**Final Project Report**

# SKYPE

**Title:** Software Development Lifecycle for Skype **Group Members:**

Romaisa (2023-BCS-073)

Javeria Rahim (2023-BCS-046)

Areeba Yasir (2023-BCS-016)

Sameen Noor (2023-BCS-077)

**Section:** BCS-3(B)

**Course:** Software Engineering

**Submitted to:** Mam Rabail **Date of Submission:** December 12, 2024

# Table of Contents

1. **Introduction** ............................................................................................................................................. 3
   1. Background of related applications ..................................................................................................... 3
   2. Main Components of Skype ................................................................................................................ 3
2. **Problem Statement** .................................................................................................................................. 3
3. **Project Objectives** ................................................................................................................................... 4
4. **Software Requirements** .......................................................................................................................... 4
   1. Functional Requirements .................................................................................................................... 4
   2. Non-functional Requirements ............................................................................................................. 5
   3. External Interface Requirements ......................................................................................................... 5
5. **Software Design** ....................................................................................................................................... 6
   1. System Architecture ............................................................................................................................ 9
   2. Data Design ....................................................................................................................................... 13
   3. User Interface Design ....................................................................................................................... 15
6. **Software Implementation** ..................................................................................................................... 16
   1. Platform Details ................................................................................................................................ 16
   2. Code Structure .................................................................................................................................. 17
   3. Screenshots ....................................................................................................................................... 17
7. **Software Testing** .................................................................................................................................... 19
   1. Testing Levels ................................................................................................................................... 19
   2. Test Cases and Results ...................................................................................................................... 20
8. **Conclusion** ............................................................................................................................................. 27
   1. Limitations ........................................................................................................................................ 27
   2. Future Work ...................................................................................................................................... 28
9. **References** .............................................................................................................................................. 28

# 1. Introduction

Skype is a widely used communication platform that provides features such as voice and video calls, instant messaging, and file sharing. First launched in 2003, it was one of the pioneers in internet-based communication, transforming how people connect across long distances. Skype offers tools that cater to both personal and professional needs, making it a versatile choice for users worldwide.

## 1.1 Background of related applications

Over the years, other applications like Zoom, Google Meet, and Microsoft Teams have emerged as competitors. While Zoom excels in large-scale video conferencing, Google Meet integrates well with Google's ecosystem, and Microsoft Teams is ideal for collaborative workplace communication. Skype differentiates itself by offering a freemium model with strong integration into Microsoft’s suite and features like screen sharing and PSTN calling.

## 1.2 Main Components of Skype

The core features of Skype include:

1. **Audio and Video Calls**: High-definition one-on-one and group calling.
2. **Instant Messaging**: Real-time text communication with support for multimedia sharing.
3. **File Sharing**: Seamless exchange of documents, images, and videos.
4. **Screen Sharing**: Collaboration through live screen sharing during calls.
5. **Cross-Platform Support**: Compatibility with Windows, macOS, Linux, iOS, Android, and web browsers.

This report delves into the software development lifecycle (SDLC) of Skype, focusing on its design, implementation, and testing phases to highlight its robust architecture and user-centric design.

# 2. Problem Statement

Effective communication is essential in both personal and professional domains. Traditional communication methods often come with limitations, including high costs, geographical constraints, and inefficiencies. These challenges necessitate a robust solution that transcends barriers and adapts to modern needs. Skype addresses these challenges by providing a versatile, internet-based communication platform that is accessible, cost-effective, and feature-rich. It empowers users to connect seamlessly through high-quality voice and video calls, instant messaging, and collaborative tools, regardless of their location.

# 3. Project Objectives

The objectives of developing Skype include:

**Facilitating Seamless Communication**:

Enable users to engage in high-quality voice and video calls across various platforms and devices.

**Supporting Collaborative Features**:

Provide tools like instant messaging, file sharing, and screen sharing for enhanced collaboration.

**Ensuring Data Security and Privacy**:

Implement encryption and adhere to global privacy standards to protect user information.

**Achieving Scalability and Reliability**:

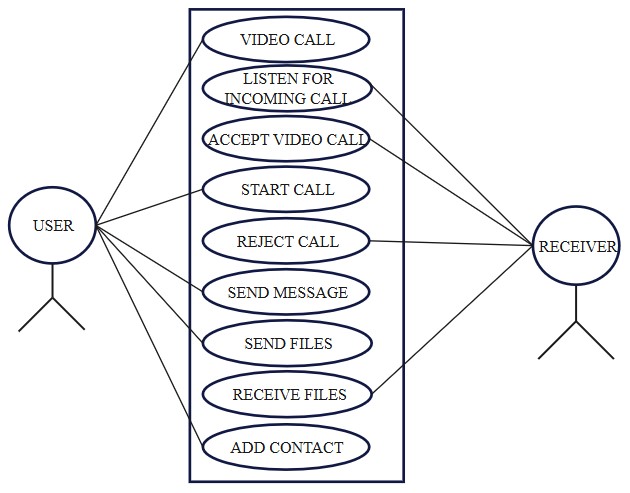
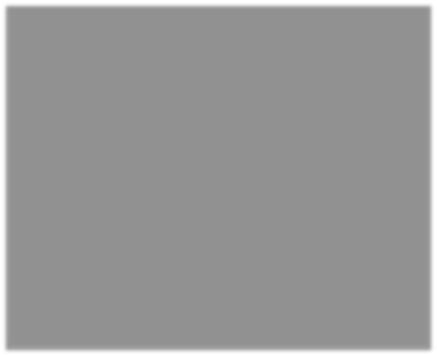
Design the platform to handle millions of concurrent users with minimal downtime.

**Enhancing User Experience**:

Deliver an intuitive and accessible interface that ensures ease of navigation and user satisfaction.

# 4. Software Requirements

**Use Case Diagram:**



## 4.1 Functional Requirements

1. **User Authentication:**
   * + Secure login with email and password.
     + Multi-factor authentication for enhanced security.
2. **Communication Features:**
   * + High-definition voice and video calls (one-to-one and group).
     + Instant messaging with multimedia sharing (images, videos, documents).
     + Screen sharing for presentations and collaboration.
3. **Profile Management:**
   * + Ability to create and edit user profiles.
     + Password recovery mechanisms.
4. **Contact Management:**
   * + Search, add, and manage contacts.
     + Integration with device’s contact list.
5. **Call Management:**
   * + Call history and logs.
     + Options for muting, holding, and recording calls.

**4.2 Non-functional Requirements**

1. **Availability:**
   * 99.99% uptime to ensure uninterrupted service.
2. **Performance:**
   * Response time <200ms for major operations like messaging and call initiation.
   * Support for thousands of concurrent users without degradation in quality.
3. **Compatibility:**
   * Cross-platform support (Windows, macOS, Linux, iOS, Android, web browsers).
4. **Security:**
   * End-to-end encryption for all communications.
   * Regular updates to address vulnerabilities and enhance security.

## 4.3 External Interface Requirements

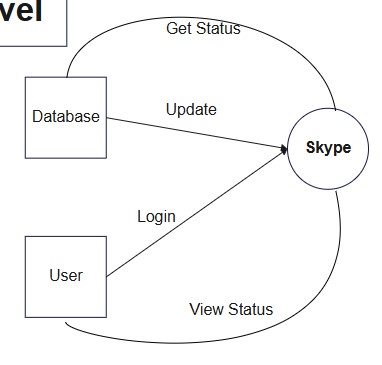
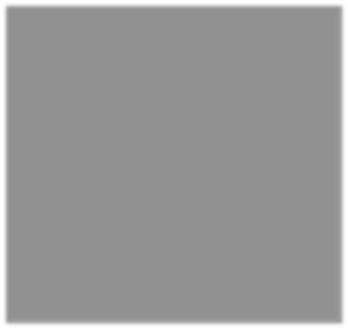
1. **User Interfaces:**
   * Login screen, profile management, chat interface, and call screens.
2. **Hardware Requirements:**
   * Microphone, camera, speakers, and a stable internet connection.
3. **Software Requirements:**
   * Compatible with popular browsers using WebRTC technology.
   * Integration with Microsoft tools like Outlook and Office.

# 5. Software Design

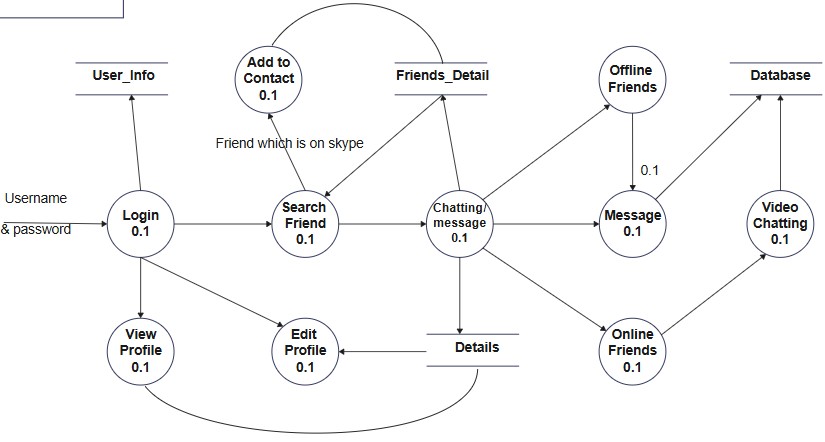
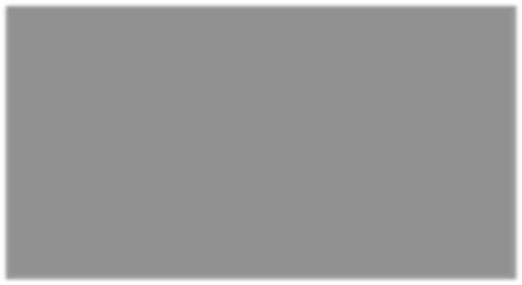
**Data**

**Flow Diagram:**

**Zero Level:**



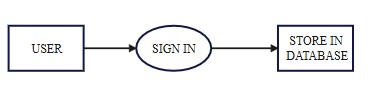
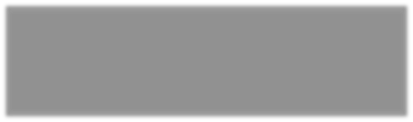
**First Level**



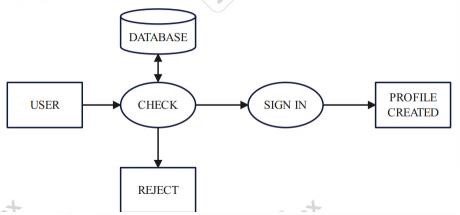
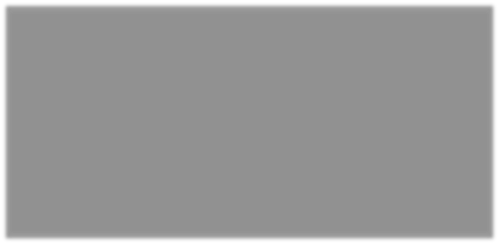
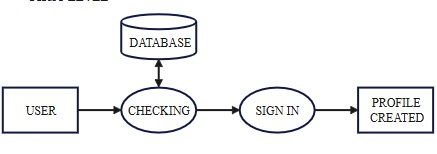
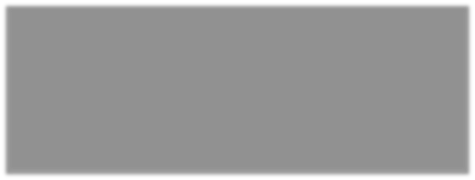
**Sign in:**

**Zero level**

**First Level**



**Second level**

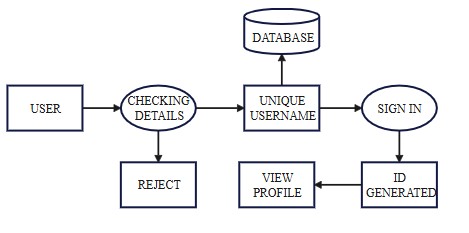
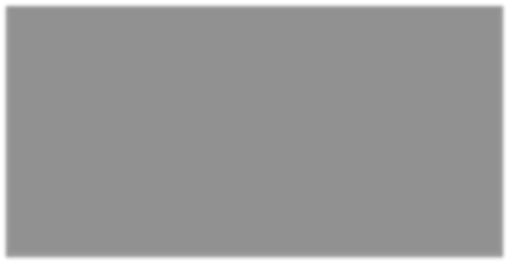


**Third level**

**2**

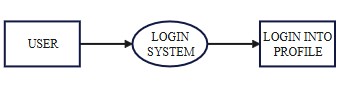
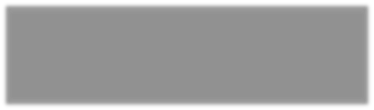
**.**

**Login Process**



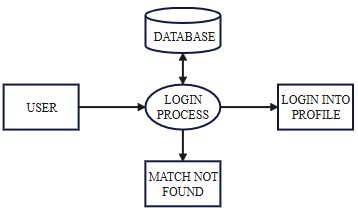
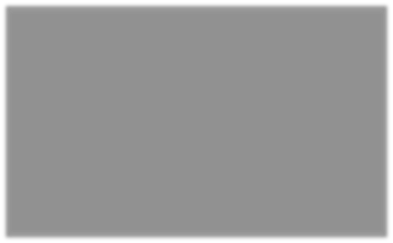
**Zero level**

**First level**



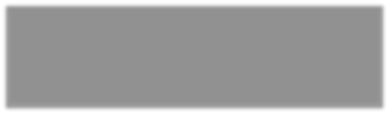
**3.**

**Chatting**



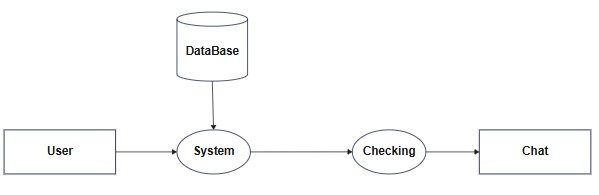
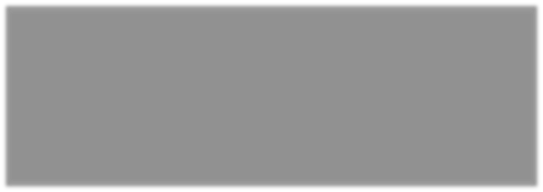
**Zero level**

**First level**



**5.1**

**System Architecture**



Skype employs a client-server architecture to handle communication and data management effectively. The architecture is composed of the following components:

1. **Client Applications:**

Interfaces for various platforms (Windows, iOS, Android, web).

1. **Communication Servers:**

Manage real-time voice and video communication.

1. **Data Storage Servers:**

Store user profiles, chat logs, call histories, and shared files.

1. **Authentication Services:**

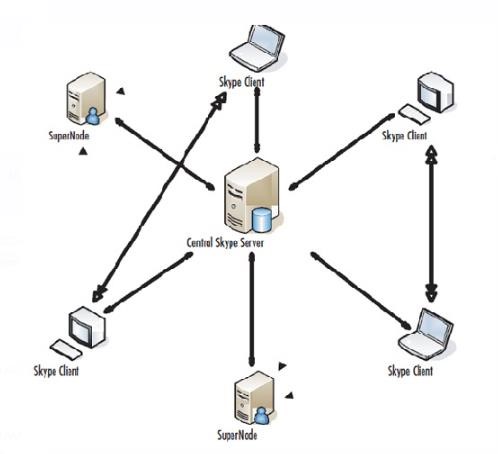
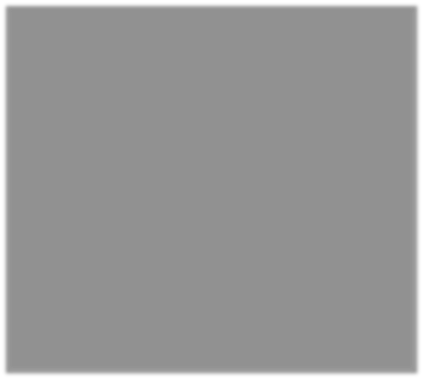
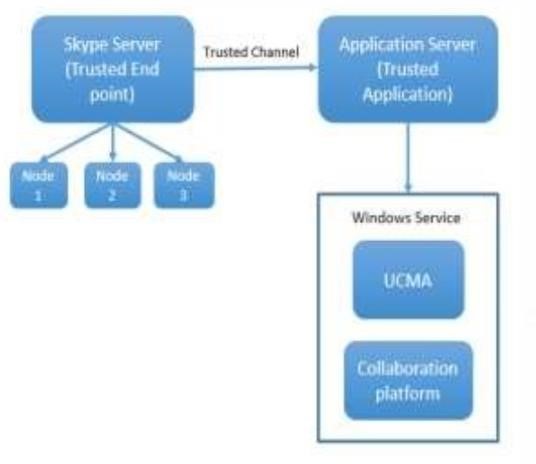
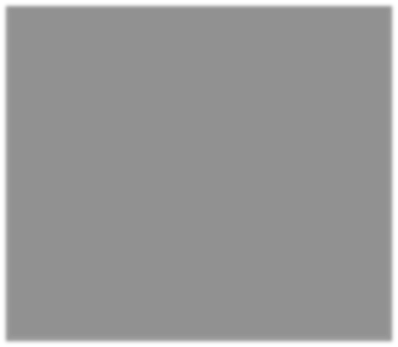
Secure user login and session management.

1. **Load Balancer:**

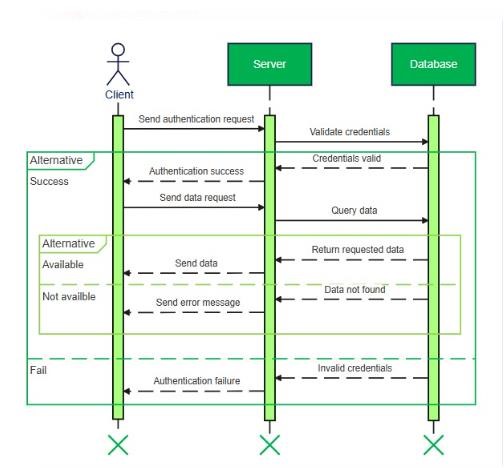
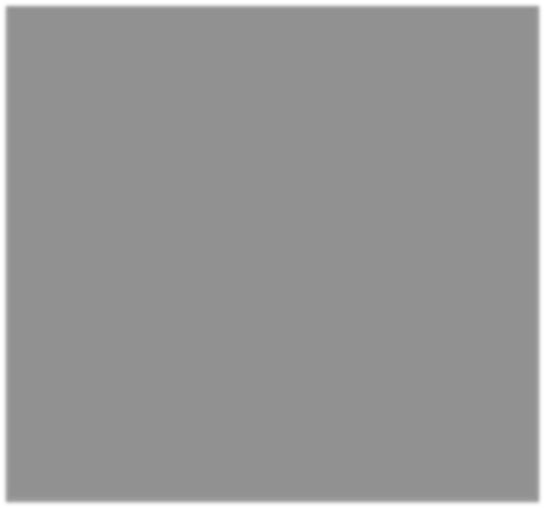
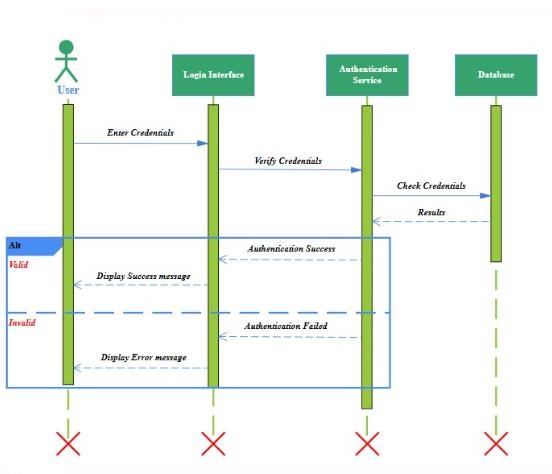
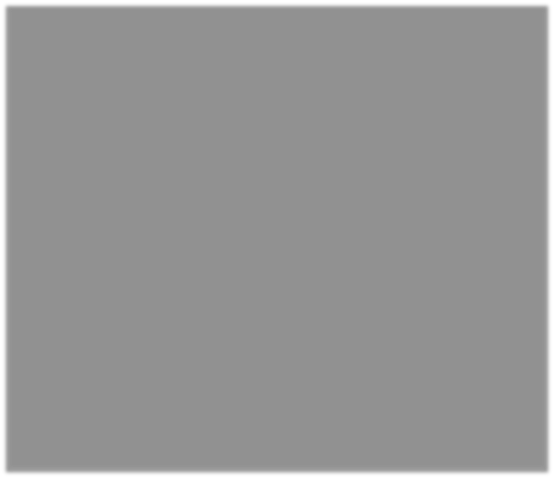
Distribute traffic across servers to maintain performance.

## Architecture Diagram

**Deployment Diagram**

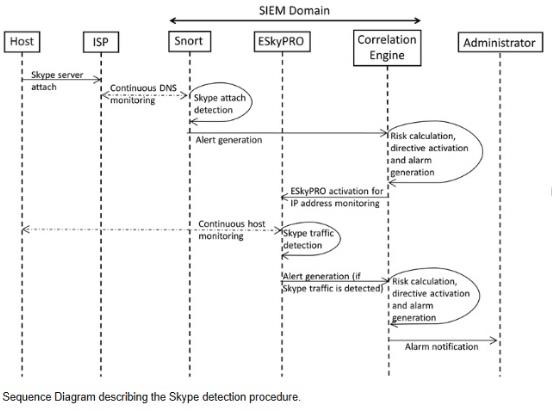
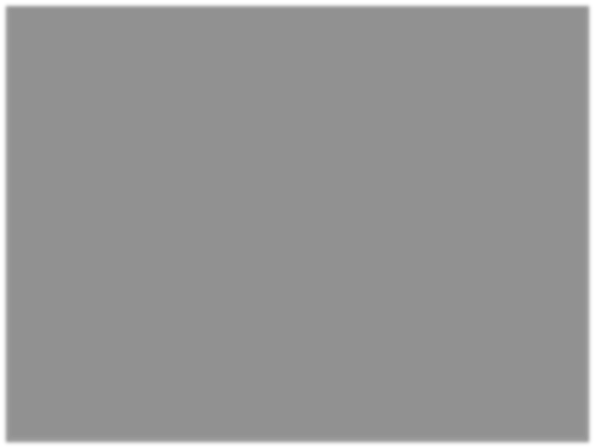


## Sequence Diagram



**5.2**

**Data Design**



Data in Skype is categorized into structured and unstructured formats:

1. **Structured Data:**

Stored in relational databases (e.g., SQL) for user profiles, chat logs, and call histories.

1. **Unstructured Data:**

Stored in NoSQL systems for shared files and multimedia content.

**Key Entities:**

* **User:**

**Attributes:** Name, Phone-No, D.O.B, User-ID, Email-ID

* **Offline Members:**

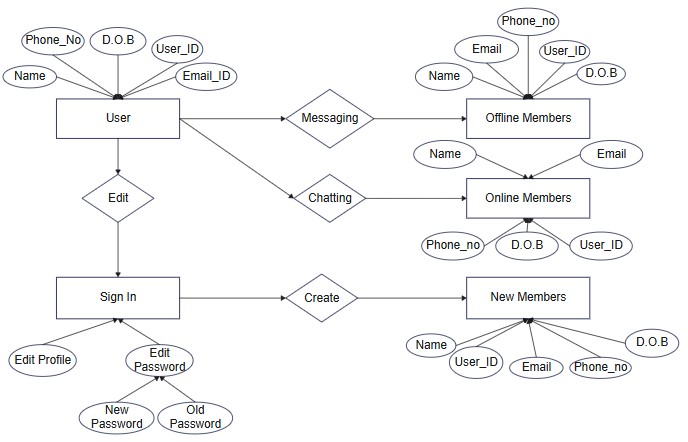
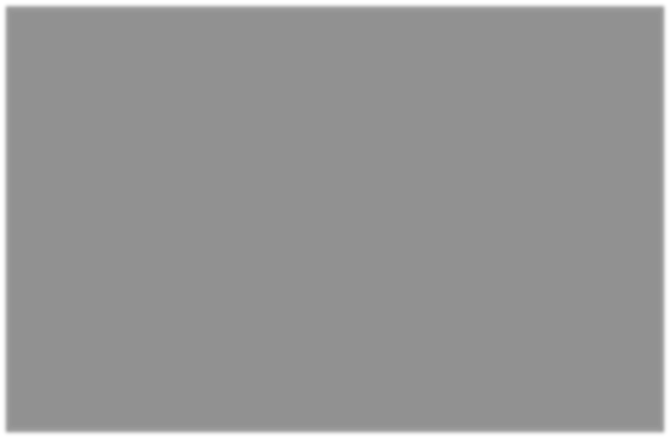
**Attributes:** Name, Email, Phone-No, User-ID, D.O.B

* **Online Members:**

**Attributes:** Name, Email, Phone-No, D.O.B, User-ID

* **Sign in:**
* **Attributes:** Edit-profile, edit-password, new-password, old-password
* **New Member**
* **Attributes:** Name, User-ID, email, D.O.B, phone-no

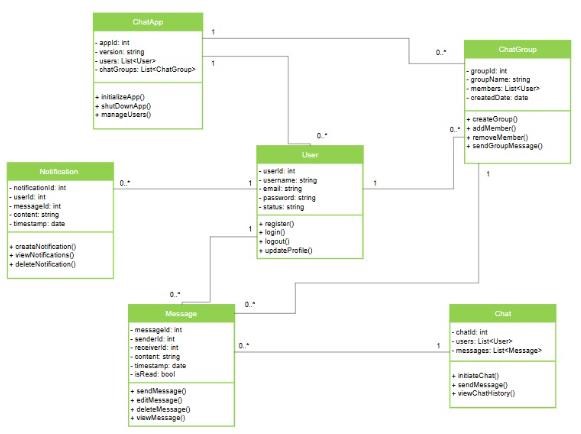
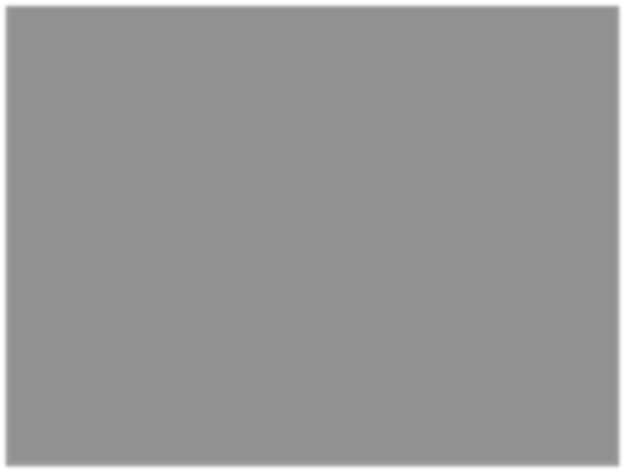
## Entity-Relationship Diagram



**Class Diagram**

**5.3**

**User Interface Design**



Skype’s interface is designed for simplicity and usability:

1. **Navigation:**

Easy access to calls, chats, and contact management.

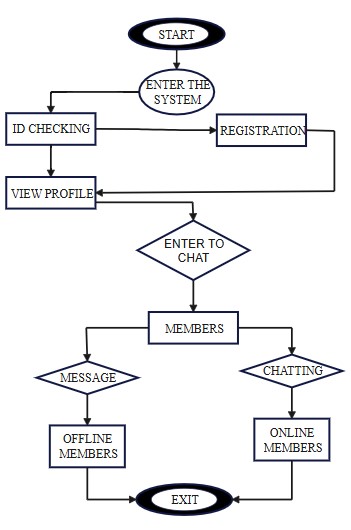
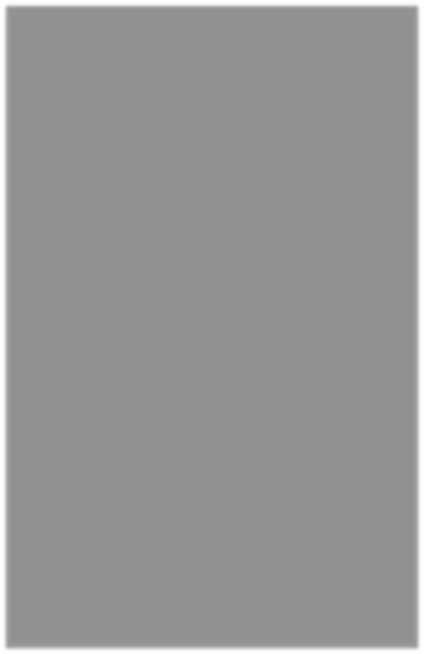
1. **Feedback Mechanisms:**

Real-time notifications for message delivery and call status.

1. **Accessibility Features:**

Support for multiple languages and screen readers.

**Activity Diagram:**



# 6. Software Implementation

## 6.1 Platform Details

Skype is implemented using a combination of technologies:

1. **Languages:**

JavaScript, C#, Objective-C.

1. **Frameworks and Tools:**

WebRTC for real-time communication.

Microsoft Azure for cloud storage and scalability.

1. **Deployment:**

Containerized using Docker for efficient deployment and scaling.

## 6.2 Code Structure

1. **UI Layer:**

Handles user interactions and input validation.

1. **Business Logic Layer:**

Manages messaging, calling, and profile operations.

1. **Data Access Layer:**

Interfaces with databases to retrieve and store information.

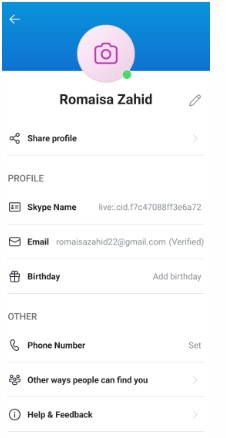
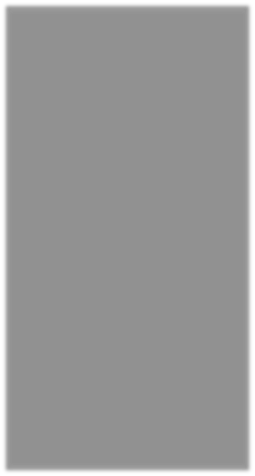
## 6.3 Screenshots

Screenshots of the application include:

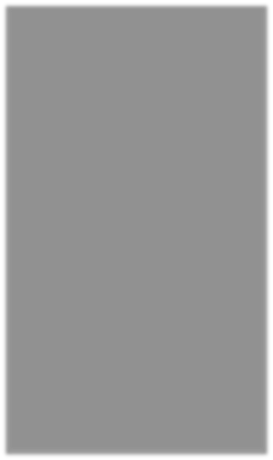
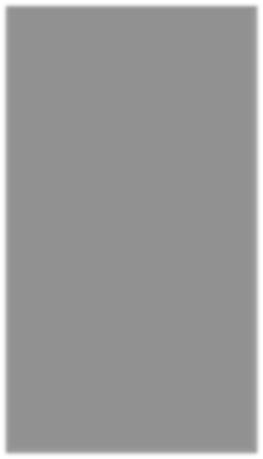
1. Login Page.



1. Profile Management Interface.



2. Chat and Call Screens.



# 7. Software Testing

## 7.1 Testing Levels

1. **Unit Testing:**

Validated individual components like login, messaging, and call features.

1. **Integration Testing:**

Ensured seamless interaction between the UI, business logic, and data layers.

1. **System Testing:**

Verified overall application functionality.

1. **User Acceptance Testing (UAT):**

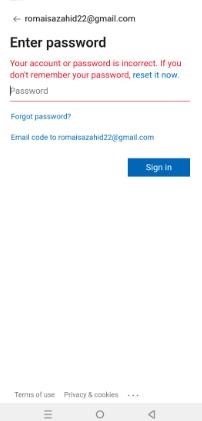
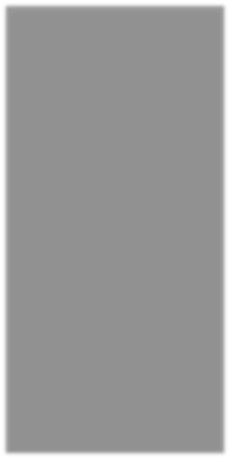
Gathered feedback from real users to confirm usability.

## 7.2 Test Cases and Results

1. **Enter incorrect credentials**.

**Expected Result**: Display error message.

**Actual Result**: Error Message displayed.

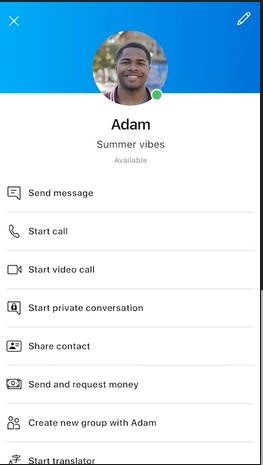
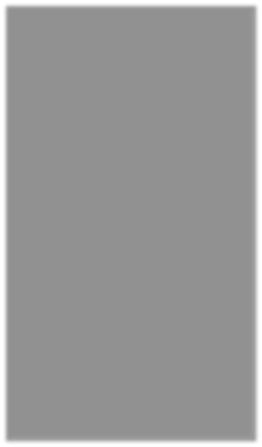


**Status:** Pass

1. **Search for friend in contact**

Click on search and write name of person **Expected Result:** user profile get opened

**Actual Result:** User Profile found



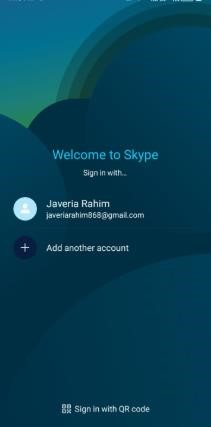
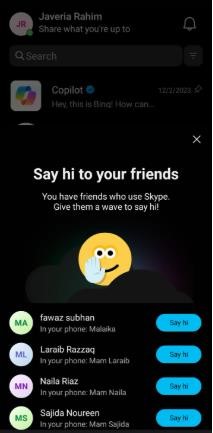
**Status:** Pass

1. **User Sign in**

Click on sign in button

**Expected Output**: Successfully login

**Observed Result**: Successfully login

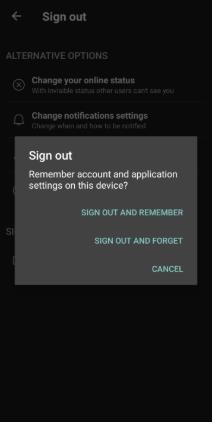
**Status:** Pass

1. **User Sign Out**

Click on Sign out Button

**Expected Result:** User should be sign out from Account

**Observed Result:** User is sign out of account



**Status**: Pass

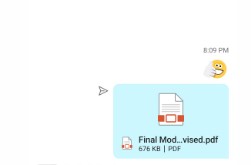
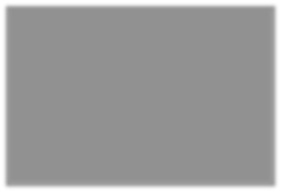
1. **Able to send Emojis**

**Expected Result:** emojis sent

**Observed Result:** emojis sent

**Status:**

Pass



1. **Message**

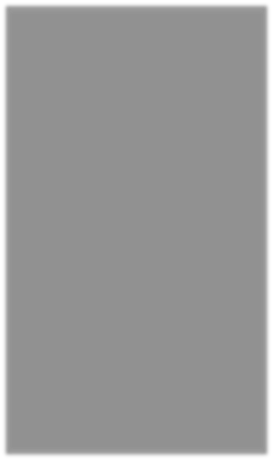
Click on send button

**Expected Result**: Message sent

**Observed Result**: Message sent

**Status:**

pass



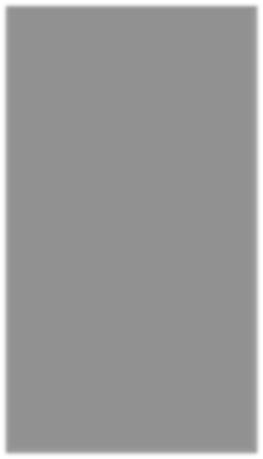
**7.Video Calling**

**Expected Result:** Incoming call sent

**Observed Result:** Call received

**Status:**

pass

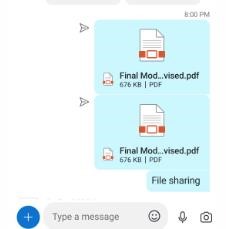
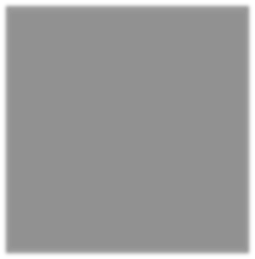


**8.File Sharing**

Click on send button with attached file with it.

**Expected Result:** File sent

**Observed Result:** File sent



**Status:** pass

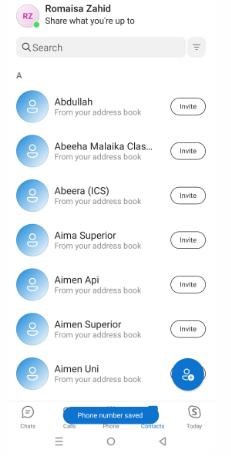
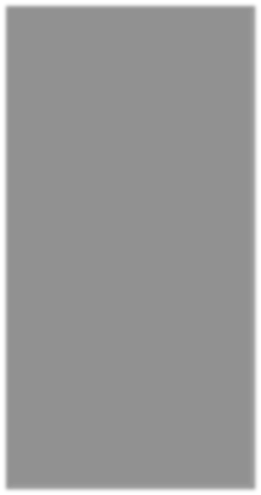
**9.Add Contact**

**Expected Result:** contact added to friend list

**Observed Result:** contact saved

**Status:**

pass



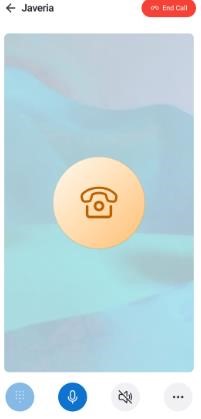
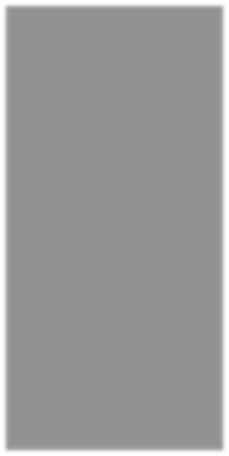
**10. Calling**

**Expected Result:** call received if online

**Observed Result:** Call received

**Status:**

pass



**Performance Metrics**

**Response Time:** 180ms for messaging.

**Call Setup Time:** 2 seconds.

**System Uptime:** 99.98% during testing period.

# 8. Conclusion

The development and implementation of Skype have addressed critical communication needs. Skype has proven to be a reliable and efficient communication platform, The application’s robust architecture, feature-rich design, and focus on user security have made it a leading choice for personal and professional communication.

## 8.1 Limitations

**Internet Dependency**:

Skype’s functionality is heavily reliant on a stable internet connection, and performance may degrade under poor network conditions.

**Limited Offline Features**:

The lack of offline capabilities restricts access to certain functionalities when not connected to the internet.

## 8.2 Future Work

To further enhance Skype’s functionality and user experience, the following improvements can be considered:

**AI Integration**:

Implement advanced AI tools for noise suppression, automatic transcription, and real-time language translation during calls.

**Expanded Collaboration Tools**

Introduce features like shared whiteboards and integrated task management for professional users.

**Enhanced Offline Features**:

Develop offline capabilities such as drafting messages and storing meeting notes for future synchronization.

**AR/VR Integration**:

Explore the use of augmented and virtual reality technologies for immersive communication experiences.

By addressing these areas, Skype can continue to evolve and maintain its position as a leading communication platform.

# 9. References

* IEEE 29148-2018 Standard for Requirements Engineering.
* Skype API Documentation - Microsoft.
* [https://learn.microsoft.com/en-us/skype-sdk.](https://learn.microsoft.com/en-us/skype-sdk) ▪ Skype Features and Benefits - Official Skype Website ▪ [https://www.skype.com/en/features/.](https://www.skype.com/en/features/)
* User Feedback and Market Research Reports (2022).
* WebRTC Official Documentation.
* [https://webrtc.org/.](https://webrtc.org/)