A
CAPSTONE (Major) Project Report
on

GameO'Chat

at

Workpulse Software Pvt Ltd

Submitted by
Roushil Singla
Registration Number – 18MCA10028

Guided by

Mr Ashish Shah
iOS Team Lead
(External Guide)

Dr M Ashwin
Assistant Professor,Senior
(Internal Guide)

Submitted in partial fulfillment of the requirement for
the degree of
"Master of Computer Applications"

Submitted to
School of Computing Science and Engineering
VIT Bhopal University
Bhopal (MP) – 466 114

April 2019

VIT BHOPAL UNIVERSITY, M P - 466114

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING

CANDIDATE'S DECLARATION

I hereby declare that the Dissertation entitled "Game'OChat" is my own work conducted under the supervision of Dr M Ashwin, Senior Assistant Professor, SCSE at VIT University, Bhopal.

I further declare that to the best of my knowledge this report does not contain any part of work that has been submitted for the award of any degree either in this university or in other university / Deemed University without proper citation.

Roushil Singla
18MCA10028

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Date: 06-04-2020

Dr M Ashwin
Senior Associate Professor

VIT UNIVERSITY BHOPAL, M P – 466114

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the work embodied in this Capstone Project Report entitled **Game'OChat** has been satisfactorily completed by **Mr Roushil Singla, 18MCA10028** in the School Computing Science and Engineering of MCA at VIT University, Bhopal. This work is a bonafide piece of work, carried out under my/our guidance in the School of Computer Science and Engineering for the partial fulfillment of the degree of Bachelor of Technology.

Dr M Ashwin
Senior Associate Professor

Forwarded by

Dr Kanchan Lata k
Program Chair

Approved by

Dr Manas Kumar Mishra
Professor & Dean

COMPLETION CERTIFICATE

Date: 13th January 2020

To Whomsoever it may concern

This is to certify that **Mr. Roushil Singla**, S/O Mr. Rajesh Singla, a student of MCA, 6th Sem from Vellore Institute of Technology, Bhopal, VIT University has successfully completed Six Months (from **24th September 2019 to 24th March 2020**) **iOS internship** program at **Workpulse Software Pvt.Ltd.**

During the period of his internship program with us he was found punctual, hardworking and inquisitive. We extend our best wishes for his bright future and wish him success in his endeavors.

For Workpulse Software Pvt. Ltd

WORKPULSE SOFTWARE PVT. LTD. INDORE

Ankita Panchal
Sr. HR Executive

ACKNOWLEDGMENT

The success and final outcome of this project required a lot of guidance and assistance from many people and I am extremely privileged to have got this all along the completion of my project. However, it would not have been possible without the kind support and help of many individuals and organizations. I would like to extend my sincere thanks to all of them.

I respect and thank Mr. Ashish Shah and Mr Mustafa Saify (iOS Consultant) for providing me an opportunity to do the project work in Workpulse Software Pvt Ltd and giving us all support and guidance which made me complete the project duly. I am extremely thankful to him for providing such a nice support and guidance, although he had busy schedule managing the corporate affairs.

I would like to express my gratitude towards my parents & member of VIT Bhopal for their kind co-operation and encouragement which help me in completion of this project.
I would like to express my special gratitude and thanks to industry persons for giving me such attention and time. I am thankful too and fortunate enough to get constant encouragement, support and guidance from all Teaching staffs of SCSE which helped us in successfully completing our project work.

EXECUTIVE SUMMARY

The project is on the topic Game'OChat which is a cloud-based mobile and desktop messaging app with a focus on security and speed. Users can send messages and exchange photos and stickers of any type. Users can add multiple devices to their account and receive messages on each one. It provides authentication to users for secure login and registration and enables user to check out the profiles of each user.

LIST OF FIGURES

Table of Contents

INTRODUCTION

Communication is a mean for people to exchange messages. It has started since the beginning of human creation. Distant communication began as early as 1800 century with the introduction of television, telegraph and then telephony. Interestingly enough, telephone communication stands out as the fastest growing technology, from fixed line to mobile wireless, from voice call to data transfer. The emergence of computer network and telecommunication technologies bears the same objective that is to allow people to communicate. All this while, much efforts has been drawn towards consolidating the device into one and therefore indiscriminate the services. Chatting is a method of using technology to bring people and ideas together despite of the geographical barriers. The technology has been available for years but the acceptance it was quit recent. Our project is an example of a chat server. It is made up of applications the client application which runs on the users mobile and server application which runs on any iPad and iPhone. To start chatting our client should get connected to server where they can do private chatting.

1.1 Problem Statement

This project is a chat application with a server and users to enable the users to chat with each others.

To develop an instant messaging solution to enable users to seamlessly communicate with each other. The project should be very easy to use enabling even a novice person to use it.

1.2 Project Scheduling

This document provides a scalable scheduling tool and associated schedule development, analysis, and monitoring methods that can be used by Implementing Agencies (IA) to prepare, monitor, and report project schedules. Our Project is not that complex so we will not use very complex scheduling method.

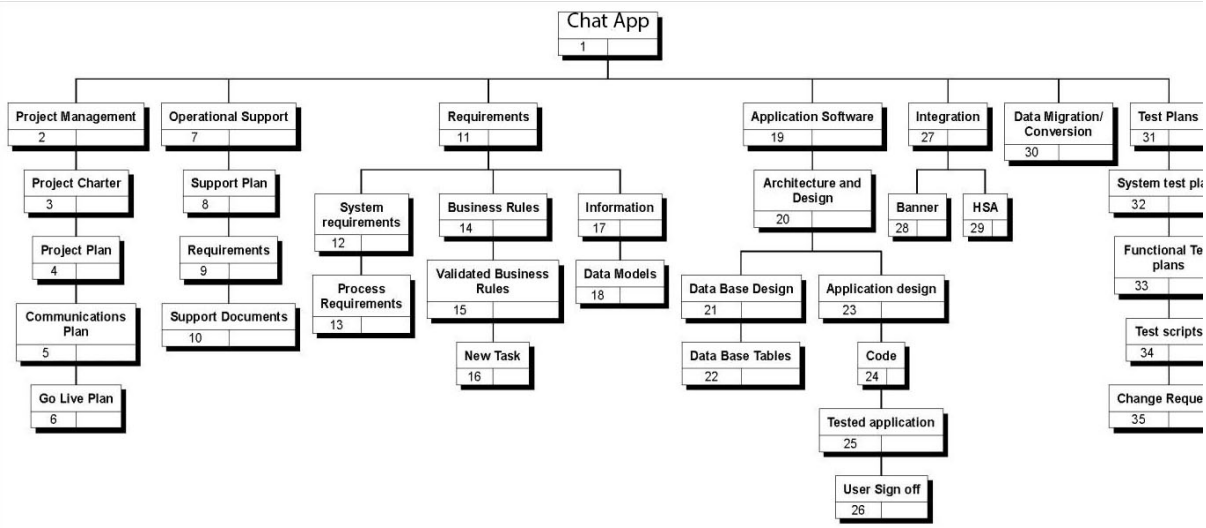

Fig. 1

LANGUAGE ADAPTATION

- **SWIFT 4.2, 5.0 -** Swift is a new programming language for iOS, macOS, watchOS, and tvOS app development. Nonetheless, many parts of Swift will be familiar from your experience of developing in C and Objective-C. Swift is a type-safe language, which means the language helps you to be clear about the types of values your code can work with different versions.

- **OBJECTIVE-C -** Objective-C is the primary programming language you use when writing software for OS X and iOS. It's a superset of the C programming language and provides object-oriented capabilities and a dynamic runtime. Objective-C inherits the syntax, primitive types, and flow control statements of C and adds syntax for defining classes and methods. It also adds language-level support for object graph management and object literals while providing dynamic typing and binding, deferring many responsibilities until runtime.

WORKPULSE PROFILE

Workpulse is your trusted operations management consultant offering a suite of innovative mobility solutions for highly efficient operations management and employee engagement. We believe in solving problems associated with traditional operations management with technology solutions that make it easier for business owners to have greater control over their business operations, even while managing multiple locations.

Our solutions have the potential to help businesses across different domains overcome their operational challenges and create a highly productive work environment to meet the demands of today and tomorrow.

About **Workpulse**

Workpulse is your trusted operations management consultant offering a suite of innovative mobility solutions for highly efficient operations management and employee engagement.

We believe in solving problems associated with traditional operations management with technology solutions that make it easier for business owners to have greater control over their business operations, even while managing multiple locations.

We have helped 2500+ restaurant and QSR establishments overcome their operational challenges by providing targeted mobile applications for managing specific processes from guest satisfaction to brand and food safety compliance.

Our solutions have the potential to help businesses across different domains overcome their operational challenges and create a highly productive work

About 1

Workpulse– Operations Management Simplified

Workpulse offers a suite of operations management software for simplifying day to day operations and for ensuring unmatched employee and guest experience.

All Workpulse mobile apps are integrated with the cloud for real time information processing and data management. We have also ensured that our mobile apps work seamlessly with any WiFi network.

With Workpulse, you can get complete peace of mind with assured improvement in the efficiency of your business operations. Our solutions are currently being used at over 2500 locations by 125+ businesses including QSR (Quick Service Restaurants) and Restaurant Franchises.

SYSTEM SPECIFICATIONS

Software Requirements-

- Xcode 10.1+
- iOS 12.1 or Above
- Swift 4.2 +

Hardware Requirements-

- MacOS Version 10.12.1+
- iPhone and iPad
- iOS Simulator Version 11.2 (SimulatorApp-912.4 SimulatorKit-570.3 CoreSimulator-681.15)

REQUIREMENT ANALYSIS

Data gathering is the step in which collect data about the system to be developed. We use different **tools and methods** depending on situations.

Written document may be reports, forms, business plans, memos, policy statement, organizational chart and many others. It provides valuable information about the existing system.

The Requirements are mentioned below:-

REQUIREMENTS:-

1) Responsive Templates from iOS Giphy

2) User can access data only if User is logged in.

3) All data stored in Database.

4) User Friendly Environment.

5) Show Profile.

6) When User give feedback then mail come to me.

7) Problem identification and project initiation

8) Background analysis

9) Inference or Findings

1. SOFTWARE REQUIREMENT SPECIFICATION

1.1 PURPOSE

However , the purpose of this project is to develop a iOS chat application. The objective of this process is as follows;

1. To develop an instant messaging solution to enable users to seamlessly communicate with each other

2. The project should be very easy to use enabling even a novice person to use it

1.2 PROJECT SCOPE AND FEATURES

1. Broadcasting Chat Server Application is going to be a text communication software, it will be able to communicate between two computers using point to point communication

2. The limitation of Live Chat is it does not support audio conversations. To overcome this limitation we are concurrently working on developing better technologies

3. Companies would like to have a communication software wherein they can communicate instantly within their organization

4. The fact that the software uses an internal network setup within the organization makes it very secure from outside attacks

2. FUNCTIONAL AND NON FUNCTIONAL REQUIREMENTS

2.1 FUNCTIONAL REQUIREMENTS

1. User Registration - User must be able to register for the application through a valid phone number. On installing the application, user must be prompted to reg- ister their phone number. If user skips this step, application should close. The users phone number will be the unique identifier of his/her account on Chat Application.

2. Adding New Contacts - New registered user will be automatically added in the contacts and makes it easier to connect them from add user option. If any of the contacts have not yet registered on Chat Application, user should be provided with an invite option that sends those contacts a regular text message asking them to join Chat Application.

3. Send Message - User should be able to send instant message to any contact on his/her Chat Application contact list. User should be notified when message is successfully delivered to the recipient by displaying a tick sign next to the message sent.

2.2 NON-FUNCTIONAL REQUIREMENTS

1. Privacy Messages shared between users should be encrypted to maintain privacy.

2. Robustness - In case users device crashes, a backup of their chat history must be stored on remote database servers to enable recoverability.

3. Performance - Application must be lightweight and must send messages instantly.

SDLC(SYSTEM DEVELOPMENT LIFE CYCLE)

System development life cycle. SDLC is a system development life cycle of software development life cycle. It include guideline policies and procedures for developing system. through their life cycle it include requirement design implementation, testing, deployment operation and maintenance.

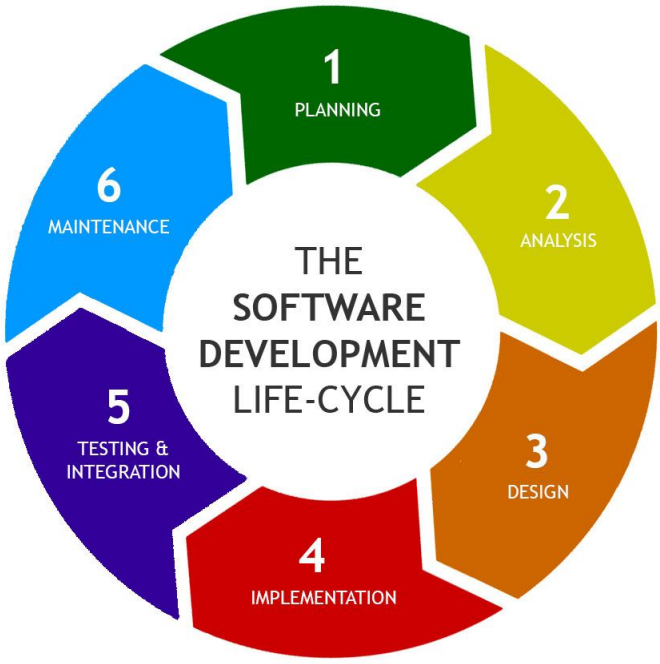

Fig. 2

Various phases were performed by Organization:

1. **Planning -** Done by Management Team from Gibbsboro which included financial studies by owner and business executives
2. **Analysis -** Done by Product Manager(s) and higher team measuring various outcomes as well as study of feasibility
3. **Design -** Prepared by iOS Consultant and Web team
4. **Implementation -** By iOS team which include coding and architectures for code flow including Database structure
5. **Testing -** By testers which includes Unit Testing as well by iOS developers
6. **Maintenance -** By iOS Team

AGILE METHODOLOGY

This project followed the Agile software development methodology which is more than frameworks such as Scrum, Extreme Programming or Feature-Driven Development (FDD). Agile software development comprises various approaches to software development under which requirements and solutions evolve through the collaborative effort of self-organizing and cross-functional teams and their customers and users. It advocates adaptive planning, evolutionary development, early delivery, and continual improvement, and it encourages rapid and flexible response to change.

Agile software development is more than practices such as pair programming, test-driven development, stand-ups, planning sessions and sprints.

It includes:

- Track of regular Updates
- Daily Scrum for Task
- Sessions twice a month
- Git Approach to track performance
- Zoom Extensions for Presentation

FEASIBILITY STUDY

Feasibility study is used to assess the strengths and weakness of a proposed project and present directions of activities which will improve a project and achieved desired result.It involves an examination of operation, HR and marketing aspects of a business on ex ante basis.

Feasibility study is designed to provide an overview of the primary issue related to the business idea.

It involves:

Appraisal of existing system and manual process- Troubleshooting, process reengineering, risk analysis and assessment, risk management, cost benefit analysis, impact analysis, integration of existing of new system, resource requirement planning and timing.

There are many different types of feasibility studies; here is a list of some of the most common:

- **Technical Feasibility** – Does the company have the technological resources to undertake the project? Are the processes and procedures conducive to project success?

- **Economic Feasibility** – Given the financial resources of the company, is the project something that can be completed? The economic feasibility study is more commonly called the cost/benefit analysis.

- **Operational Feasibility** – This measures how well your company will be able to solve problems and take advantage of opportunities that are presented during the course of the project

SYSTEM DESIGN METHODOLOGIES

Systems design is the process of defining the architecture, modules , interfaces, and data for a system to satisfy specified requirements. Systems design could be seen as the application of systems theory to product development. There is some overlap with the disciplines of system analysis, system architecture and systems engineering.

- **Architectural design :-** The architectural design of a system emphasizes the design of the system architecture that describes the structure, behavior and more views of that system and analysis.
- **Logical design :-** The logical design of a system pertains to an abstract representation of the data flows, inputs and output of the system. This is often conducted via modeling, using an over-abstract (and sometimes graphical) model of the actual system. In the context of systems, designs are included. Logical design includes entity-relationship diagrams (ER diagrams).
- **Physical design :-**

The physical design relates to the actual input and output processes of the system. This is explained in terms of how data is input into a system, how it is verified/authenticated, how it is processed, and how it is displayed. In physical design, the following requirements about the system are decided.

1. Input requirement
2. Output requirements
3. Storage requirements
4. Processing requirements
5. System control and backup or recovery

Put another way, the physical portion of system design can generally be broken down into three sub-tasks:

1. User Interface Design
2. Data Design
3. Process Design

SYSTEM ANALYSIS

- **Identification of Objects :-**

1. Registration & Login Page
2. Users
3. Chat Partners
4. Profile
5. Text Messages

- **Identification of Entities :-**

1. Users - Current logged User
2. Chat Partners - Chatting Partner through Id
3. Messages - Text Messages List
4. Media Type - Gallery

- **Attributes for the following Entities :-**

1. Users :- userName, userEmail, profileImageURL
2. Chat Partner :- userName, userEmail, profileImageURL
3. Messages :- fromID, toID, timeStamp, text
4. Media Type :- profileImageURL, emoji

- **Design :-**

1. Use Case Diagram (UML)
2. ER diagram
3. DFD

UML ARCHITECTURE

UML is a standard language for specifying, visualizing, constructing, and documenting the artifacts of software systems.A picture is worth a thousand words, this idiom absolutely fits describing UML. Object-oriented concepts were introduced much earlier than UML. At that point of time, there were no standard methodologies to organize and consolidate the object-oriented development. It was then that UML came into picture.

UML diagrams are not only made for developers but also for business users, common people, and anybody interested to understand the system. The system can be a software or non-software system. Thus it must be clear that UML is not a development method rather it accompanies with processes to make it a successful system.

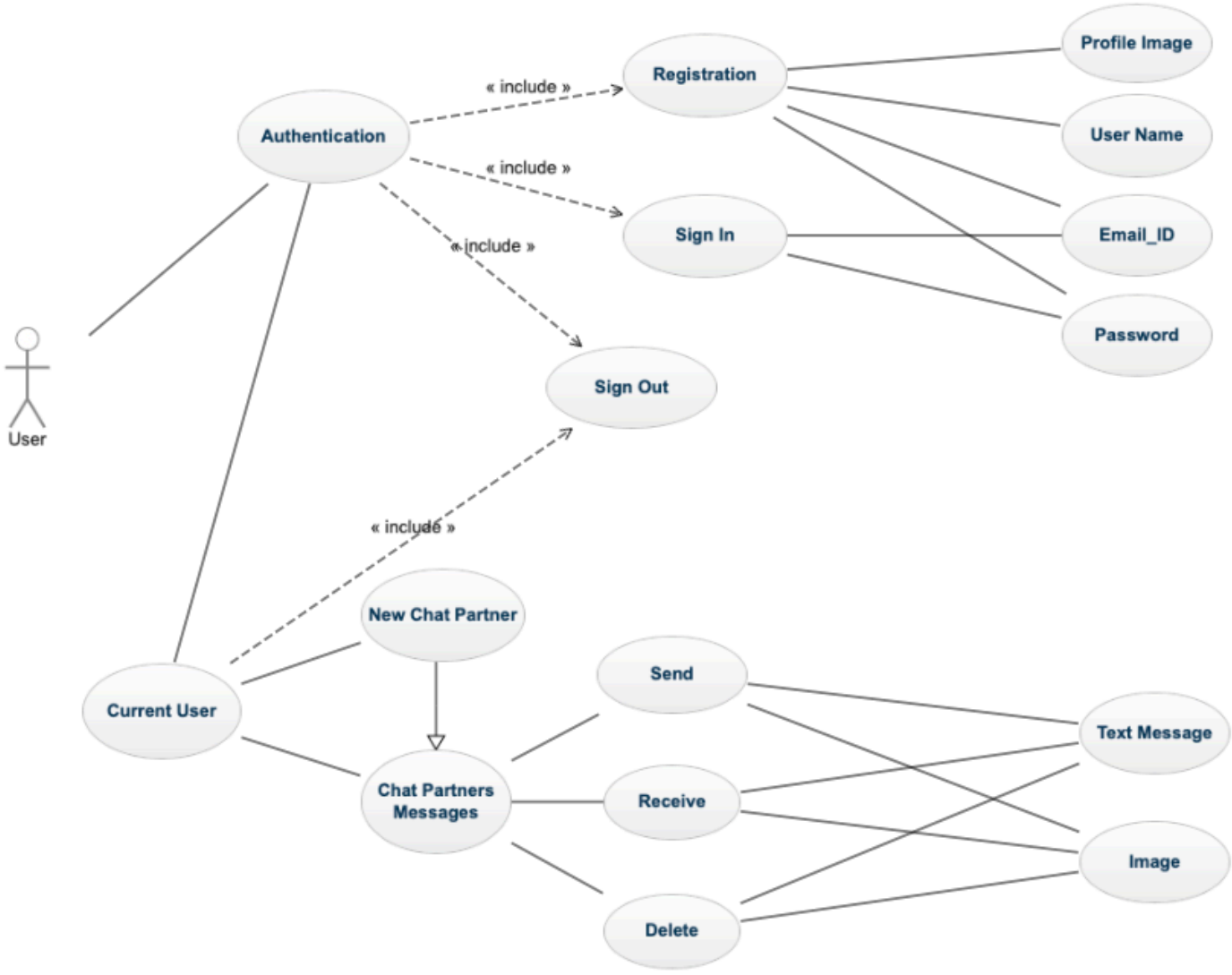

Fig. 3

ER - DIAGRAM

An entity relationship diagram (ERD) shows the relationships of entity sets stored in a database. An entity in this context is an object, a component of data. An entity set is a collection of similar entities. These entities can have attributes that define its properties. By defining the entities, their attributes, and showing the relationships between them, an ER diagram illustrates the logical structure of databases.

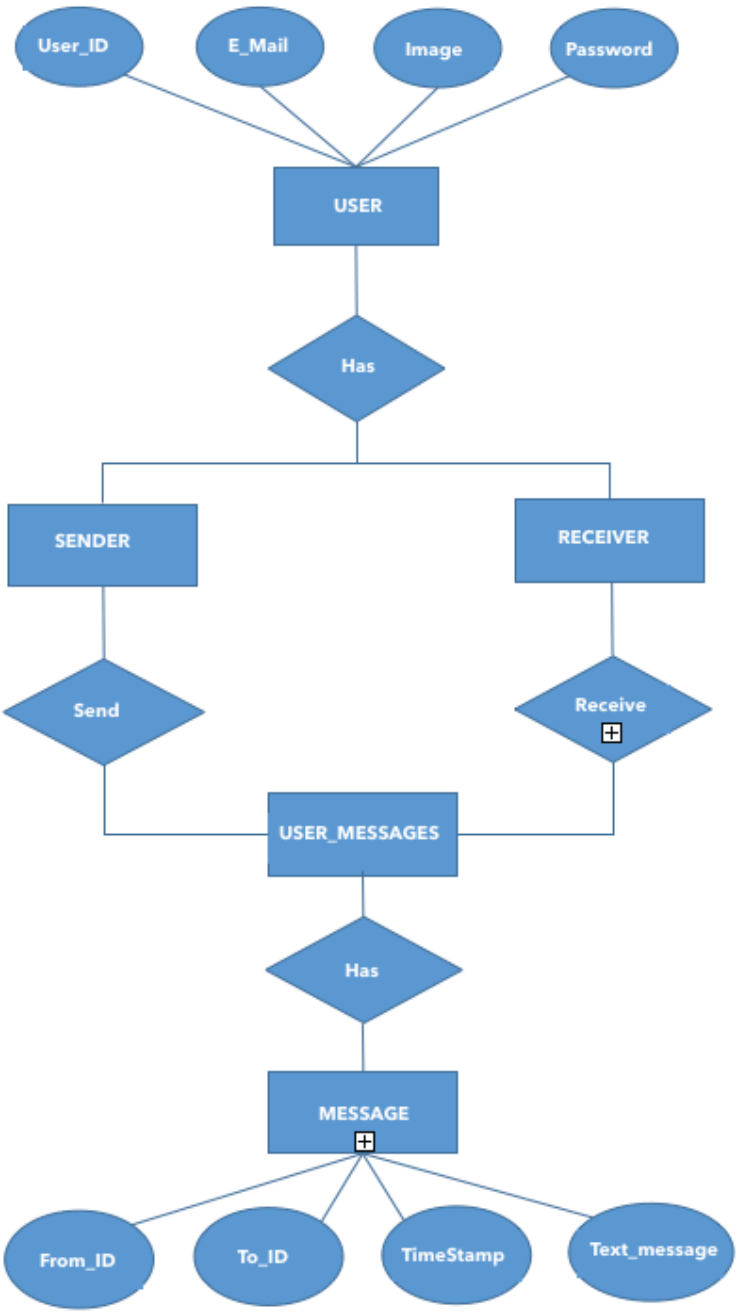

Fig. 4

DATA FLOW DIAGRAM

A data flow diagram shows the way information flows through a process or system. It includes data inputs and outputs, data stores, and the various subprocesses the data moves through. DFDs are built using standardized symbols and notation to describe various entities and their relationships.

Data flow diagrams are also categorized by level. Starting with the most basic, level 0, DFDs get increasingly complex as the level increases.

As we are building our own data flow diagram, we decided which level our diagram will be suitable for organization.

Using Level 2 DFD :-

It simply break processes down into more detailed subprocesses. In theory, DFDs could go beyond level 3, but they rarely do. Level 3 data flow diagrams are detailed enough that it doesn't usually make sense to break them down further

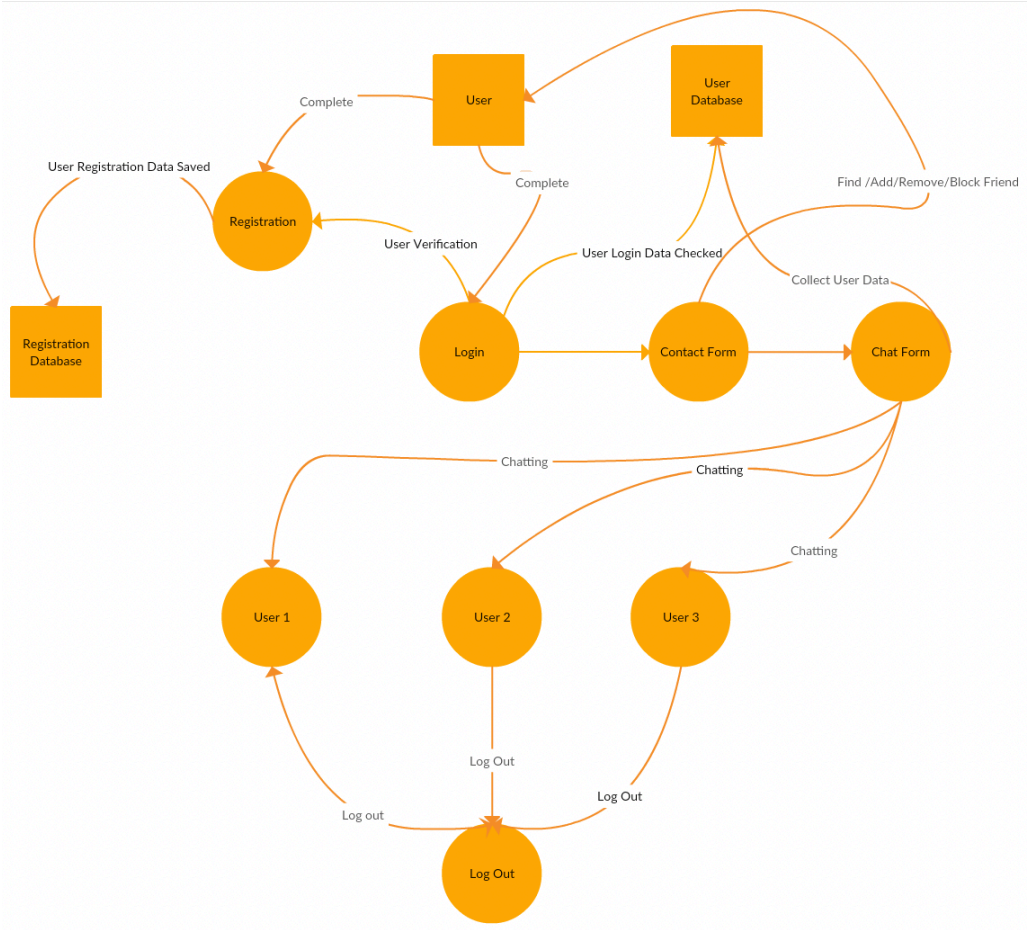

Fig. 5

WORKING LAYOUT

1. USER INTERFACE : -

1.1. REGISTERING USER :-

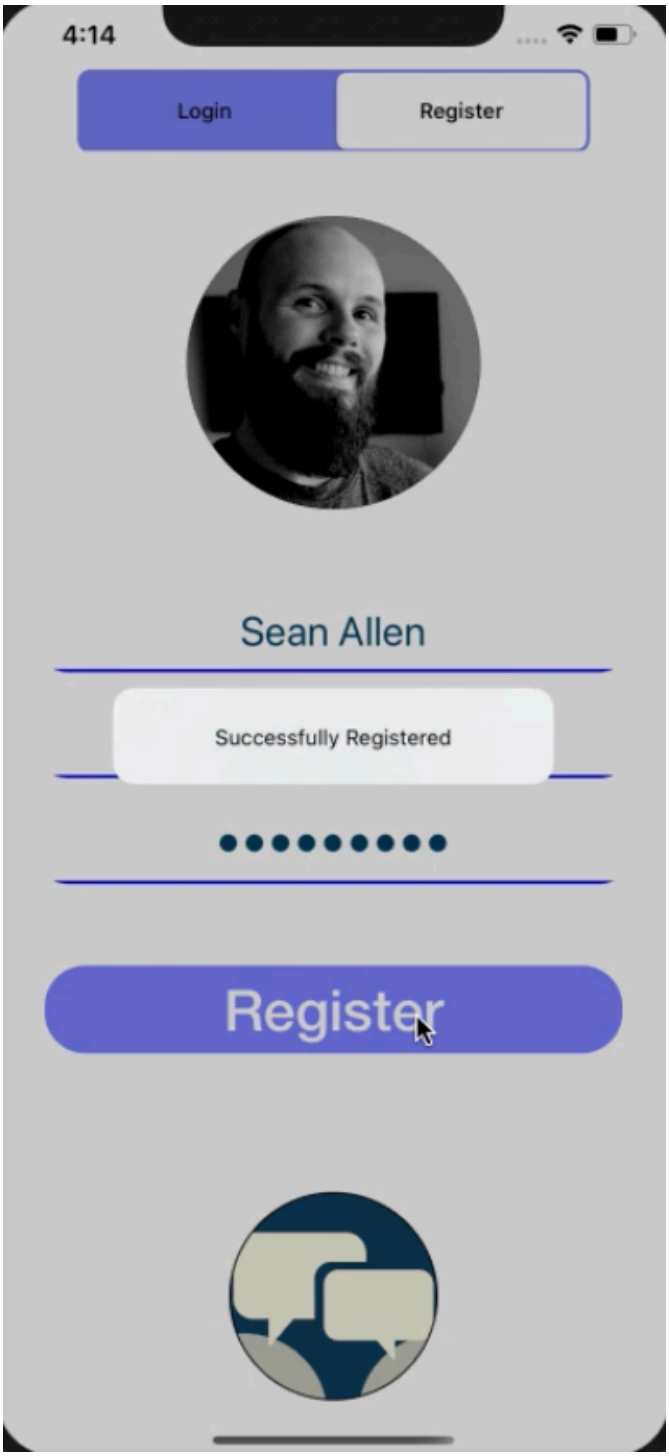

Fig. 6.1

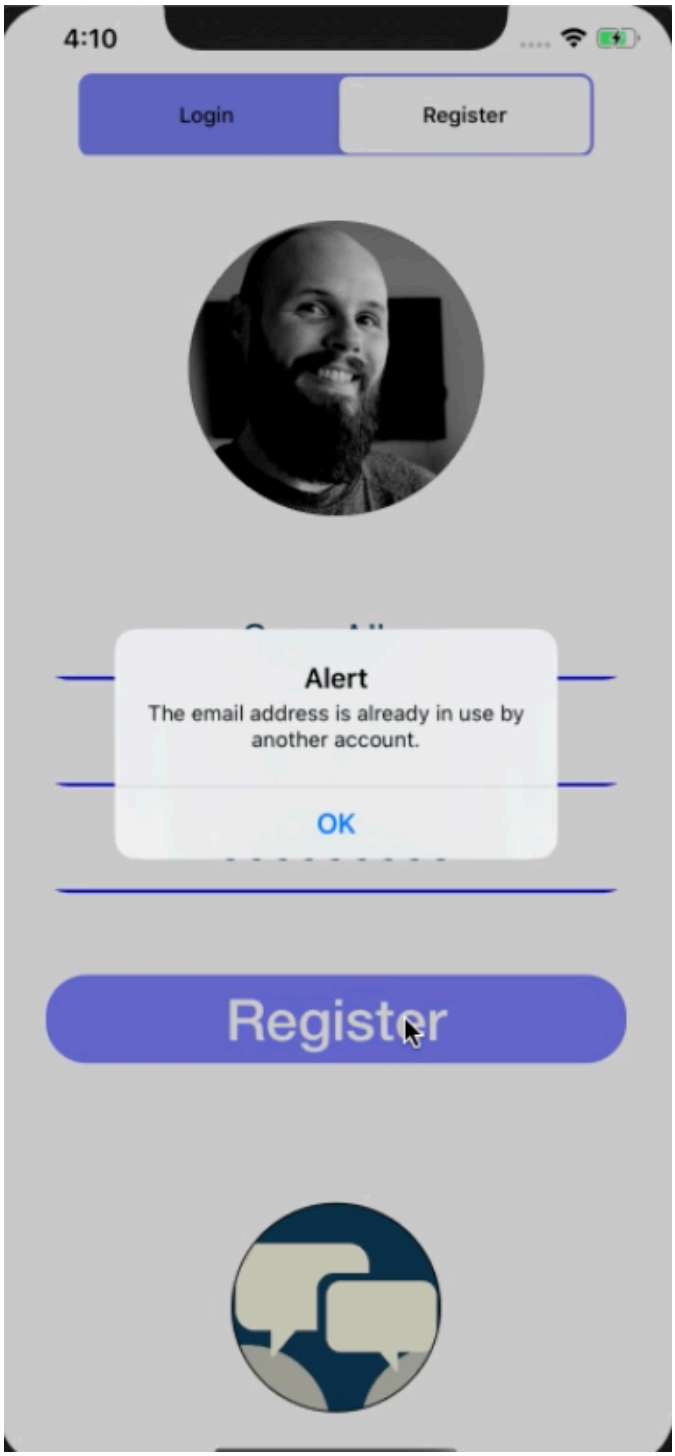

Fig. 6.2

1.2. LOGGING USER :-

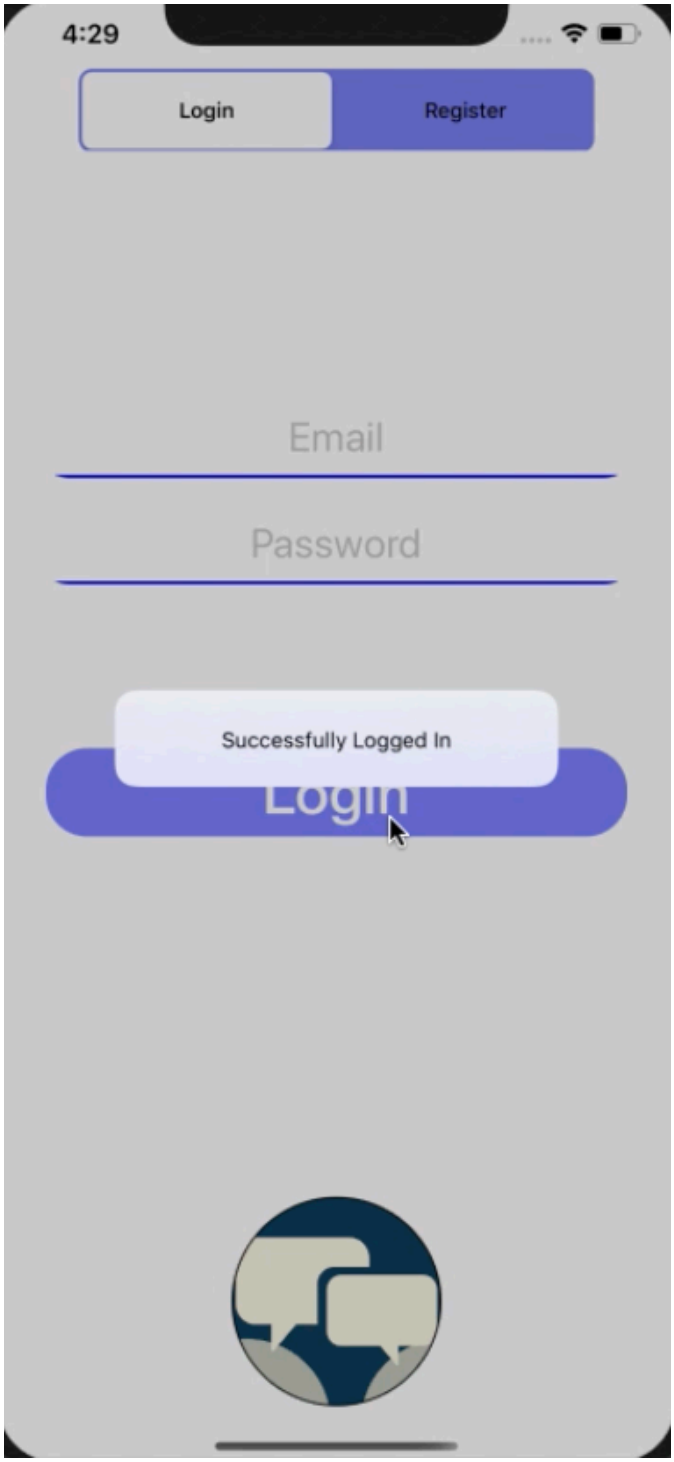

Fig. 7.1

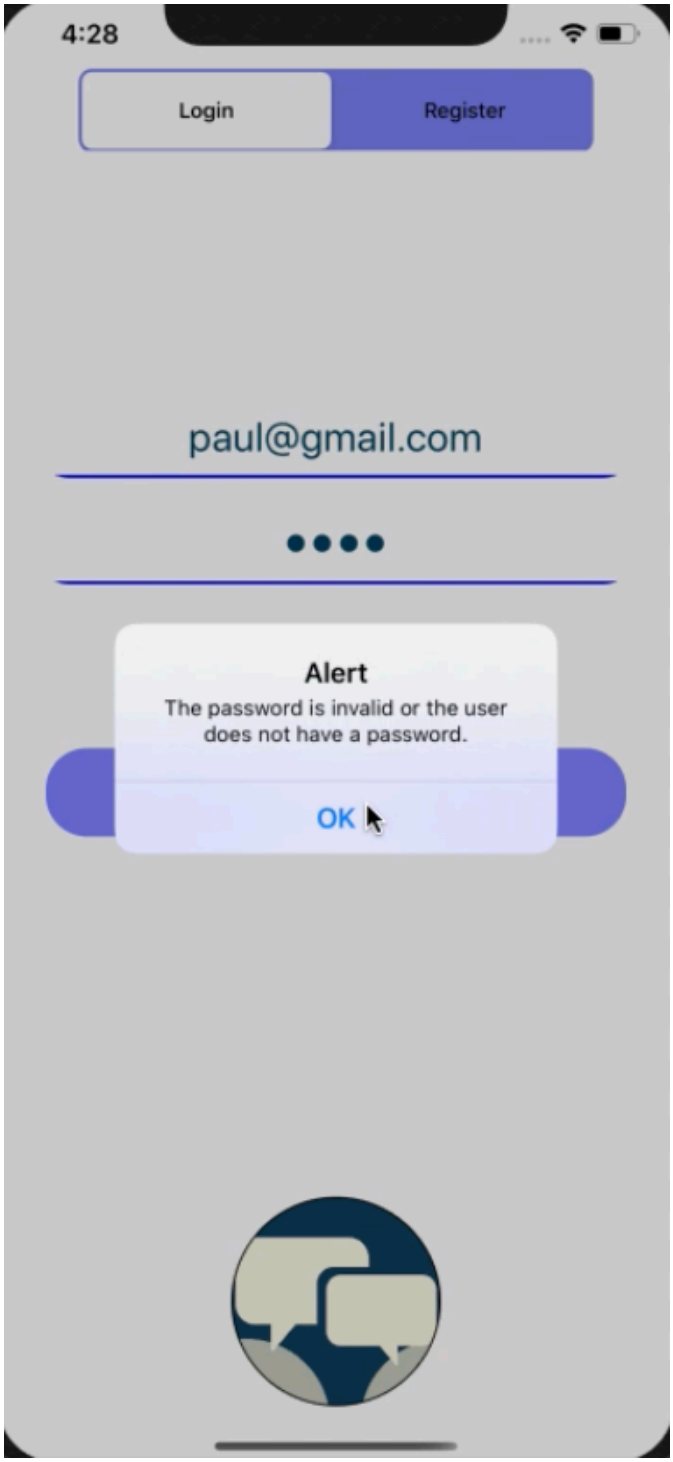

Fig. 7.2

1.3 CHAT LIST :-

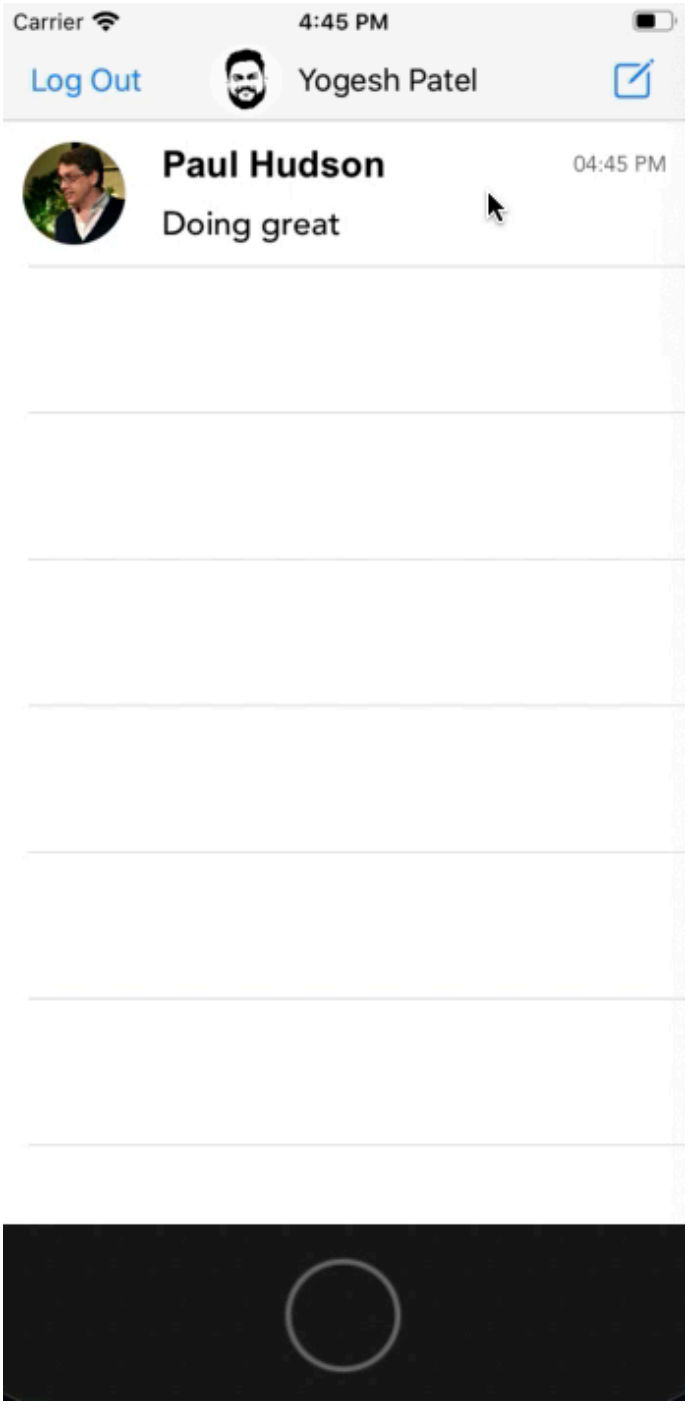

Fig. 8.1

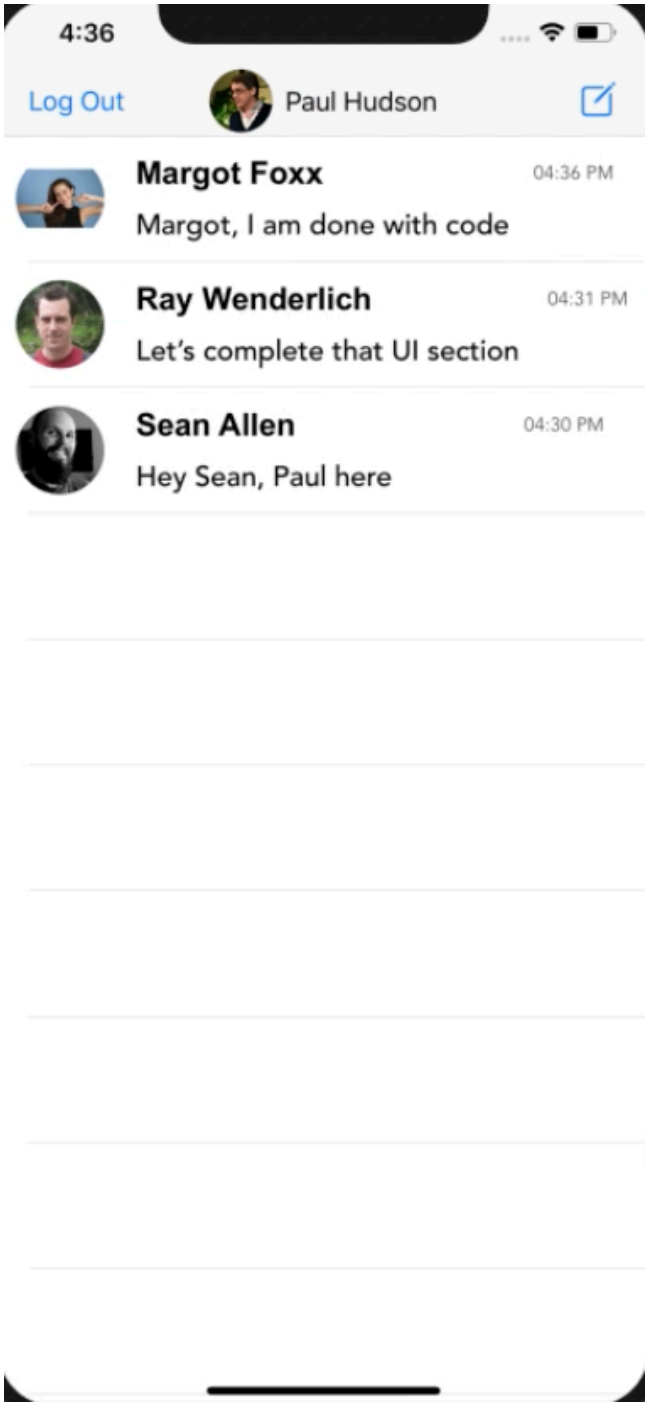

Fig. 8.2

1.4. USER MESSAGES :-

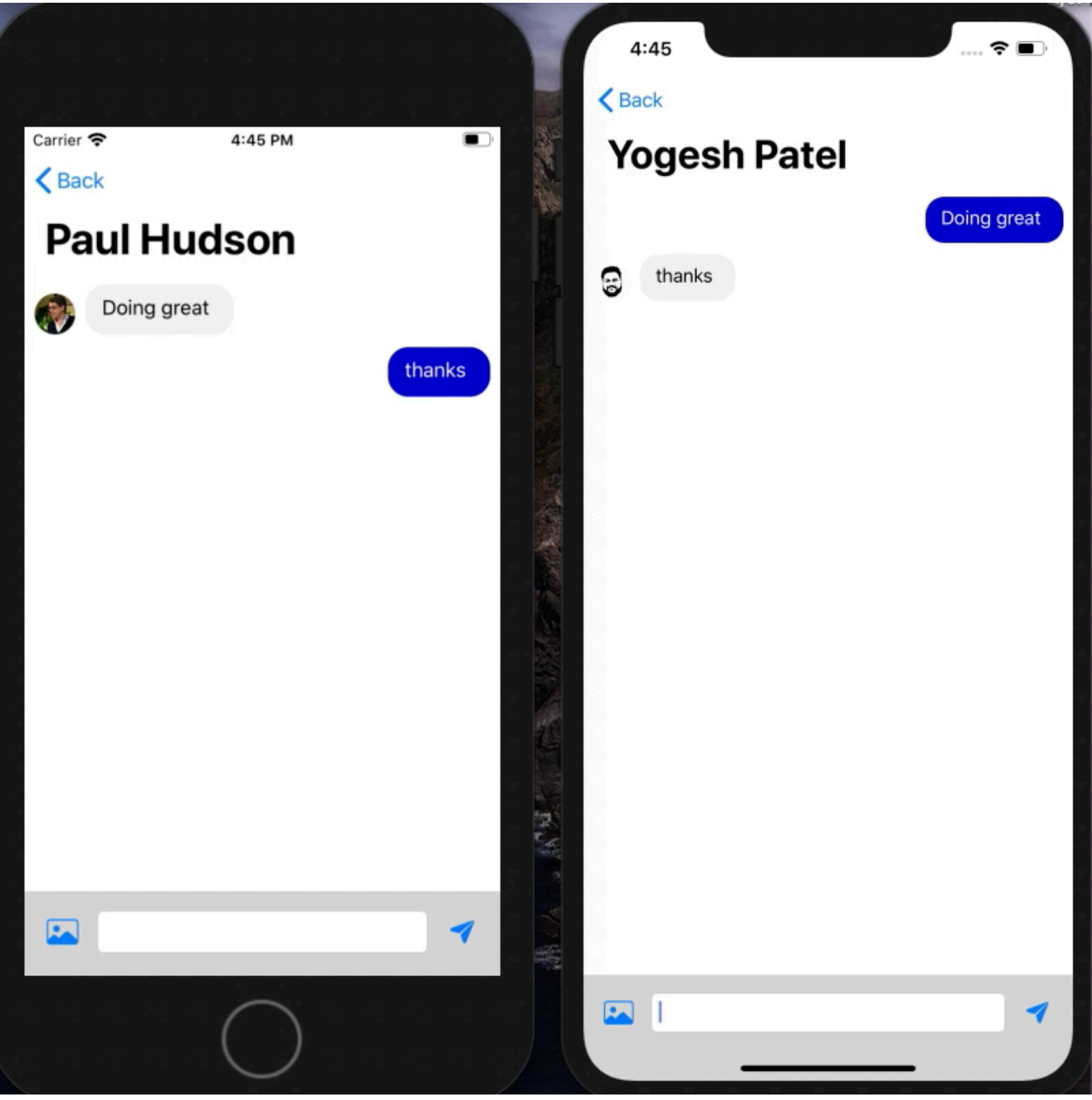

Fig. 9

1.5. SENDING MULTIMEDIA :-

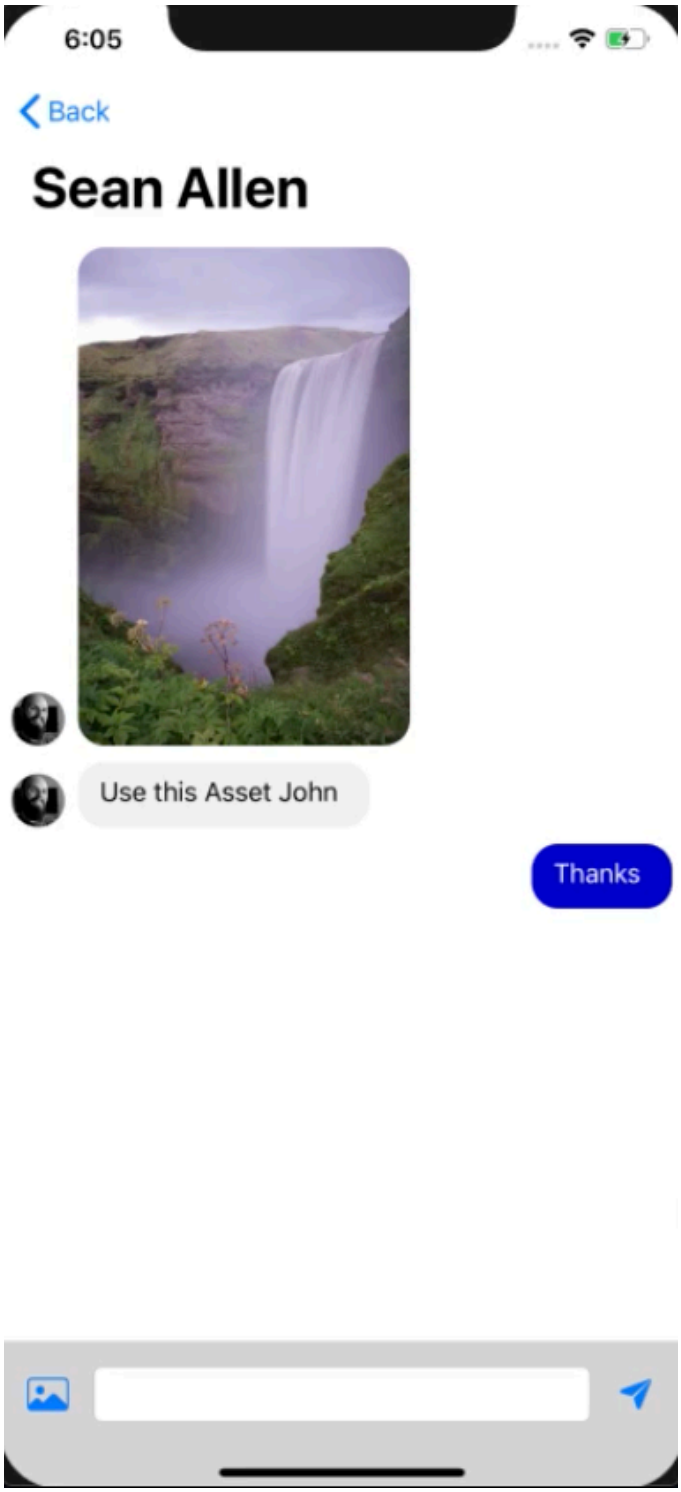

Fig. 10.1

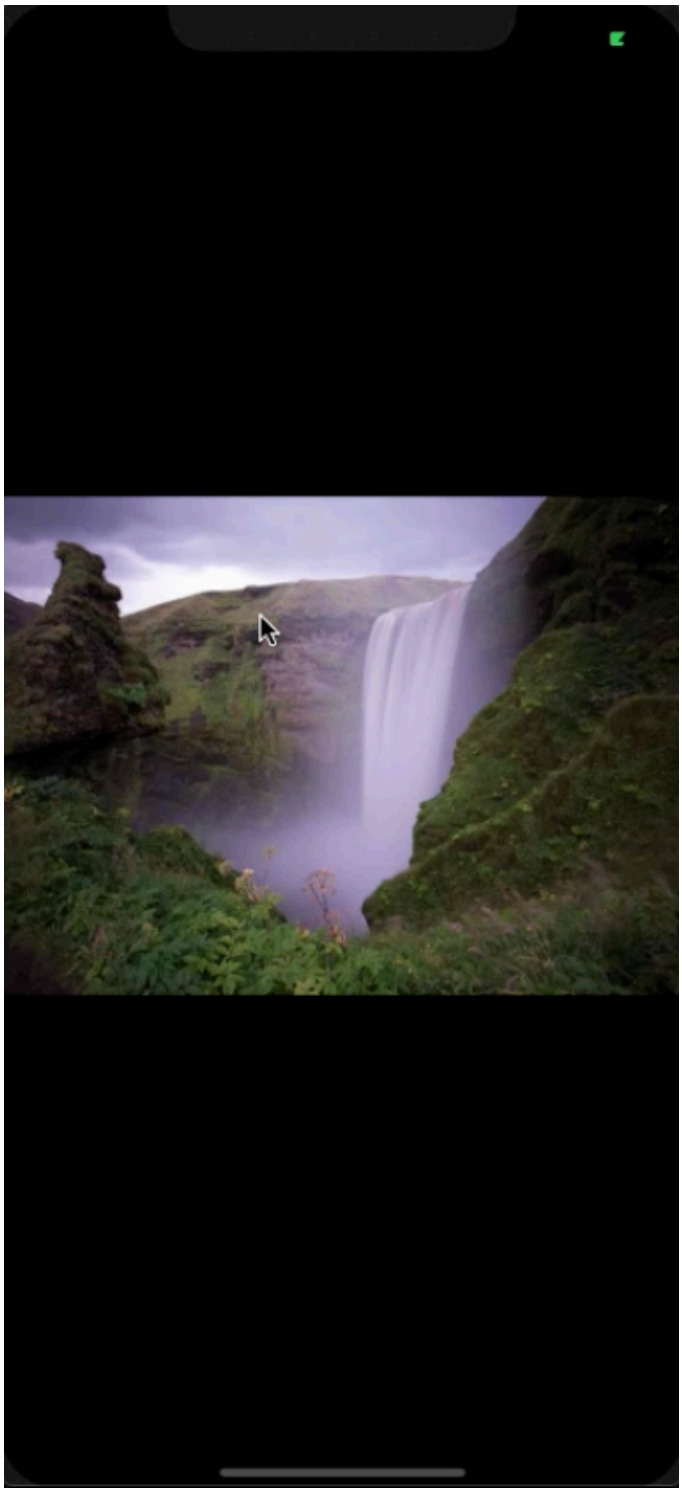

Fig. 10.2

2. DATABASE FLOW : -

2.1. AUTHENTICATING USER :-

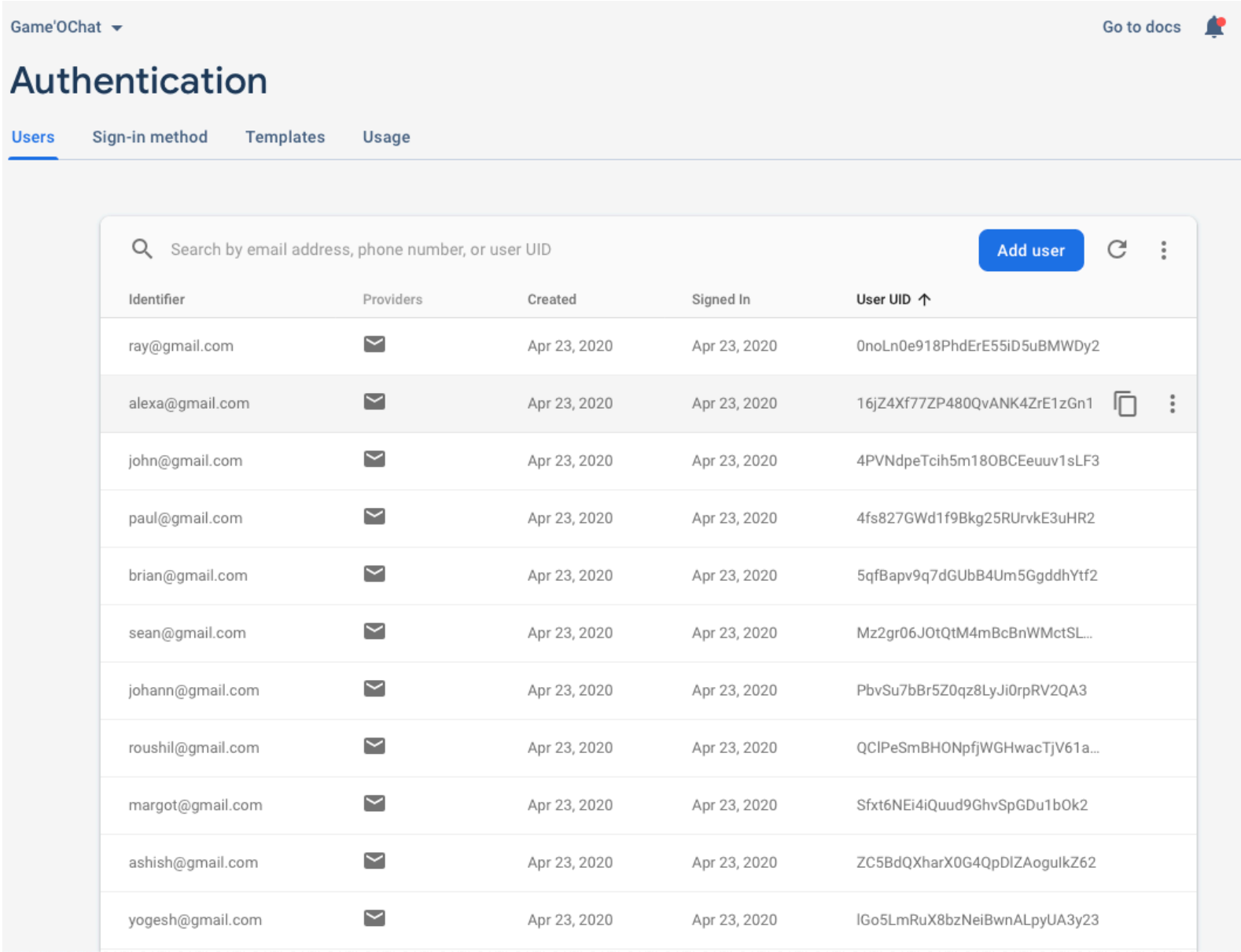

Fig. 11

2.2. STRUCTURING USER PROFILE AND MESSAGES :-

2.2.1. Users :-

- Email
- Name
- ProfileImageURL

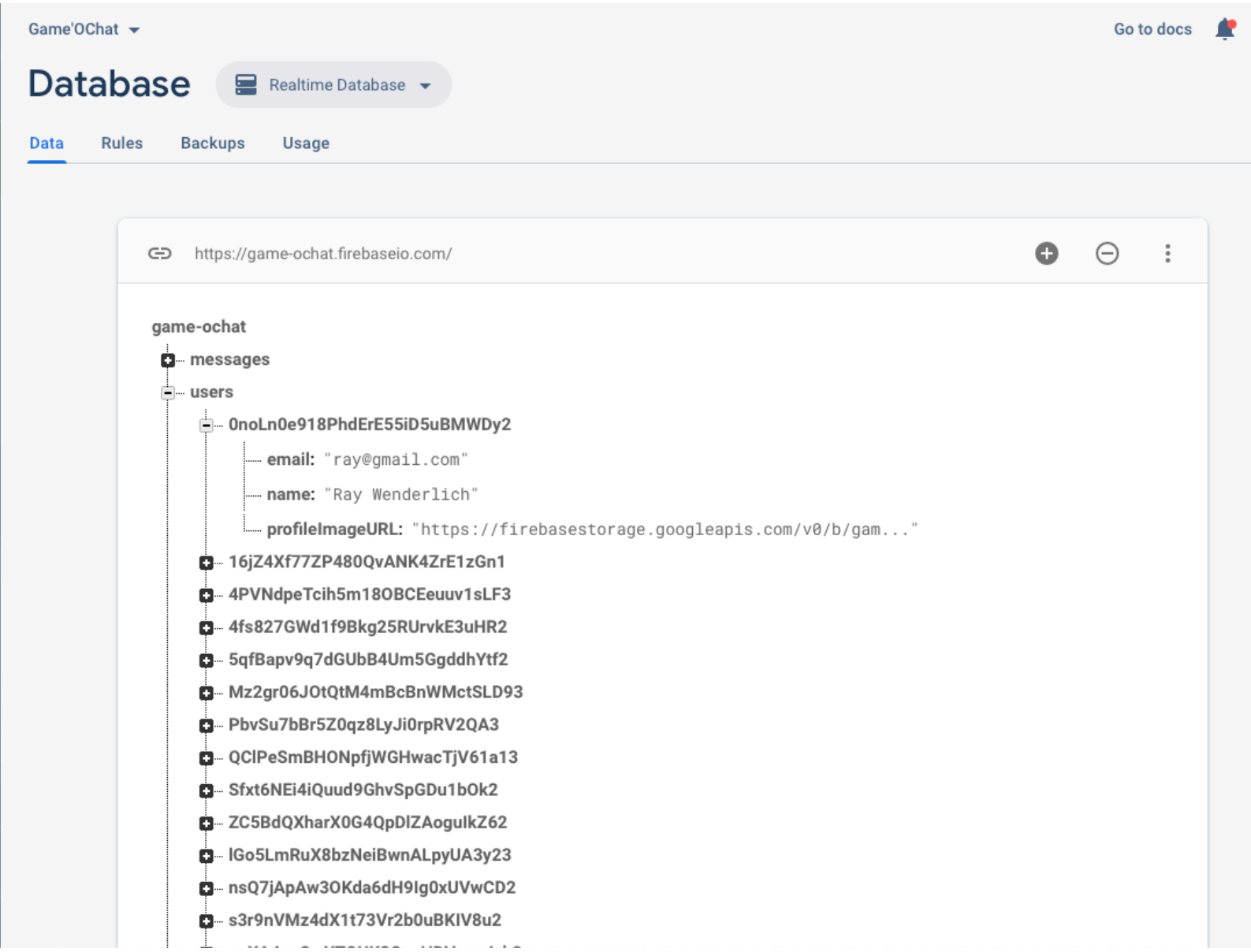

Fig. 12

2.2.2. Messages :-

- From_ID
- Text
- TimeStamp
- To_ID

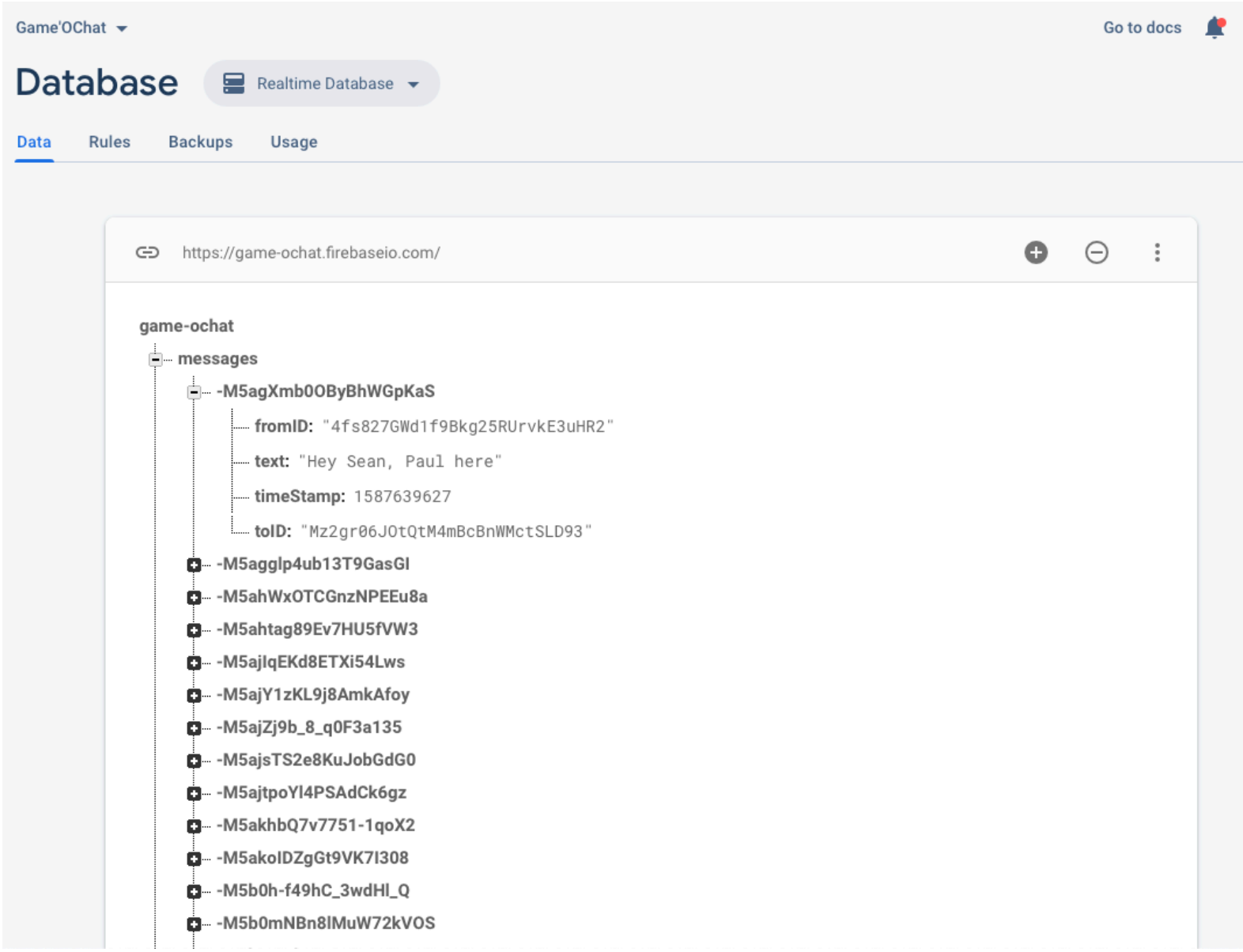

Fig. 13

2.2.3. User Messages :-

- Users_ID
  - ChatPartners_ID
    - Messages_ID

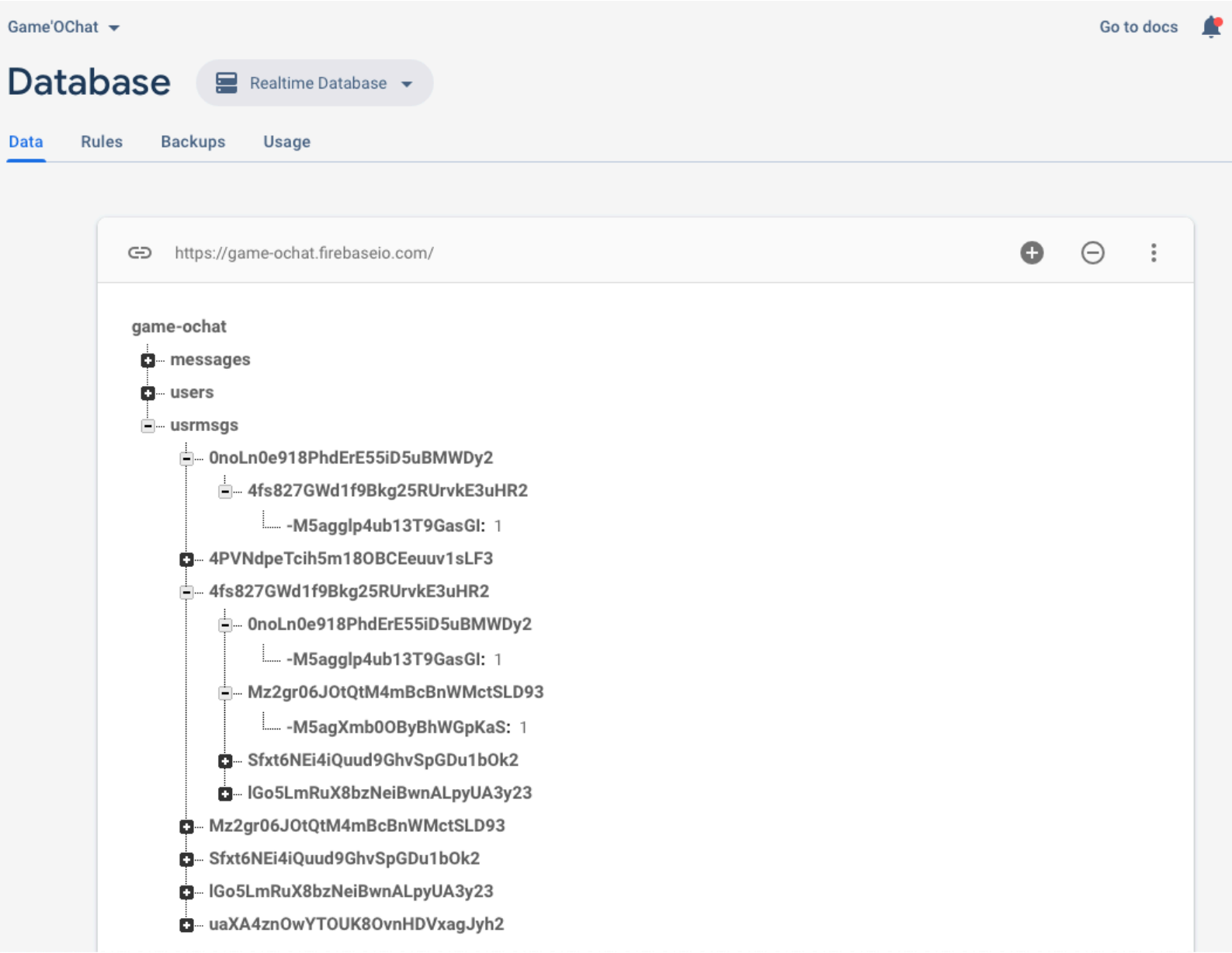

Fig. 14

2.3. MULTIMEDIA STORAGE :-

2.3.1. Profile Images :-

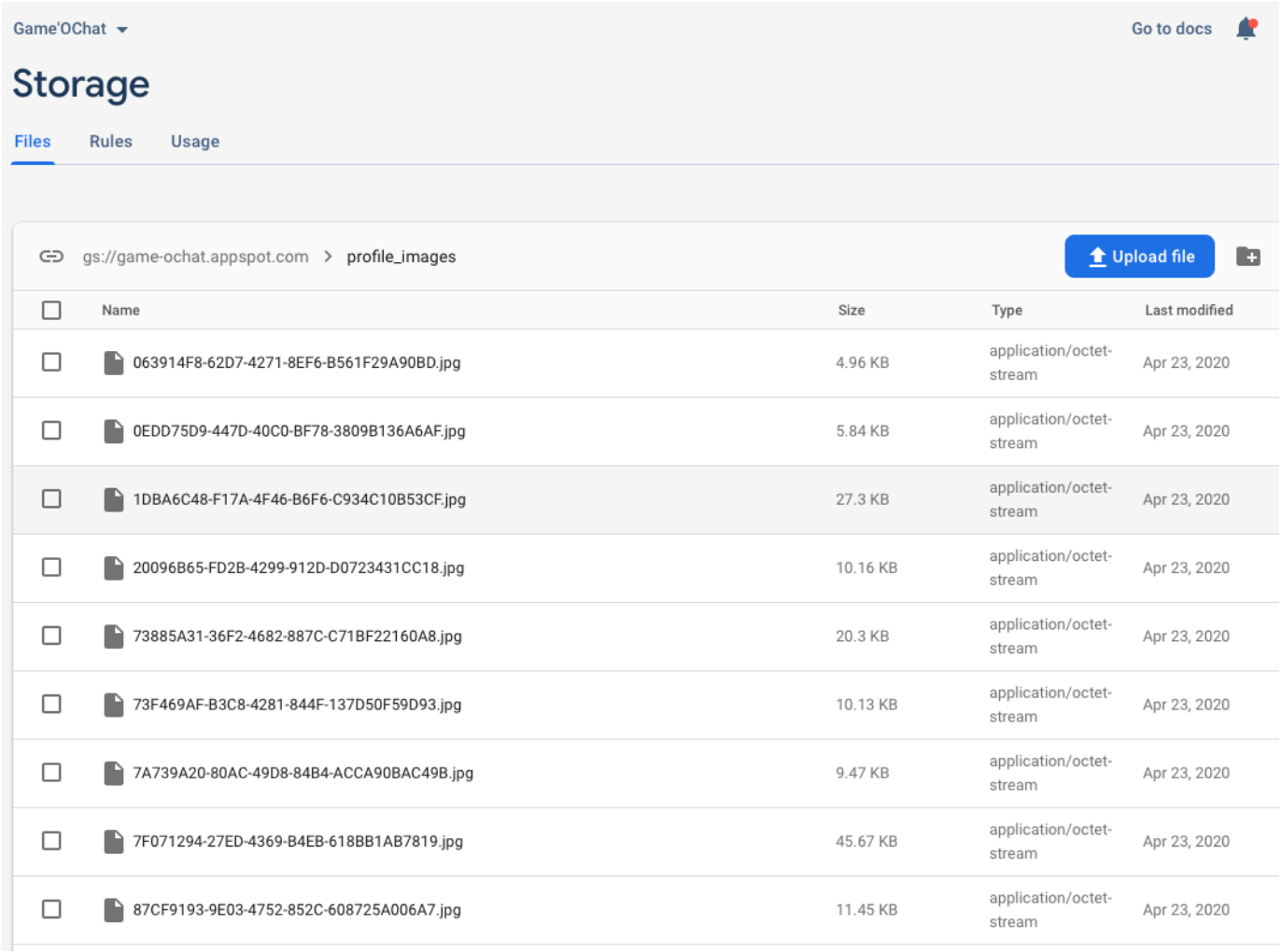

Fig. 15

2.3.2. Message Images :-

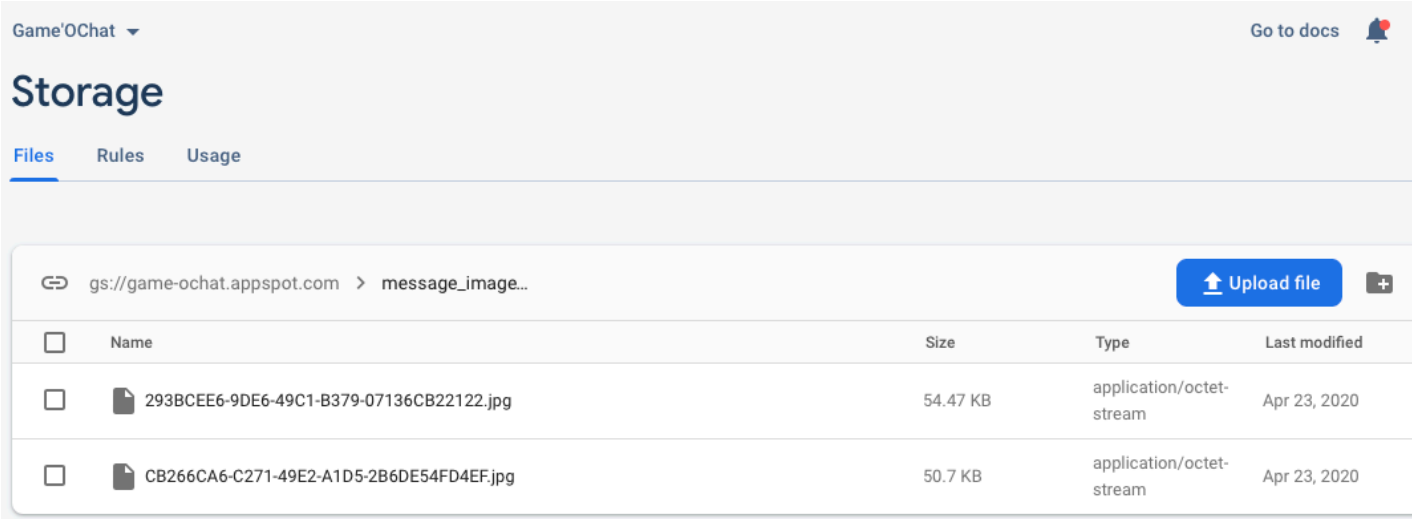

Fig. 16

TESTING & IMPLEMENTATION

1. UNIT TESTING IN iOS : -

Unit Testing do small and specific functionality tests for a certain unit of code and make sure that every unit passes the tests. If it passes the test, a green logo will appear next to it. If, for any reason, it was not successful, Xcode will mark a test as 'failed'. This is a sign to look into the code and search for the reasons of failure.

A unit test is (ideally) for testing a single 'unit' of code. Exactly what makes up a unit varies, but the important thing to keep in mind is that a unit test should be testing exactly one 'thing' at a time.

1.1 Tools & Technology :

Our technical team master a comprehensive set of tools and technologies during project development. A topic-oriented structure is provided below.

Product innovation: Brainstorming, LogicMap, XCAssert, Algorithms

Software analysis, architecture and design:

Brainstorming
UI XCAssertion
Design patterns
UML tools and techniques
Jira, Xcode, Vision

Database modeling tools:

DBBrowser for SQLite, Xcode CoreModel Tool, Firebase Analytics

Project management:

Project planning and management: MS Project, ScrumDesk

Networking protocols and data security:

TCP/IP, HTTP/HTTPS, POP3, FTP, etc.

1.2 Test Plan :

Introduction

Chat Application will be the interpreter to bring people and ideas together. We have been designing our Chat Application with well- equipped technology. This project is now at development phase, so readers can read the Software Requirement Specification document for details. As we know, master test plan is a living and breathing document that summarizes the overall effort required to test a software product. Master test plan will actually contain the details of individual tests to be run during the testing cycle like unit test, system test, beta test etc. However, our document will categorize and describe each test case. We strictly follow the instructions provided by our respective course teacher. The estimated time line for this project is a semester. The testing activities are to be done in parallel with the development process.

Items to be tested :

1. Application running on different client's iPhone including iPhone SE (2nd Generation)
2. User Profile
3. Chatting
4. Group Chatting
5. Add Friend
6. Remove Friend

Items Not To be Tested:

1. SRS of Chat Application
2. User Manual of Chat Application
3. Already Exist Chat Application
4. Any Legacy System

1.3 Item Pass Fail Criteria :

The test process will be completed when the project leader will be satisfied with the result of the test. For this, at least 98% of test cases must pass; all functionalities must be covered in those test cases and most of all, high and medium severity defects must be detected and fixed. Minor defects can be ignored, but with the assurance that it does not lead to severe defect. The project leader will decide whether the detected defects and criticality will cause the release of Chat Application.

Every unit test you write makes up part of a test suite. A test suite houses all unit tests related to a logical grouping of functionality (like your combat unit tests). If any individual test in a test suite fails, the entire test suite fails. A test suite is where you logically divide your tests. You want to divide your test code among different, logical suites (e.g. a test suite for physics and a separate one for combat). If you run into a situation where Unity can't find your test files or tests, double check to make sure there's an assembly definition file that includes your test suite. The next step is setting this up:

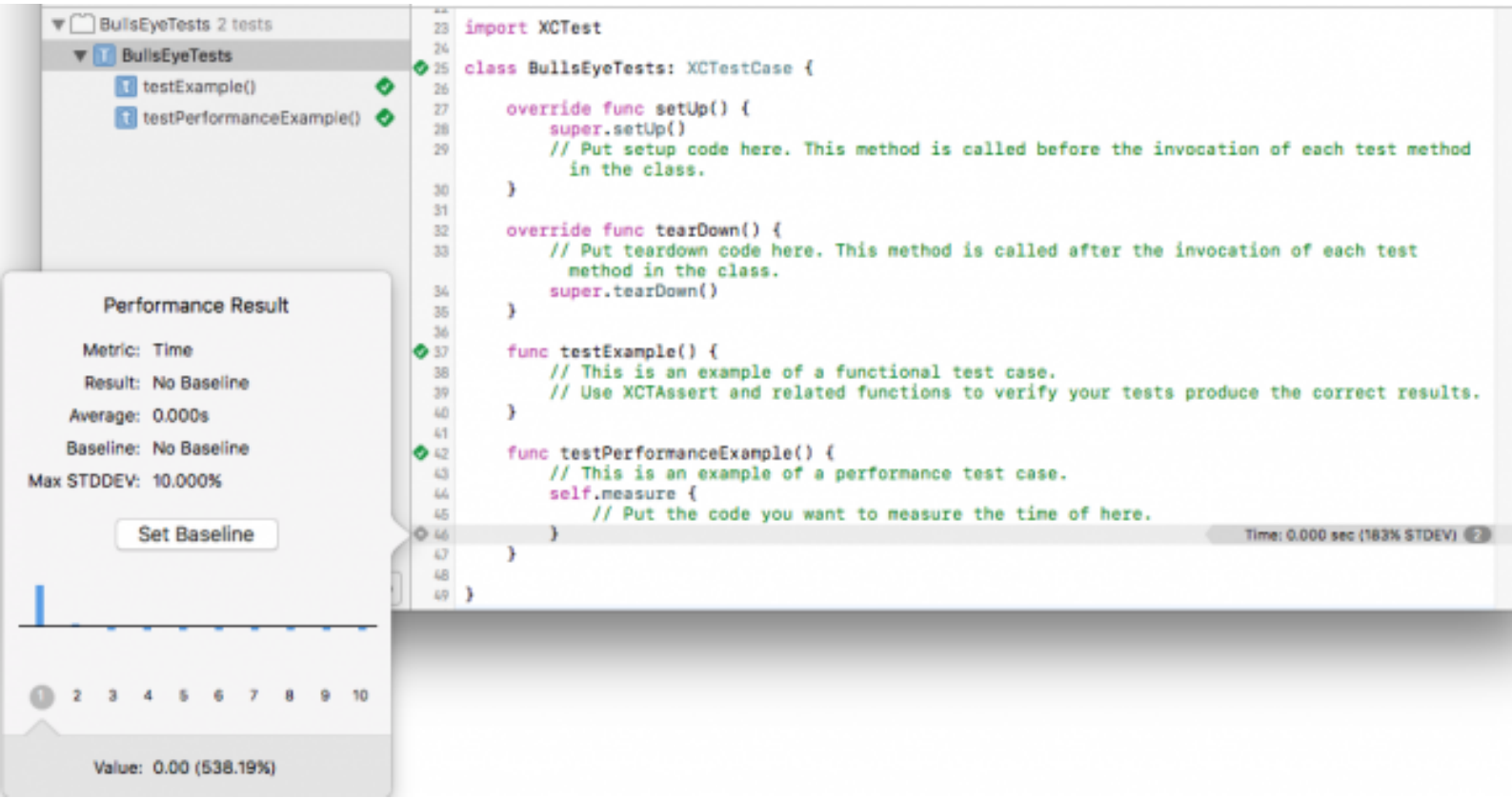

Fig. 17

1.4 Features To Be Tested :

The feature and attributes to be focused on during testing of the application:

Few features got depreciated later due to Test Failure: Ensuring Email, Group Chat, Search history, Edit Profile, Block friend, Deleting profile Picture

Features	Priority	Description
Create Account	1	To make user able to registrar in order to use Chat Application.
Log In	1	Log in as Authenticate user
Log Out	1	Log out from the system
Add Profile	3	Create profile for user
Edit Profile	3	Change any information of profile
Delete Account	3	Deleting Account
Find Friend	2	Searching for friend / user
Add Friend	2	Connecting Friends
Remove Friend	3	Disconnecting Friends
Block Friend	3	To get rid of unbearable friends
Personal Chat	1	Messaging Friend
Group Chat	2	Group Messaging
Add Friends In Group Chat	3	To add user in Group Messaging
Remove Friend From Group Chat	3	To Remove Someone From Group
Clear Chat History	3	To Remove Messages
Clear Search History	3	To Clear Search History
Upload Profile Picture	4	To add photos in the profile
Delete Profile Picture	4	To Remove Profile Picture
Change Password	4	To Change Password by user
Appropriate error message processing	3	It's important for both user & admin
Database	2	Technical features should be tightly in control as accessing database is frequently needed operation.
Ensure Email is sent to expected receiver	2	To ensure privacy
Authentication	2	Without authentication confidentiality integrity are not guaranteed.
Ensure Message is Sent in Minimal Time	2	To Ensure Policy & Privacy

Fig. 18

CONCLUSION

There is always a room for improvements in any apps. Right now we are just dealing with text communication. There are several iOS apps which serve similar purpose as this project, but these apps were rather difficult to use and provide confusing interfaces. A positive first impression is essential in human relationship as well as in human computer interaction. This project hopes to develop a chat service iOS app with high quality user interface. In future we may be extended to include features such as:

1. File Transfer
2. Voice Message
3. Video Message
4. Audio Call
5. Video Call
6. Group Chat
7. Edit Profile
8. Synchronizing Contacts

BIBLIOGRAPHY

- https://www.swiftbysundell.com
- https://www.hackingwithswift.com/articles/94/how-to-refactor-your-app-to-add-unit-tests
- https://www.raywenderlich.com/9261385-alamofire
- youtube.com
- https://medium.com/flawless-app-stories/a-complete-list-of-articles-on-unit-testing-with-swift-from-2017-9be8f046ef25
- https://www.letsbuildthatapp.com