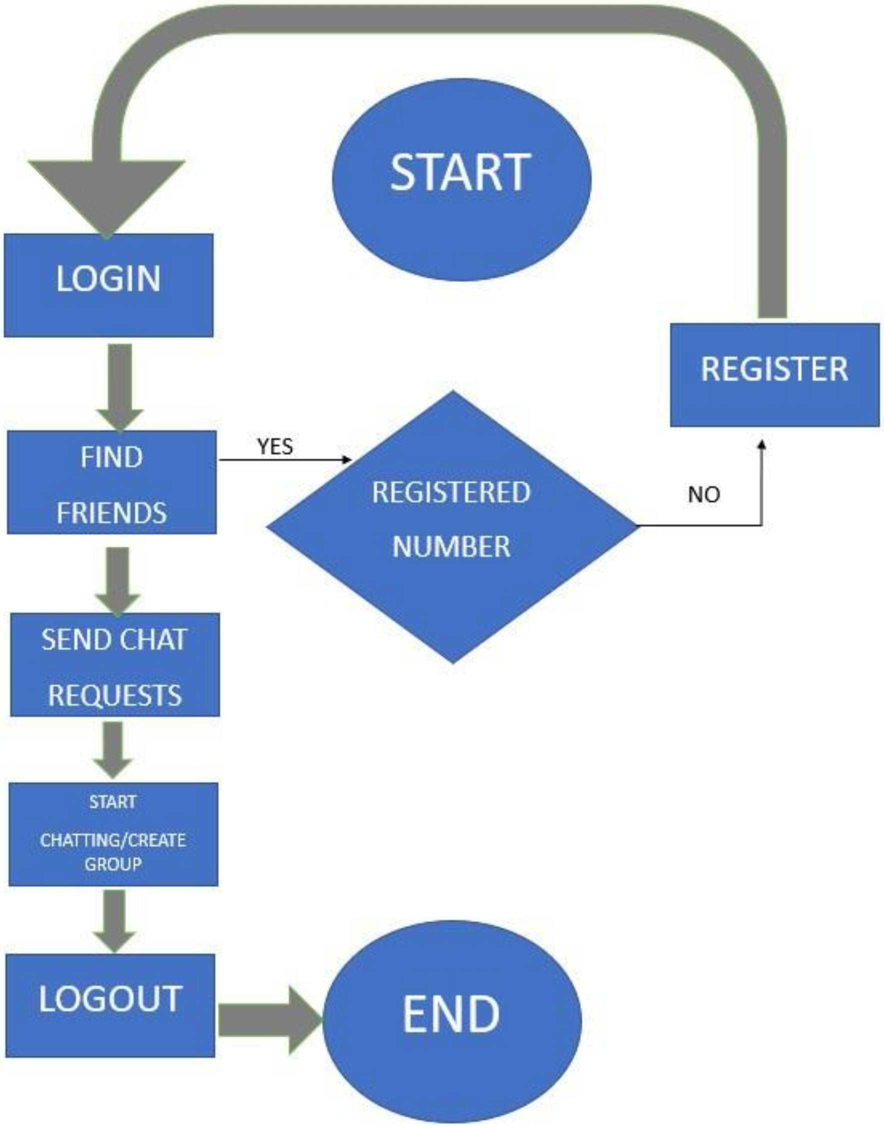
- Chat Room Creation and Management: Users will be able to create chat rooms, set access permissions, and manage the chat room settings. Users will also be able to join existing chat rooms.

- Database Management: Chat project will store user details, chat room settings, and message history in a database which enable users to easily retrieve their chat history and allow the project to maintain state and provide notifications when new messages are received.

- User Interface: The user interface of the chat project is designed to be intuitive and user-friendly, allowing users to easily navigate and use the project. The interface will provide access to chat rooms, message exchange, and chat room management.

- Deployment and Scalability: Our project is designed to be easily deployed and scalable to handle a large number of users and messages.

- Overall, the proposed system of our chat application aims to provide a reliable and fast communication platform that enables users to exchange messages in real-time while ensuring their privacy and security. This project will be user-friendly and scalable, providing an efficient and effective platform for digital communication.


START
LOGIN
REGISTER
FIND FRIENDS
YES
REGISTERED NUMBER
NO
SEND CHAT REQUESTS
START CHATTING/CREATE GROUP
LOGOUT
END

3.1 Features and Functionalities

The features and functionality of a real-time chat project in Python can include the following:

- User Registration and Authentication: The project would allow users to create an account and authenticate their identity before they can access the chat rooms. This can be implemented using various authentication mechanisms, such as email verification or OAuth.

- Chat Room Creation and Management: Users would be able to create new chat rooms and manage them, including setting the name, description, and access permissions for the chat room. Chat room management features should also include the ability to invite or remove users and modify settings.

- Real-Time Messaging: The project should allow users to send and receive messages in real-time within chat rooms. The messages should be delivered instantly and reliably to all participants in the chat room. The project should also support features such as message editing and deleting.

- User Privacy and Security: The project should implement appropriate security measures to protect user data, including encryption of messages and user authentication. The project should also provide users with control over their privacy settings, including the ability to block or report other users.

- Emojis and GIFs: The project can provide users with the ability to use emojis and GIFs in their messages to enhance the user experience.

- Search and Filter: The project should provide users with the ability to search and filter through chat rooms and messages to find relevant information.

4. Project outcomes

The project outcomes of developing a real-time chat project in Python can include the following:

- A functional and reliable real-time chat project that allows users to communicate with each other instantly in a secure and private manner.

- Improved communication and collaboration among users, especially in remote work settings.

- Increased productivity due to the ability to quickly and efficiently exchange information and ideas.

- Enhanced user privacy and security through encryption of messages and user authentication.

- Increased user engagement and retention due to a user-friendly interface and notification system.

- Enhanced learning experience for the developers involved in building the project.

- Improved technical skills of the developers involved in building the project, especially in the areas of Python programming, web development, and server-side architecture.

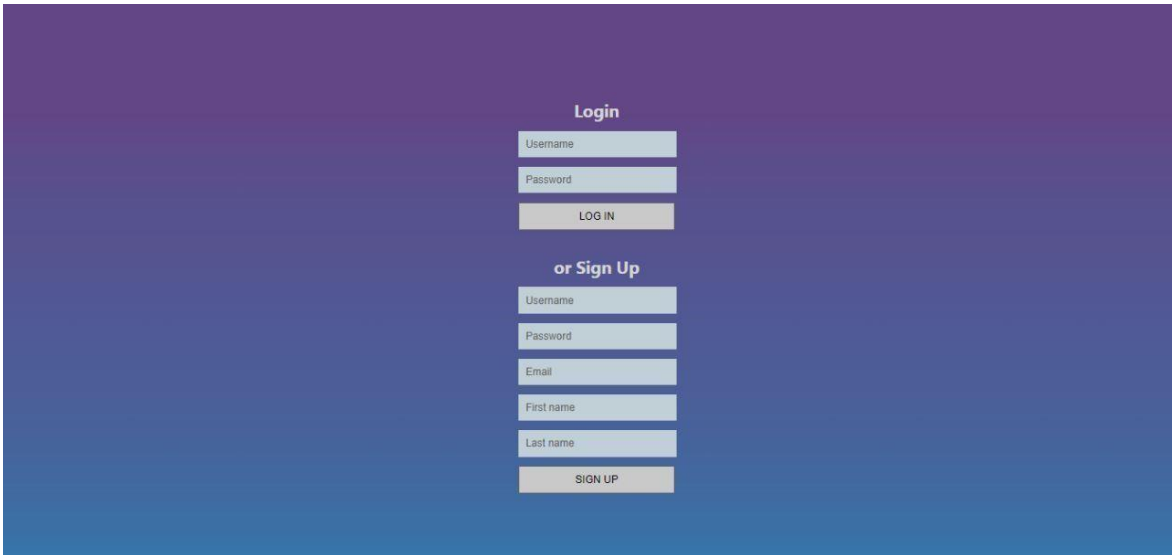
5. Software Requirements

- **Front End**:- A user interface framework/library such as Tkinter.

- **Back End**:-A database access layer such as AST for communicating with the database. For Database purpose we have used Pymysql.

6. Project Design

i. Login Page



Login

Username

Password

LOG IN

or Sign Up

Username

Password

Email

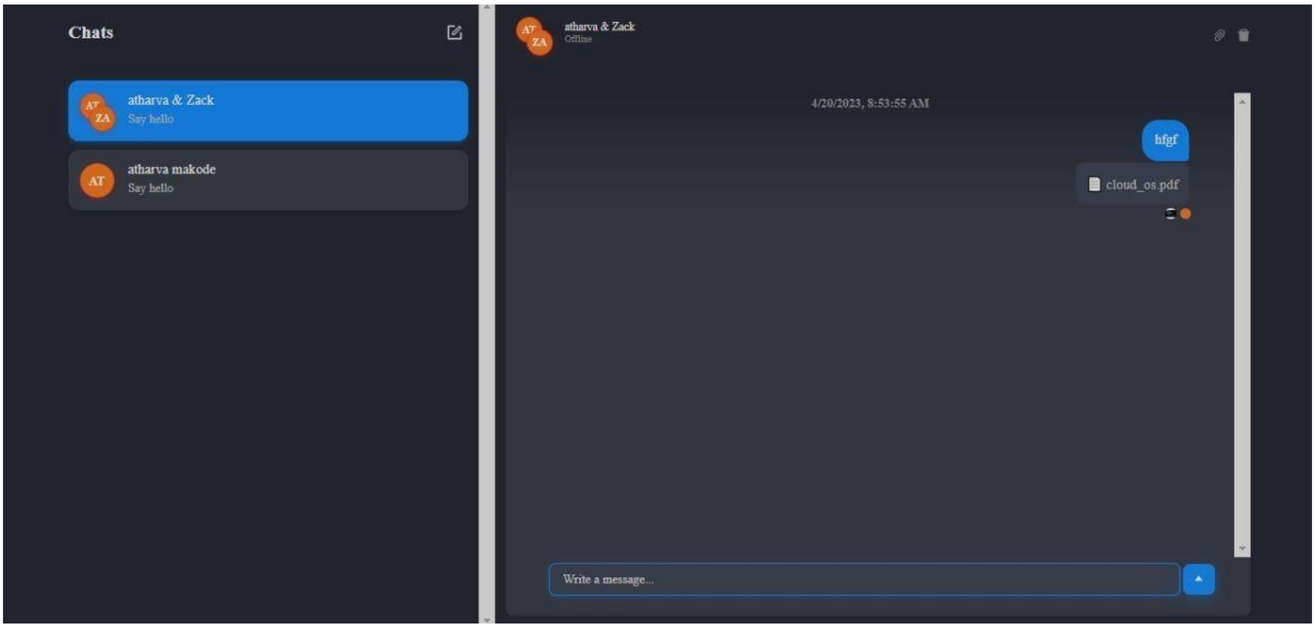
First name

Last name

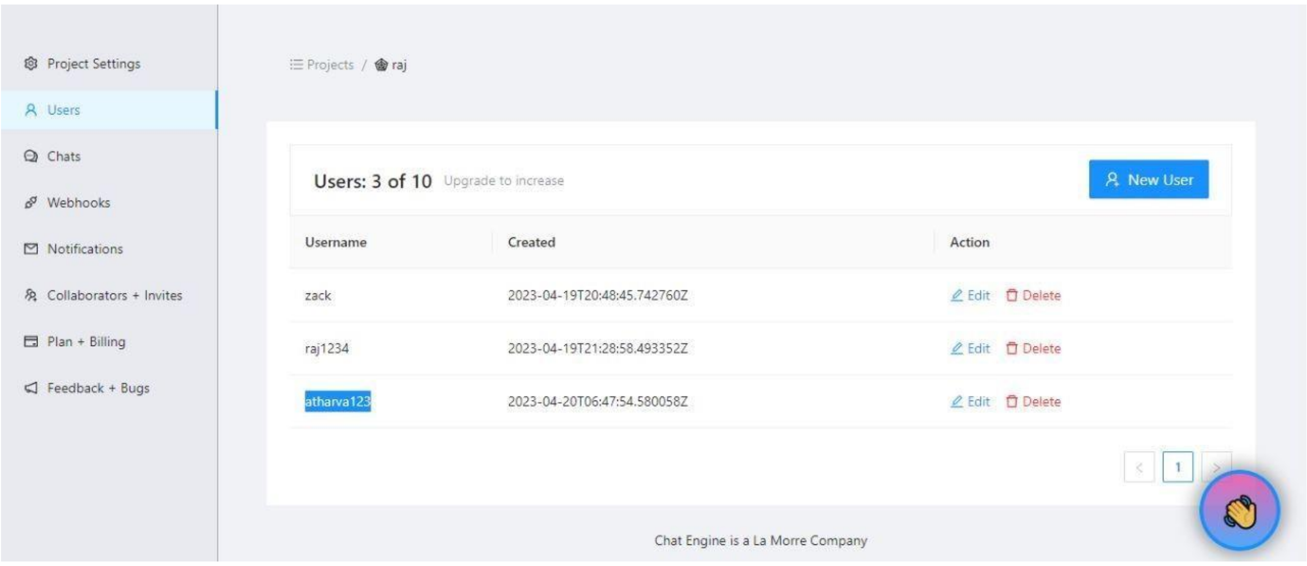
SIGN UP

ii. Home

Page



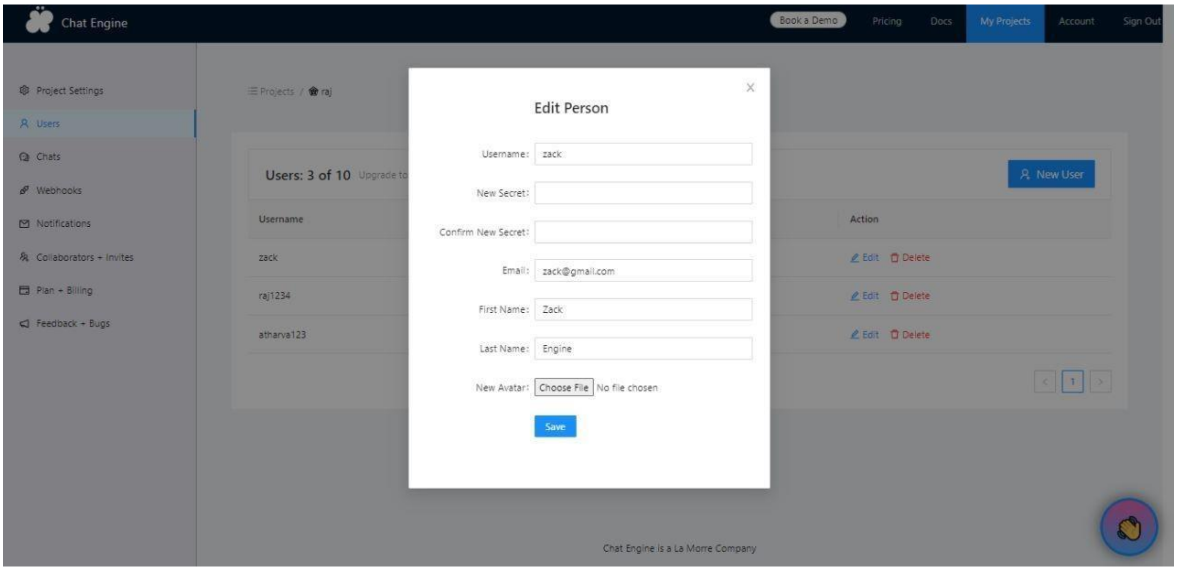
iii. Chat Engine



Users: 3 of 10 Upgrade to increase

Username	Created	Action
zack	2023-04-19T20:48:45.742760Z	Edit Delete
raj1234	2023-04-19T21:28:58.493352Z	Edit Delete
atharva123	2023-04-20T06:47:54.580058Z	Edit Delete

iv. Profile Editing



Edit Person

Username: zack

New Secret:

Confirm New Secret:

Email: zack@gmail.com

First Name: Zack

Last Name: Engine

New Avatar: Choose File No file chosen

Save

7. Project Scheduling

Group member	Time duration	Work to be done
Raj Nikam Atharva Makode Yash Kasar Hitesh Dubey	Week 1 - Week 3	Implementing 1st module Implemented GUI using Tkinter
	Week 3- Week 4	Testing 1st Module Added some features after the feedback
	Week 4 - Week 6	Implementing 2nd module Implementation of database connectivity using AST and Pymysql
	By End Of The Week	Implementing 3rd module Learnt basics of connectivity and Report making

8. Conclusion

Our chat project is a valuable tool for communication and collaboration among users in various settings, including remote work, education, and social networking. The development of such a project requires a thorough understanding of the objectives, scope, and requirements of the project, as well as proficiency in Python programming, web development, and server-side architecture.

The proposed system for a real-time chat project is to provide users with a reliable, efficient, and user-friendly platform for digital communication, while ensuring their privacy and security. The features and functionality of the project is designed to enhance the user experience and provide a seamless platform for messaging and collaboration.

The project outcomes of developing a real-time chat project would be beneficial to both users and developers, providing a functional and efficient platform for digital communication while enhancing technical skills and potential market opportunities. With the increasing demand for digital communication tools, developing our project can be a valuable project with significant potential for growth and impact.

.

References

a. Python module of the week by Duong Hellman

b. https://youtu.be/IzcEH8QgoqY