

A Project Report On

Vidzy

Submitted in partial fulfillment of the
requirement for the award of the degree

**MASTER OF COMPUTER
APPLICATIONS
(M.C.A.)**

Academic Year 2025 – 26

Harsh Bhupatkar 92400584097

Internal Guide

Amit Gohel



Marwadi
University
Marwadi Chandarana Group





Marwadi
University
Marwadi Chandarana Group



Faculty of Computer Applications (FCA)

Certificate

This is to certify that the project work entitled
Vidzy

submitted in partial fulfillment of the requirement for
the award of the degree of

MASTER OF COMPUTER APPLICATIONS
(M.C.A.)

of the

Marwadi University

is a result of the bonafide work carried out by

Harsh Bhupatkar (92400584097)

during the academic year 2025-26

Faculty Guide

HOD

Dean

CERTIFICATE

DECLARATION

I hereby declare that this project work entitled **Vidzy** is a record done by me.

I also declare that the matter embodied in this project is genuine work done by me and has not been submitted whether to this University or to any other University / Institute for the fulfillment of any course of study.

Place: Rajkot

Date:

Harsh Bhupatkar (92400584097)

Signature:_____

ACKNOWLEDGEMENT

It is indeed a great pleasure to express my thanks and gratitude to all those who helped me. No serious and lasting achievement one can ever achieve without the help of friendly guidance and co-operation of so many people involved in the work.

I am very thankful to my faculty guide **Prof. Amit Gohel**, the person who makes me follow the right steps during project work. I express my deep sense of gratitude to his guidance, suggestions, and expertise at every stage. Apart from that his valuable and expertise suggestion during documentation of my report indeed helped me a lot.

Thanks to my friends who have been a source of inspiration and motivation that helped me during my project work. I am heartily thankful to our external guide **Mr.Meet Patel** for providing me an opportunity to work over this report and for their endless and great support. I am also thankful to my Head **Dr. Sunil Bajaja** and Dean **Prof. (Dr.) Rijwan Khan** for their support as and when required. I would like to extend my gratitude towards all who directly or indirectly supported and helped me to accomplish my project work.

Harsh Bhupatkar (92400584097)

Signature:_____

CONTENTS

Chapters	Particulars	Page No.
1	SYNOPSIS	1
2	PREAMBLE	3
2.1	General Introduction	3
2.2	Statement of Problem	4
2.3	Objective and Scope of the Study	5
2.4	Module Description with functionality	6
2.5	Methodology	8
2.6	Feasibility Study	9
3	REVIEW OF LITERATURE (Study of Existing Similar Types of Systems)	10
4	TECHNICAL DESCRIPTION	11
4.1	Hardware Requirement	11
4.2	Software Requirement	11
5	SYSTEM DESIGN AND DEVELOPMENT	12
5.1	Architectural Design / System Flow	12
5.2	Data Flow Diagram (DFD)	13
5.3	Structural Modeling	15
5.3.1	● Class Diagram	15
5.4	Behavioral Modeling	16
5.4.1	● Use Case Diagram	16
5.4.2	● Sequence Diagram	18
5.4.3	● Activity Diagram	19
5.5	Database Design	20
5.5.1	● Table Structure	20
5.5.2	● Entity Relationship Diagram	21
5.6	Screen Design (Screen shots with short description)	22
6	SYSTEM TESTING	28
6.1	Testing and Implementation	28
6.2	Testing Methodology	29
6.3	Test Case Design	30
6.4	System Implementation Strategy	32
7	CONCLUSION	33
8	LEARNING DURING PROJECT WORK	34
9	FUTURE ENHANCEMENTS (if applicable)	35
10	BIBLIOGRAPHY	36
10.1	Online References	36
10.2	Offline References	37

FIGURE INDEX

Sr. No.	Figure No.	Particulars	Page No.
1	5.1	Flow Chart	12
2	5.2.1	Context DFD	13
3	5.2.2	DFD level 1	14
4	5.3.1	Class Diagram	15
5	5.4.1.1	Use Case Diagram (User)	16
6	5.4.1.2	Use Case Diagram (Admin)	17
7	5.4.2	Sequence Diagram	18
8	5.4.3	Activity Diagram	19
9	5.5.2	Entity Relationship Diagram	21
10	5.6.1	Login Screen	22
11	5.6.2	Meet & Chat Screen	23
12	5.6.3	Chat Screen	24
13	5.6.4	Meeting Screen	25
14	5.6.5	Settings Screen	26
15	5.6.6	Logout Screen	27

TABLE INDEX

Sr. No.	Table No.	Particulars	Page No.
1	4.1.1	Hardware Requirements (Developer)	11
2	4.1.2	Hardware Requirements (User)	11
3	4.2.1	Software Requirements (Developer)	11
4	4.2.2	Software Requirements (User)	11
5	6.3	Test Cases	30

1. SYNOPSIS

Title: Vidzy – A Unified Real-Time Communication System Using Flutter, Firebase, and Zego Cloud

Introduction:

Vidzy is a simple and easy to use one-to-one and group communication app. Vidzy lets users to chat with another user. Apart from chat functionality user can create or join the video call meeting which lets users to communicate via App. Our app uses Zego cloud SDK for video call functionality. In short Vidzy is combination of Whatsapp and Google Meet.

Problem Statement:

There are many platforms available in the market that let user to chat with user. There are platforms to do perform video conferencing. But user need to use two different app to perform such task. To overcome this hurdle we have created Vidzy all in one chat and video call app.

Objectives:

- To let user to send and receive message.
- Let user to communicate with other user via video call.
- Let user to create or join external meetings.
- Provide simple and easy to use user interface.

Scope:

Vidzy provides users to chat with other users who are using our app. User can create or join external meeting. In which we provide host control. Where without permission of the host one can't join the meeting. One can also call the other user via video call. This platform does not provide any kind of admin controls.

Technology Stack:

Flutter, Zego Cloud SDK, Firebase.

Architecture:

Our application is developed using Flutter. Flutter provides write once and run on multiple platform functionality. We use Firebase for Cloud data storage. We use Zego Cloud SDK for video conferencing.

Feasibility:

This project is feasible as flutter is open source. Zego cloud provide 10,000 minutes free video calling API. Firebase also free if we want to build just prototype level application.

Expected Outcome:

A functional mobile app in which there will be features like one-to-one video call, one-to-one chat functionality, External create or joining meeting functionality. Simple and easy to use User Interface.

Limitations:

Performance is totally dependent on quality of internet connectivity. We rely on Zego Cloud for providing video conferencing service. If it goes down then it will affect the video call feature.

Future Enhancements:

Group creation for chat, Recording and live transcription of video call. Firebase push notification for message, etc.

Conclusion:

We provide easy and simple to use chat and video call feature in one single app. Such that user does not need to visit many different apps to perform video conferencing and chat.

2. PREAMBLE

2.1 General Introduction.

Since 2020, Communication via video has increased. There are platforms such as Google Meet, Zoom, etc. provides functionality for video conferencing. But to chat with user we must platforms like Whatsapp, etc. Use two or more platforms to chat and video conferencing is hurdle and pure gap.

Our platform Vidzy is resolving this gap. Our platform Vidzy is designed to provide feature of one-to-one chat functionality and external meeting feature. When one can chat with other user and create or join external meeting which we call video conference in simple terms.

Vidzy is developed on Flutter. Flutter is an open-source framework build by Google. The main benefit of using flutter is that developer must write code once and it can be run on mobile, tablet, windows, macOS, or as a website. For video conferencing we are using SDK of Zego Cloud. They are providing feature of one-to-one video call and group video call. For Database we are using Firebase which Baas (Backend as a service) developed by Google.

The primary purpose of Vidzy is to fill the gap of using two different app for video calling and chat functionality. Our aim is to create an app which is easy to use with simple user interface. We have provided basic functionality in this app.

This project serves how open-source framework like Flutter can help to create app and can be used to different Mobile devices like Android and IOS. We have integrated video conferencing functionality using Zego Cloud. Which has helped us to utilize the functionality of Video Conferencing. Overall we have tried to serve the simple and easy to use video conferencing and chat app.

2.2 Statement of Problem

In today's era, communication via digital platforms is booming. The main reason behind rise was Covid where everyone was locked inside their home. In that time communication via video conferencing boomed. Specially in education industry as all the schools and universities were closed due to Covid. To perform video conferencing there are platforms like Google Meet, Zoom, etc. But each has their own drawback.

Back then, communication via chat was already there. But to communicate via chat and video call we must use two different app. For example. We use whatsapp for chat and Google meet or other platforms for video conferencing. There was no direct app which provides feature of both chat and video conferencing in one single app.

That's where Vidzy come to solve this gap. Vidzy is easy to use communication app. Which provides functionality of one-to-one chat and video call functionality, and one can create or join the meeting or video conference just like Zoom or Google Meet.

Vidzy is developed on Flutter. Flutter is an open-source framework build by Google. The main benefit of using flutter is that developer must write code once and it can be run on mobile, tablet, windows, macOS, or as a website. For video conferencing we are using SDK of Zego Cloud. They are providing feature of one-to-one video call and group video call. For Database we are using Firebase which Baas (Backend as a service) developed by Google.

In simple manner, Vidzy is fulfilling the gap of using two or more different app for getting feature of Chat and Video Conferencing.

2.3 Objective and Scope of the Study

Objective of building this app was to create an easy to use and simple mobile application. By using which user can use functionality of one-to-one chat and video call functionality in one application.

Scope of the Study

The scope defines what the project will and will not cover.

In-Scope Features

- User login using Google authentication
- Creating a meeting or joining via meeting code
- Real-time video and audio communication
- One-to-one chat with real-time messaging
- Simple and clean user interface

Out-of-Scope Features

- Cloud recording and storage of meetings
- Advanced enterprise tools like breakout rooms, noise suppression, or attendance reports
- Admin-level meeting moderation
- Self-hosted servers
- Web browser version of Vidzy (mobile app only)

2.4 Module Description with Functionality

1. Login Module

Functionality:

- Allows users to log in using Google Authentication.
- Verifies identity and creates a session for the user.
- Stores only basic profile information required to use the app.

Purpose:

Ensures secure and quick access without traditional sign-up forms.

2. Chat Module

Functionality:

- Enables one-to-one and group chat between registered users.
- Supports real-time message exchange using Cloud Firestore.

Purpose:

Provides continuous text-based communication and acts as the entry point for direct calls.

3. Direct Call Module

Functionality:

- Allows users to initiate and receive video calls directly from chat.
- Uses call signalling to manage incoming, accepted, and rejected calls.
- Provides call alerts and waiting-state handling before connection.

Purpose:

Direct call module let users to create one-to-one video call functionality.

4. External Meeting Module

Functionality:

- This module generates a unique meeting ID.
- Participants can join the meeting by using meeting ID.
- One must wait until host approves him/her in waiting screen.

Purpose:

Provides functionality to create or join external meeting.

5. Meeting Interface Module

Functionality:

- It provides user interface of video conference.
- Provides controls like mic/camera turn on and off.

Purpose:

Provides meeting interface.

6. Backend/Database Module

Functionality:

- Stores data of meeting, chat and user.

Purpose:

It ensures proper access and storage of the data.

2.5 Methodology

The development of vidzy follows a proper structure of making and creating mobile application.

1. Requirement Analysis

- Identified the clear gap of using two or more features for chat and meeting functionality.
- Identified what are requirements of the project by comparing existing project like Whatsapp, Zoom,etc.

2. Implementation

- Designed core user interface
- Integrated Google Sign-In using Firebase Authentication.
- Integrated Zego Cloud SDK for audio and video communication.
- Implemented call handling and chat logic.

3. Deployment

- Created and built apk file for testing on different physical devices.
- Verified core functionality on physical devices.

2.6 Feasibility Study

1. Technical Feasibility

Vidzy is technical feasible as it is built on open-source technology Flutter.

- Flutter provides a stable cross-platform framework for app development.
- Zego Cloud SDK offers ready-made video calling infrastructure, removing the need to build complex RTC systems from scratch.
- Login via google is provided by firebase which free till some extend.

Conclusion: The project is fully achievable with available technology.

2. Operational Feasibility

The aim of the project is to create easy and simple User Interface. Which eventually led easy operation and maintenance of the application.

- Sign in via google is provided by Firebase
- Data is stored in Firebase.
- We have provided many controls such as mic/camera on and off feature, Chat in the meeting, etc. features.

Conclusion: The system is user-friendly and operationally efficient.

3. Economic Feasibility

Vidzy uses cost-effective tools:

- Flutter (free)
- Firebase (free tier options)
- Zego Cloud SDK(free tier).There are no licensing fees or server expenses unless a custom Zego Cloud server is deployed. Development and testing can be done on personal devices without additional cost.

Conclusion: The project is economically viable for a student project and low-budget deployments.

3. REVIEWS OF LITERATURE

While creation of Vidzy we have tried to provide solution of one single application for chat and video conferencing. To study the problem, we have studied different Platform and SDK. Following are the platforms and SDK that we have taken into consideration for identifying the gap.

Zoom

Zoom is video conferencing. One can create or join the meeting. It provides many features such as host controls, camera/mic on and off feature, audio transcription, in meeting chat functionality. Main problem of existing system is that it is not completely free.

Google Meet

Google Meet is also platform of Google. It is a part of Google workspace. In this one can join or create meeting. It is similar like Zoom. Google meet allows users to share their screen. In meeting chat feature and host controls.

Microsoft Teams

Microsoft Teams is product of Microsoft. It is widely used as enterprise level application. It has feature of chat and video conferencing in one single app. We can say that we are similar version of Teams

Zego Cloud (Proprietary Cloud SDK)

Zego cloud provides SDK for video conference. By using this one can communicate via one or more than one people. It is not completely free. It provides 10,000 minutes free access of the SDK.

4. TECHNICAL DESCRIPTION

4.1 Hardware Requirements

Ram	8GB
Storage	256 GB SSD or higher
Processor	Intel i5 / Amd Ryzen 5 or higher
System	Laptop / Desktop capable to run android emulator.

Table 4.1.1 Hardware Requirements (Developer Side)

Ram	2GB Minimum
Storage	200MB Minimum
Camera & Mic	Camera & mic required

Table 4.1.2 Hardware Requirements (User Side)

4.2 Software Requirements

OS	Windows 10/macOs/linux
Flutter Version	Latest stable version
Android Studio	With android sdk & emulator
JDK	Java JDK 11 or above
Tools	Android Studio / Vs Code

Table 4.2.1 Software Requirements (Developer Side)

OS	Android Version 8.0+
Google Services	Google services enabled
Permission	Microphone, Camera

Table 4.2.2 Software Requirements (User Side)

5. SYSTEM DESIGN AND DEVELOPMENT

5.1 System Flow

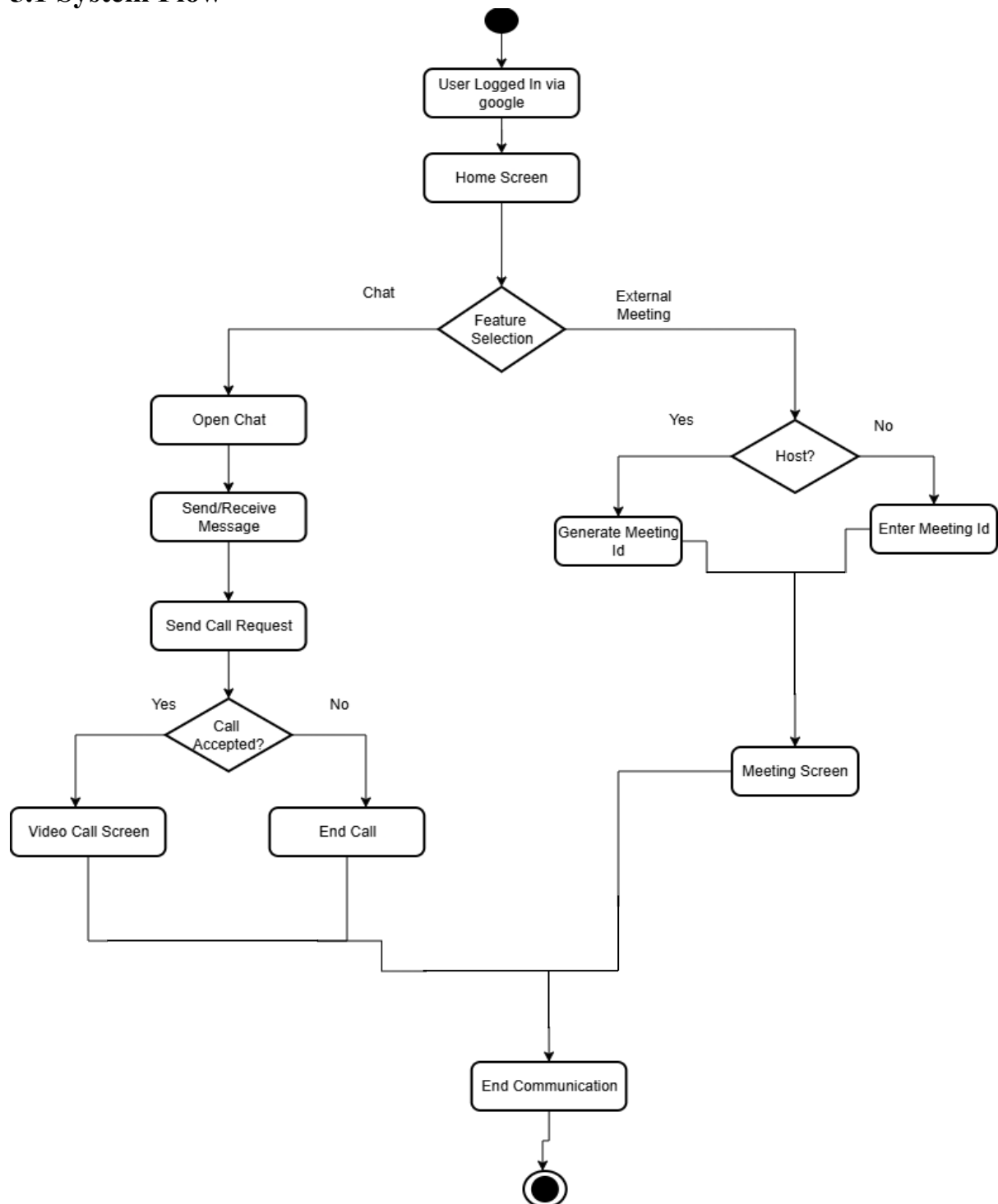


Figure 5.1 Flow Chart

This Diagram is flow chart. It displays the flow of project when user logs in he/she can perform two things chats and meetings. Our platform gives additional functionality of one-to-one video call.

5.2 Data Flow Diagram (DFD)

5.2.1 Context DFD

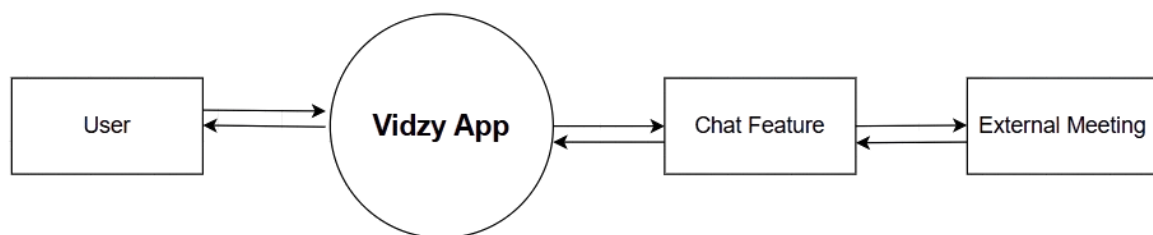


Figure 5.2.1 Context DFD

This is a context level Data Flow Diagram. It displays functionality of the project. It shows that project has two main features. Chat and External Meeting.

5.2.2 DFD Level 1

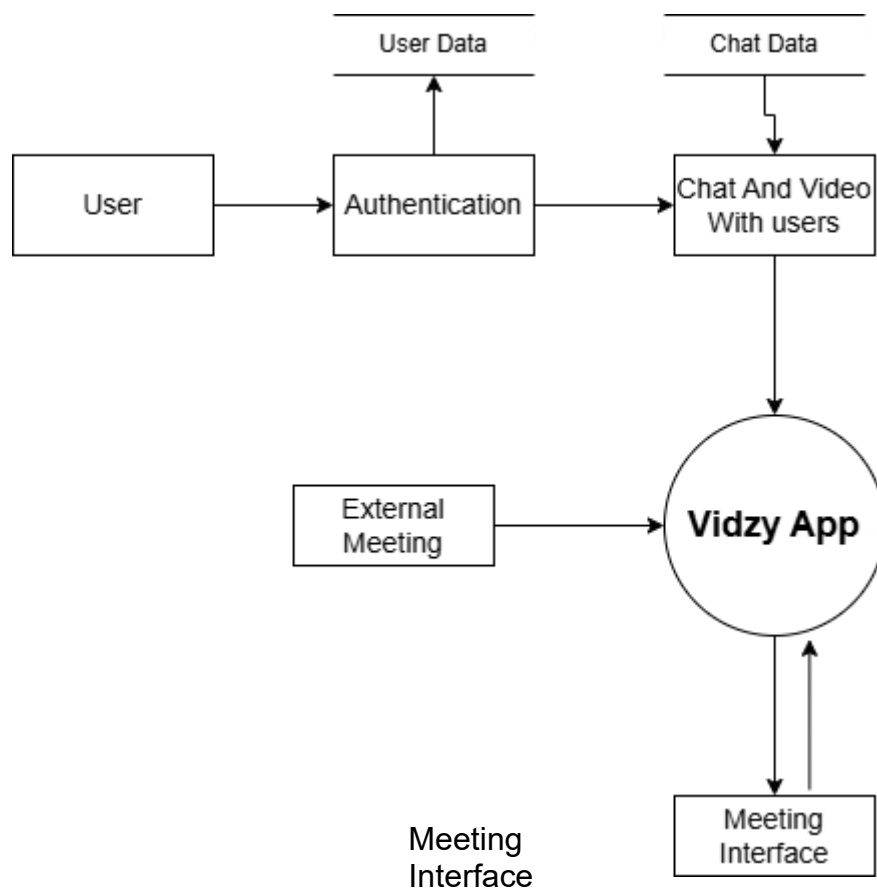


Figure 5.2.2 DFD Level 1

This is a level 1 Data Flow Diagram. When user log in he/she is authenticated by google. After that he/she can chat with any available user. Video call can be made using video call feature. One can avail functionality of external meeting in which one can create or join meeting.

5.3 Structural Modeling

5.3.1 Class Diagram

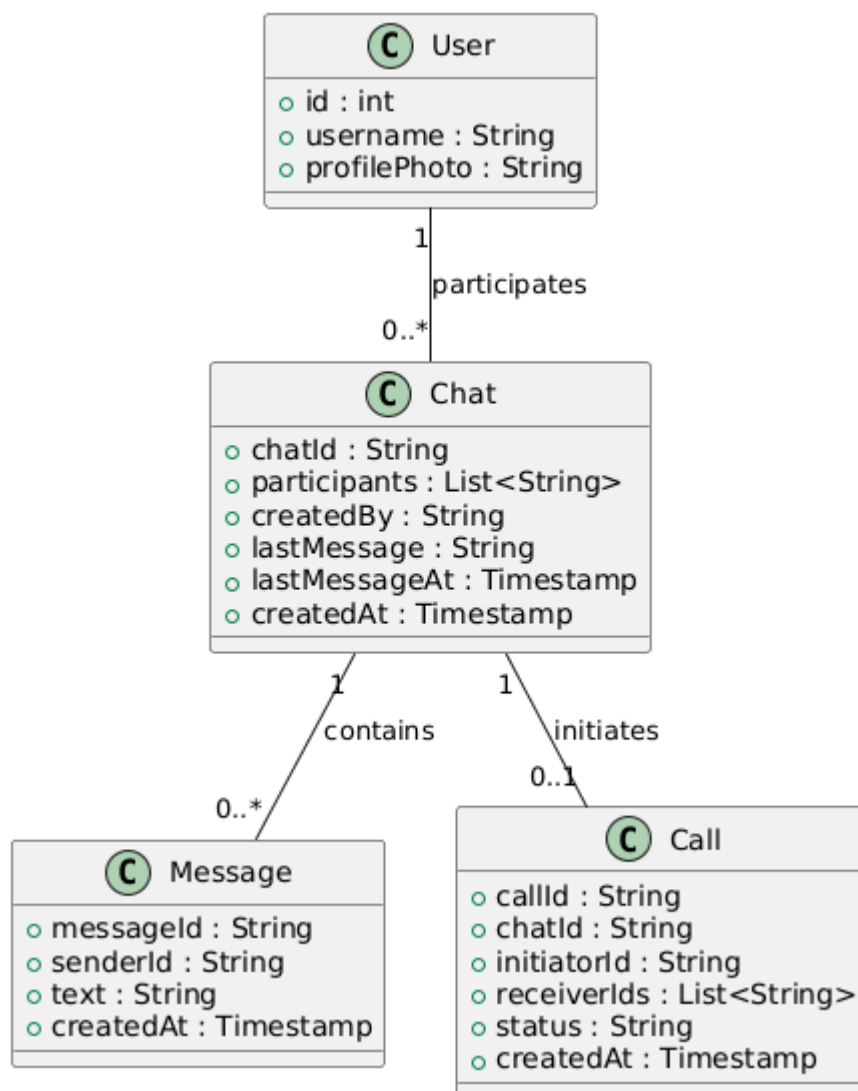


Figure 5.3.1 Class Diagram

This is a class diagram. It shows that we have user class which has following fields, id, username, profilephoto. User participates in chat. Chat class has related field to help to store data of chat in firebase. It also has message and call class which give functionality for message and call.

5.4 Behavioral Modeling

5.4.1 Use Case Diagram (User side)

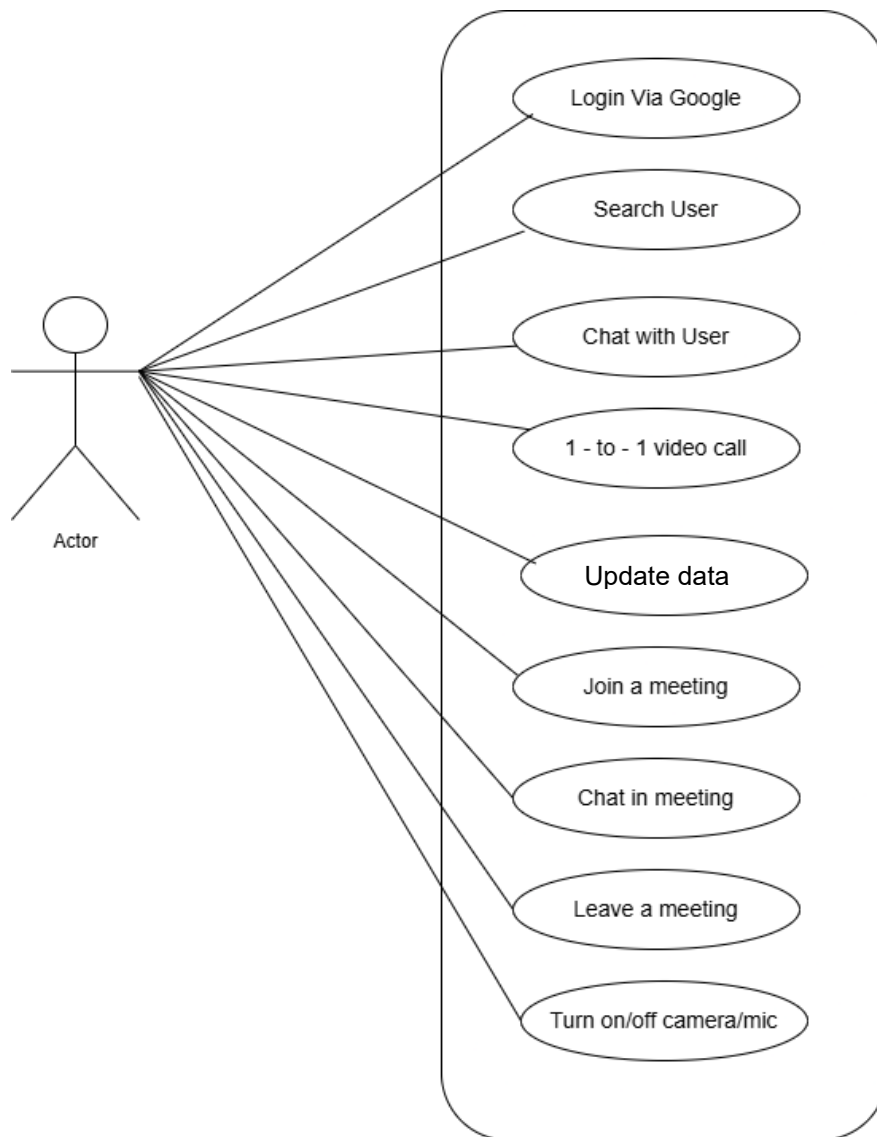


Figure 5.4.1.1 Use Case Diagram (User side)

This is a user case diagram of user side. It shows different actions performed by the user throughout the app. One can log in, chat with user, 1-to-1 video call, update data, join a meeting, chat in meeting, leave a meeting. Turn on/off camera/mic.

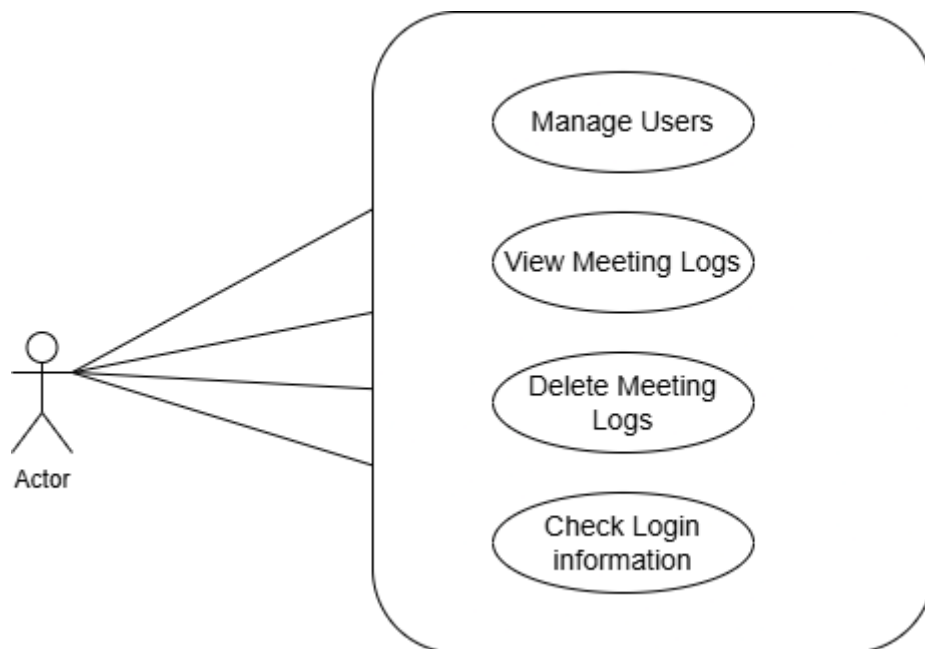


Figure 5.4.1.2 Use Case Diagram (Admin Side)

This is a use case diagram of admin side. Admin can perform tasks like manage users, view meeting logs, delete meeting logs, check login information, etc.

5.4.2 Sequence Diagram

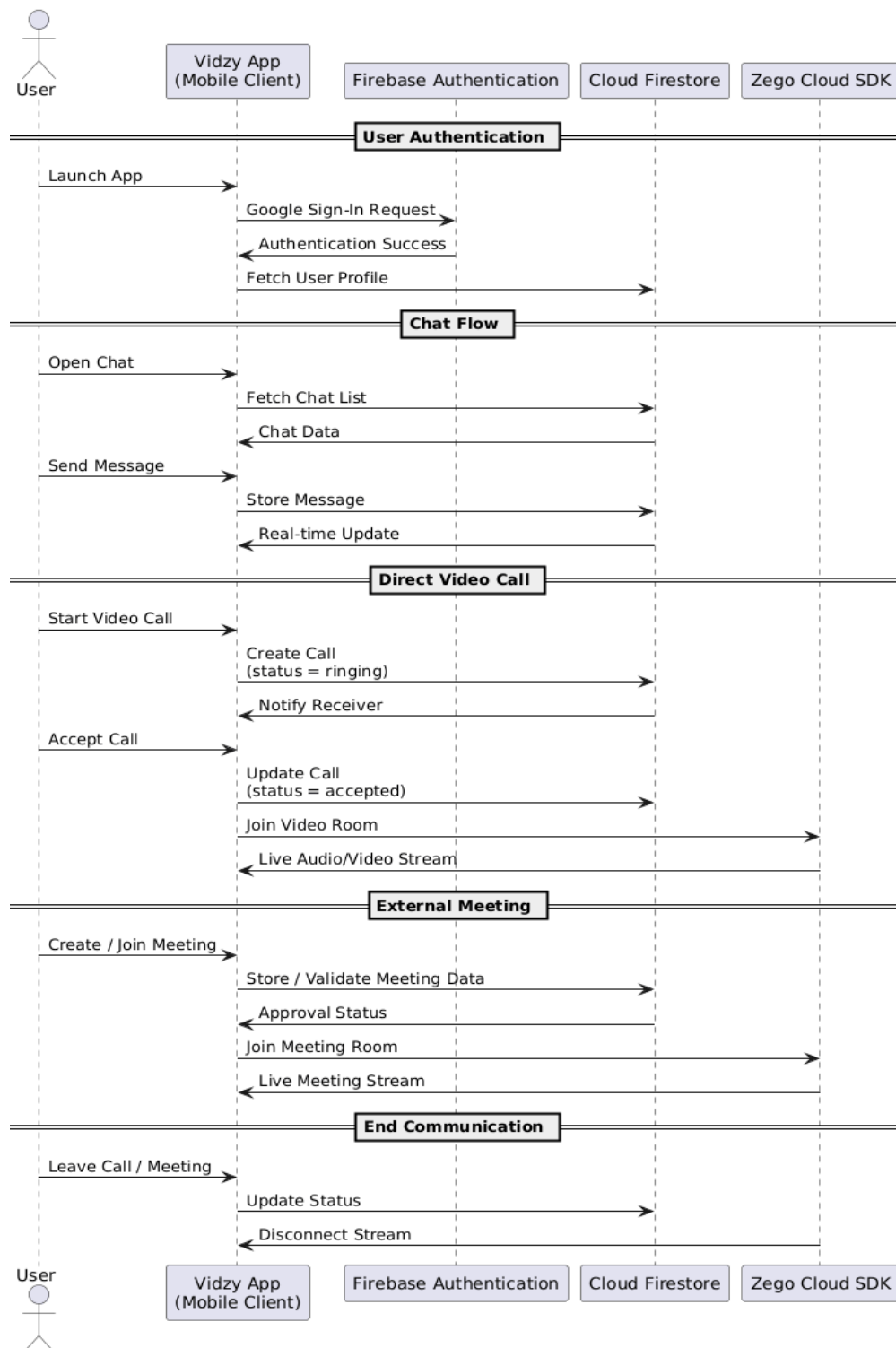


Figure 5.4.2 Sequence Diagram

This is a sequence diagram display the sequence of the various task. How they are executed. It displays step by step navigation between components. How every component works.

5.4.3 Activity Diagram



Figure 5.4.3 Activity Diagram

This is an Activity Diagram. It displays how user navigates to perform different activity. Activity diagram shows sequence of actions performed by the user. The activity diagram shows the complete workflow of the system.

5.5 Database Design

5.5.1 Table Structure

Collection: users

```
{
  "id" : int,
  "profilephoto" : String,
  "username" : String
}
```

Collection: chats

```
{
  "chatId" : String,
  "participants" : List<String>,
  "createdBy" : String,
  "lastMessage" : String,
  "lastMessageAt" : Timestamp,
  "createdAt" : Timestamp
}
```

Sub-Collection: chats/{chatId}/messages

```
{
  "messageId" : String,
  "senderId" : String,
  "text" : String,
  "createdAt" : Timestamp
}
```

Collection: calls

```
{
  "callId" : String,
  "chatId" : String,
  "initiatorId" : String,
  "receiverIds" : List<String>,
  "status" : String, // ringing | accepted | rejected | ended
  "createdAt" : Timestamp
}
```

The users collection stores basic user profile information required for identification and communication within the system. The chats collection manages conversation metadata, while the messages sub-collection stores individual messages exchanged between participants. The calls collection handles voice/video call details, including participants, call status, and timestamps for tracking call lifecycle.

5.5.2 Entity Relationship Diagram

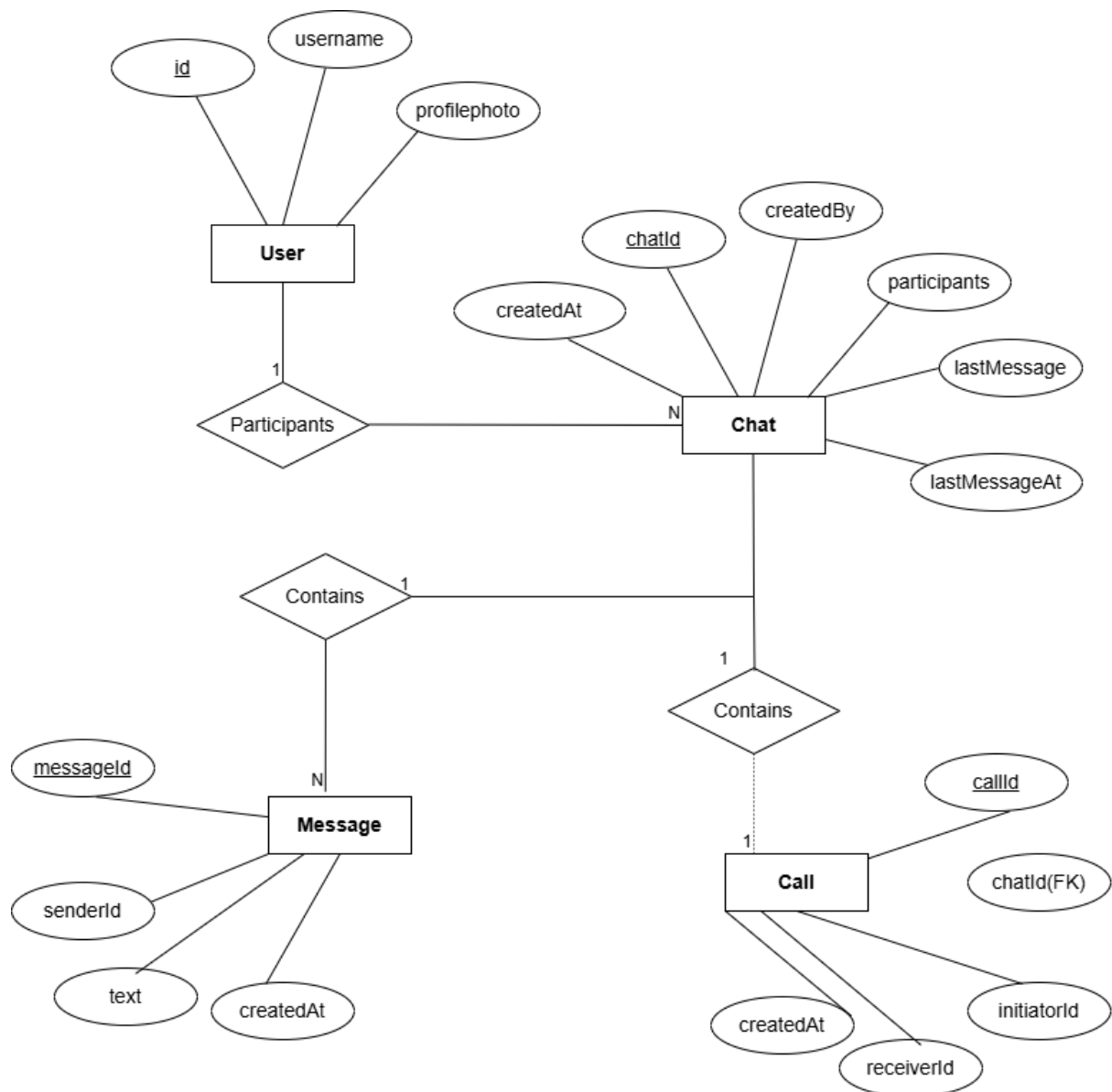


Figure 5.5.2 Entity Relationship Diagram

This is an Entity Relationship Diagram. This diagram explains relationships between entities and how data flows during messaging and calling.

5.6 Screen Design

Login Screen

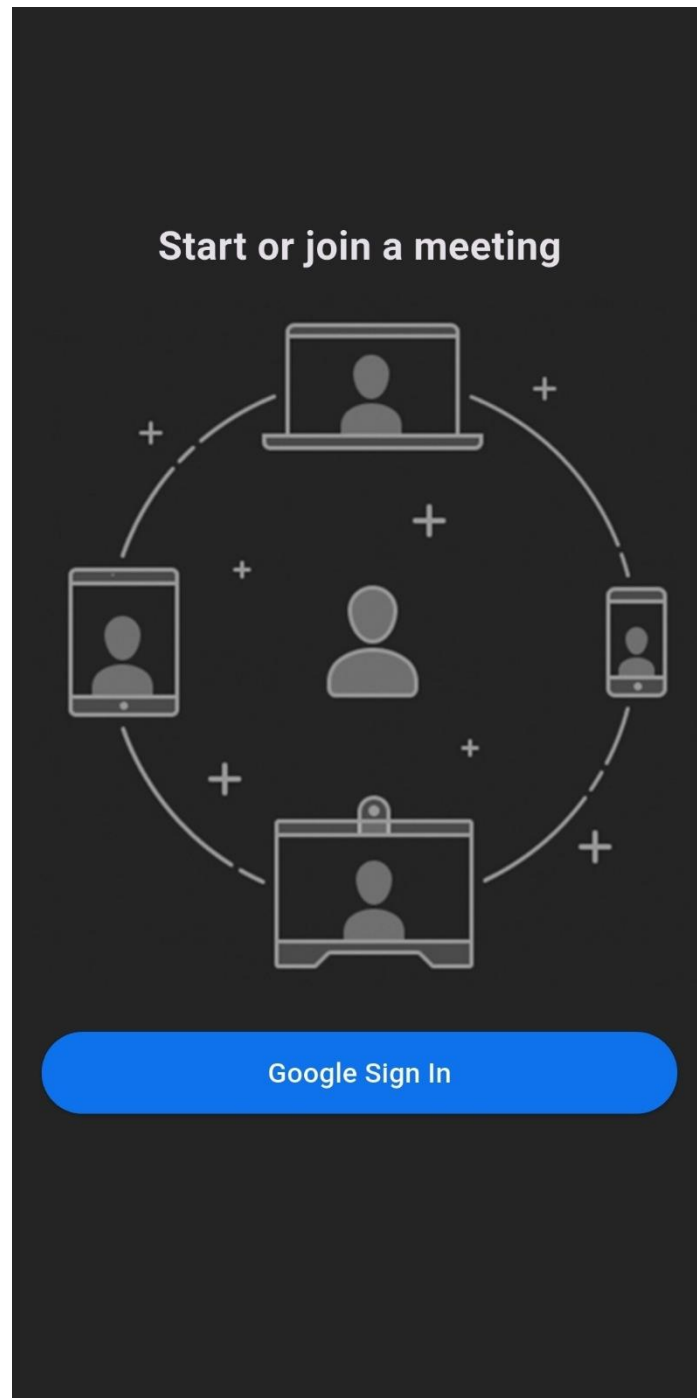


Figure 5.6.1 Login Screen

This image is of login screen. It provides functionality of login into the system via Google. Our system uses Google Mail Authentication.

Meet & Chat Screen

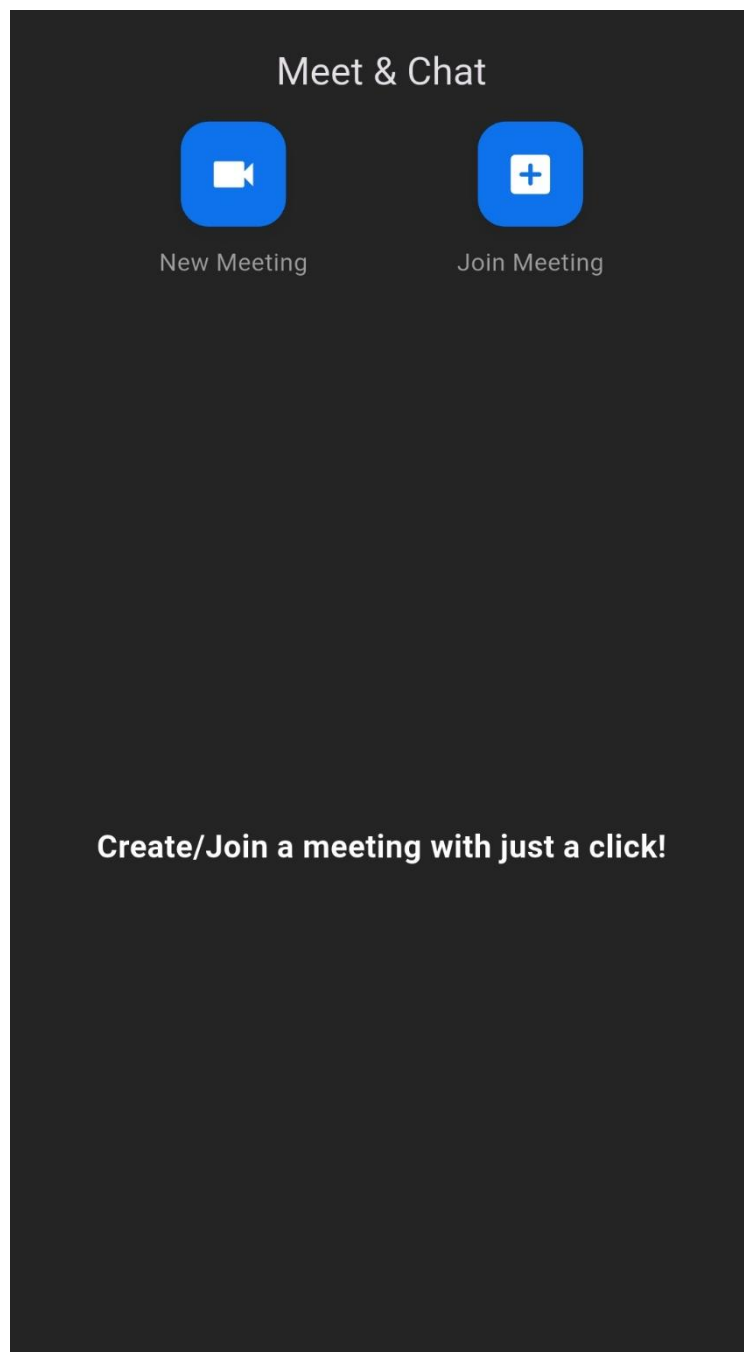


Figure 5.6.2 Home Screen

This screen displays two options to user. User can either join a meeting or create a meeting. If user clicks on create meeting user will be considered as host else user will be navigated to the join meeting option.

Chat Screen

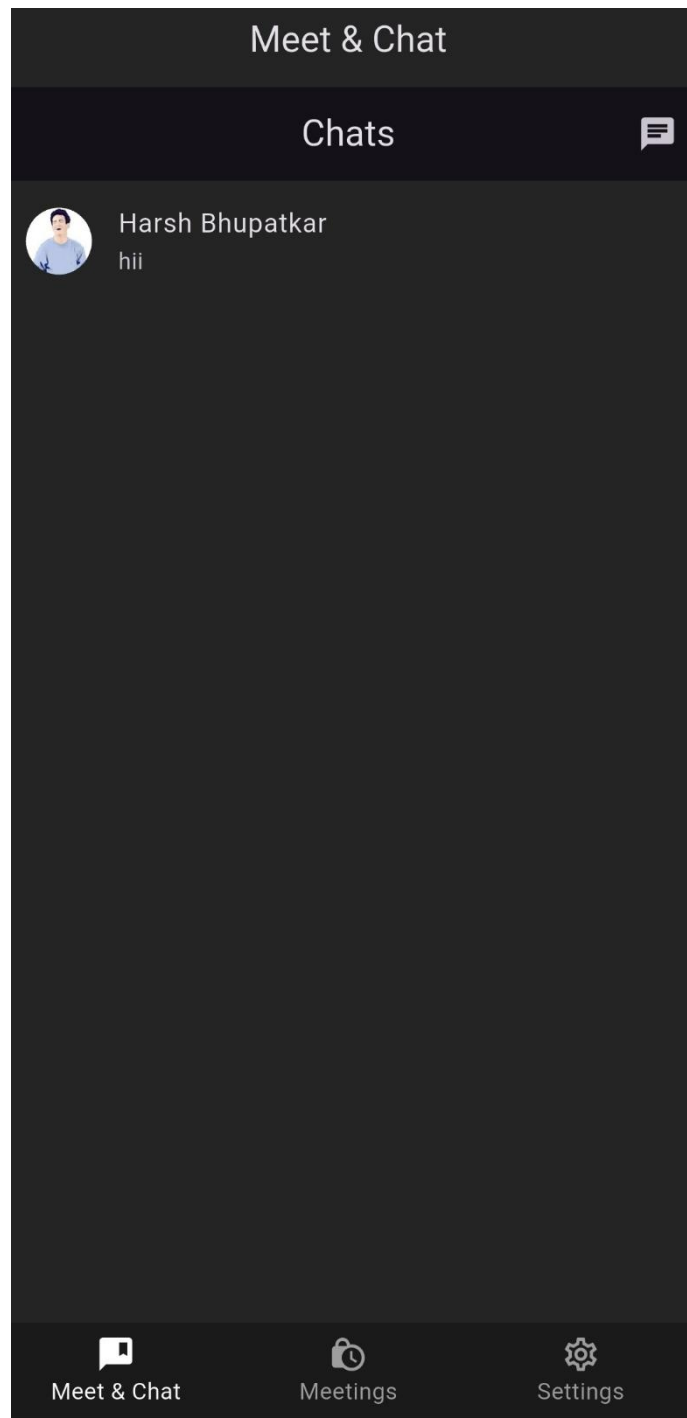


Figure 5.6.3 Chat Screen

This is a chat screen. It displays list of users with whom user has started chatting. User can see the last message sent by sender.

Meeting Screen

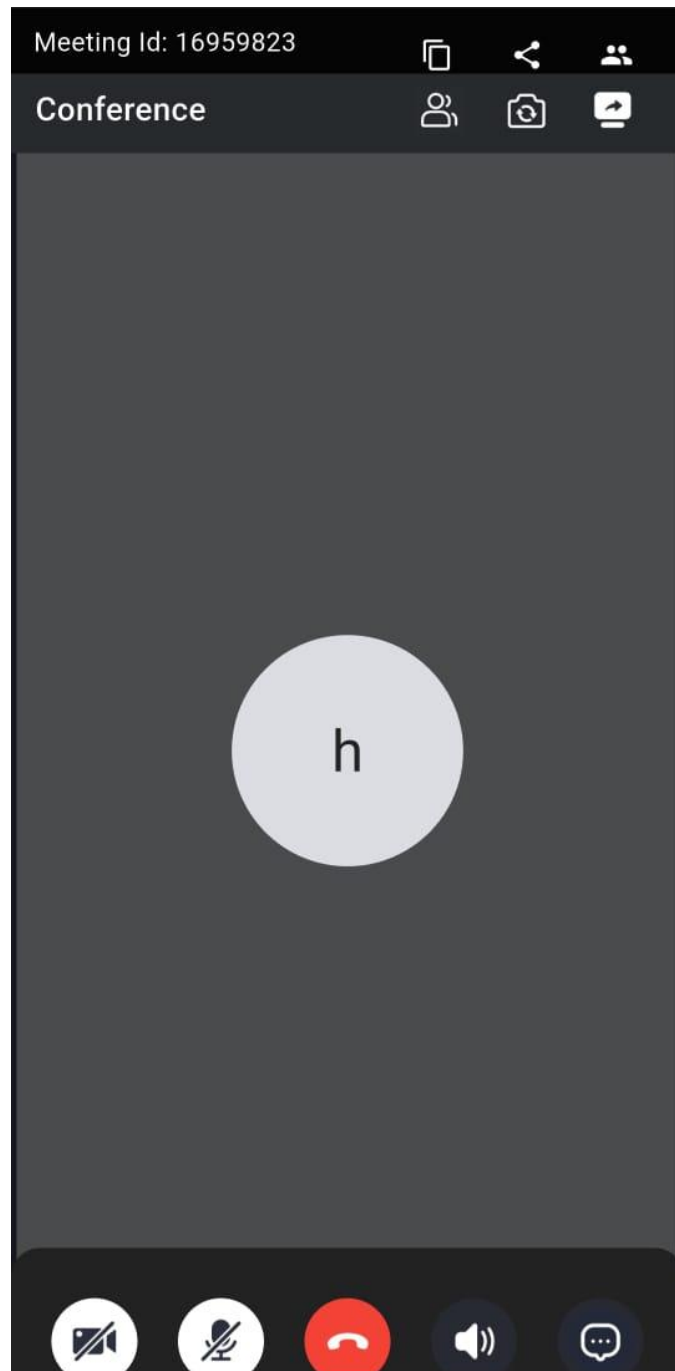


Figure 5.6.4 Meeting Screen

This is a meeting screen. It displays various options. One can chat with other users and talked with other participants. Participant can turn on/off his mic/camera.

Settings

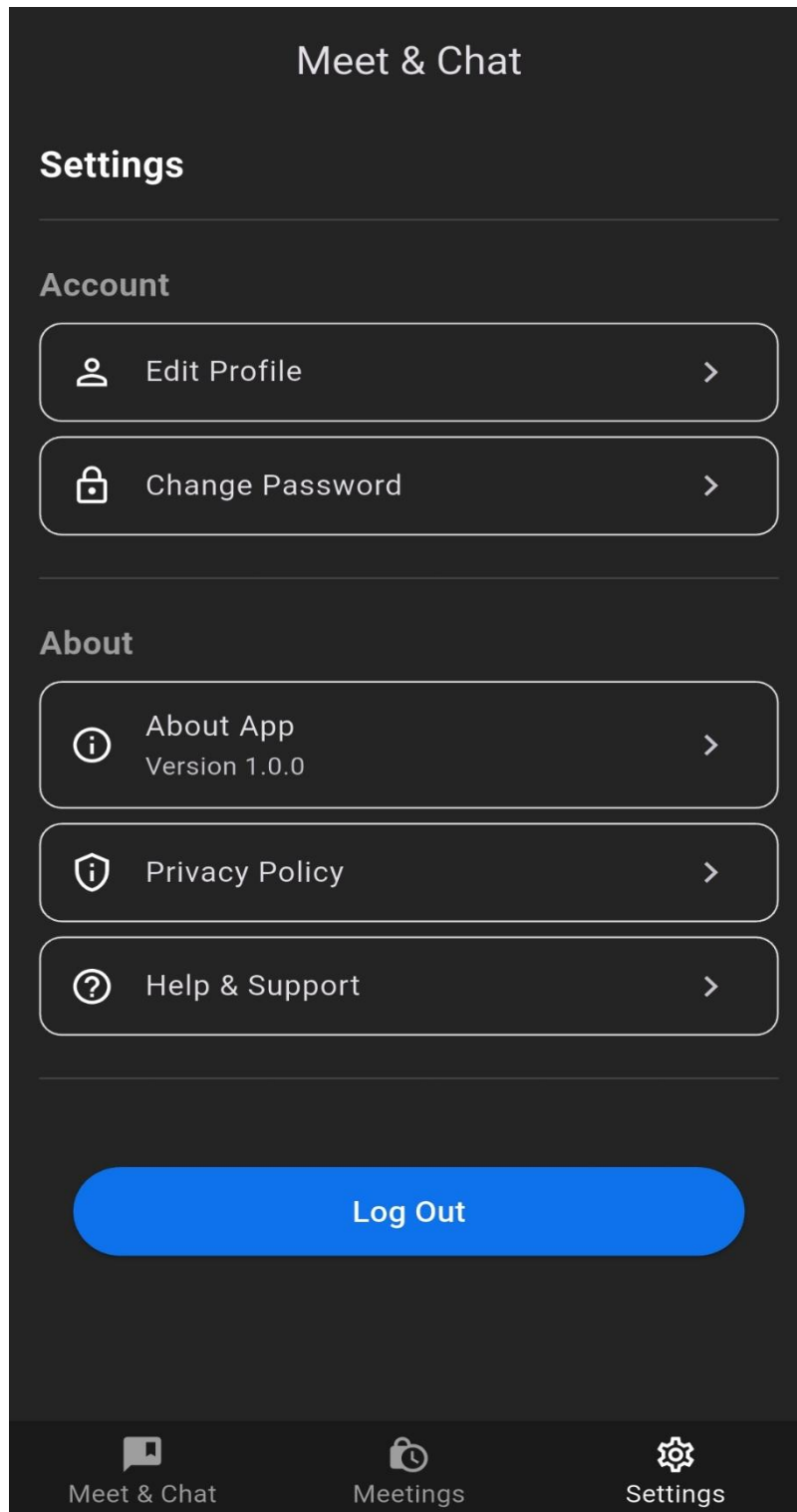


Figure 5.6.5 Settings

This is a settings screen where user gets various options available to perform. Such as edit profile, read privacy policy, help & support, Log Out from the app, etc.

Logout

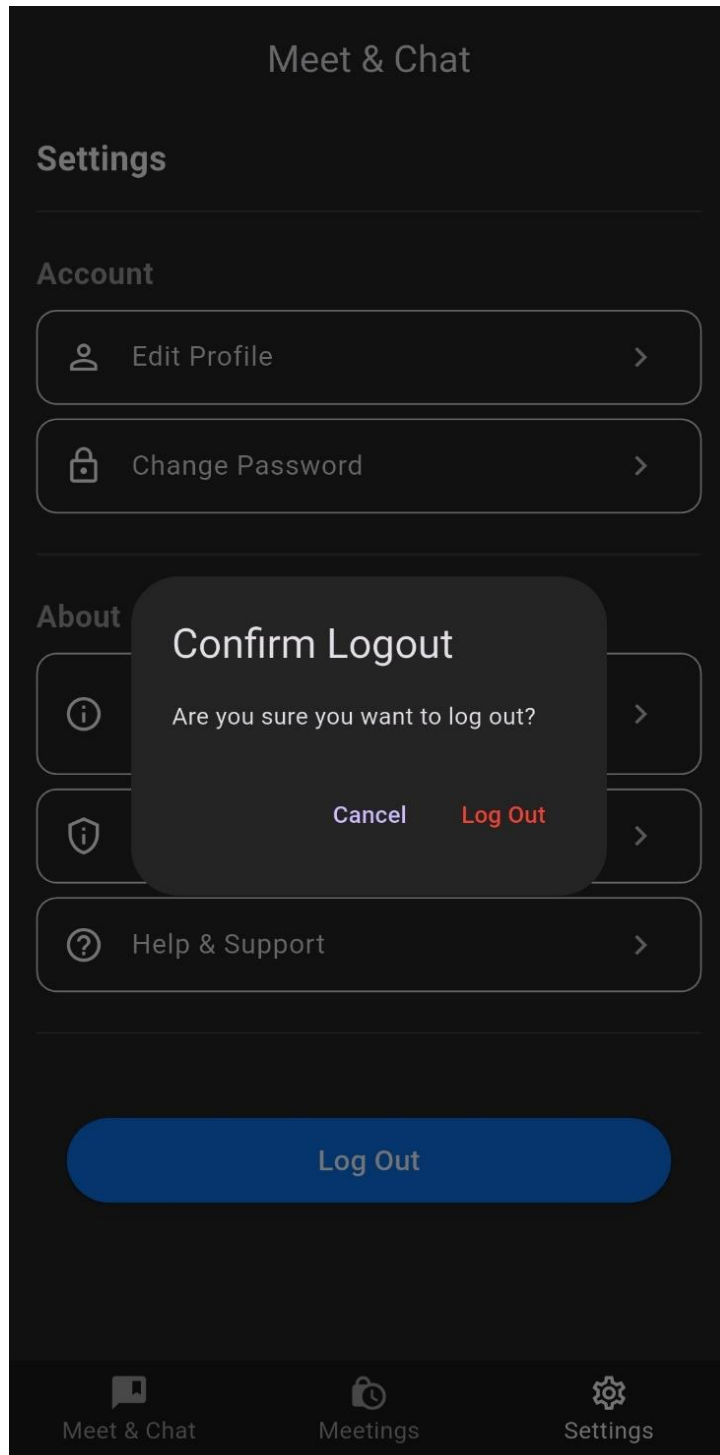


Figure 5.6.6 Logout

This is a logout screen. When user press on logout button it will ask confirmation pop whether user wants to log out from the system or not.

6.SYSTEM TESTING

6.1 Testing and Implementation

System Testing

System testing ensures that all modules of Vidzy work together correctly and meet the functional requirements of the application. The testing process validates chat, video calling, and external meeting features under real usage conditions.

Major Test Scenarios

- Login with Google
- Create meeting / Join meeting
- In-meeting controls (mute, camera, chat)
- Testing of application on different devices.
- App stability on different Android versions (API 26+)

Implementation

Following are implementation steps of how we test different modules.

Implementation Steps

1. Authentication Module: Testing of Google authentication by log in via different account.
2. Meeting Logic: Test on different physical devices on join/creating meeting.
3. In-Meeting Functions: Checking of meeting controls like mute, unmute, chat, camera turn on or off.
4. UI Integration: Designed easy to understand user interface.

6.2 Testing Methodology

The testing methodology ensures that every feature of the application should work same on every device. We have tested our application on different devices.

1.Unit Testing

Unit testing is part of testing in which individual modules are tested. We have test various modules in this testing phase. Such as working of authentication module, meeting creation and joining module, chat module, etc.

2.System Testing

System testing ensures working of core functionality of the application form start to end.

- Login with Google
- Create or join a meeting
- One to one chat functionality
- Perform in-meeting actions.

3. Performance & Reliability Testing

Performance test evaluates performance of the application on different devices and different network conditions.

- Video call under weak network connection.
- Time taken to send and receive the message.
- App performance on different devices.

6.3 Test Cases Design

Test Case Id	Description	Preconditions	Steps	Expected Result	Actual Result	Status (pass/fail)
TC-01	Google Login	App installed, internet available	1.Open App. 2. Tap on log in button.	User logs in and reaches home screen	User logged in.	Pass
TC-02	Create Meeting	User Logged in	1. Tap “Create Meeting”. 2. wait for zego session	Meeting starts with unique meeting ID	User joined in meeting	Pass
TC-03	Join Meeting	Existing meeting available	1. Enter meeting id. 2. Tap join	User successfully join the meeting	User joined meeting	Pass
TC-04	Mute/ Unmute Or Camera On/Off	User inside meeting	1. Tap on Mute or camera icon.	Mic or Camera toggles	Action completed.	Pass
TC-05	Send Chat Message	User inside meeting	1.Open chat. 2. Type message. 3. Send	Message delivered and visible.	Chat delivered.	Pass
`	Leave Meeting	User inside meeting	1. Tap Leave.	User returns to home screen	User left the meeting	Pass

TC-07	Chat fails (internet lost)	Inside meeting	1. Disable internet 2. Send chat	App shows failed message indicator	No such indicator.	Fail
TC-08	Login fails (no internet)	Internet off	1. Tap “Sign in with Google”	App shows network error; login blocked	User will stay in login screen.	Fail
TC-09	Incoming video call notification	Both users logged in, chat exists	1. User initiates video call. 2. Receives Call	Incoming call screen shown is displayed to receiver	Incoming call screen shown	Pass
TC-10	Message Unread count	User must start chat	1. User sends message. 2. Receiver sees unread count	Unread message count will be displayed.	No such indicator.	Fail

Table 6.3 Test Cases

6.4 System Implementation Strategy

The implementation strategy of vidzy is simple. We have focused on providing our core functionality in this app. We have tried to implement chat and video conferencing features. Which made our app unique among other existing players in the market standing right now.

We have implemented Google authentication as a sole login medium which increases the trust and reliability of the application. We have tested this module by login via different account.

We have created separate chat feature to provide chat functionality to user. One can chat with other users. We have also provided two ways to use video conference functionality. First way is user can directly video call the other user from chat screen directly. Second way is by using external meeting functionality. In that one can join or create an external meeting. We have tested the modules on different devices to check functionality and performance.

We have tried to give host controls to host. One can't enter in the meeting without permission of host. For this we have created one waiting screen where participant will stay until he/she allowed to attend a meeting by host. We have also created feature of approving and disapproving the participant while host is in the meeting so that host does not need to navigate to participant list to approve or disapprove user.

Final thing was testing modules in physical devices. For this we have build apk file and then installed in two different physical device to check the performance and core functionality.

7. CONCLUSION

The aim of the Vidzy is to give user chat and video conference functionality in one single app. We have tried to achieve this functionality. We have built this app using Flutter. Which is open source and backed by Google. We have implemented Zego Cloud SDK for video conferencing feature.

We have followed a structured approach throughout the project to ensure smooth development and testing of components such as chat feature, Authentication module, call signaling, meeting creation, in-meeting controls, etc.

Eventually we have achieved our objective of creation of the Vidzy. A simple and easy to use app which gives chat and video conferencing functionality in one single app.

8. LEARNING DURING PROJECT WORK

During the development of Vidzy, one of the learning was to understand integration of video call feature given in this application. We have divided the core components into modules. Like, Authentication module, Chat module, Video call module, Data storage module, Meeting module. It has made easier to develop and improve the functionality of the app.

We have used Zego Cloud SDK for video conferencing feature. It has helped us to gain experience of integrating third party services and SDKs. We have implemented Google authentication feature. It has helped to gain experience integration of services associated with Firebase.

Overall, this project has helped me to build clean and easy user interface, optimize the user experience. This project provided me hands-on experience to real world application development.

9. FUTURE ENHANCEMENTS

Vidzy is developed using Flutter. The main benefit of flutter is that developer has to write code once and it can be run on iOS, Android, Web platforms, etc. But right now we can run this project in android device only. In future we will upgrade it to work with the iOS ecosystem.

Future versions of Vidzy will focus on providing advance features, like security enhancement. Development of more host controls like removing the user, etc. We will try to give feature like cloud recording, push notification for calls and messages, Analytical Dashboards etc.

Overall, These improvements will allow vidzy to evolve as more robust, secure and scalable communication platform.

10. BIBLIOGRAPHY

10.1 Online References

- [1] Flutter, “Flutter Documentation,” Flutter.dev, 2025. [Online]. Available: <https://flutter.dev/docs>. [Accessed:Nov. 30, 2025].
- [2] Google Developers, “Google Sign-In for Android and Web,” Google Authentication Documentation, 2025. [Online]. Available: <https://developers.google.com/identity/sign-in>. [Accessed:Nov. 30, 2025].
- [3] Jitsi, “Jitsi Meet Developer Guide,” Jitsi.org, 2025. [Online]. Available: <https://jitsi.github.io/handbook/docs/dev-guide/dev-guide-android-sdk>. [Accessed:Dec. 3, 2025].
- [4] Firebase, “Cloud Firestore & Realtime Database Documentation,” Firebase Google, 2025. [Online]. Available: <https://firebase.google.com/docs>. [Accessed: Dec. 10, 2025].
- [5] Dart.dev, “Dart Language Documentation,” Dart Programming Guide, 2025. [Online]. Available: <https://dart.dev/guides>. [Accessed: Dec. 25, 2025].
- [6] Material Design, “Material Design Guidelines,” Google Material Design, 2025. [Online]. Available: <https://m3.material.io/>. [Accessed:Jan. 23, 2026].

10.2 Offline References

- [1] J. Smith, *Flutter App Development for Beginners*. Tech Publications, 2022.
- [2] A. Kumar and R. Patel, *Mobile Application Development Using Flutter*. SunRise Publishers, 2023.
- [3] M. Johnson, *Real-Time Communication Systems: Concepts and Applications*. Global Tech Press, 2021.
- [4] S. Williams, *UI/UX Design Principles for Modern Mobile Apps*. Creative Minds Publications, 2020.
- [5] R. Gupta, *Software Engineering: Principles and Practices*. National Academic Press, 2022.
- [6] P. Brown, *Cloud & Backend Integration for Mobile Applications*. TechWorld Publishing, 2021.