SOFTWARE TEST CASES DOCUMENT

CHATZEN

NOVEMBER 11, 2023

RUBIX!

CZ!

Version 1.0

CHATZEN

Team: RUBIX!

Shreya Singh (LCS2022046)

Raghunandan Bansal (LCS2022058)

Rishi Raj Maheshwari (LCB2022005)

Table of Contents

1. Introduction

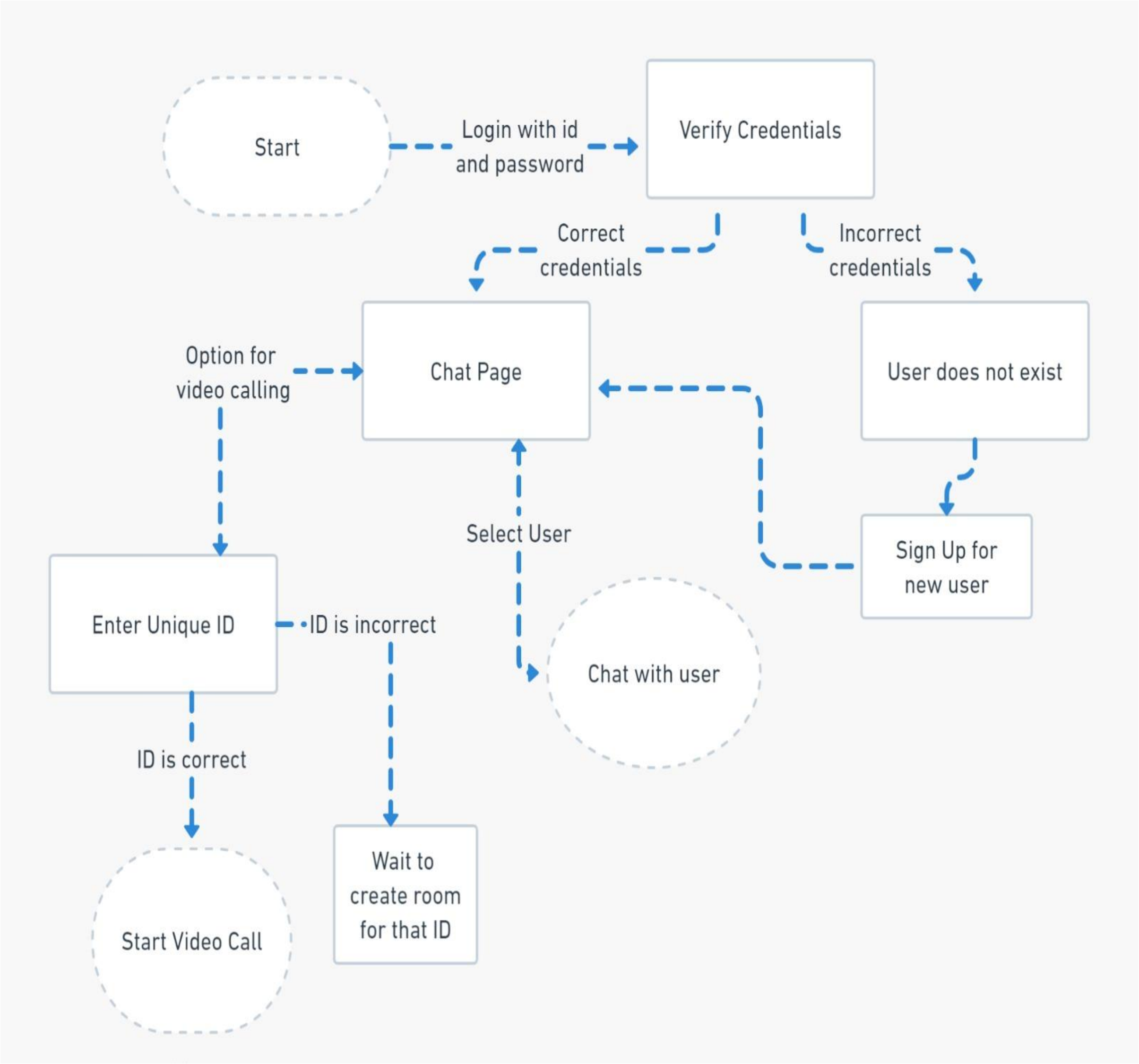
This document outlines the various test cases that will be conducted to validate the quality and correctness of the ChatZen messaging application. Thorough testing is critical to ensure ChatZen meets all requirements and provides a smooth, user-friendly experience.

This document will cover test cases for different types of testing such as structural, functional, system, and regression testing. Both positive and negative test cases will be included to verify intended functionality and identify defects. The test cases aim to evaluate ChatZen from a unit, integration, and system level.

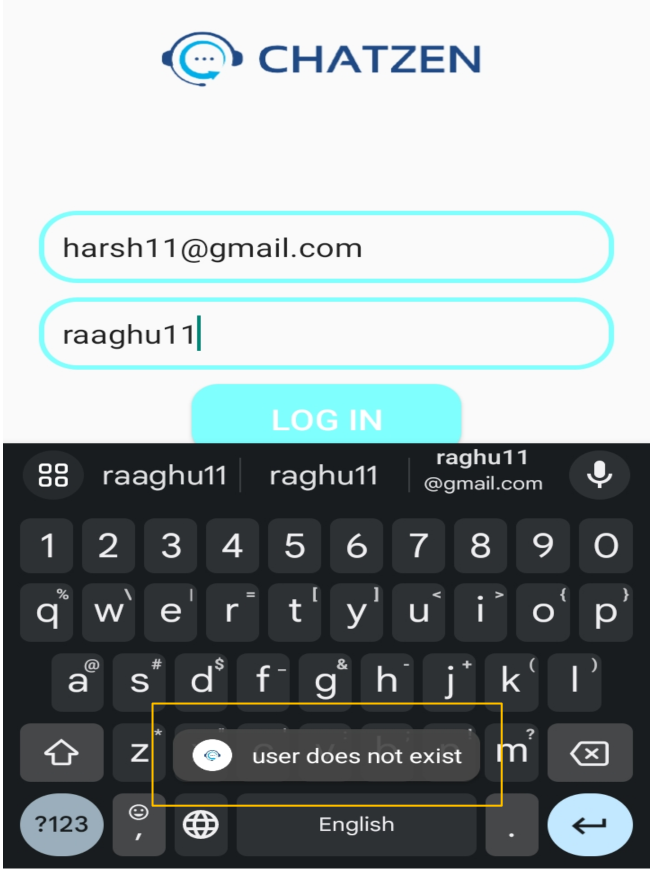
The intended audience for this document is the quality assurance and development teams who will be executing the test cases and using them to guide the testing effort. These test cases may also be referenced by other stakeholders like product managers and business analysts.

2. Structural Testing

2.1 Basic Path Testing



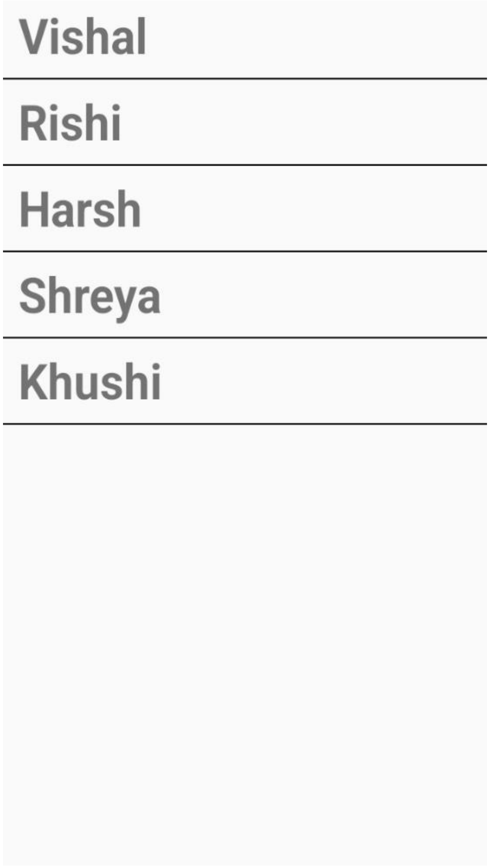
2.2 Control Statement Testing



harsh11@gmail.com

raaghu11

user does not exist



Vishal

Rishi

Harsh

Shreya

Khushi

When users provide accurate login information, they will be successfully added to the chat list under 'contacts.' In case of incorrect information, the system will notify users that their credentials are invalid, and they will be informed that the user does not exist in the system.

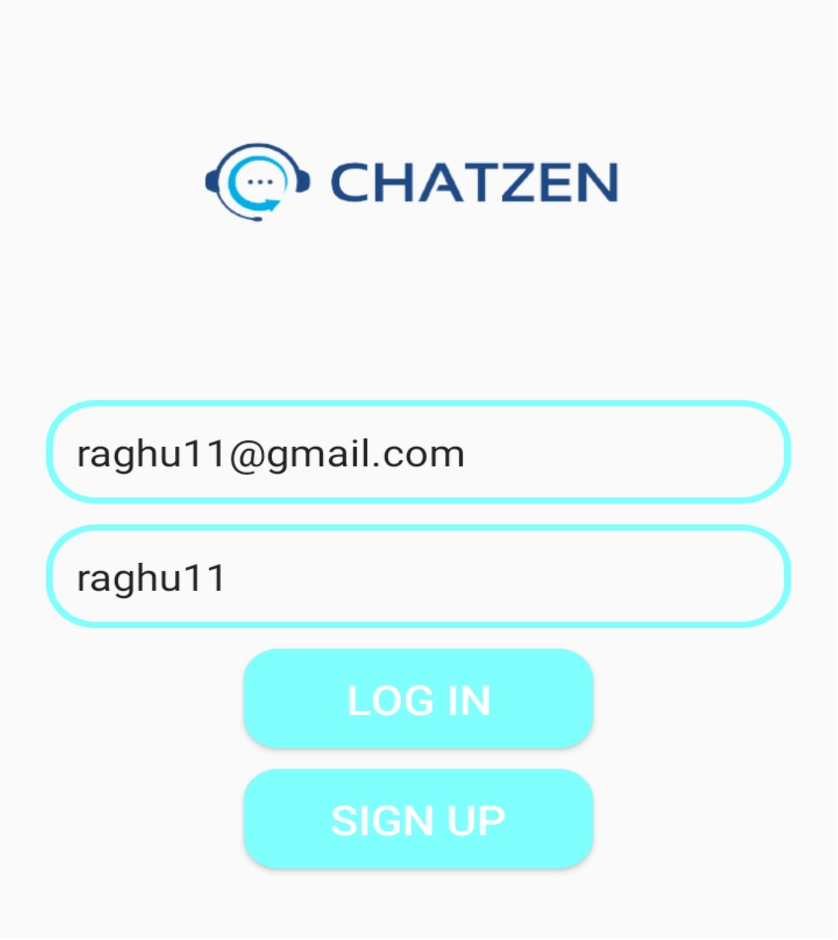
3. Functional Testing

3.1 Test case Development

- UID check at every login
- Unique UID is created at new user sign up
- Unique ID for every send message
- Details of the users are stored
- Unique room name for Video Call

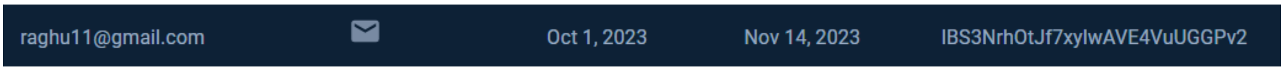
3.2 Test case implementation

- **UID check at every login:**



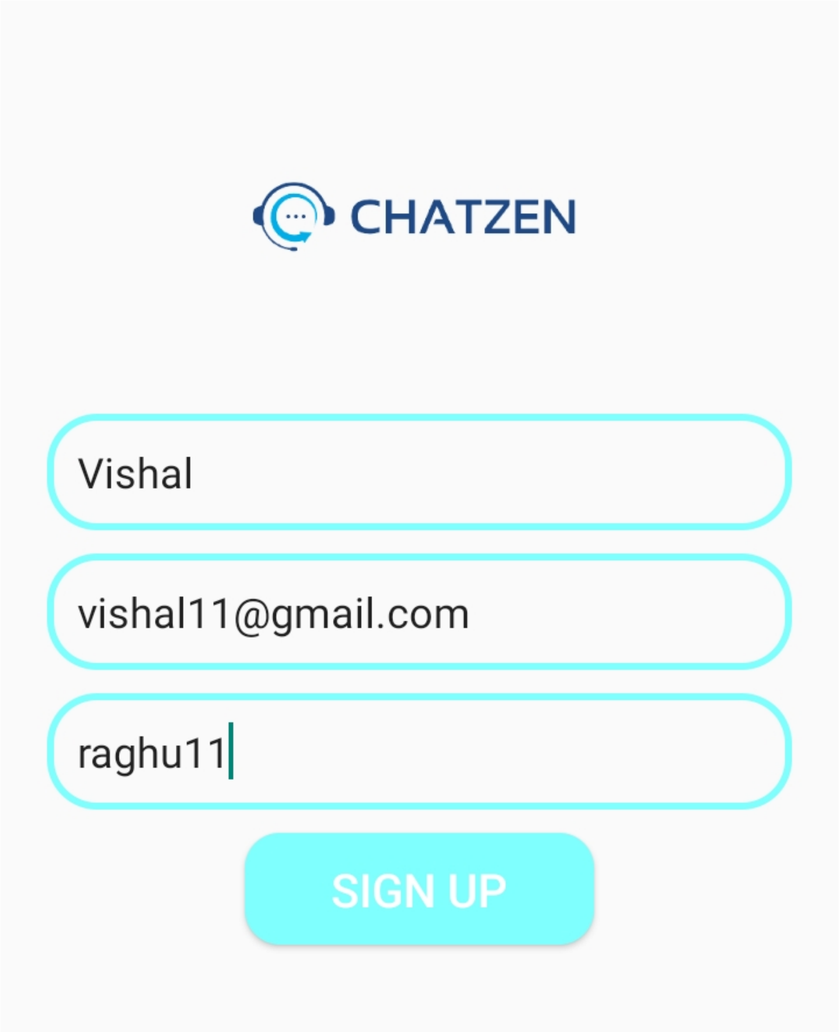
raghu11@gmail.com

raghu11



raghu11@gmail.com	✉	Oct 1, 2023	Nov 14, 2023	IBS3NrhOtJf7xylwAVE4VuUGGPv2

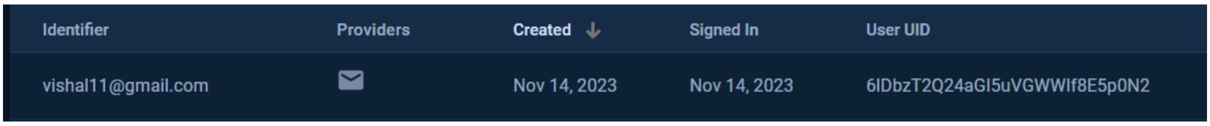
- **Unique UID is created at new user sign up:**



Vishal

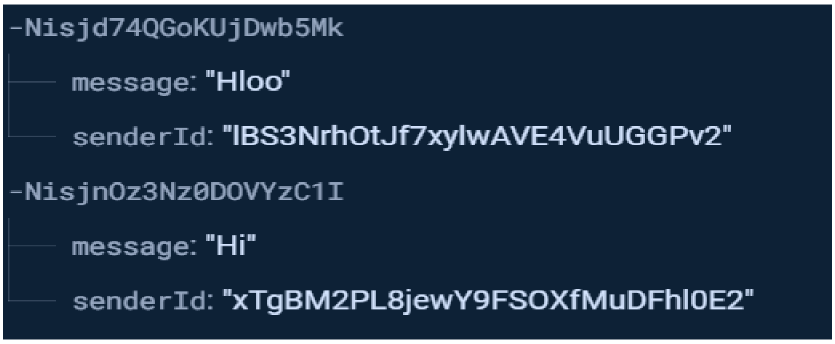
vishal11@gmail.com

raghu11



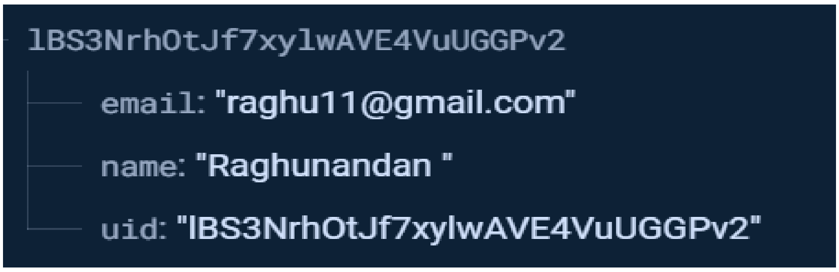
Identifier	Providers	Created ↓	Signed In	User UID
vishal11@gmail.com	✉	Nov 14, 2023	Nov 14, 2023	6lDbzT2Q24aGI5uVGWWIf8E5p0N2

- **Unique ID for every send message:**



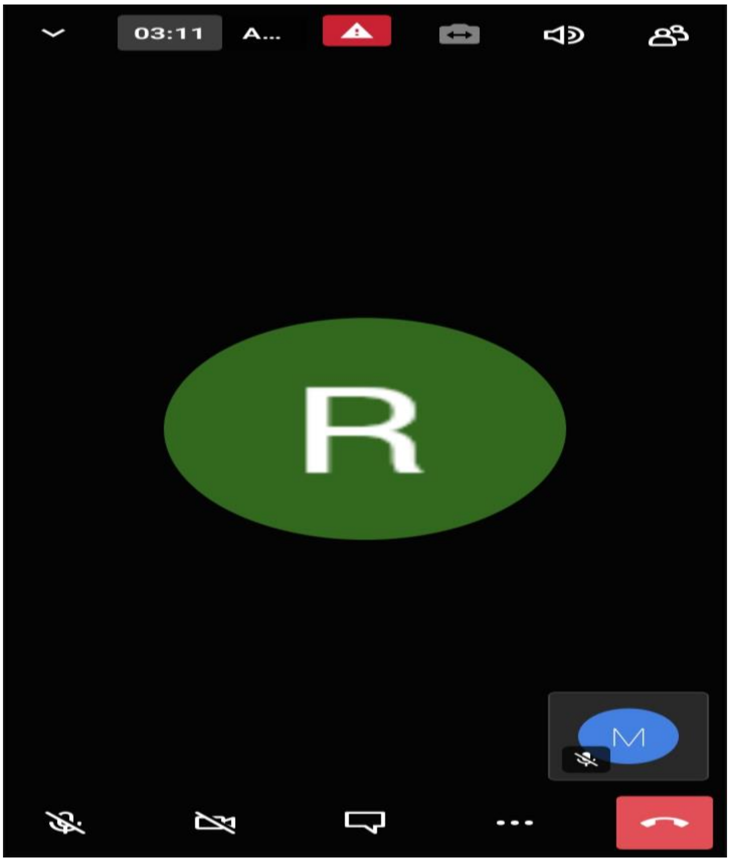
```
-Nisjd74QGoKUjDwb5Mk
    message: "Hloo"
    senderId: "lBS3NrhOtJf7xylwAVE4VuUGGPv2"
-Nisjn0z3Nz0DOVYzC1I
    message: "Hi"
    senderId: "xTgBM2PL8jewY9FSOXfMuDFhl0E2"
```

- **Details of the users are stored:**



```
1BS3Nrh0tJf7xylwAVE4VuUGGPv2
    email: "raghu11@gmail.com"
    name: "Raghunandan "
    uid: "lBS3NrhOtJf7xylwAVE4VuUGGPv2"
```

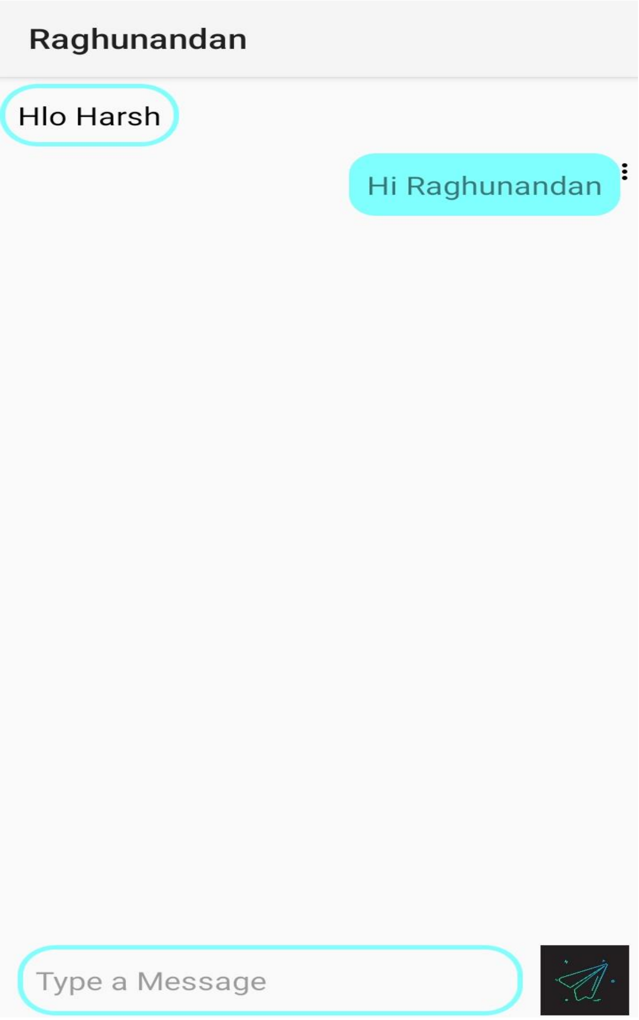
- **Unique room name for Video Call:**



4. SDLC Testing

4.1 Unit Testing

i. **Message Handling:** Message was smoothly sent and received

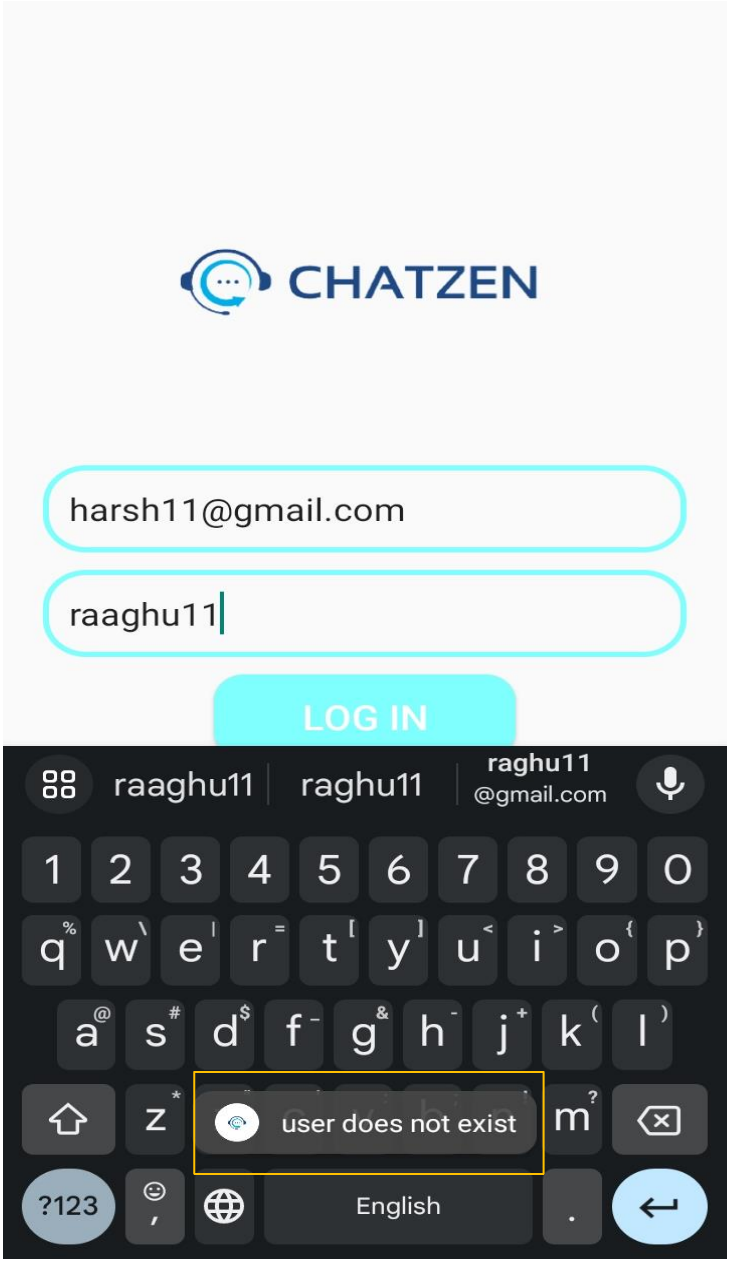


Raghunandan

Hlo Harsh

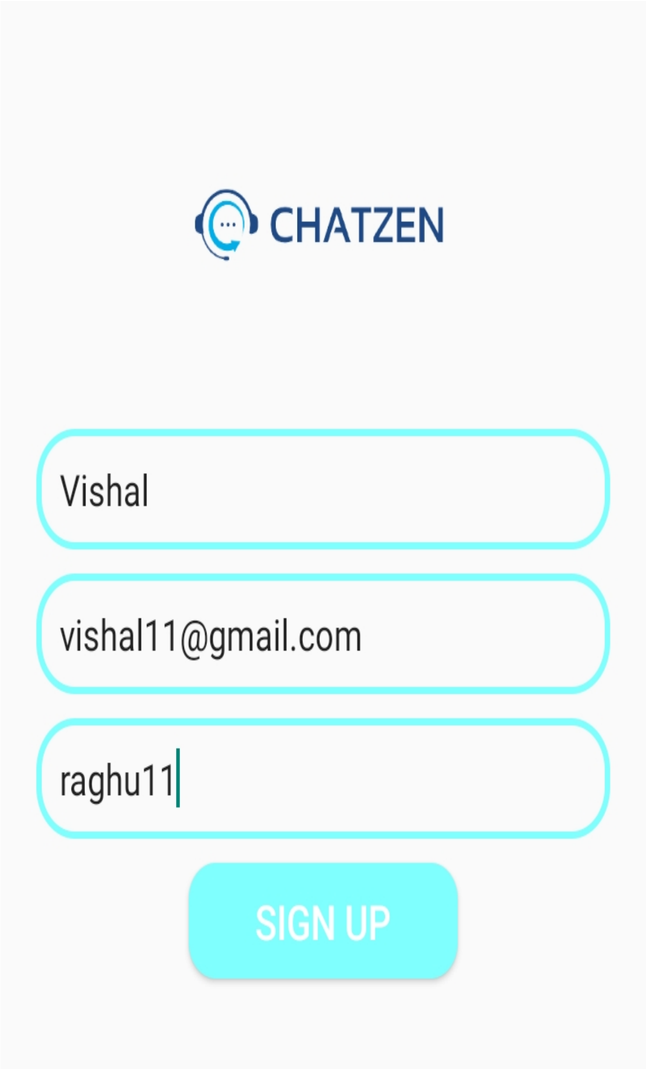
Hi Raghunandan

ii. **User Authentication:** Can login and also handle incorrect login



user does not exist

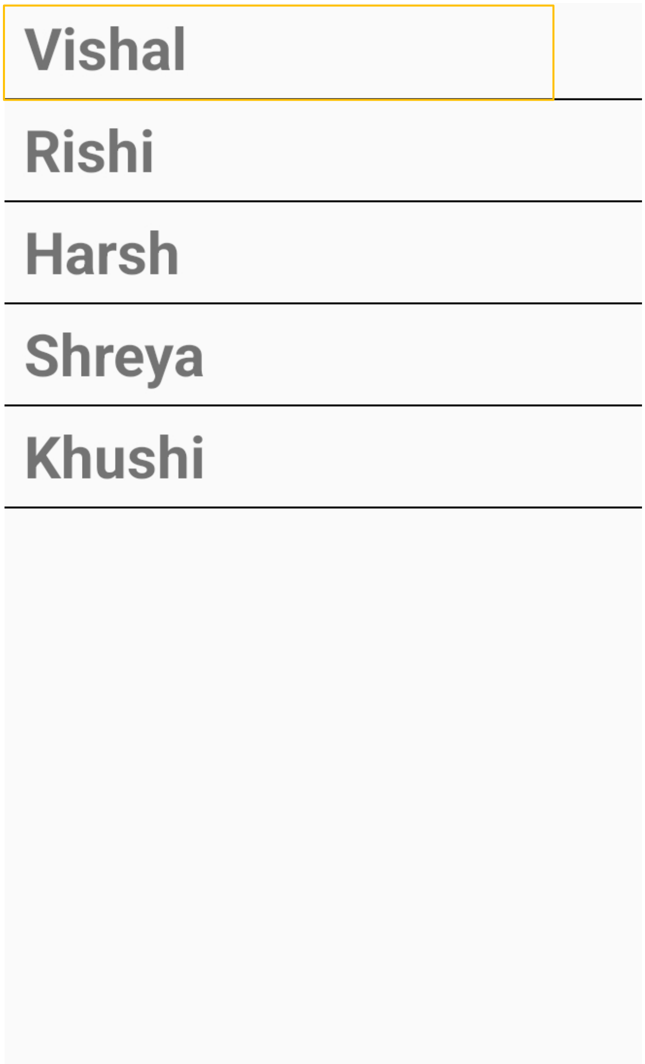
iii. **Chatroom Management:** New user is automatically added to the chat room



Vishal

vishal11@gmail.com

raghu11



Vishal

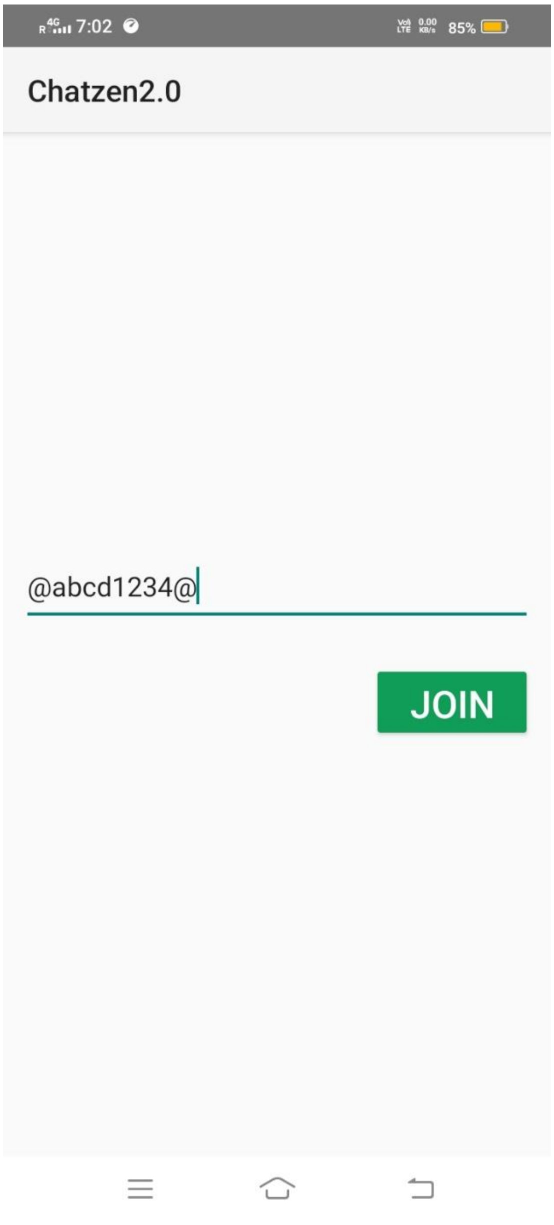
Rishi

Harsh

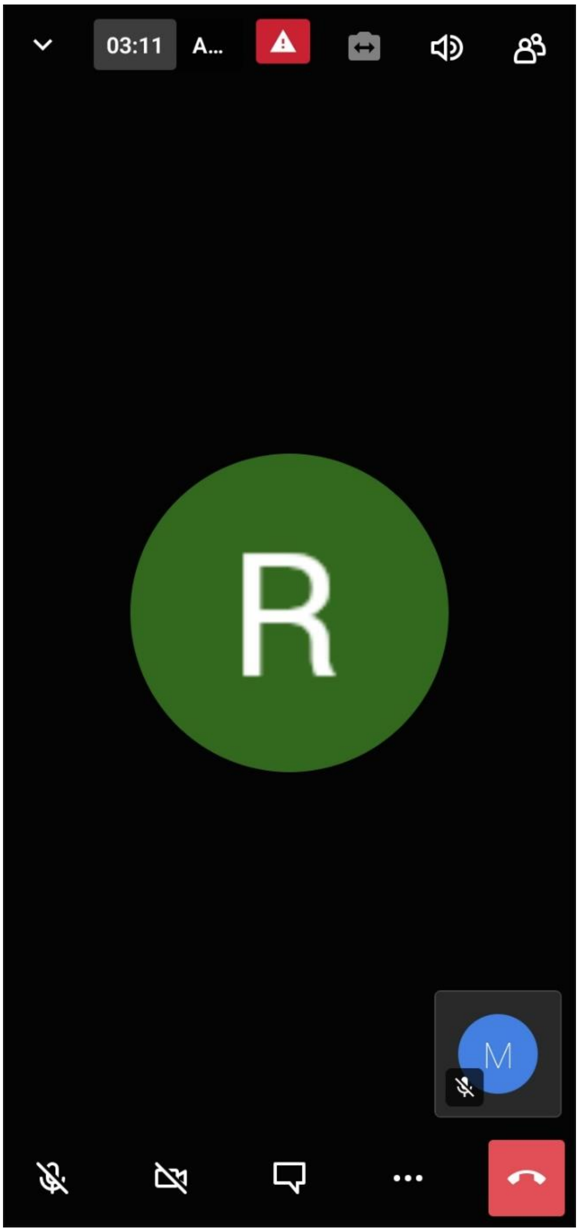
Shreya

Khushi

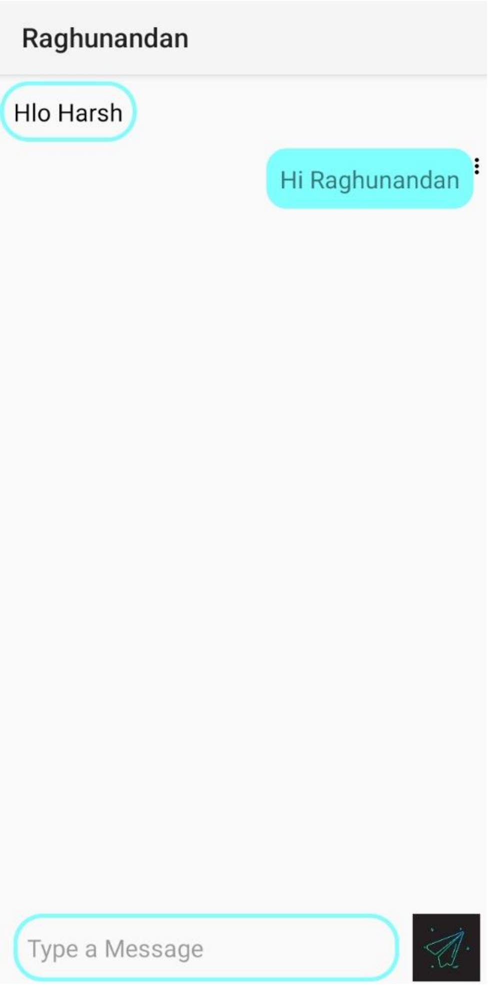
iv. **Video Call Management:** Enter the video call room on entering the unique ID


Chatzen2.0

@abcd1234@



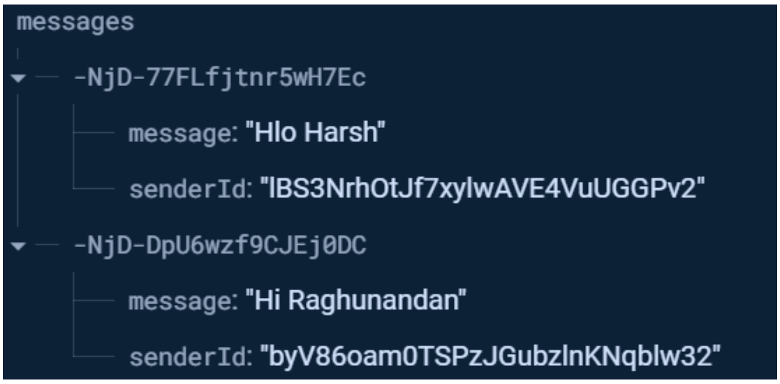
v. **<u>Real-time Database Communication</u>:** Allow real time communication, so all the data is stored at backend



Raghunandan

Hlo Harsh

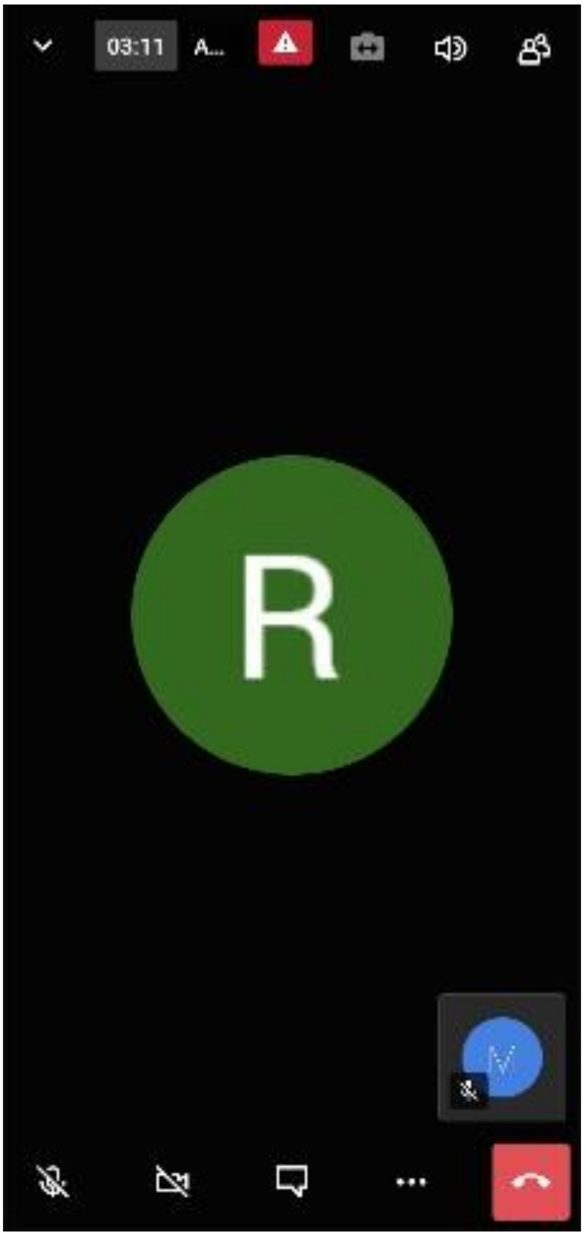
Hi Raghunandan



```
messages
-NjD-77FLfjtnr5wH7Ec
    message: "Hlo Harsh"
    senderId: "lBS3NrhOtJf7xylwAVE4VuUGGPv2"
-NjD-DpU6wzf9CJEj0DC
    message: "Hi Raghunandan"
    senderId: "byV86oam0TSPzJGubzlnKNqblw32"
```

4.2 Integration Testing

- **Video Call:** A new functionality of video calling is added to the chat room activity



5. System Testing

5.1 Security

- **Secure User Authentication:** Every time when a new user is registering, they have to add their name, email, password
- **Admin can modify anonymous user:** Admin can add or delete the user
- **Login using email and password:** Every time to login, user have to use their id and password
- **Unique id for video calling:** To join a video call, user need to enter the unique room number
- **Data stored at backend:** All the data is stored at backend

5.2 Stress:

Many users can use the service at a time

5.3 Performance:

System can work for a long time without failure and on the huge traffic

6. Conclusion

The test cases outlined in this document encompass a wide variety of tests ranging from usability, security, performance, to edge case handling. They are designed to thoroughly evaluate the functional and non-functional aspects of the Chatzen application. Successful execution of these test cases will help ensure all requirements are met before launch.

As issues are uncovered through testing, additional test cases may be added to the regression suite to prevent regressions in the future. The test cases are modular and customizable to adapt to changing project needs. Proper implementation of these test cases will improve the quality of Chatzen and help deliver a robust, secure, user-friendly messaging application to customers.