

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**  
**TRƯỜNG ĐẠI HỌC MỞ THÀNH PHỐ HỒ CHÍ MINH**



**NGUYỄN TRỌNG HẢO**

**XÂY DỰNG ỨNG DỤNG NHẮN TIN TRỰC TUYẾN**  
**MOONCHAT**

**Mã số sinh viên:**

**1851050040**

**ĐỒ ÁN NGÀNH**

**NGÀNH CÔNG NGHỆ THÔNG TIN**

**Giảng viên hướng dẫn : LÊ NGỌC HIẾU**

**TP. HỒ CHÍ MINH, 2021**

## LỜI CẢM ƠN

Hoàn thành đồ án tốt nghiệp với đề tài “Xây dựng ứng dụng nhắn tin trực tuyến *MoonChat*”, cũng như các kết quả nghiên cứu của đồ án này là sự giúp đỡ rất nhiệt tình của thầy **ThS.Lê Ngọc Hiếu**. Em xin trân trọng cảm ơn thầy đã đồng hành cùng em trong dự án lần này.

Thêm vào đó, không thể không nhắc tới những thầy cô đã trao cho em kiến thức ở những môn học trước, những người đã góp phần không nhỏ củng cố kiến thức cũng như bước đệm để em có thể phát triển kỹ năng cũng như kiến thức của mình như bây giờ.

Xin chân thành cảm ơn Lãnh đạo trường Đại học Mở Thành phố Hồ Chí Minh và Khoa Công nghệ thông tin đã tạo điều kiện và gia hạn thời gian để chúng em có thể hoàn thành tốt đồ án của mình trong khoảng thời gian dịch bệnh này.

Cuối cùng, em xin gửi lời cảm ơn chân thành tới những thầy cô, những bạn sinh viên cùng lớp, đã giúp sức, ủng hộ và động viên em trong quá trình khó khăn thực hiện đồ án này.

TP.HCM, ngày      tháng      năm 20

*Sinh viên thực hiện đồ án*

***Nguyễn Trọng Hảo***

## LỜI CAM ĐOAN

Em xin cam đoan rằng đề án “Xây dựng ứng dụng nhắn tin trực tuyến MoonChat” là đề tài do em thực hiện.

Ngoại trừ những tài liệu tham khảo được trích dẫn trong đề án này, em xin cam đoan rằng toàn phần hoặc phần nhỏ của đề án này chưa được công bố hoặc sử dụng để nhận bằng cấp ở nơi khác.

Không có sản phẩm riêng nào của người khác ở trong đề án này mà không được trích dẫn theo đúng quy định.

Đề án này chưa bao giờ được nộp để nhận bằng cấp hay văn bằng tại các cơ sở học tập nào khác.

TP.HCM, ngày    tháng    năm 20

*Sinh viên thực hiện đề án*

***Nguyễn Trọng Hảo***

[illegible]

-----

# MỤC LỤC

LỜI CẢM ƠN .....	2
LỜI CAM ĐOAN.....	3
DANH MỤC CÁC TỪ VIẾT TẮT .....	8
MỞ ĐẦU .....	11
Chương 1 : GIỚI THIỆU TỔNG QUAN VỀ ĐỀ TÀI.....	12
1.1 Lý do chọn đề tài .....	12
1.2 Mục tiêu nghiên cứu .....	12
1.3 Đối tượng nghiên cứu .....	12
1.4 Phạm vi nghiên cứu.....	12
1.5 Phương pháp nghiên cứu .....	13
1.6 Ý nghĩa thực tiễn .....	13
Chương 2 : CƠ SỞ LÝ THUYẾT .....	14
2.1 Firebase.....	14
2.1.1 Firebase là gì ? .....	14
2.1.2 Firebase Authentication .....	15
2.1.3 Firebase Realtime Database .....	15
2.1.4 Firebase Cloud Storage .....	16
2.1.5 Firebase Cloud Messaging .....	16
2.2 Android Studio.....	17
2.2.1 Android Studio là gì ? .....	17
2.2.2 Tổng quan Project .....	18
2.2.3 Chi tiết cấu trúc thư mục .....	19
2.2.4 Chi tiết cấu trúc thư mục .....	21
2.2.5 Giới thiệu về các View .....	23
2.3 Liên kết Firebase với ứng dụng.....	26
2.3.1 Setup Firebase.....	26

2.3.2	Cài đặt môi trường Android :	27
2.3.3	Hoàn tất cài đặt	29
2.4	Một số thư viện cần thiết trong ứng dụng	30
<b>Chương 3 : PHÂN TÍCH THIẾT KẾ HỆ THỐNG</b>		31
3.1	Khảo sát hệ thống.....	31
3.2	Phân tích nghiệp vụ hệ thống.....	31
3.3	Phân tích thiết kế hệ thống.....	33
3.3.1	Lược đồ Usecase.....	33
3.3.2	Usecase đăng ký.....	33
3.3.3	Usecase hiển thị người dùng.....	33
3.3.4	Usecase gửi tin nhắn.....	33
3.3.5	Usecase nhận thông báo tin nhắn mới.....	34
3.3.6	Usecase hiển thị tin nhắn .....	34
3.3.7	Usecase Đổi ảnh đại diện, trạng thái .....	34
3.3.8	Usecase Firebase DB.....	34
3.4	Sơ đồ hoạt động (Activity Diagram).....	36
3.5	Thiết kế và xây dựng cơ sở dữ liệu.....	41
3.5.1	Bảng 1 của hệ thống .....	41
3.5.2	Bảng 2 của hệ thống .....	41
3.5.3	Bảng 3 của hệ thống .....	41
3.5.4	Bảng 4 của hệ thống .....	42
3.6	Sơ đồ ER Diagram: .....	42
<b>Chương 4 : THỰC NGHIỆM</b> .....		43
4.1	Cấu hình cài đặt .....	43
4.1.1	Java JDK .....	43
4.1.2	Android Studio .....	43
4.1.3	Firebase.....	43

4.2	Giao diện ứng dụng.....	43
<b>Chương 5 : KẾT LUẬN .....</b>		<b>57</b>
5.1	Kết quả đạt được.....	57
5.2	Những mặt hạn chế.....	57
5.3	Kiến nghị và hướng phát triển.....	57
<b>TÀI LIỆU THAM KHẢO.....</b>		<b>59</b>

# DANH MỤC CÁC TỪ VIẾT TẮT

**API** - Application Program Interface

**IDE** - Integrated Development Environment

**JSON** – JavaScript Object Notation

**NoSQL** – Non-Structured Query Language

**XML** – Extensible Markup Language

**SDK** – Software Development Kit

**FCM** – Firebase Cloud Messaging

**Admin** – Administrator

**UI** – User Interface



## DANH MỤC HÌNH VẼ

Hình 2.1.1 Firebase.....	14
Hình 2.1.2 Firebase Authentication .....	15
Hình 2.1.3 Firebase Realtime Database .....	15
Hình 2.1.4 Firebase Cloud Storage.....	16
Hình 2.1.5 Push Notification .....	16
Hình 2.2.1 Giới thiệu về Android Studio .....	17
Hình 2.2.2 Cấu trúc Project .....	18
Hình 2.2.3 Đặt id trong View .....	19
Hình 2.2.4 Gọi id trong class.....	19
Hình 2.2.5 Cấu trúc thư mục Layout.....	19
Hình 2.2.6 Cấu trúc thư mục Java .....	21
Hình 2.2.7 Cấu trúc của View[4].....	23
Hình 2.2.8 XML file .....	24
Hình 2.2.9 Giao diện Preview .....	25
Hình 2.2.10 Giao diện Preview .....	25
Hình 2.3.1 Thêm project mới .....	26
Hình 2.3.2 Khởi tạo project Firebase .....	26
Hình 2.3. 3 Firebase tự động khởi tạo .....	26
Hình 2.3. 4 Firebase tự động khởi tạo .....	26
Hình 2.3.5 Lấy chứng thực SHA-1.....	27
Hình 2.3.6 Đăng ký ứng dụng của Firebase .....	27
Hình 2.3.7 Cài đặt google services .....	28
Hình 2.3.8 Thêm Firebase SDK .....	28
Hình 2.3.9 Thêm Firebase SDK .....	29
Hình 2.3.10 Giao diện Analytics của Firebase .....	29
Hình 2.4.1 Thư viện cần thiết trong ứng dụng .....	30
3.3.1.1 Lược đồ Usecase.....	33
Hình 3.4.1 Sơ đồ Đăng ký người dùng.....	36
Hình 3.4. 2 Sơ đồ đăng nhập bằng email-password .....	37
Hình 3.4. 3 Sơ đồ đăng nhập bằng Google Account .....	37
Hình 3.4. 4 Sơ đồ xem người dùng.....	38

Hình 3.4. 5 Sơ đồ hiển thị tin nhắn.....	38
Hình 3.4. 6 Sơ đồ gửi tin nhắn.....	39
Hình 3.4. 7 Sơ đồ nhận tin nhắn .....	39
Hình 3.4. 8 Sơ đồ nhận thông báo tin nhắn mới.....	40
Hình 3.4. 9 Sơ đồ đổi ảnh đại diện .....	40
Hình 3.5.1. 1 Bảng User .....	41
Hình 3.5.1. 2 Bảng Chat .....	41
Hình 3.5.1. 3 Bảng ChatList .....	42
Hình 3.5.1. 4 Bảng Token.....	42
Hình 3.6. 1 Sơ đồ ERD .....	42
Hình 4.2.1 Đăng nhập tài khoản với Google Account .....	44
Hình 4.2.2 Đăng nhập tài khoản với Email-Password .....	44
Hình 4.2.3 Đăng ký tài khoản cho người dùng mới .....	45
Hình 4.2.4 Giao diện xác minh tài khoản .....	46
Hình 4.2.5 Xác minh tài khoản từ email .....	46
Hình 4.2.6 Giao diện tin nhắn khi có tin nhắn cũ.....	47
Hình 4.2.7 Giao diện tin nhắn khi chưa có tin nhắn nào .....	47
Hình 4.2.8 Giao diện hiển thị Users .....	48
Hình 4.2.9 Giao diện Profile.....	49
Hình 4.2.10 Đặt ảnh đại diện mới .....	50
Hình 4.2.11 Chọn ảnh đại diện trong LocalFiles.....	50
Hình 4.2.12 Giao diện nhắn tin chính.....	51
Hình 4.2. 13 Trạng thái "seen" .....	52
Hình 4.2.14 Trạng thái "delivered" .....	52
Hình 4.2.15 Trạng thái người dùng offline .....	52
Hình 4.2.16 Trạng thái người dùng online .....	52
Hình 4.2.17 Thông báo tin nhắn mới.....	53
Hình 4.2.18 Thay đổi mật khẩu .....	54
Hình 4.2.19 Giao diện Analytics của Firebase .....	55
Hình 4.2.20 Giao diện Firebase Authentication .....	55
Hình 4.2.21 Giao diện Firebase Cloud Storage.....	56

## MỞ ĐẦU

Trong xã hội ngày nay, các nền tảng trực tuyến đã không còn xa lạ với mỗi người, kể cả những thế hệ đi trước, những người mới tiếp cận Internet. Thêm vào đó, trong vài năm gần đây, tình hình dịch bệnh trên toàn thế giới nói chung và Việt Nam nói riêng đã làm ngăn cách và chặn đứng khả năng giao tiếp, gặp gỡ, tiếp xúc với những người xung quanh. Đi cùng với đó là sự đi lên về sự phát triển của mạng xã hội, những hình thức trực tuyến, các lớp học online,... Nhưng nhu cầu giao tiếp của con người vẫn không thể thay đổi, đó là động lực cho những công ty, doanh nghiệp phát triển những ứng dụng giao tiếp cho riêng mình, và *MoonChat* ra đời để đáp ứng mục đích đó.

Với những tính năng cơ bản, phục vụ cho một cộng đồng vừa và nhỏ, *MoonChat* là sự đáp ứng cơ bản với những nhu cầu giao tiếp, trao đổi qua tin nhắn trên một thiết bị điện thoại mà ai cũng sở hữu trong xã hội ngày nay.

Ứng dụng cho phép người dùng có thể đăng nhập, tương tác, nhắn tin với những người trong cộng đồng sử dụng nó. Gặp gỡ và trò chuyện với mọi người xung quanh là điều ai cũng mong muốn được làm nhưng không thể với tình hình dịch bệnh như hiện nay. Nhưng thông qua ứng dụng, chúng ta có thể gặp gỡ những người bạn mới, nhắn tin trò chuyện với họ thông qua nền tảng trực tuyến một cách thuận tiện nhất.

Với những khát khao đó, đề tài lần này mong muốn sẽ đáp ứng được nhu cầu của người dùng, xây dựng một cộng đồng vững mạnh, để từ đó có thể phát triển xa hơn để phục vụ, đáp ứng những nguyện vọng lớn hơn của người dùng khi sử dụng ứng dụng này.

# **Chương 1 : GIỚI THIỆU TỔNG QUAN VỀ ĐỀ TÀI**

## **1.1 Lý do chọn đề tài**

Là một người sử dụng hệ điều hành android đã lâu, em cảm thấy rất hứng thú với những gì mà hệ điều hành này mang lại cho người dùng: hệ điều hành mở, tùy biến nhiều nội dung, có thể phát triển những gì mà người dùng thích mà không bị gò bó quá nhiều về mặt pháp lý,... Vì thế trong đề án lần này em muốn thử sức tiếp cận với ngôn ngữ cũng như hệ điều hành này để khám phá những thứ mới mẻ cũng như thử sức mình trong một môi trường mới.

Xét về khía cạnh “Tại sao lại xây dựng một ứng dụng nhắn tin trực tuyến”, để trả lời câu hỏi này thì chúng ta phải xem xét đến hoàn cảnh thực tế hiện nay, tình hình dịch bệnh COVID rất phức tạp, mọi hoạt động đều diễn ra trên nền tảng trực tuyến. Nhận thấy nhu cầu đó, em muốn tạo ra một ứng dụng có thể giao tiếp mọi người trong một cộng đồng nhỏ nhất định. Vì thế em đã chọn đề tài “Ứng dụng nhắn tin” này.

## **1.2 Mục tiêu nghiên cứu**

Đề án tập trung vào việc phát triển một ứng dụng nhắn tin, giao tiếp trực tiếp qua nền tảng Android.

Từ đó, phục vụ cho việc nghiên cứu phát triển một ứng dụng trên nền tảng điện thoại, phân tích về thiết kế, xây dựng cơ sở dữ liệu, xử lý dữ liệu ở trong nền tảng này.

## **1.3 Đối tượng nghiên cứu**

Đối tượng nghiên cứu chính của đề tài “Ứng dụng nhắn tin trực tuyến” bao gồm hệ điều hành Android, nền tảng Firebase và các chức năng phục vụ cho người dùng sử dụng ứng dụng.

## **1.4 Phạm vi nghiên cứu**

Phạm vi nghiên cứu của đề án lần này là xây dựng một ứng dụng giao tiếp tập trung vào một cộng đồng quy mô vừa và nhỏ, từ đó để có thể phát triển lên một ứng dụng có thể đáp ứng cho việc liên kết những người có mong muốn phát triển một cộng đồng riêng cho bản thân mình.

Nghiên cứu này tập trung vào các chức năng cơ bản của một mạng xã hội giao tiếp cơ bản.

## **1.5 Phương pháp nghiên cứu**

Đề tài lần này nghiên cứu về một ứng dụng khá phổ biến hiện nay nên dựa vào đó để chọn lọc những phương pháp tối ưu nhất:

- Tham khảo các ứng dụng đi trước, quan sát những chức năng cần thiết, những hoạt động và các đối tượng của ứng dụng đó, ví dụ như: Messenger, Zalo, Telegram,... Từ đó chất lọc và đưa ra một thiết kế tối ưu cho đề án.
- Nghiên cứu về hệ điều hành Android, nền tảng lưu trữ Firebase và cách phát triển một ứng dụng kết hợp tốt tất cả những điều trên.
- Thực nghiệm, chạy thử ứng dụng và kiểm tra lấy kết quả.

Qua những phương pháp và các bước cụ thể như trên, từ đó phát triển lên một ứng dụng hoàn chỉnh có thể ứng dụng thực tế.

## **1.6 Ý nghĩa thực tiễn**

Đề tài đề án lần này không chỉ là cơ hội để trau dồi kỹ năng, kiến thức cần có để phát triển bản thân mà còn góp phần nghiên cứu nhu cầu về mảng truyền thông, giao tiếp xã hội trực tuyến trong hoàn cảnh thực tiễn ngày nay.

## Chương 2 : CƠ SỞ LÝ THUYẾT

### ❖ Giới thiệu

Trong phần này, em sẽ giới thiệu những kiến thức cần có để phát triển một ứng dụng trên nền tảng Android, bên cạnh đó là cách kết hợp sử dụng Firebase làm cơ sở dữ liệu, lưu trữ thông tin hay xác minh tài khoản.

Đầu tiên, em sẽ trình bày về nền tảng lưu trữ Firebase, các công cụ hoạt động cũng như cách để liên kết Firebase với dự án của mình.

Sau đó em sẽ trình bày về các khái niệm cơ bản của Android, xây dựng một ứng dụng thuần Android với sự hỗ trợ của ngôn ngữ Java và các thư viện cần thiết để tạo nên ứng dụng này.

### ❖ Cơ sở lý thuyết

#### 2.1 Firebase

##### 2.1.1 Firebase là gì ?

Firebase thuộc dạng Backend-as-a-service(BaaS)[1] là một nền tảng cung cấp lưu trữ dữ liệu đám mây được phát triển bởi Google. Nó cung cấp cho các lập trình viên rất nhiều công cụ và chức năng để phát triển một ứng dụng hoàn chỉnh và chất lượng.

Nhưng trên thực tế, Firebase chỉ hoạt động một cách ổn định trên những dự án vừa và nhỏ. Để tiến xa đến những dự án lớn, cần phải có sự tối ưu về mặt lưu trữ và sử dụng Firebase một cách hiệu quả nhất.

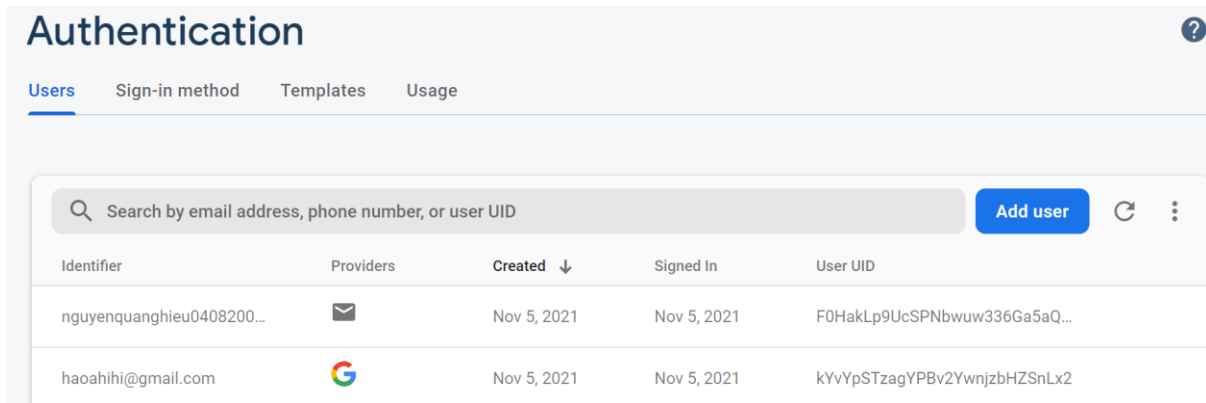
Trong dự án lần này, MoonChat chỉ tiếp cận đối tượng sử dụng vừa và nhỏ nên Firebase là sự lựa chọn mang tính hiệu quả cao để tối ưu ứng dụng.



*Hình 2.1.1 Firebase*

### 2.1.2 Firebase Authentication

Firebase còn hỗ trợ chức thực tài khoản qua Firebase Authentication bằng nhiều hình thức : email-passwords, phone numbers, Google, Facebook, Twitter, ... Bên cạnh đó Firebase Authentication (SDK) còn hỗ trợ tích hợp nhiều phương thức chức thực trong cùng một dự án, một ứng dụng.

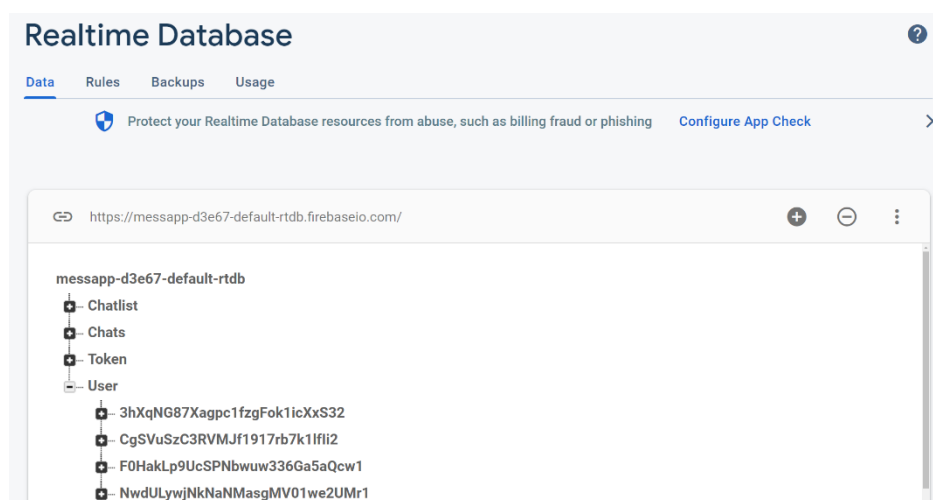


Hình 2.1.2 Firebase Authentication

### 2.1.3 Firebase Realtime Database

Firebase Realtime Database là một cơ sở dữ liệu lưu trữ đám mây NoSQL. Dữ liệu được lưu dưới dạng JSON và được đồng bộ hóa trong thời gian thực mỗi khi người dùng truy cập vào nó, kể cả khi ứng dụng offline[2].

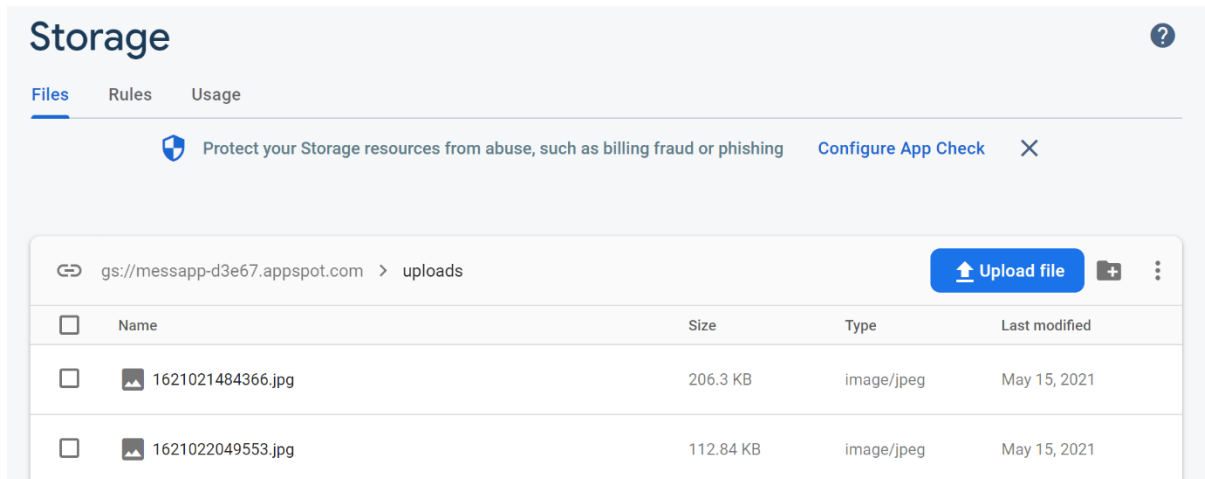
Một khi ứng dụng được kết nối với Realtime Database, tất cả nền tảng ứng dụng như Apple, Android và JavaScript SDKs, đều được đồng bộ hóa và cập nhật liên tục theo thời gian thực.



Hình 2.1.3 Firebase Realtime Database

### 2.1.4 Firebase Cloud Storage

Cloud Storage là dịch vụ lưu trữ trực tuyến của Firebase, nó cho phép các lập trình viên có thể lưu trữ những dữ liệu liên quan đến người dùng như ảnh, âm thanh hoặc video. Firebase SDKs dành cho Cloud Storage sẽ bảo đảm về bảo mật để tải lên hoặc tải xuống các tập tin trong ứng dụng đã liên kết Firebase.

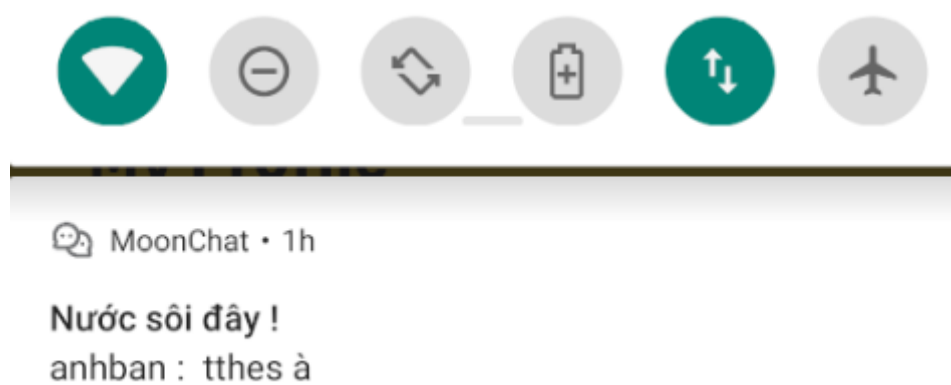


Hình 2.1.4 Firebase Cloud Storage

### 2.1.5 Firebase Cloud Messaging

Firebase Cloud Messaging (FCM) hỗ trợ lập trình viên có thể gửi tin nhắn tới những thiết bị sử dụng ứng dụng mà không mất bất kì khoản chi phí nào.

Ở trong đồ án này, FCM được sử dụng để gửi Push Notification đến các thiết bị để nhận tin nhắn mới.



Hình 2.1.5 Push Notification



## 2.2 Android Studio

### 2.2.1 Android Studio là gì ?

Android Studio[3] là một IDE dành riêng cho phát triển ứng dụng Android dựa trên IntelliJ IDEA. Hơn thế, Android Studio đã đem đến những tính năng tích hợp riêng để có thể phát triển một ứng dụng một cách hoàn thiện nhất :

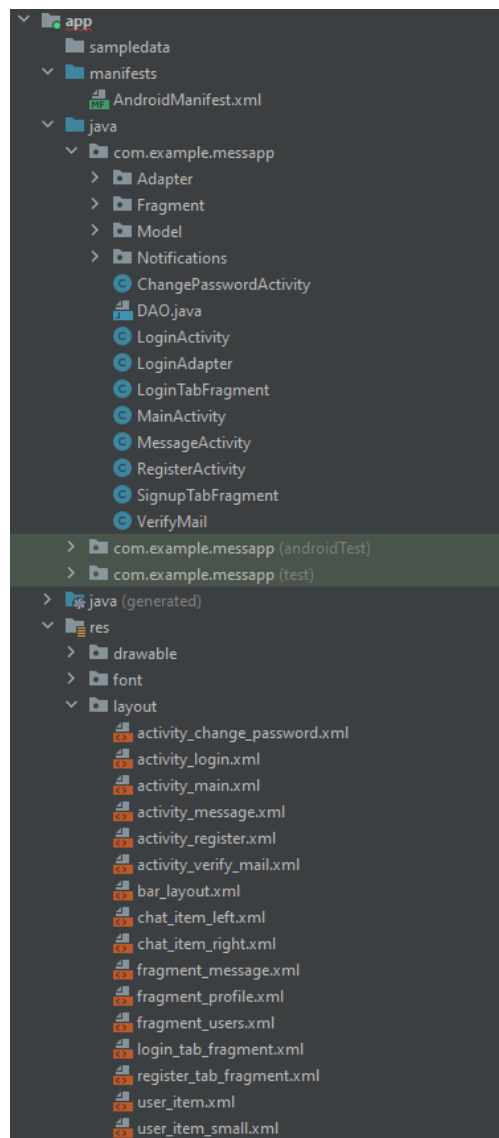
- Tùy biến hệ thống Gradle-based
- Kho giả lập tương thích nhiều thiết bị cũng như API
- Thư viện rộng lớn
- Môi trường ảo để lập trình viên có thể phát triển tất cả ứng dụng Android
- Hệ thống phát triển để hỗ trợ cho Google Cloud Platform nên việc kết hợp với Firebase sẽ được tối ưu nhất

Việc phát triển một ứng dụng trên Android Studio sẽ dễ dàng hơn nhiều so với các phần mềm không thông dụng khác. Bên cạnh đó, một cộng đồng với những lập trình viên đi trước sẽ là những nguồn hỗ trợ đắc lực khi gặp vấn đề trong lúc phát triển ứng dụng.



*Hình 2.2.1 Giới thiệu về Android Studio*

### 2.2.2 Tổng quan Project



Hình 2.2.2 Cấu trúc Project

Trong Android Studio, lập trình viên phải thiết kế 2 thành phần riêng biệt nhưng được liên kết hoạt động với nhau đó là *Java* và *Layout*. Trong đó:

- Layout chứa các file .xml là các thiết kế giao diện dành cho người dùng. Ở đây, lập trình viên sẽ thiết kế các *Activity*, đây là thành phần quan trọng trong một ứng dụng. Một ứng dụng hoàn chỉnh là sự kết hợp một cách mượt mà giữa các *Activity*. Mỗi *Activity* sẽ đảm nhận một module, hoặc thành phần, chức năng độc lập tương ứng với một giao diện người dùng ( UI ).

- Java là nơi lưu trữ các class .java, chịu trách nhiệm xử lý và phát triển những chức năng của ứng dụng.
- Trong các class, để xử lý những thành phần trên giao diện, chúng phải được liên kết với nhau qua *id*, được đặt trong các thành phần của Layout, và được gọi thông qua phương thức *findViewById()*.

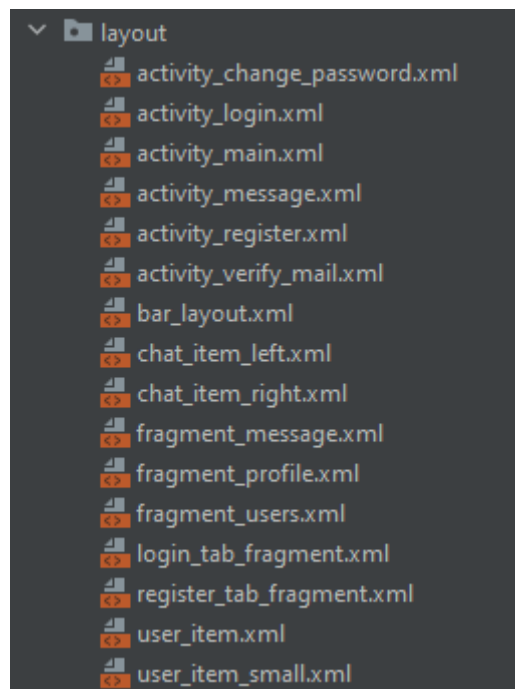
```
<TextView
    android:id="@+id/textView"
```

Hình 2.2.3 Đặt id trong View

```
TextView textView = findViewById(R.id.textView);
```

Hình 2.2.4 Gọi id trong class

### 2.2.3 Chi tiết cấu trúc thư mục

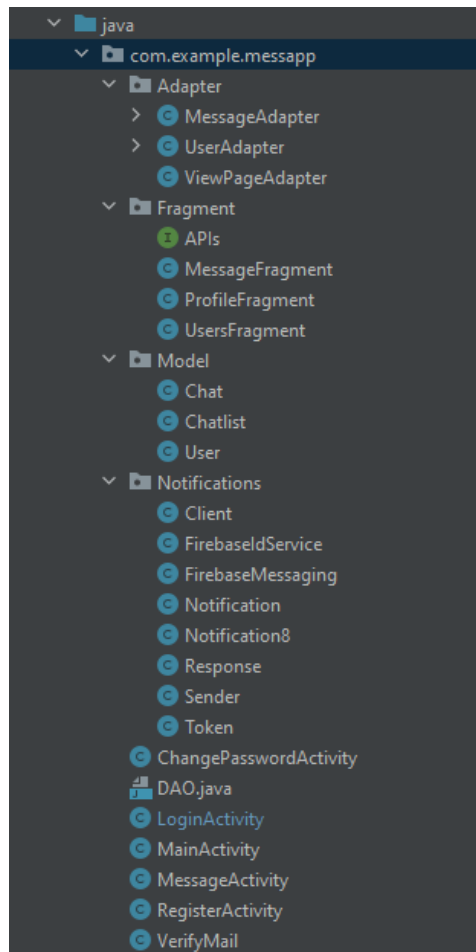


Hình 2.2.5 Cấu trúc thư mục Layout

- Các folder và file chính :

- activity\_change\_password.xml : Giao diện thay đổi mật khẩu cho người dùng.
- activity\_login.xml : Giao diện đăng nhập khi mở app lần đầu.
- activity\_main.xml : Giao diện chính của người dùng khi sử dụng app.
- activity\_message.xml, chat\_item\_left, chat\_item\_right : Giao diện nhắn tin, bao gồm các thành phần của người gửi và người nhận.
- activity\_register.xml : Giao diện đăng ký người dùng mới
- fragment\_message.xml : Giao diện tin nhắn hiển thị trên activity\_main.xml, tại đây sẽ hiển thị những tin nhắn của người dùng đã nhắn với bạn trước đó.
- fragment\_profile.xml : Giao diện hiển thị tên, ảnh đại diện và bio của người dùng.
- fragment\_users.xml : Giao diện hiển thị những người dùng đã đăng ký ở trong ứng dụng.
- user\_item.xml : Thành phần người dùng, hiển thị tên, ảnh đại diện và tin nhắn cũ.

## 2.2.4 Chi tiết cấu trúc thư mục



Hình 2.2.6 Cấu trúc thư mục Java

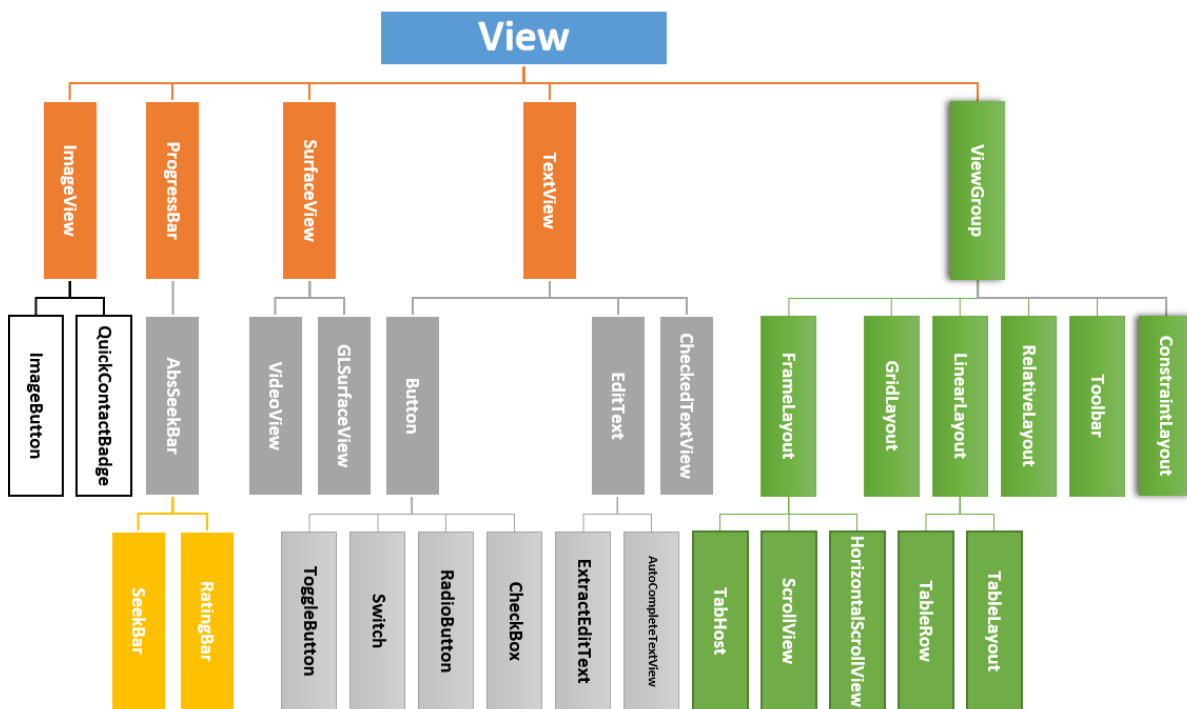
- Các folder và file chính xử lý Activity
  - Adapter : Xử lý các thành phần item trong RecyclerView
    - MessageAdapter : Xử lý các thành phần trong giao diện nhắn tin, hiển thị người gửi, người nhận và các tin nhắn giữa hai đối tượng.
    - UserAdapter : Xử lý hiển thị các thành phần trong user\_item như tên, ảnh đại diện, online, offline.
    - ViewPagerAdapter : Điều hướng xử lý thanh Tablayout.
  - Fragment : Đây là các thành phần trong Tablayout, mỗi Fragment xử lý cho một Activity riêng và hiển thị từng thành phần trong Tablayout. Tất cả những thành phần này đều giao tiếp với phía database của Firebase.
    - APIs : Interface chứa Authorization key nhằm mục đích sử dụng FCM, hay Push Notification.

- MessageFragment : Xử lý tin nhắn, sau đó lọc ra những user đã nhắn tin với người dùng trước đó để hiển thị lên giao diện Chats.
- UsersFragment : Tập hợp user ở trong database và hiển thị lên giao diện Users.
- ProfileFragment : Hiển thị thông tin người dùng, bao gồm tên, ảnh đại diện và giới thiệu. Ở fragment này còn xử lý upload ảnh từ phía thiết bị, thiết lập ảnh đại diện và gửi ảnh này về phía server của Firebase.
- Model : Thiết lập các model để tạo dữ liệu trong database
  - Chat : Model tất cả tin nhắn của người dùng
  - Chatlist : Model của những user đã nhắn tin với người dùng
  - User : Model thông tin của người dùng
- Notifications : Thiết lập thông báo của FCM, Push Notification, hiển thị thông báo khi có tin nhắn mới
  - Client : Thiết lập Retrofit ( truy xuất và tải lên JSON )
  - Notification : Model của notification cho hệ điều hành Oreo trở lên
  - Notification8 : Model của notification cho hệ điều hành Oreo trở xuống
  - FirebaseIdService : Lấy token của user và update token mỗi khi người dùng đăng nhập lại
  - FirebaseMessaging : Xử lý các thành phần thông báo : icon, title, user, nội dung tin nhắn,...
  - Token : Model của token
- ChangePasswordActivity : Xử lý hoạt động thay đổi mật khẩu của người dùng
- LoginActivity : Xử lý trạng thái đăng nhập của người dùng
- MainActivity : Xử lý giao diện chính của người dùng
- MessageActivity : Xử lý hành động nhắn tin giữa người dùng với người dùng
- RegisterActivity : Xử lý trạng thái đăng ký của người dùng mới bằng phương thức email-password

- VerifyMail : Xử lý trạng thái xác thực tài khoản sau khi đăng ký người dùng.

### 2.2.5 Giới thiệu về các View

Các thành phần cơ bản để thiết kế nên một giao diện người dùng ( UI ) là ViewGroup, View và Layout. Tất cả các thành phần đều được xây dựng từ lớp cơ sở View, kết hợp chúng lại ta có thể xây dựng các UI phức tạp cho ứng dụng Android.



Hình 2.2.7 Cấu trúc của View[4]

**XML** (Extensible Markup Language - ngôn ngữ đánh dấu mở rộng), là một văn bản để lưu dữ liệu. Các giá trị phần tử này được gọi bởi các thẻ `< %tag% >` mở và `</ %tag% >` đóng. Các thẻ này là do ứng dụng quy định hoặc do người sử dụng đặt ví dụ như : TextView, ImageView, ...

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res-auto"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/linearLayout4"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:gravity="center"
    tools:context=".VerifyMail">

    <TextView
        android:id="@+id/txt_verify"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:fontFamily="@font/sanfrancisco_regular"
        android:text="Please verify your email"
        android:textAlignment="center"
        android:textSize="29dp"
        android:textStyle="bold"
        android:visibility="gone"
        app:layout_constraintBottom_toTopOf="@+id/btn_verify"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.895" />

    <Button...>

    <Button...>

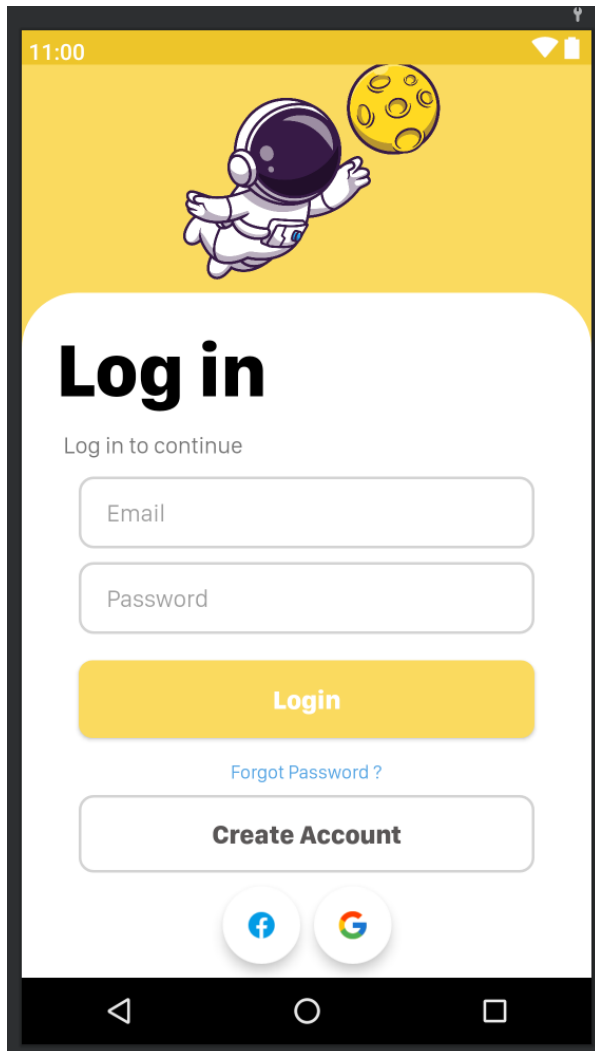
</androidx.constraintlayout.widget.ConstraintLayout>

```

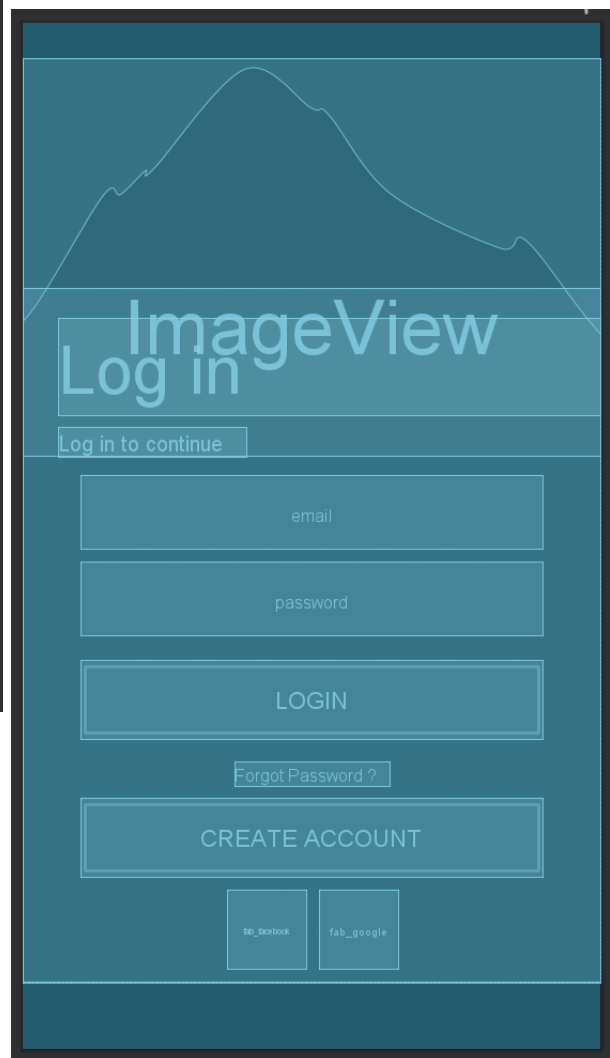
Hình 2.2.8 XML file



Layout Preview : Sau khi lập trình viên sử dụng các thẻ để gọi các thành phần và kết hợp chúng lại với nhau, Android Studio sẽ hiển thị preview trực tiếp một cách trực quan nhất để hỗ trợ người sử dụng.



Hình 2.2.10 Giao diện Preview

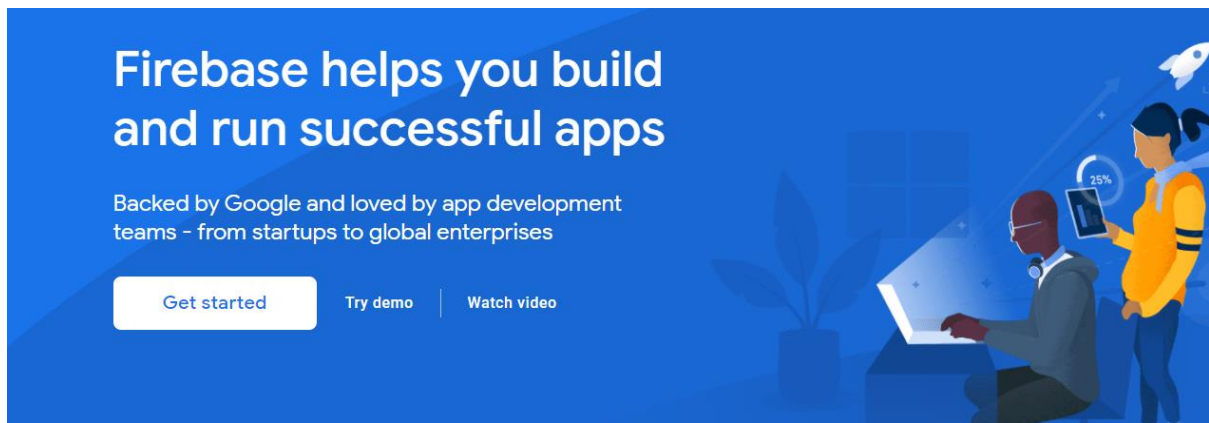


Hình 2.2.9 Giao diện Preview

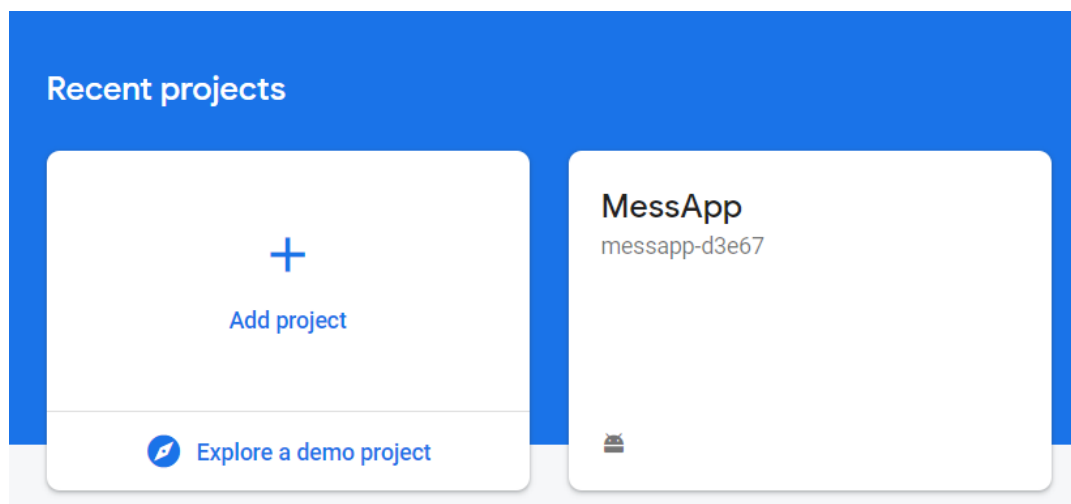
## 2.3 Liên kết Firebase với ứng dụng

### 2.3.1 Setup Firebase

- Tạo project : Khởi tạo project trong Firebase
  - Sau khi vào trang chủ Firebase, chọn bắt đầu với dự án mới

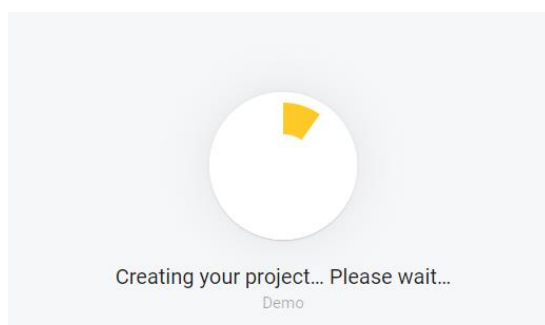


Hình 2.3.2 Khởi tạo project Firebase

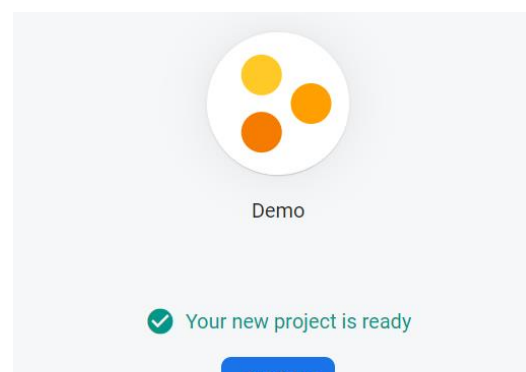


Hình 2.3.1 Thêm project mới

- Khi đặt tên và setup xong Firebase sẽ tự động khởi tạo một môi trường project mới



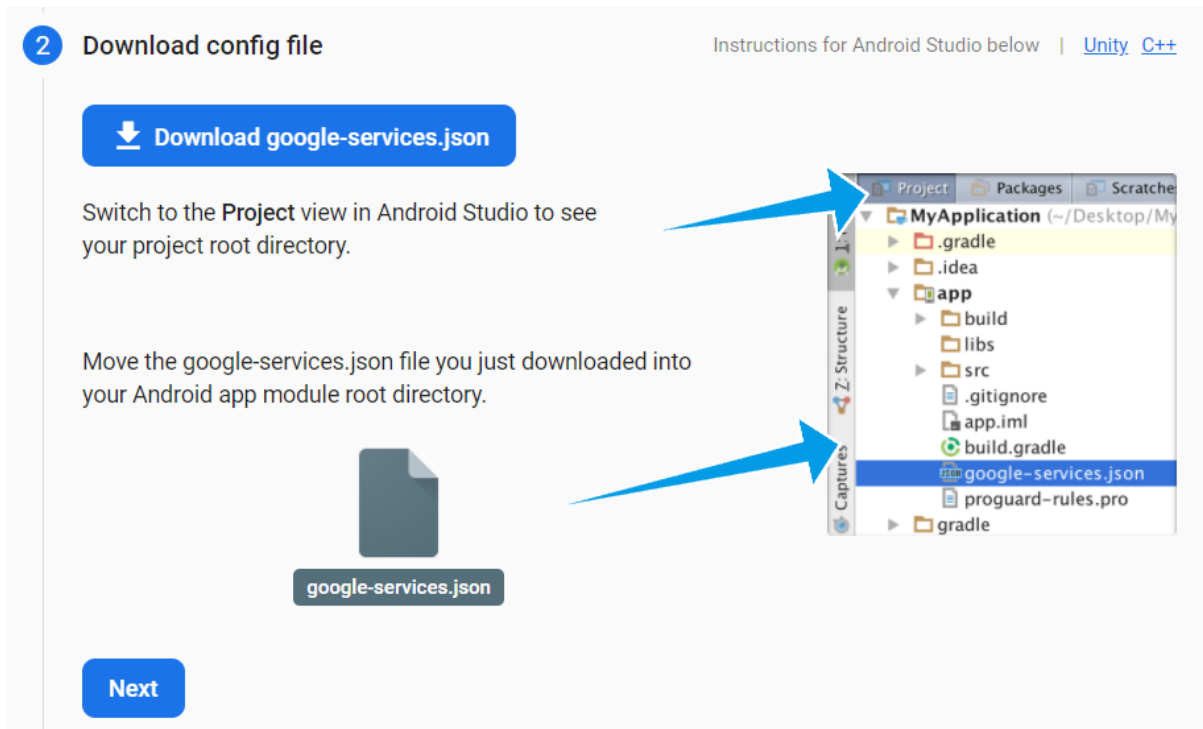
Hình 2.3. 4 Firebase tự động khởi tạo



Hình 2.3. 3 Firebase tự động khởi tạo



- Tải file google-services.json và đưa nó vào trong project



Hình 2.3.7 Cài đặt google services

- Thêm Firebase SDK vào project : Copy Firebase SDK vào Gradle của project

```
repositories {  
    google()  
    mavenCentral()  
}  
dependencies {  
  
    classpath 'com.google.gms:google-services:4.3.10'
```

Hình 2.3.8 Thêm Firebase SDK

```

apply plugin: 'com.android.application'
apply plugin: 'com.google.gms.google-services'

dependencies {

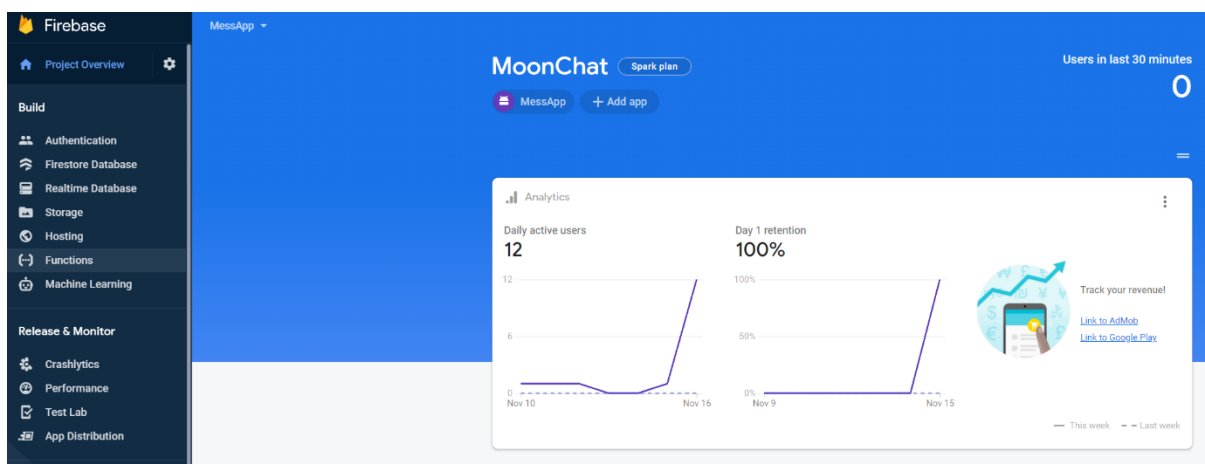
    implementation platform('com.google.firebase:firebase-bom:29.0.0')
    implementation 'com.google.firebase:firebase-analytics'
}

```

*Hình 2.3.9 Thêm Firebase SDK*

### 2.3.3 Hoàn tất cài đặt

Sau khi cài đặt môi trường, Firebase sẽ tự động kết nối với project, công việc của lập trình viên chỉ là sử dụng những công cụ có sẵn của Firebase để hỗ trợ tối ưu nhất ứng dụng của bản thân.



*Hình 2.3.10 Giao diện Analytics của Firebase*

## 2.4 Một số thư viện cần thiết trong ứng dụng

```
//Google Sign in
implementation 'com.google.android.gms:play-services-auth:19.2.0'

//RecyclerView
implementation 'androidx.appcompat:appcompat:1.3.1'
implementation 'androidx.recyclerview:recyclerview:1.2.1'
implementation 'com.google.android.material:material:1.4.0'

//CircleImage
implementation 'com.mikhaellopez:circularimageview:4.2.0'
implementation 'de.hdodenhof:circleimageview:3.1.0'
implementation 'com.github.bumptech.glide:glide:4.11.0'

//Notification
implementation 'com.squareup.retrofit2:retrofit:2.9.0'
implementation 'com.squareup.retrofit2:converter-gson:2.9.0'
```

*Hình 2.4.1 Thư viện cần thiết trong ứng dụng*

- Google Sign in : Thư viện đăng nhập thông qua chứng thực trực tiếp bằng Gmail của Google mà không cần mật khẩu.
- RecyclerView : Thư viện sử dụng RecyclerView – phiên bản nâng cao của ListView bởi tính linh hoạt của nó[5].
- CircleImage : Thư viện cho phép lập trình viên hiển thị hình ảnh dưới dạng hình tròn thay vì hiển thị thông thường.
- Push Notification : Thư viện hỗ trợ của FCM cho phép sử dụng thông báo đẩy ( push notification ).

## **Chương 3 : PHÂN TÍCH THIẾT KẾ HỆ THỐNG**

### **3.1 Khảo sát hệ thống**

Lần lượt tiến hành khảo sát những ứng dụng nhắn tin trực tuyến phổ biến như : Messenger, Zalo, Telegram, Whatsapp, ... trong đó phổ biến nhất là Messenger. Bên cạnh đó để phát triển UI phải tham khảo những thiết kế giao diện trên những trang web như Dribbble, Figma, Behance, ...

Qua khảo sát thực tế để đánh giá và phát triển những chức năng cơ bản và giao diện tối ưu nhất cho ứng dụng.

### **3.2 Phân tích nghiệp vụ hệ thống**

Qua những khảo sát trên, xây dựng một ứng dụng cơ bản với những chức năng như sau :

- Đăng ký, đăng nhập sử dụng email – password : Nếu chưa có tài khoản, người dùng thực hiện đăng ký với username, email, password. Sau đó đăng nhập với tài khoản đã được đăng ký.
- Đăng ký, đăng nhập với Google Account : Nếu chưa từng đăng nhập, sau khi chọn chức năng đăng nhập với Google Account, người dùng sẽ chọn tài khoản muốn đăng nhập, sau đó hệ thống sẽ tự động đăng ký tài khoản cho người dùng và tự động đăng nhập.
- Hiển thị người dùng : Trong mục Users hệ thống sẽ hiển thị những người dùng đã đăng ký ứng dụng, người sử dụng có thể chọn đối tượng và bắt đầu hội thoại.
- Hiển thị tin nhắn cũ : Trong mục Chats hệ thống sẽ hiển thị những user đã có tin nhắn với người dùng trước đây, đồng thời hiển thị tin nhắn mới nhất mà người dùng gửi đến.
- Hiển thị trạng thái người dùng : Nếu người dùng đang sử dụng ứng dụng, hệ thống sẽ hiển thị trạng thái màu xanh ứng với online và màu xám ứng với offline.
- Đổi ảnh đại diện, trạng thái : Người dùng có thể đổi ảnh đại diện hoặc trạng thái bằng cách nhấn vào ảnh đại diện trong mục Profile, hệ thống sẽ mở tới

Local file của điện thoại, sau đó người dùng có thể chọn ảnh có sẵn trong máy để đặt làm ảnh đại diện.

- Nhắn tin : Người dùng chọn đối tượng user muốn trò chuyện, ứng dụng sẽ mở cửa sổ nhắn tin cho 2 đối tượng người gửi và người nhận, người dùng nhập tin nhắn muốn gửi và chọn biểu tượng gửi. Tin nhắn sẽ được gửi tới người nhận ngay lập tức.
- Nhận thông báo tin nhắn mới : Nếu người dùng không sử dụng ứng dụng, khi có tin nhắn mới, ứng dụng sẽ tự động gửi tin nhắn tới bằng một thông báo đẩy ( push notification ) với tên người gửi và nội dung tin nhắn. Người dùng có thể nhấp vào thông báo để tới trực tiếp tin nhắn với người gửi.

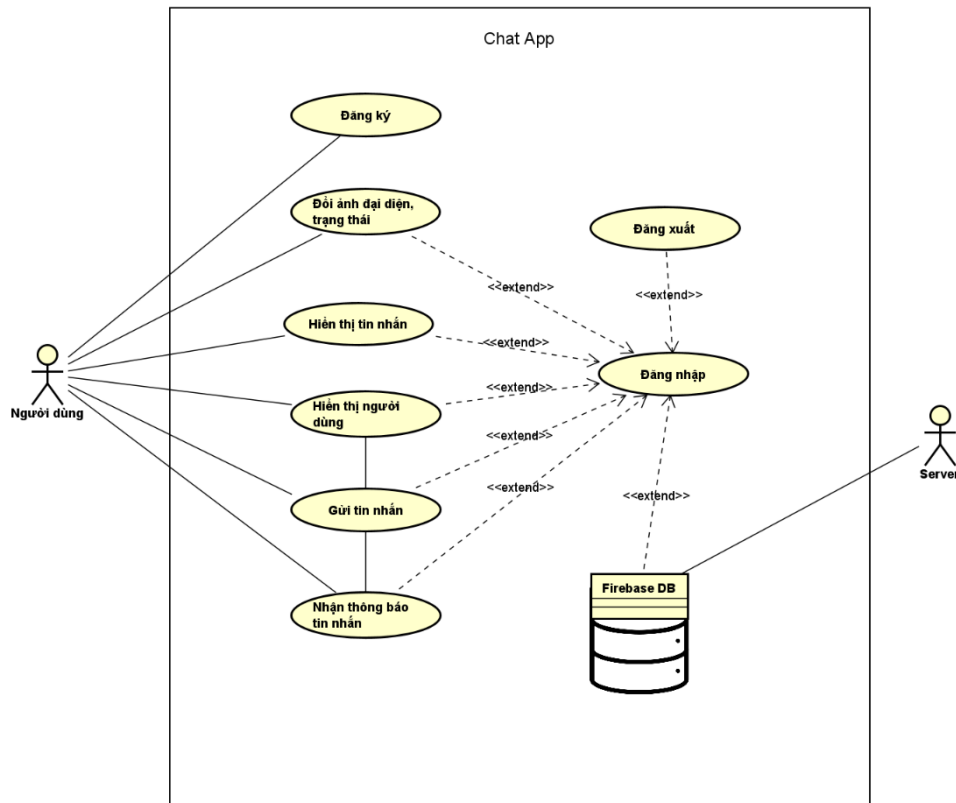
Bên cạnh đó, phía server Firebase cung cấp cho Admin những chức năng quản lý hệ thống như :

- Quản lý người dùng : quản lý tài khoản người dùng, hiển thị các phương thức đăng nhập mà người dùng sử dụng, hoặc thêm sửa xóa người dùng.
- Báo cáo thống kê : Admin sẽ xem được phân tích của ứng dụng bao gồm: lượt truy cập, tần suất sử dụng, lượng lưu trữ trong Storage hay Database, ...



### 3.3 Phân tích thiết kế hệ thống

#### 3.3.1 Lược đồ Usecase



3.3.1.1 Lược đồ Usecase

#### 3.3.2 Usecase đăng ký

- Mô tả : Người dùng thực hiện đăng ký thêm một tài khoản mới vào hệ thống
- Luồng hoạt động : Sau khi nhập thông tin đăng ký, hệ thống sẽ đưa người dùng tới cửa sổ chứng thực email, người dùng phải xác thực email. Hệ thống sẽ gửi tin nhắn về email của người dùng đăng ký, sau khi xác thực thành công, người dùng có thể đăng nhập vào ứng dụng.

#### 3.3.3 Usecase hiển thị người dùng

- Mô tả : Người dùng thực hiện xem những user có mặt trong hệ thống
- Luồng hoạt động : Sau khi đăng nhập thành công, người dùng có thể xem những Users đang sử dụng ứng dụng. Người dùng có thể chọn user muốn giao tiếp để thực hiện hành động trò chuyện.

#### 3.3.4 Usecase gửi tin nhắn

- Mô tả : Người dùng thực hiện gửi tin nhắn cho người đối tượng muốn gửi

- **Luồng hoạt động :** Sau khi chọn đối tượng muốn nhắn tin, người dùng sẽ được đưa tới cửa sổ nhắn tin. Tại đây người dùng có thể gửi tin nhắn cho đối tượng, hoặc nhận tin nhắn từ đối tượng đó. Bên cạnh đó, hệ thống còn hiển thị trạng thái tin nhắn đã được gửi ( delivered ) hay đã được người nhận xem hay chưa (seen).

### **3.3.5 Usecase nhận thông báo tin nhắn mới**

- **Mô tả :** Hệ thống gửi thông báo tin nhắn mới nếu người dùng không sử dụng ứng dụng
- **Luồng hoạt động :** Nếu người dùng thoát ra ngoài, khi nhận được tin nhắn mới, hệ thống sẽ gửi một thông báo đầy đủ âm thanh thông báo tới thanh thông báo trên điện thoại của người dùng, thông báo này sẽ hiển thị những nội dung chính : Tiêu đề, người gửi tin nhắn, nội dung tin nhắn. Người dùng có thể bấm vào thông báo để đi tới cửa sổ nhắn tin với người gửi.

### **3.3.6 Usecase hiển thị tin nhắn**

- **Mô tả :** Người dùng thực hiện xem những tin nhắn cũ của những đối tượng nhắn tin trước đó
- **Luồng hoạt động :** Sau khi gửi tin nhắn với một đối tượng bất kỳ, hệ thống sẽ chuyển người được nhắn tin qua cửa sổ Chats. Tại đây, người dùng có thể xem những đối tượng mà mình đã nhắn tin trước đó, đồng thời hệ thống sẽ hiển thị tin nhắn gần nhất mà người dùng nhắn với đối tượng.

### **3.3.7 Usecase Đổi ảnh đại diện, trạng thái**

- **Mô tả :** Người dùng thực hiện đổi ảnh đại diện mà mình mong muốn
- **Luồng hoạt động :** Khi người dùng tạo tài khoản mới đăng nhập lần đầu, hệ thống sẽ tự động để ảnh đại diện mặc định của hệ thống. Người dùng có thể thay đổi ảnh đại diện của riêng mình trong cửa sổ Profile, người dùng bấm vào ảnh đại diện, hệ thống sẽ đưa người dùng tới Local files trên máy, sau đó người dùng tiến hành chọn ảnh đại diện thích hợp, hệ thống sẽ tự động thay đổi ảnh đại diện và đồng bộ cho người dùng.

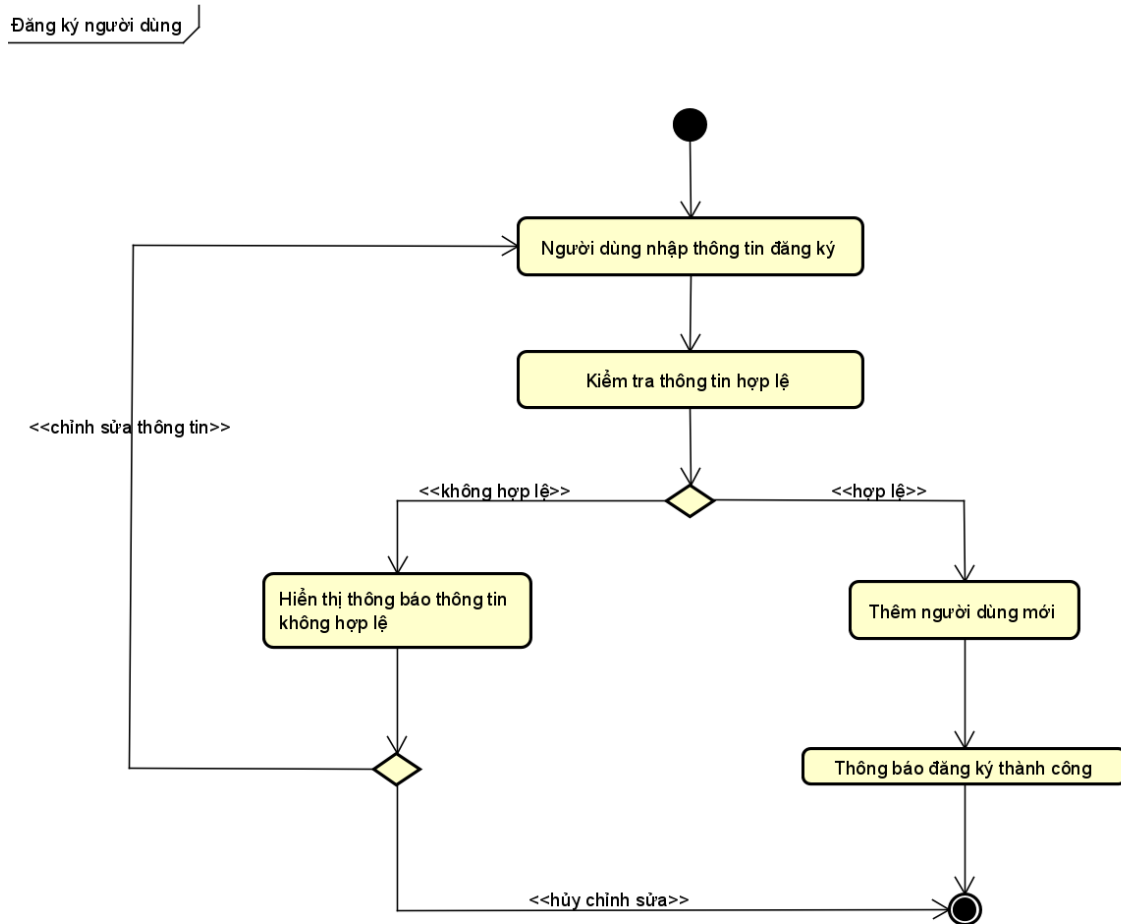
### **3.3.8 Usecase Firebase DB**

- **Mô tả :** Quản lý ứng dụng thông qua công cụ của Firebase

- Luồng hoạt động : Admin có thể quản lý người dùng, quản lý database, quản lý kho lưu trữ dữ liệu thông qua Firebase một cách trực quan nhất. Admin có thể xem phân tích số liệu, analytics mà Firebase tích hợp sẵn trong Overview để kiểm soát, tính toán và phân tích sử dụng ứng dụng một cách tối đa hóa nhất.

### 3.4 Sơ đồ hoạt động (Activity Diagram)

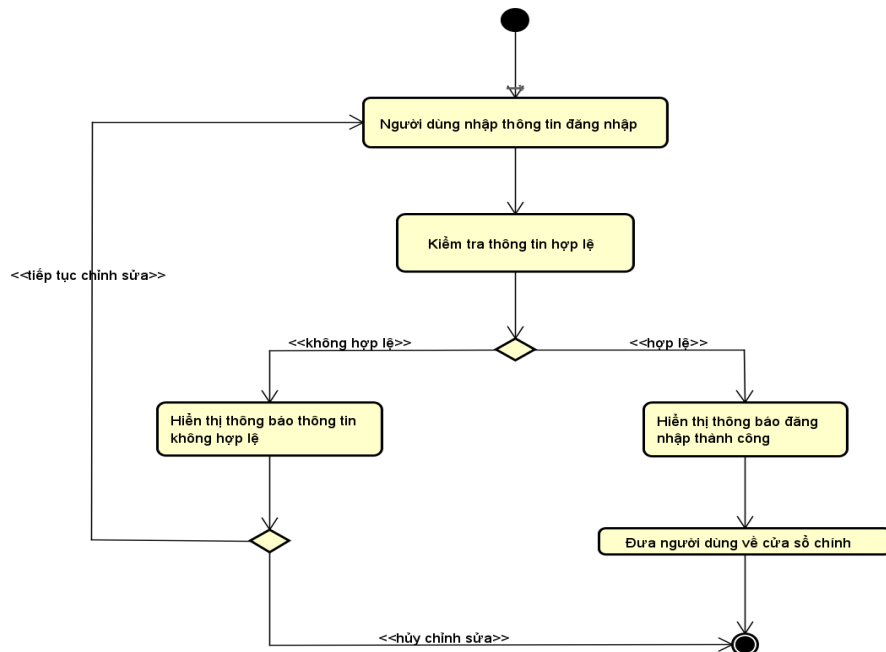
- Đăng ký người dùng :



Hình 3.4.1 Sơ đồ Đăng ký người dùng

- Đăng nhập bằng email-password:

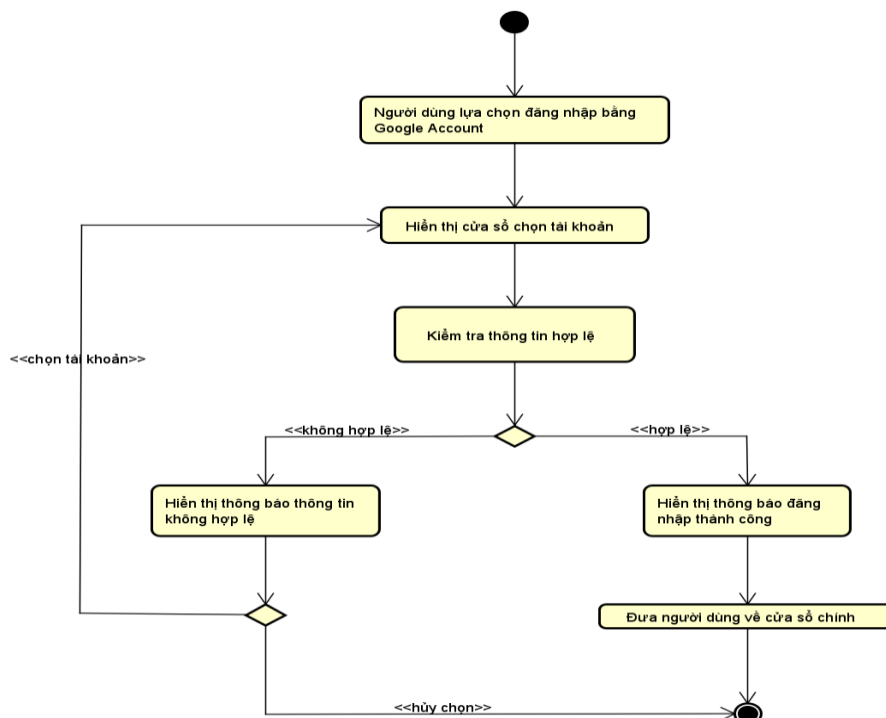
Đăng nhập bằng email-password



Hình 3.4. 2 Sơ đồ đăng nhập bằng email-password

- Đăng nhập bằng Google Account:

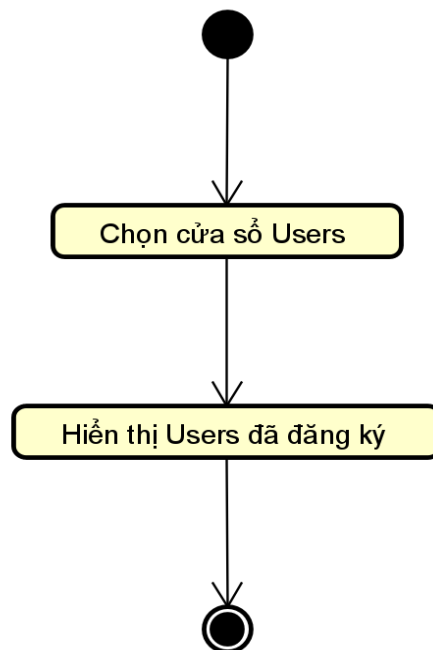
Đăng nhập bằng Google Account



Hình 3.4. 3 Sơ đồ đăng nhập bằng Google Account

- Xem các người dùng đã đăng ký :

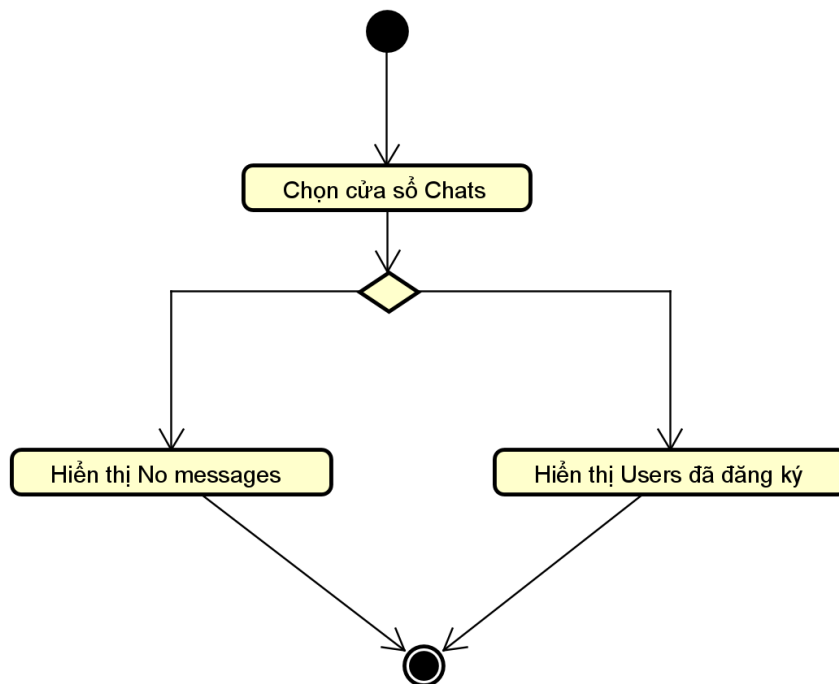
Xem các người dùng



*Hình 3.4. 4 Sơ đồ xem người dùng*

- Xem tin nhắn cũ

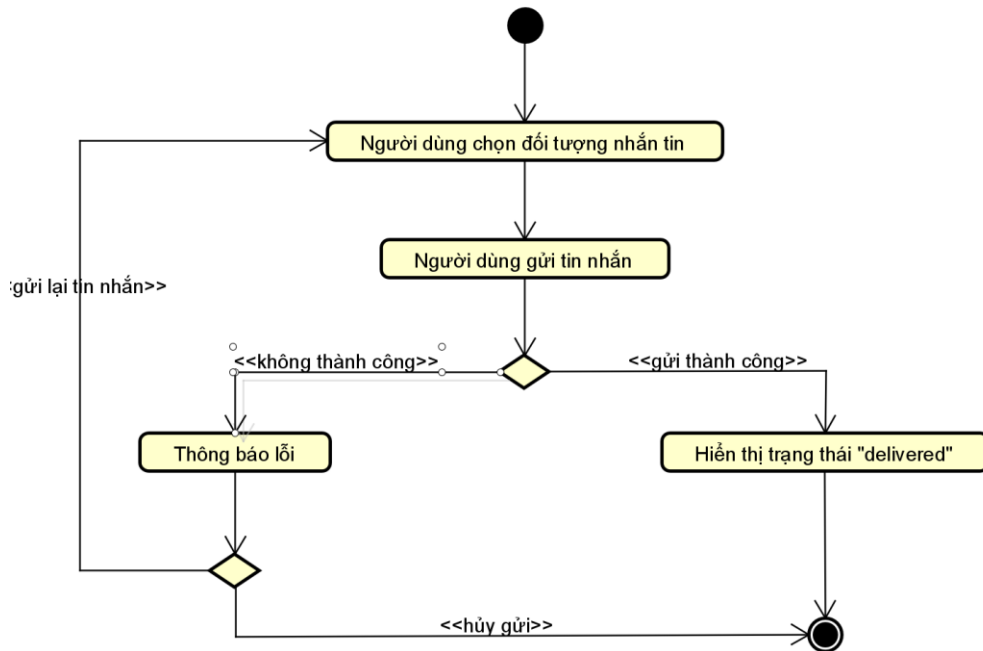
Xem tin nhắn



*Hình 3.4. 5 Sơ đồ hiển thị tin nhắn*

- Sơ đồ gửi tin nhắn :

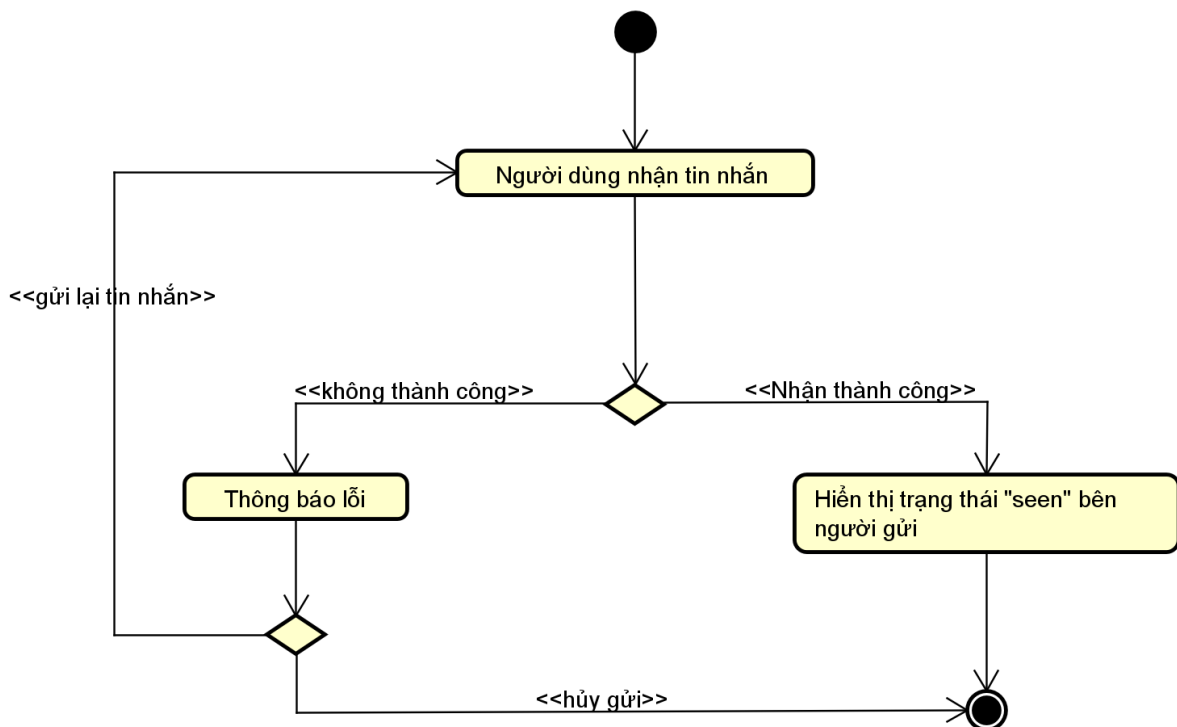
Gửi tin nhắn



Hình 3.4. 6 Sơ đồ gửi tin nhắn

- Sơ đồ nhận tin nhắn:

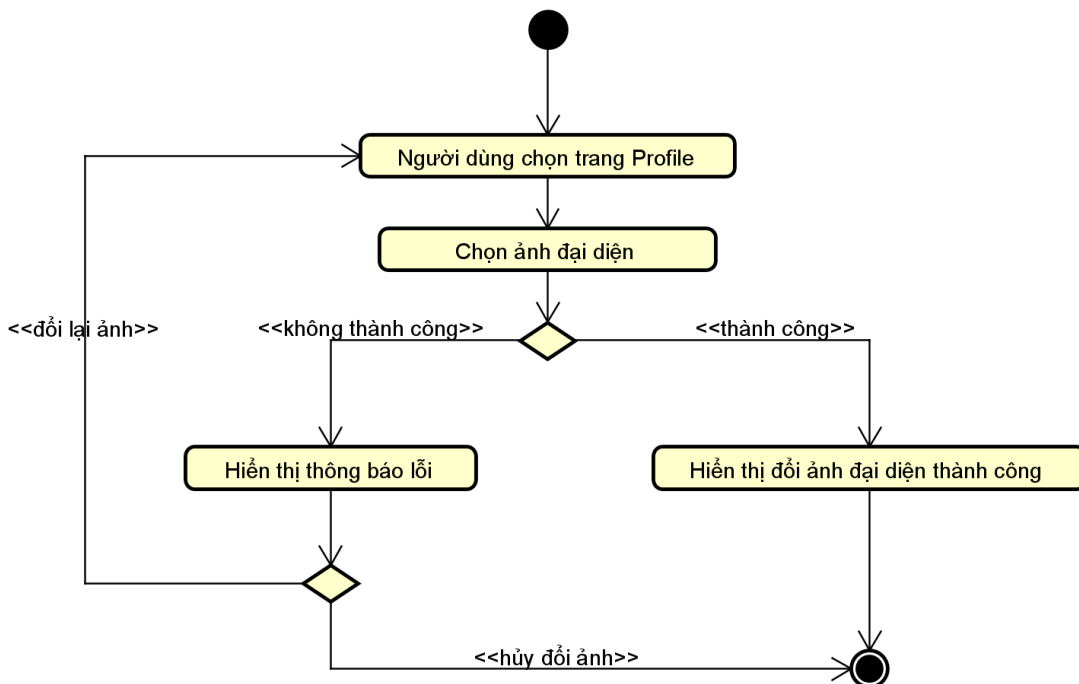
Nhận tin nhắn



Hình 3.4. 7 Sơ đồ nhận tin nhắn

- Sơ đồ đổi ảnh đại diện

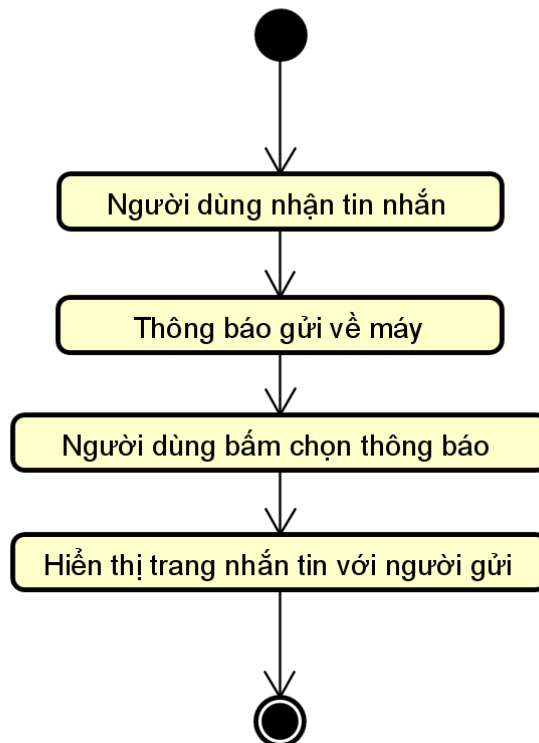
Đổi ảnh đại diện



Hình 3.4. 8 Sơ đồ nhận thông báo tin nhắn mới

- Sơ đồ nhận thông báo tin nhắn mới

Nhận thông báo tin nhắn mới



Hình 3.4. 9 Sơ đồ đổi ảnh đại diện



### 3.5 Thiết kế và xây dựng cơ sở dữ liệu

#### 3.5.1 Bảng 1 của hệ thống

- Bảng User lưu trữ thông tin của người dùng

user	
🔍 id	int
username	char
email	char
imageUrl	char
status	int

Hình 3.5.1. 1 Bảng User

#### 3.5.2 Bảng 2 của hệ thống

- Bảng Chat lưu trữ thông tin của mỗi tin nhắn

chat	
🔍 id	int
isSeen	int
sender	int
receiver	int
message	int

Hình 3.5.1. 2 Bảng Chat

#### 3.5.3 Bảng 3 của hệ thống

- Bảng chatList lưu trữ danh sách người nhận tin nhắn của người dùng

chatList	
<u>id</u>	int
user_send	int
user_receive	int

Hình 3.5.1. 3 Bảng ChatList

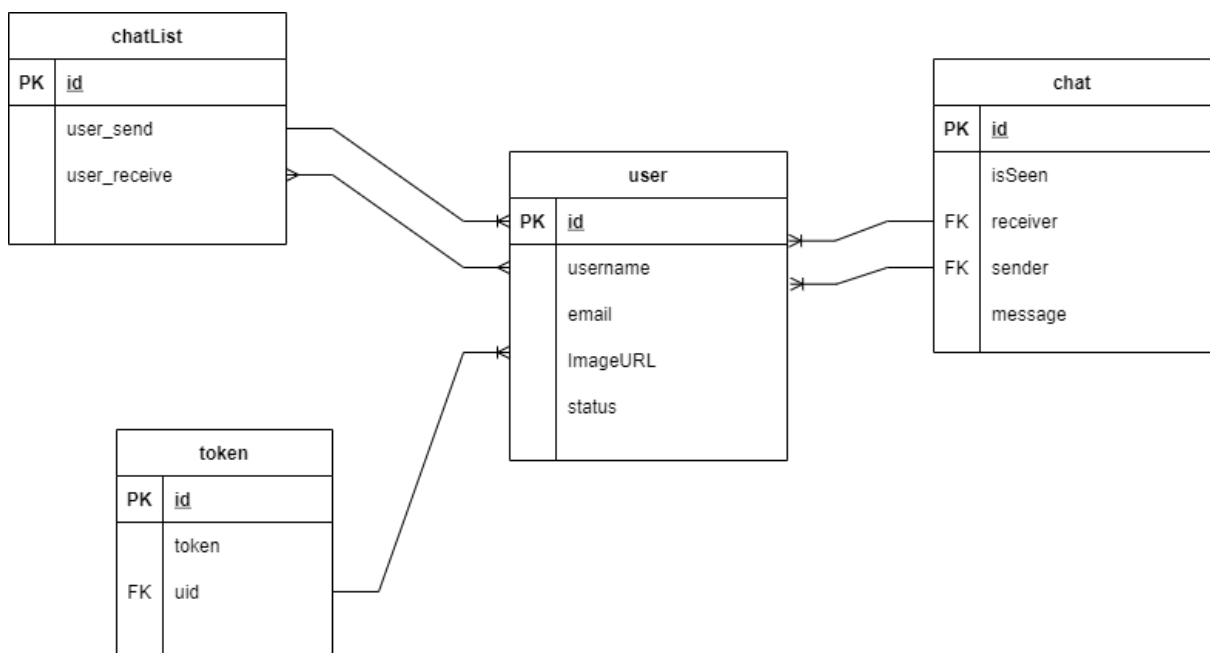
### 3.5.4 Bảng 4 của hệ thống

- Bảng Token lưu trữ thông tin token của từng người dùng

token	
<u>id</u>	int
token	int
uid	int

Hình 3.5.1. 4 Bảng Token

## 3.6 Sơ đồ ER Diagram:



Hình 3.6. 1 Sơ đồ ERD

## Chương 4 : THỰC NGHIỆM

### 4.1 Cấu hình cài đặt

#### 4.1.1 Java JDK

- Yêu cầu phần cứng tối thiểu:
  - Hệ điều hành : Window Vista, Mac OS X 10.9, Linux 5.5
  - Processor : Pentium 2 Processor
  - RAM : 128 MB
  - Bộ nhớ : 126 MB
  - Browsers : IE9

#### 4.1.2 Android Studio

- Yêu cầu phần cứng tối thiểu:
  - Hệ điều hành : Window 7 ( Emulator chỉ hỗ trợ cho 64-bit ), MacOS X 10.10, Linux GNOME hoặc KDE desktop
  - RAM : Tối thiểu 4 GB RAM ( khuyến nghị 8 GB)
  - Bộ nhớ : 4GB
- Để sử dụng Emulator, cấu hình khuyến nghị là :
  - SDK Tool 26.1.1 hoặc cao hơn
  - Vi xử lý 64-bit
  - Vi xử lý Intel hỗ trợ Intel VT-x, Intel 64

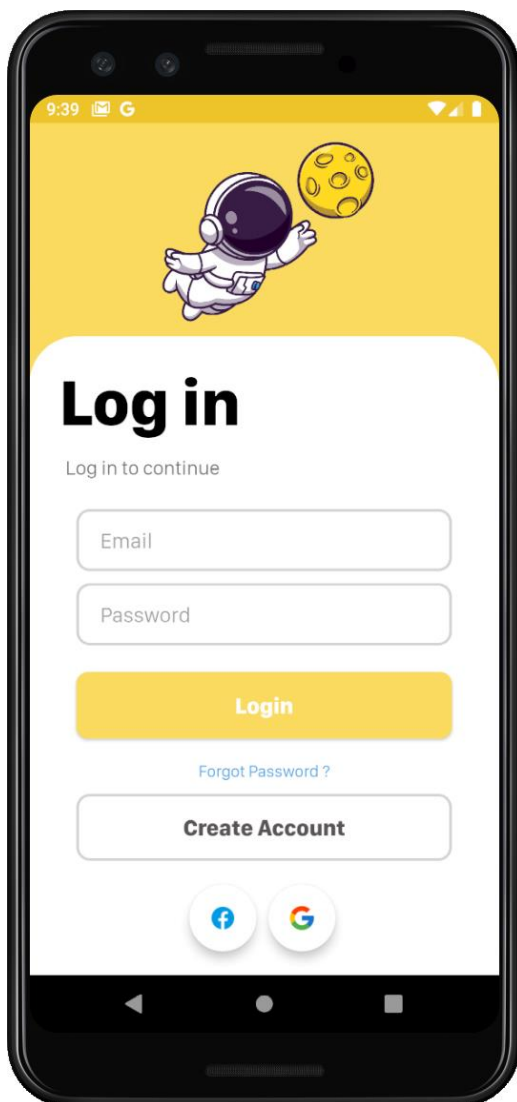
#### 4.1.3 Firebase

Cấu hình và setup Firebase như mục 2.3 để liên kết với project một cách chính xác và tối ưu nhất

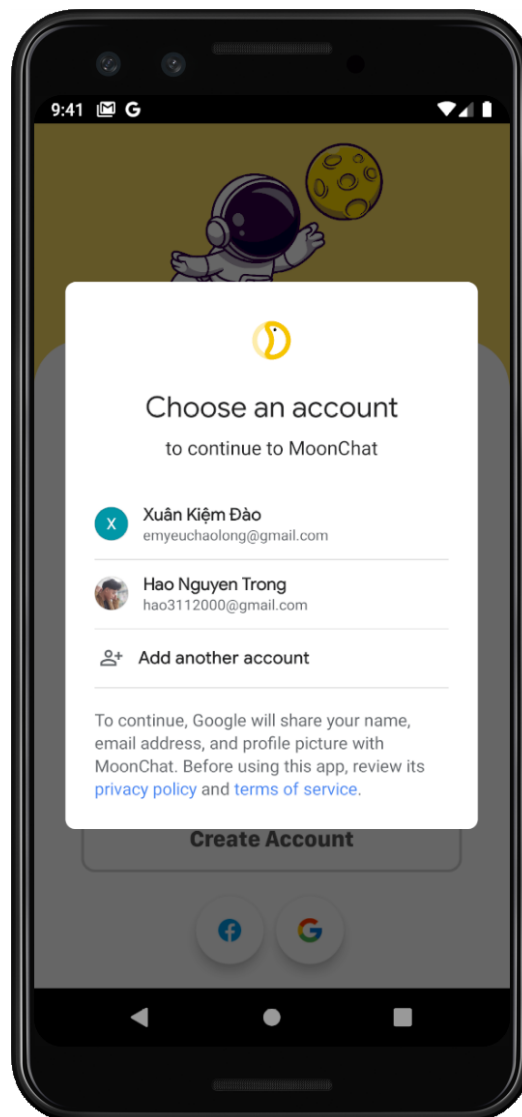
### 4.2 Giao diện ứng dụng

Ứng dụng được phát triển trên nền tảng android, cho nên giao diện tùy biến dựa trên các kích thước màn hình máy khác nhau. Khi người dùng sử dụng, hệ thống luôn có tiêu đề để người dùng có thể nhận biết vị trí một cách trực quan và dễ dàng.

- Trang đăng nhập : cửa sổ đầu tiên mở ra khi người dùng sử dụng ứng dụng.

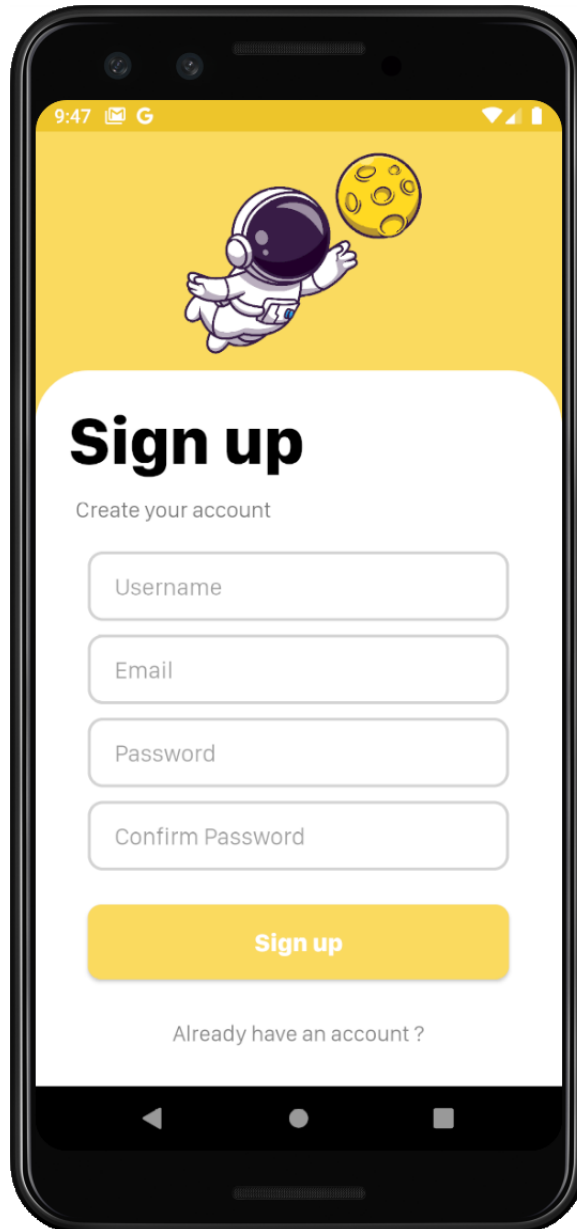


Hình 4.2.2 Đăng nhập tài khoản với Email-Password



Hình 4.2.1 Đăng nhập tài khoản với Google Account

- Trang đăng ký : Người dùng chọn Create Account và trang đăng ký mở ra để người dùng có thể đăng lý qua Email-Password



9:47

**Sign up**

Create your account

Username

Email

Password

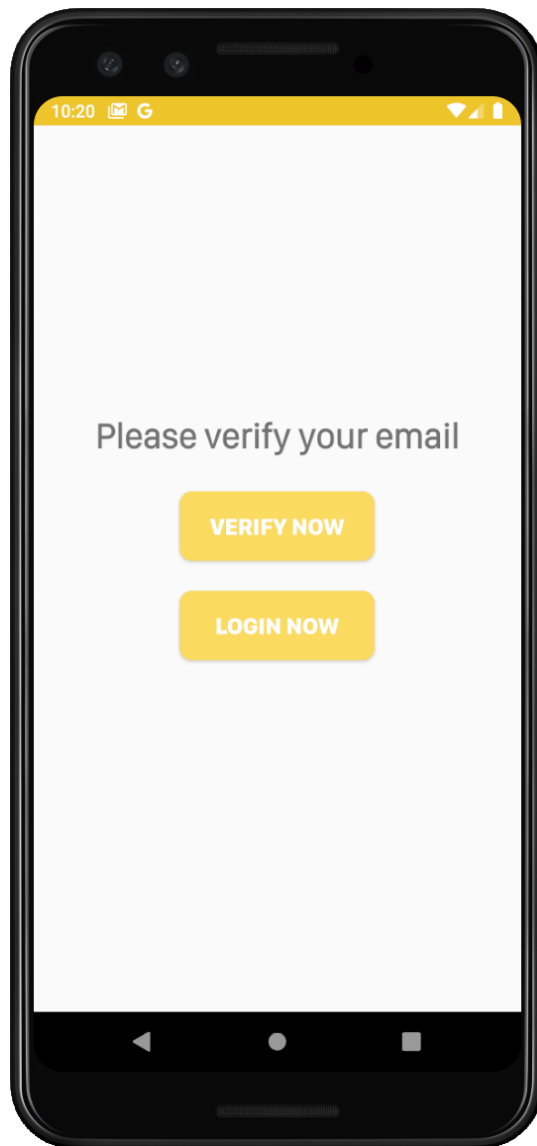
Confirm Password

**Sign up**

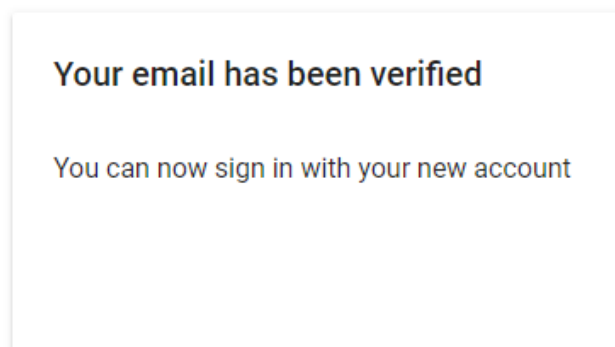
Already have an account ?

*Hình 4.2.3 Đăng ký tài khoản cho người dùng mới*

- Trang xác thực tài khoản : Khi người dùng đăng ký bằng Email, hệ thống sẽ bắt người dùng xác thực tài khoản của mình. Sau đó một tin nhắn sẽ được gửi về mail đăng ký của người dùng để tiến hành xác minh tài khoản.

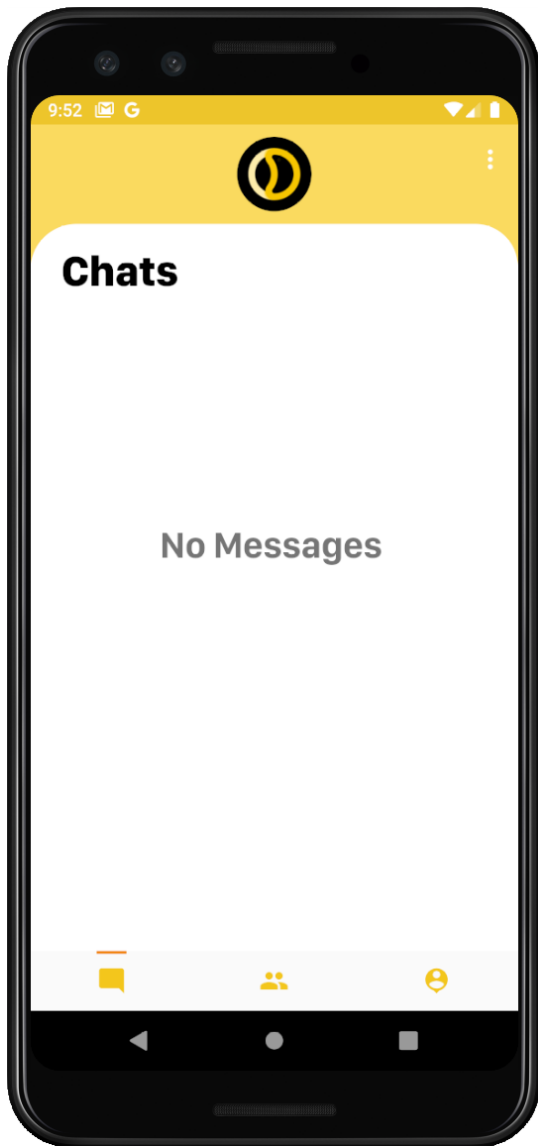


*Hình 4.2.4 Giao diện xác minh tài khoản*

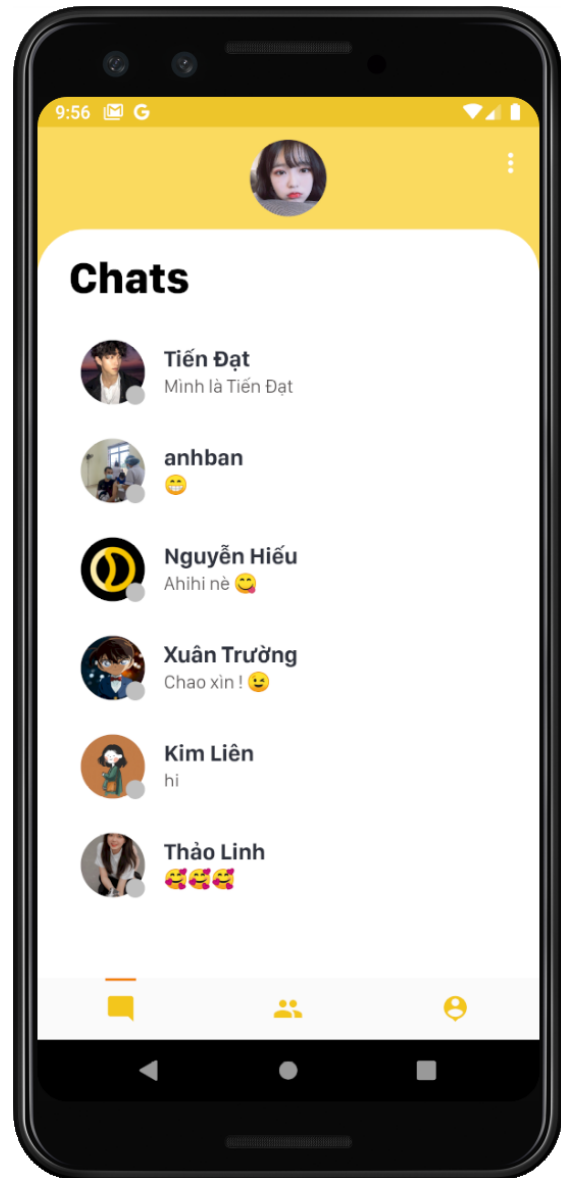


*Hình 4.2.5 Xác minh tài khoản từ email*

- Trang chính : Khi người dùng mới đăng nhập vào ứng dụng, hệ thống sẽ mở cửa sổ màn hình chính, nơi lưu trữ tin nhắn cũ mà người dùng đã giao tiếp trước đó. Nếu là người dùng mới, hệ thống sẽ hiển thị không có tin nhắn.

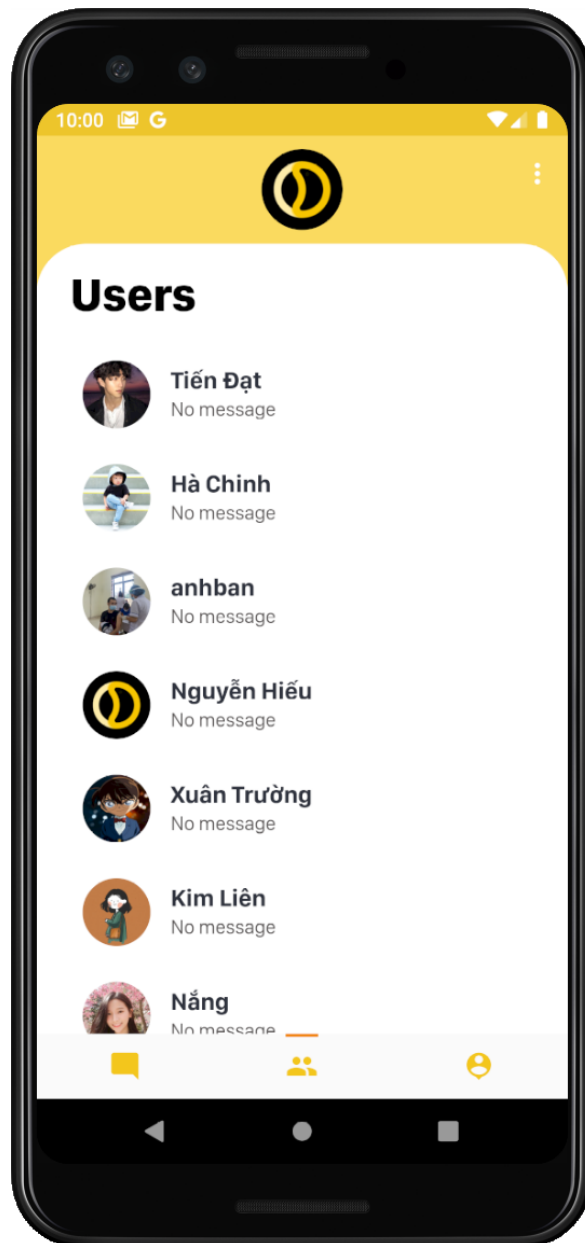


Hình 4.2.7 Giao diện tin nhắn khi chưa có tin nhắn nào



Hình 4.2.6 Giao diện tin nhắn khi có tin nhắn cũ

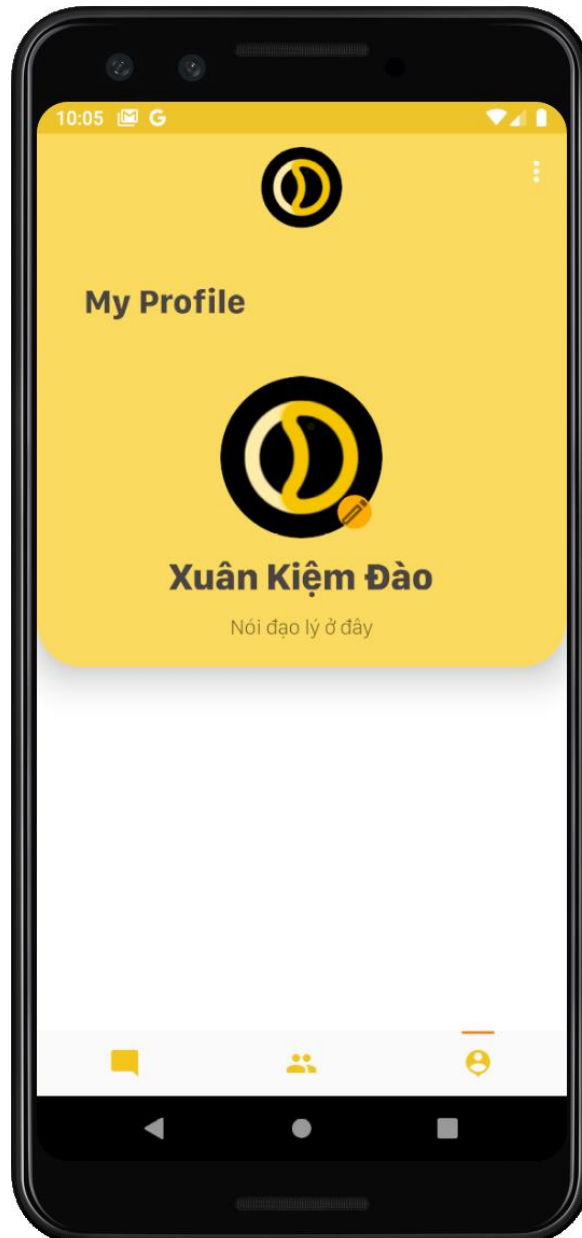
- Trang hiển thị User : trang hiển thị những User đã đăng ký và sử dụng ứng dụng.



Hình 4.2.8 Giao diện hiển thị Users

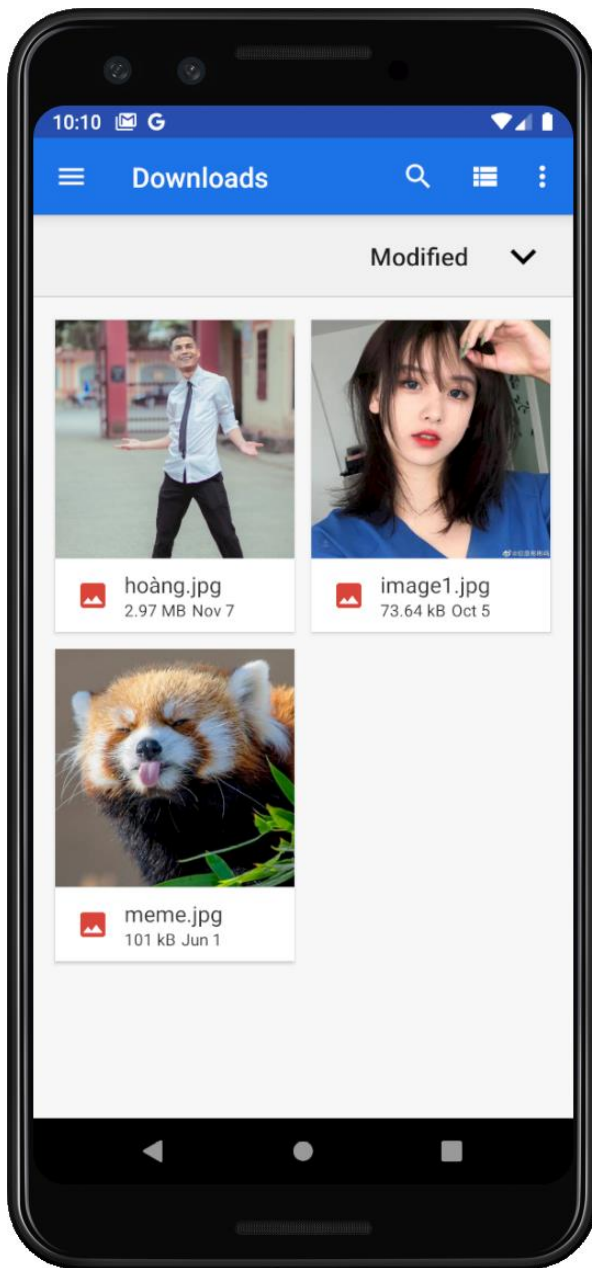


- Giao diện Profile :

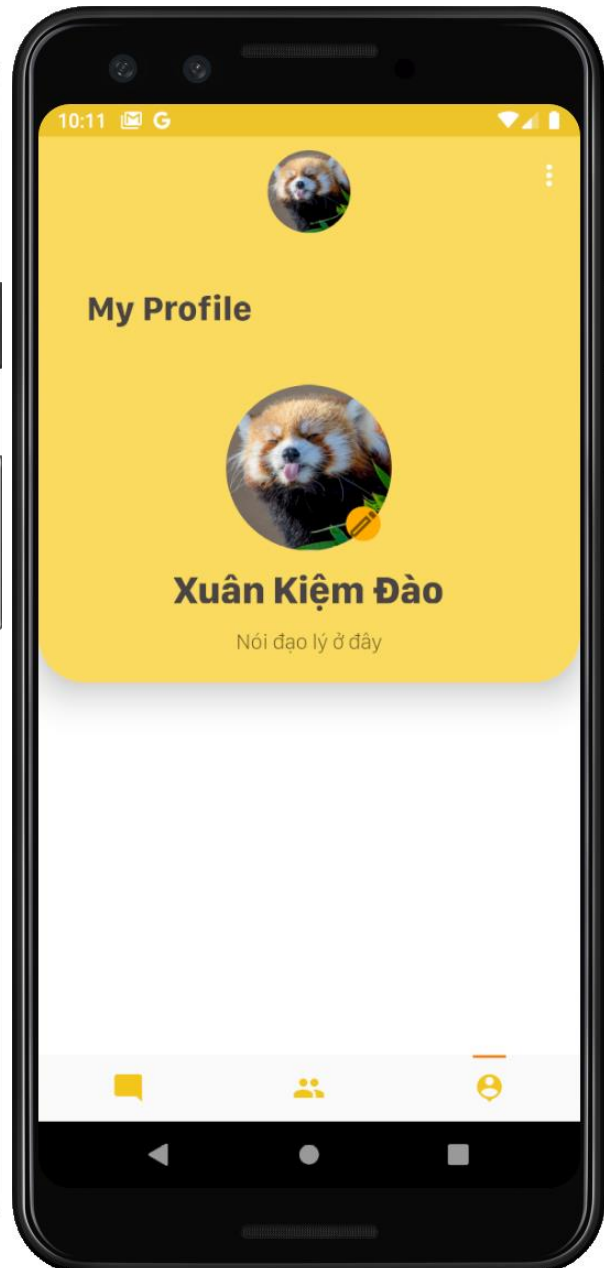


Hình 4.2.9 Giao diện Profile

- Thay đổi ảnh đại diện : Người dùng bấm chọn vào ảnh đại diện, hệ thống sẽ cho phép người dùng chọn ảnh đại diện mới của riêng mình

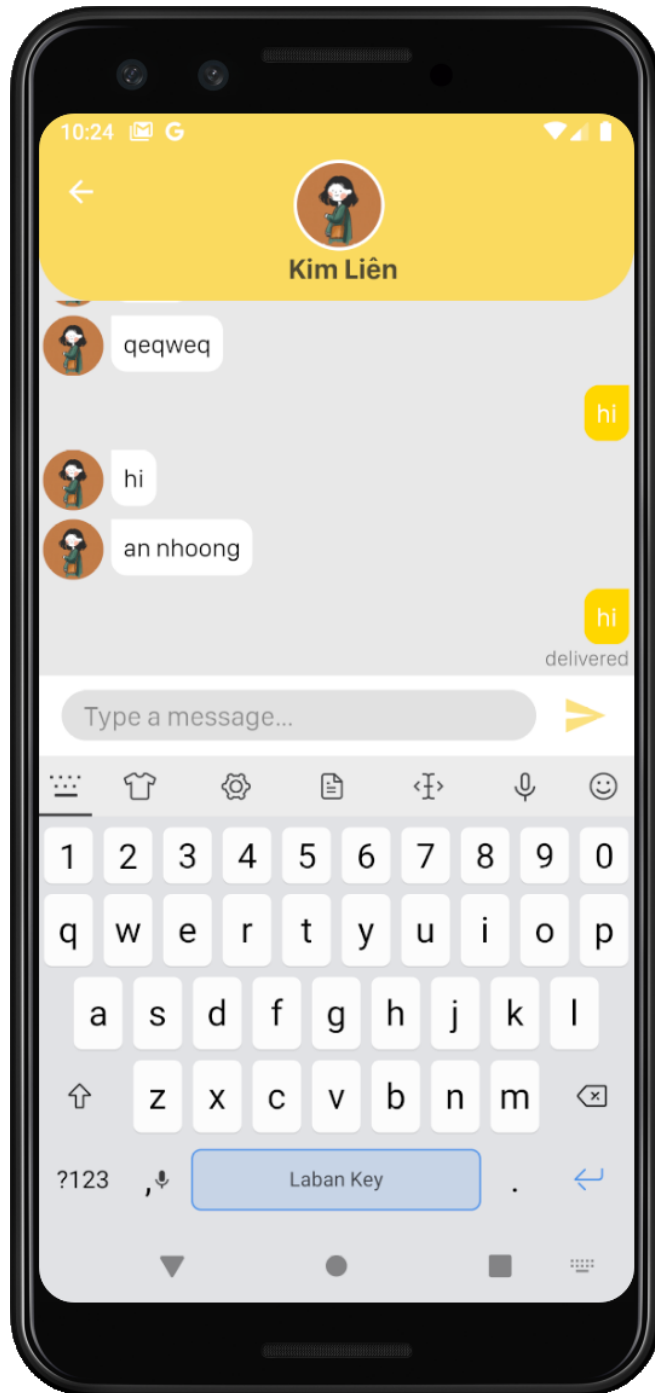


Hình 4.2.11 Chọn ảnh đại diện trong LocalFiles



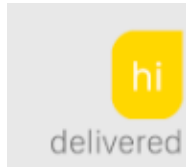
Hình 4.2.10 Đặt ảnh đại diện mới

- Giao diện nhắn tin : Người dùng chọn người dùng muốn trò chuyện, hệ thống sẽ đưa người dùng tới giao diện nhắn tin.

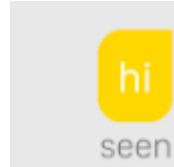


*Hình 4.2.12 Giao diện nhắn tin chính*

- Trạng thái tin nhắn : tin nhắn sau khi gửi sẽ ở trạng thái “delivered”, nếu người nhận xem tin nhắn thì tin nhắn sẽ đc chuyển qua “seen” (đã xem)



*Hình 4.2.14 Trạng thái "delivered"*



*Hình 4.2. 13 Trạng thái "seen"*

- Trạng thái người dùng : khi người dùng sử dụng ứng dụng, thanh trạng thái của người dùng sẽ hiện xanh thông báo rằng người dùng đang trực tuyến, ngược lại trạng thái của người dùng sẽ hiện xám



**Trọng Hào**  
hi

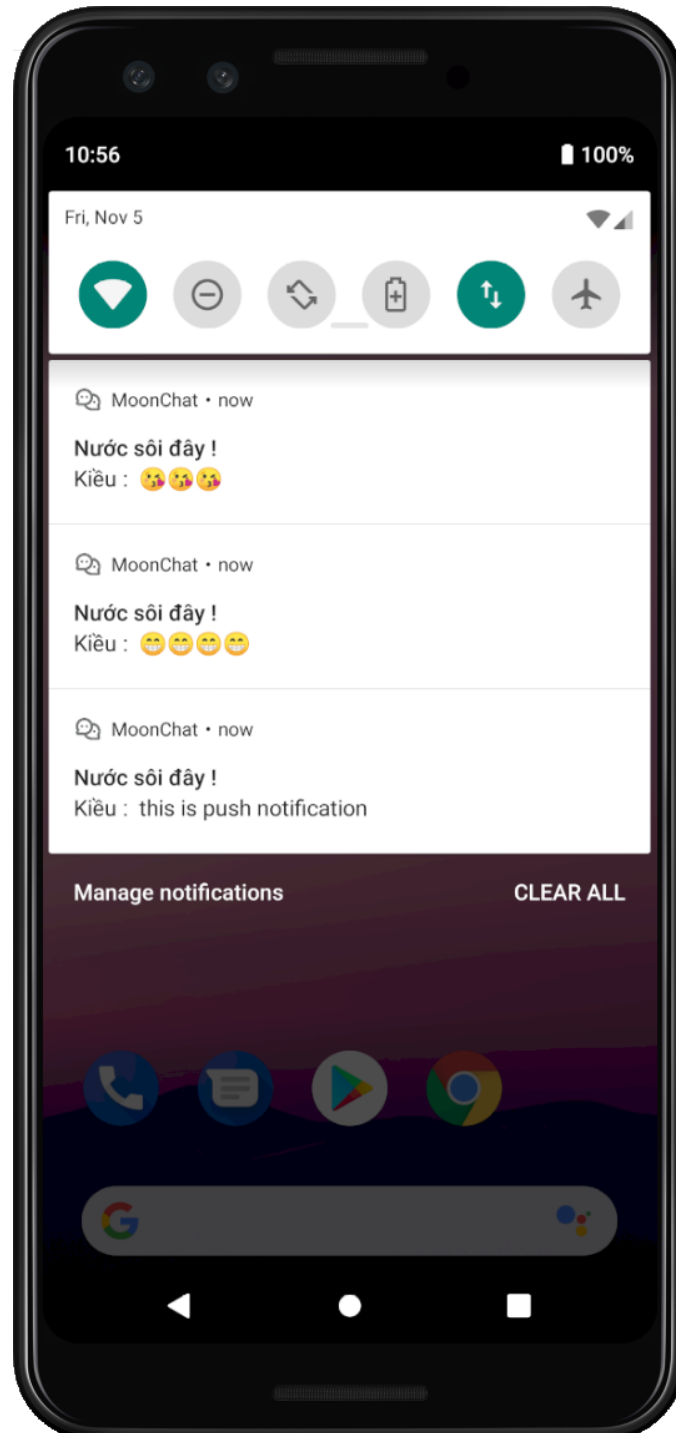
*Hình 4.2.15 Trạng thái người dùng offline*



**Trọng Hào**  
hi

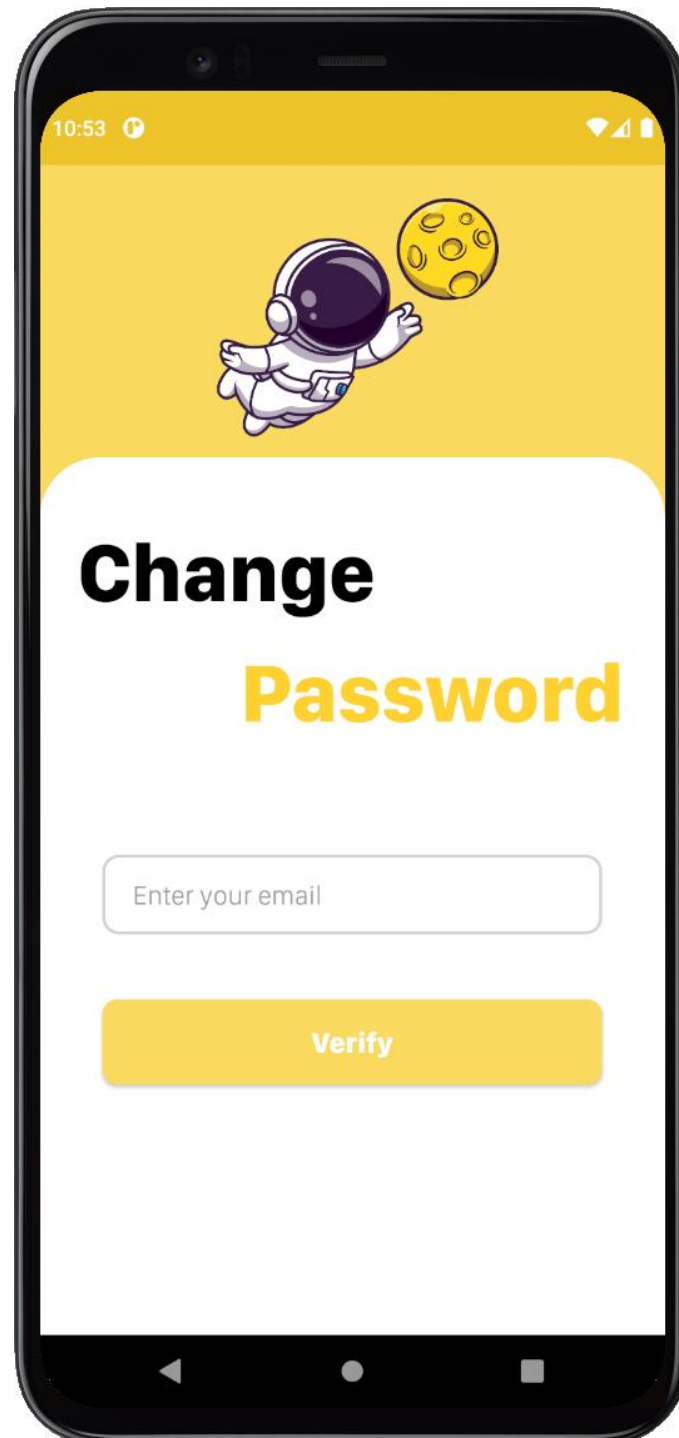
*Hình 4.2.16 Trạng thái người dùng online*

- Thông báo tin nhắn mới : Khi người dùng không sử dụng ứng dụng, thông báo sẽ đc gửi tới khi có tin nhắn mới. Người dùng có thể bấm vào để đi đến cuộc trò chuyện



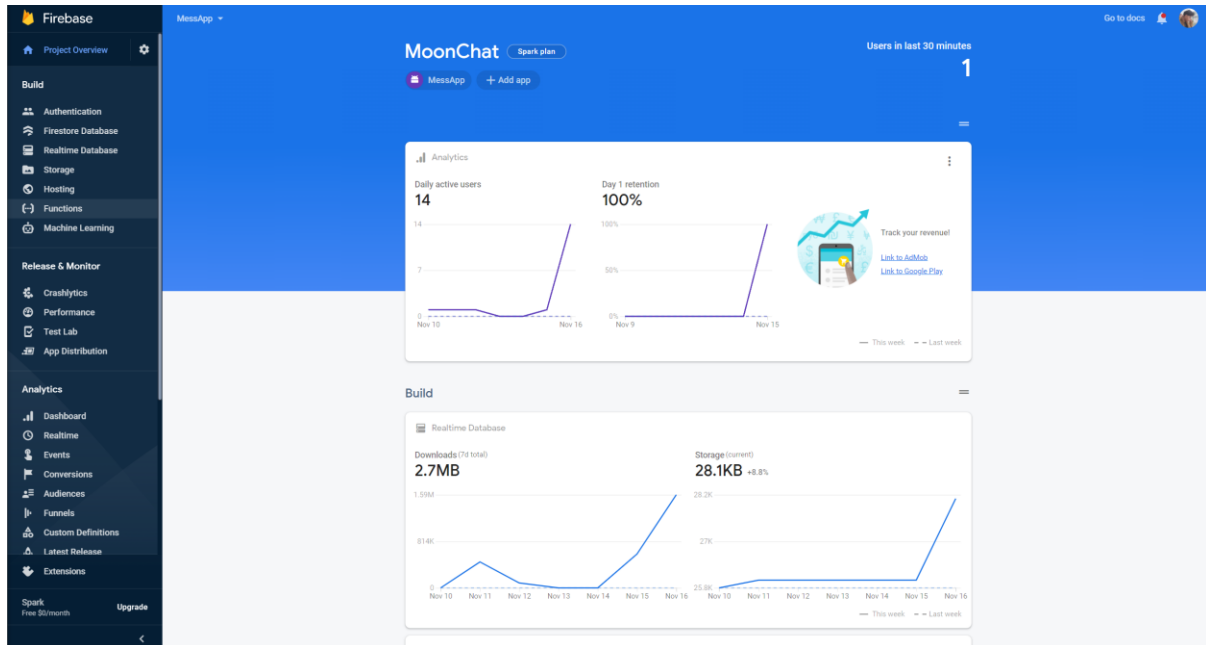
Hình 4.2.17 Thông báo tin nhắn mới

- Thay đổi mật khẩu : Chức năng thay đổi mật khẩu cho tài khoản của người dùng



Hình 4.2.18 Thay đổi mật khẩu

- Bên cạnh đó, giao diện phân tích và quản lý phía server Firebase được thể hiện một cách trực quan phục vụ cho việc xử lý thông tin của admin :



Hình 4.2.19 Giao diện Analytics của Firebase

- Admin có thể sửa xóa thông tin người dùng:

Search by email address, phone number, or user UID

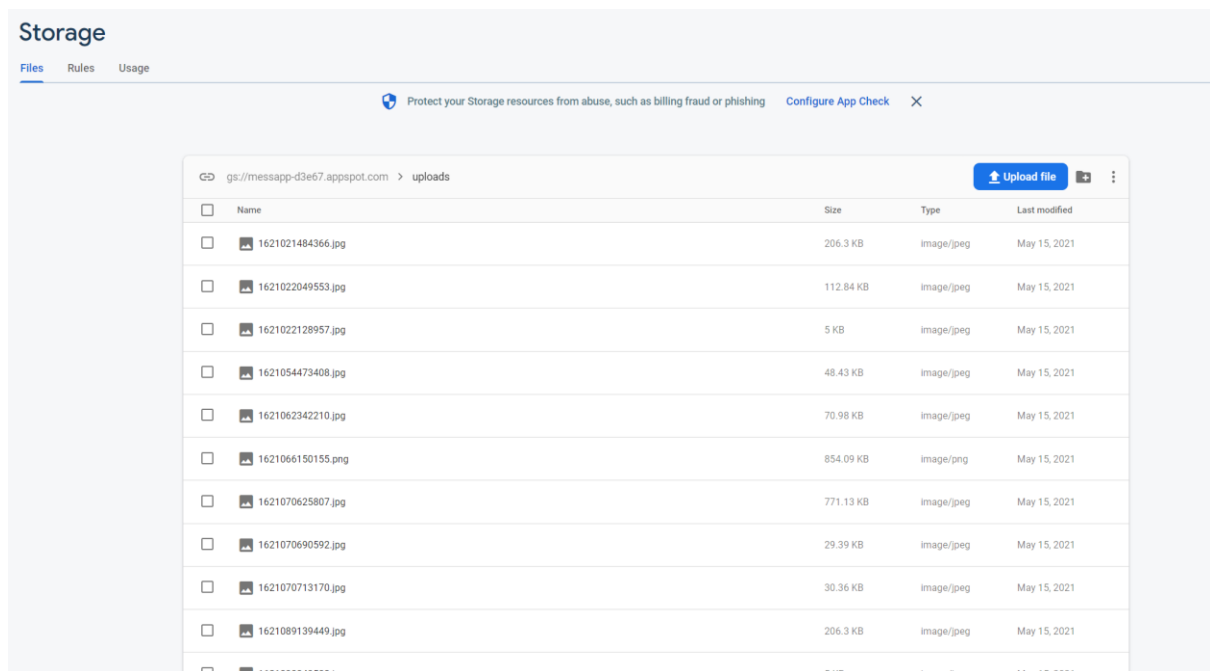
Add user

Identifier	Providers	Created <div></div>	Signed In	User UID	
erislive3112k@gmail.com	<div></div>	Nov 8, 2021	Nov 8, 2021	P1JIH4Ff1cDsQ5zgmFzi2QXZ7	<div>Reset password</div>
nguyenquanghieu0408200...	<div></div>	Nov 5, 2021	Nov 5, 2021	F0HakLp9UcSPNbwuw336Ga5	<div>Disable account</div>
emyeuchaolong@gmail.com	<div></div>	Nov 5, 2021	Nov 5, 2021	kYvYpSTzagYPBv2YwnjzbHZSnLX2	<div>Delete account</div>

- Reset password
- Disable account
- Delete account

Hình 4.2.20 Giao diện Firebase Authentication

- Quản lý các tập tin được tải lên :



Hình 4.2.21 Giao diện Firebase Cloud Storage



## **Chương 5 : KẾT LUẬN**

### **5.1 Kết quả đạt được**

Dự án lần này đã hoàn thành một cách cơ bản các chức năng của một ứng dụng nhắn tin cơ bản. Bắt đầu từ việc đăng ký đăng nhập, thể hiện thông tin người dùng, thay đổi thông tin, giao tiếp trò chuyện giữa người dùng, thông báo tin nhắn, ... để có thể xem như là một ứng dụng nhắn tin có thể hoạt động được tốt chức năng của mình.

Thêm vào đó là sự nghiên cứu về Firebase để tích hợp vào dự án, kết hợp Realtime Database, Cloud Storage hay Cloud Messaging một cách mượt mà trong ứng dụng này. Những việc này có thể nhỏ nhưng đó là bước đệm để thuận tiện cho việc phát triển dự án xa hơn trong tương lai sau này.

### **5.2 Những mặt hạn chế**

Việc làm đồ án với tư cách cá nhân hạn chế khá nhiều khả năng của bản thân cũng như tối ưu về mặt thời gian và hiệu năng để cho ra kết quả tốt nhất, dự án lần này còn có nhiều thiếu sót:

- Chức năng của ứng dụng còn đơn giản, thiếu sót về nhiều chức năng bản thân mong muốn nhưng bị hạn chế như : gửi ảnh, gửi file, tạo nhóm chat, tính năng kết bạn với từng người dùng,...
- Ứng dụng còn nhiều lỗi chưa khắc phục được như : tính năng Push Notification chưa được tối ưu, xử lý chậm, tính năng online, offline cũng hoạt động không ổn định.
- Liên kết với Firebase còn nhiều lỗi chưa được tối ưu
- Giao diện chưa có sự thống nhất khi người dùng thay đổi thiết bị có kích thước màn hình khác nhau.

Những thiếu sót này hoàn toàn có thể khắc phục được nên đây là điểm hạn chế cố hữu trong dự án lần này.

### **5.3 Kiến nghị và hướng phát triển**

Để phát triển ứng dụng này nói riêng cũng như kỹ năng của bản thân cần phải khắc phục được một cách tốt nhất những mặt hạn chế ở phía trên.

Những chức năng được nêu ở phía trên là cần có trong ứng dụng lần này nhưng bị hạn chế về nhiều mặt nên bản thân chưa thể củng cố thêm những điều cần làm ấy. Nhận thức rõ được vấn đề và định hướng được hướng phát triển cho mình, tuy nhiên cần thời gian để có thể nghiên cứu và tìm tòi phát triển những yêu cầu trên.

Tiếp thu, tìm tòi, học hỏi thêm nhiều kiến thức về mảng Android hơn nữa là điều cần làm, đầu tư thời gian để cải tiến những kỹ năng mà mình mong muốn để càng ngày càng hoàn thiện bản thân mình.

## TÀI LIỆU THAM KHẢO

- [1] Edpresso Team, “What is Firebase?”, [Trực tuyến]. Địa chỉ: [What is Firebase? \(educative.io\)](https://www.educative.io/what-is-firebase) . [Truy cập 05/11/2021].
- [2] Firebase, “Firebase Realtime Database”, [Trực tuyến]. Địa chỉ: [Firebase Realtime Database | Firebase Documentation \(google.com\)](https://firebase.google.com/docs/database/realtime). [Truy cập 05/11/2021].
- [3] Android Studio, “Meet Android Studio”, [Trực tuyến]. Địa chỉ : [Meet Android Studio | Android Developers](https://developer.android.com/studio). [Truy cập 05/11/2021].
- [4] XuanThuLab, “Hiểu về View, ViewGroup, Layout trong Android” [Trực tuyến]. Địa chỉ : [Hiểu về View, ViewGroup, Layout trong Android \(xuanthulab.net\)](https://xuanthulab.net/hiu-vue-viewgroup-layout-trong-android/). [Truy cập 05/11/2021].