

TRƯỜNG ĐẠI HỌC MỞ THÀNH PHỐ HỒ CHÍ MINH
KHOA CÔNG NGHỆ THÔNG TIN



BÁO CÁO GIỮA KỲ
ĐỀ TÀI: ỨNG DỤNG CHAT TRỰC TUYẾN

Môn học: LẬP TRÌNH TRÊN THIẾT BỊ DI ĐỘNG

GVHD: Dương Thái Bảo

LỚP: DH22CS02

SINH VIÊN THỰC HIỆN:

Nguyễn Lê Thanh Thắng – 2251012127

Mai Thị Hồng Duy – 2251012047

Đinh Bích Tiên - 2251012132

Thành phố Hồ Chí Minh
12/2024

LỜI CẢM ƠN

Kính gửi thầy,

Nhóm em xin chân thành cảm ơn thầy đã dành thời gian để đọc và xem xét báo cáo của chúng em. Mặc dù đề tài đã được hoàn thành đến một mức độ nhất định và chúng em đã cố gắng thực hiện đúng yêu cầu, nhưng nhóm nhận thấy vẫn còn một số điểm sai sót và chưa hoàn thiện. Qua quá trình thực hiện, chúng em đã nhận ra rằng có những vấn đề cần phải cải thiện để đạt được kết quả tốt hơn và đảm bảo chất lượng cao hơn.

Chúng em rất mong nhận được sự chỉ bảo và những góp ý từ thầy để có thể khắc phục những thiếu sót và hoàn thiện hơn nữa đề tài của mình. Nhóm em xin cảm ơn thầy và rất mong nhận được những ý kiến đóng góp từ thầy.

MỤC LỤC

CHƯƠNG 1: TỔNG QUAN	4
1.1. Giới thiệu đề tài.....	4
1.2. Lý do chọn đề tài.....	4
1.3. Yêu cầu đề tài	5
1.4. Mục tiêu và phạm vi	5
1.5. Phương pháp nghiên cứu	5
CHƯƠNG 2: CƠ SỞ LÝ THUYẾT	7
2.1. Giới thiệu về Android Studio.....	7
2.1.1. Giới thiệu	7
2.1.2 Cài đặt Android Studio	7
2.1.3 Cấu trúc của Project trong Android Studio.....	7
2.2 Giới thiệu về Firebase.....	9
2.2.1 Email/Password Authentication	10
2.2.2. Firestore.....	10
2.2.3 RealtimeDatabase	11
2.3. Các công nghệ khác	11
2.3.1 Glide	11
2.3.2 Clouinary.....	12
CHƯƠNG 3: ĐỀ TÀI.....	13
3.1. Các bước thực hiện đề tài.....	13
3.2. Mô tả chức năng của đề tài	14
3.3. Mô tả dữ liệu	35
3.4. Mô tả sản phẩm.....	36
CHƯƠNG 4: KẾT LUẬN	37
4.1. Kết quả đạt được	37
4.2. Hạn chế của đề tài.....	37
4.3. Phát triển đề tài trong tương lai.....	37

CHƯƠNG 1: TỔNG QUAN

1.1. Giới thiệu đề tài

Trong bối cảnh công nghệ thông tin và mạng xã hội ngày càng phát triển, nhu cầu giao tiếp trực tuyến đã trở thành một phần không thể thiếu trong cuộc sống hiện đại. Việc xây dựng các ứng dụng chat với đầy đủ tính năng giao tiếp không chỉ giúp kết nối con người, mà còn tạo nên môi trường an toàn, tiện lợi và mang lại trải nghiệm giao tiếp hiệu quả. Đề tài 'Ứng dụng chat trực tuyến' ra đời nhằm đáp ứng nhu cầu đó. Hệ thống này không chỉ cho phép người dùng kết bạn, nhắn tin, mà còn tích hợp các tính năng nâng cao như tìm kiếm lịch sử trò chuyện, và quản lý mối quan hệ giữa các user. Đồng thời, hệ thống đặt trọng tâm vào giao diện thân thiện, hướng đến việc mang lại một giải pháp giao tiếp hiện đại và đáng tin cậy. Đề tài này không chỉ có ý nghĩa thực tiễn trong việc đáp ứng nhu cầu xã hội mà còn mở ra cơ hội phát triển các ứng dụng công nghệ giao tiếp trong tương lai.

1.2. Lý do chọn đề tài

Việc xây dựng các ứng dụng chat trực tuyến hiện đại không chỉ nhằm đáp ứng nhu cầu giao tiếp mà còn tạo ra môi trường tương tác tiện lợi và hiệu quả cho người dùng. Đề tài "Ứng dụng chat trực tuyến" được lựa chọn với mục đích phát triển một hệ thống hỗ trợ kết nối bạn bè, trò chuyện cá nhân hoặc nhóm, đồng thời cung cấp các tính năng như tìm kiếm tin nhắn, quản lý danh sách bạn bè và tạo nhóm chat.

Thông qua đề tài, nhóm mong muốn ứng dụng những kiến thức đã học vào thực tế, rèn luyện kỹ năng lập trình, thiết kế giao diện và xử lý dữ liệu, đồng thời xây dựng một sản phẩm đáp ứng nhu cầu thực tế. Đây cũng là cơ hội để nhóm phát triển tư duy sáng tạo và khả năng làm việc nhóm trong lĩnh vực phát triển phần mềm trên thiết bị di động.

1.3. Yêu cầu đề tài

Đề tài yêu cầu xây dựng một hệ thống ứng dụng chat trực tuyến với các chức năng dành cho user bao gồm: đăng ký tài khoản với kiểm tra dữ liệu, đăng nhập, kết bạn với các user khác và quản lý danh sách bạn bè. Người dùng có thể trò chuyện với bạn bè qua tin nhắn văn bản và hình ảnh, xem lại lịch sử tin nhắn, đồng thời hỗ trợ tìm kiếm bạn bè trong danh sách hoặc nội dung tin nhắn đã trao đổi.

Đối với admin, hệ thống cần cung cấp các chức năng quản lý tài khoản người dùng, bao gồm thêm, xóa, sửa và thay đổi vai trò, cũng như quản lý các user. Ngoài ra, yêu cầu đề tài còn đặt ra việc thiết kế giao diện thân thiện, dễ sử dụng, đảm bảo hệ thống có khả năng mở rộng và dễ dàng bảo trì. Dữ liệu trong hệ thống phải được xử lý và lưu trữ hiệu quả, đáp ứng độ chính xác và tin cậy cao để phục vụ các nhu cầu thực tế của người dùng.

1.4. Mục tiêu và phạm vi

Mục tiêu: Đề tài hướng đến việc xây dựng một ứng dụng chat trực tuyến hiện đại, đáp ứng nhu cầu giao tiếp và kết nối của người dùng. Hệ thống sẽ cung cấp các chức năng chính như đăng ký, đăng nhập, kết bạn, trò chuyện cá nhân hoặc nhóm, tìm kiếm tin nhắn và quản lý bạn bè. Đồng thời, ứng dụng cần đảm bảo trải nghiệm thân thiện, dễ sử dụng và tích hợp các công cụ quản lý hiệu quả dành cho admin, như quản lý tài khoản người dùng và quan hệ giữa các user.

Phạm vi: Hệ thống được thiết kế dành cho hai nhóm đối tượng chính là user và admin. Người dùng có thể sử dụng các chức năng giao tiếp cơ bản và nâng cao như nhắn tin văn bản, hình ảnh, tìm kiếm thông tin, xem lịch sử trò chuyện, và quản lý mối quan hệ bạn bè. Đối với admin, hệ thống cung cấp quyền quản lý user, bao gồm việc thêm, sửa, xóa tài khoản và thay đổi vai trò, cũng như xử lý mối quan hệ giữa các user. Ứng dụng tập trung vào môi trường giao tiếp trực tuyến nội bộ và các tính năng quản lý, không mở rộng sang tích hợp với các mạng xã hội hoặc nền tảng bên ngoài.

1.5. Phương pháp nghiên cứu

Để thực hiện đề tài "Ứng dụng chat trực tuyến", nhóm nghiên cứu đã áp dụng phương pháp kết hợp giữa lý thuyết và thực tiễn. Đầu tiên, nhóm sẽ tiến hành nghiên cứu các ứng dụng chat hiện có trên thị trường, phân tích các tính năng, công nghệ và giao diện người dùng để lựa chọn công nghệ phù hợp cho hệ thống. Tiếp theo, nhóm sẽ phân tích yêu cầu

và thiết kế hệ thống, bao gồm xây dựng kiến trúc hệ thống, cơ sở dữ liệu, và giao diện người dùng sao cho dễ sử dụng và đáp ứng đầy đủ các yêu cầu chức năng.

Sau khi có thiết kế, nhóm sẽ triển khai lập trình ứng dụng sử dụng các công nghệ như Java cho Android và Firebase cho lưu trữ dữ liệu. Trong quá trình phát triển, nhóm sẽ liên tục kiểm tra, điều chỉnh và sửa lỗi để đảm bảo các chức năng hoạt động ổn định. Sau khi hoàn thiện, nhóm sẽ tiến hành kiểm thử toàn diện về chức năng, hiệu suất và tính chính xác của ứng dụng. Phương pháp này giúp đảm bảo tính khả thi và hiệu quả của ứng dụng từ thiết kế đến triển khai.

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

2.1. Giới thiệu về Android Studio

2.1.1. Giới thiệu

Android Studio là môi trường phát triển tích hợp (IDE) chính thức dành cho lập trình viên phát triển ứng dụng Android, được xây dựng dựa trên nền tảng IntelliJ IDEA. IDE này cung cấp các công cụ và tính năng cần thiết để thiết kế, phát triển, kiểm tra và triển khai ứng dụng trên hệ điều hành Android.

2.1.2 Cài đặt Android Studio

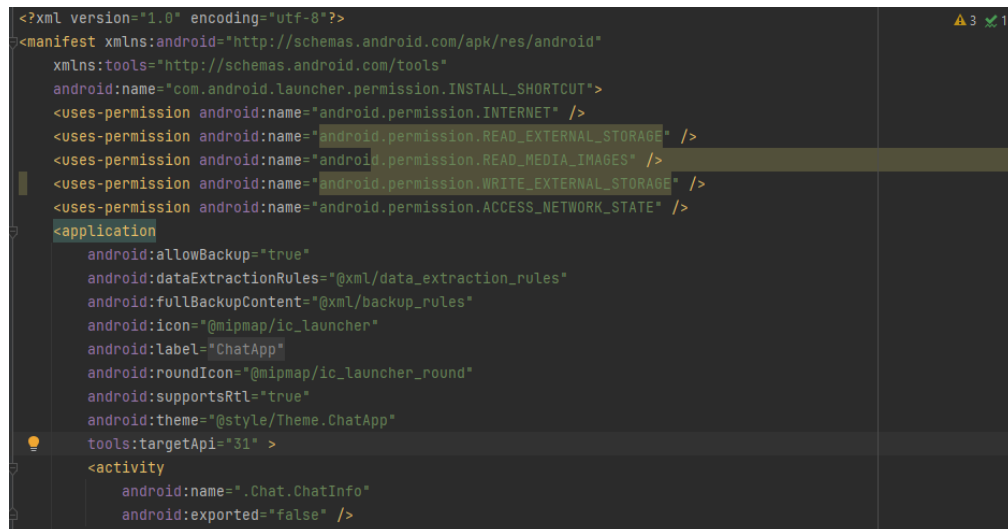
Để bắt đầu sử dụng Android Studio, người dùng cần thực hiện các bước sau:

- **Bước 1:** Truy cập trang web chính thức của Android Studio (developer.android.com) và tải xuống phiên bản phù hợp với hệ điều hành (Windows, macOS, Linux).
- **Bước 2:** Chạy tệp cài đặt và làm theo hướng dẫn trong trình cài đặt (wizard).
- **Bước 3:** Cài đặt các thành phần bổ sung như:
 - **Android SDK:** Bộ công cụ phần mềm hỗ trợ phát triển ứng dụng Android.
 - **Android Virtual Device (AVD):** Công cụ giả lập thiết bị Android để thử nghiệm ứng dụng.

2.1.3 Cấu trúc của Project trong Android Studio

Khi tạo một dự án mới trong Android Studio, cấu trúc thư mục sẽ được tổ chức như sau:

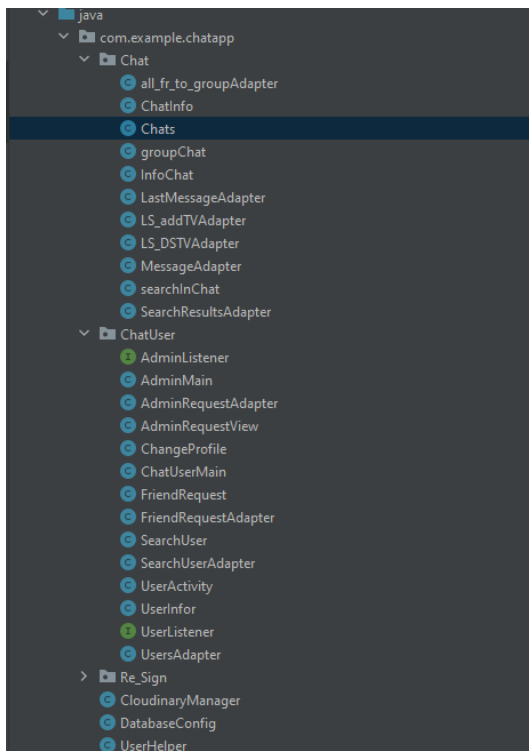
- **Manifests:** Thư mục chứa tệp AndroidManifest.xml, nơi định nghĩa cấu hình cơ bản của ứng dụng như quyền truy cập (permissions), thông tin về các Activity.



```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:name="com.android.launcher.permission.INSTALL_SHORTCUT">
    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />
    <uses-permission android:name="android.permission.READ_MEDIA_IMAGES" />
    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@mipmap/ic_launcher"
        android:label="ChatApp"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportRtl="true"
        android:theme="@style/Theme.ChatApp"
        tools:targetApi="31" >
        <activity
            android:name=".Chat.ChatInfo"
            android:exported="false" />
    </application>
</manifest>
```

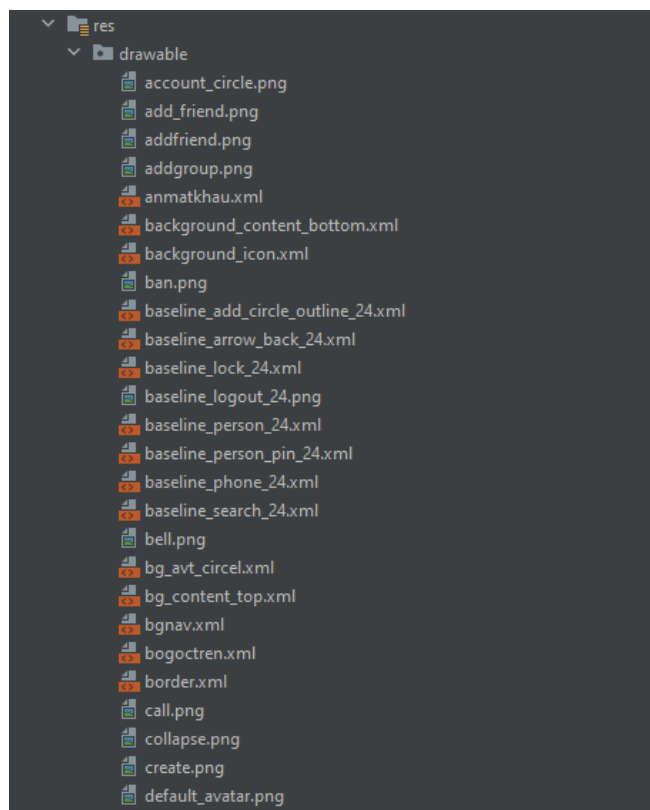
Hình 1. Quyền truy cập mạng và bộ nhớ, khai báo các activity

- **java:** Chứa mã nguồn Java của ứng dụng, bao gồm các tệp xử lý logic.



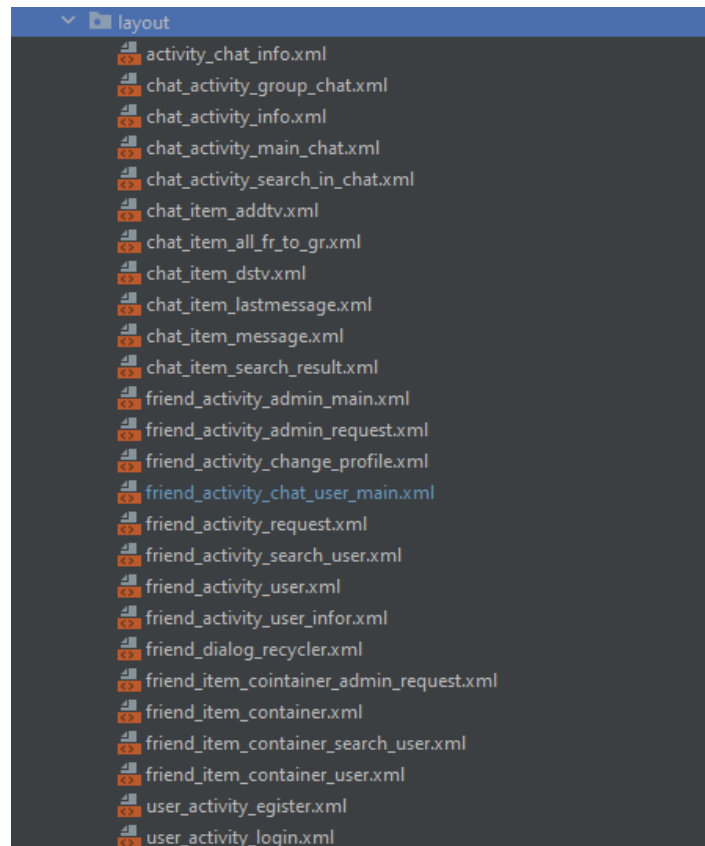
Hình 2. Cấu trúc các file.java

- **res/drawable:** Thư mục chứa tài nguyên hình ảnh và các tệp đồ họa (PNG, XML, v.v.).



Hình 3. Cấu trúc các tài nguyên trong drawable

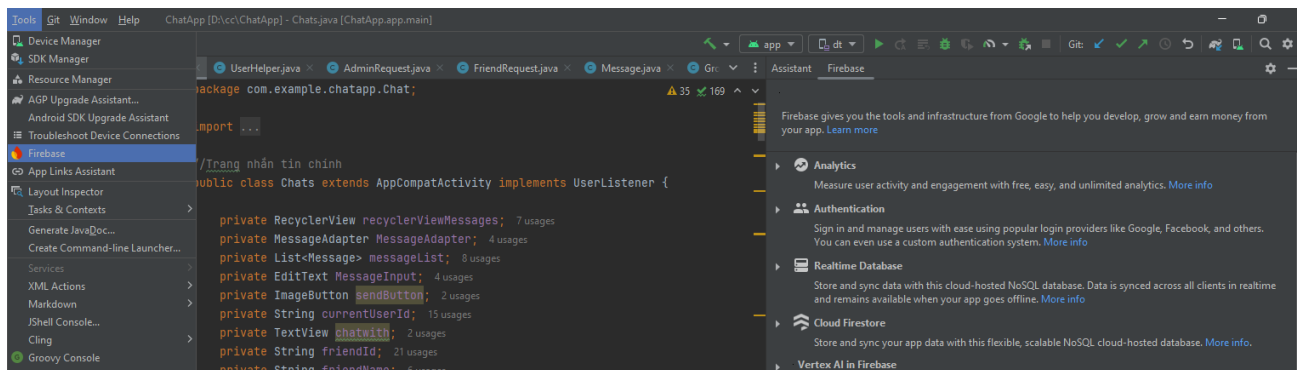
- **res/layout:** Thư mục chứa các tệp giao diện XML định nghĩa bố cục màn hình ứng dụng.



Hình 4. Các tệp giao diện

2.2 Giới thiệu về Firebase

Firebase là nền tảng phát triển ứng dụng di động được Google cung cấp, hỗ trợ các dịch vụ như cơ sở dữ liệu thời gian thực, xác thực người dùng, lưu trữ, và phân tích ứng dụng.

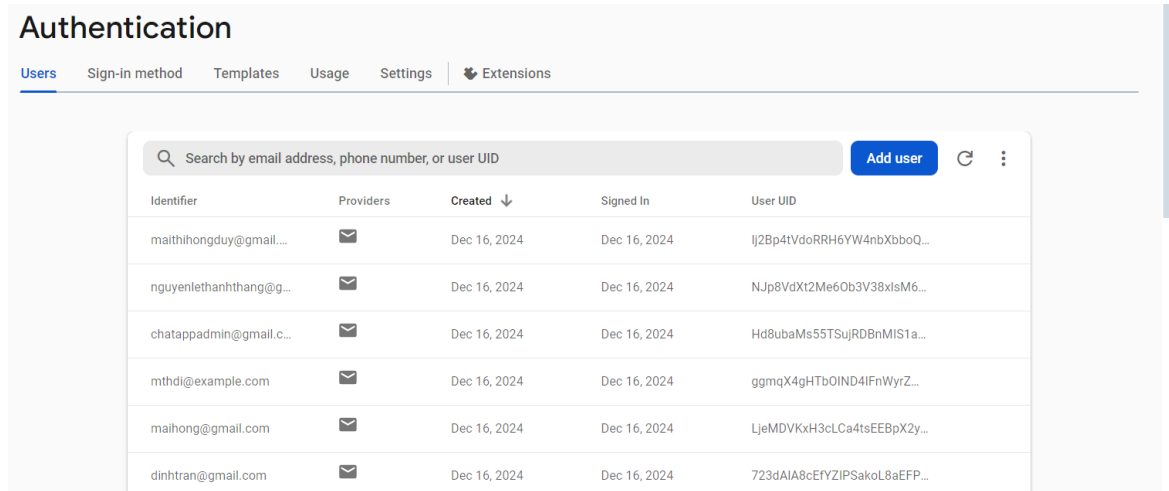


Hình 5. Firebase trong Android Studio

2.2.1 Email/Password Authentication

Firebase Authentication cho phép người dùng đăng nhập vào ứng dụng thông qua email và mật khẩu. Đây là phương thức xác thực phổ biến và dễ tích hợp. Các tính năng chính bao gồm:

- Tạo tài khoản người dùng mới.
- Đăng nhập và đăng xuất tài khoản.
- Đặt lại mật khẩu thông qua email.

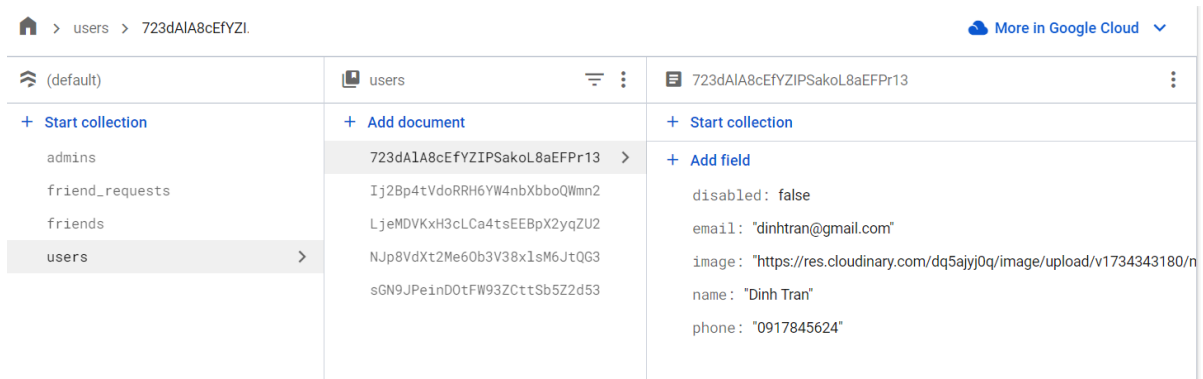


Hình 5. Các email đã được chứng thực

2.2.2. Firestore

Firestore là cơ sở dữ liệu NoSQL hiện đại được cung cấp bởi Firebase. Nó lưu trữ dữ liệu dưới dạng **document** và **collection**, phù hợp cho việc xây dựng ứng dụng yêu cầu khả năng đồng bộ hóa dữ liệu theo thời gian thực. Một số đặc điểm nổi bật:

- Hỗ trợ truy vấn mạnh mẽ, linh hoạt.
- Tích hợp liền mạch với các dịch vụ Firebase khác.
- Hỗ trợ offline, cho phép ứng dụng hoạt động ngay cả khi không có kết nối mạng.

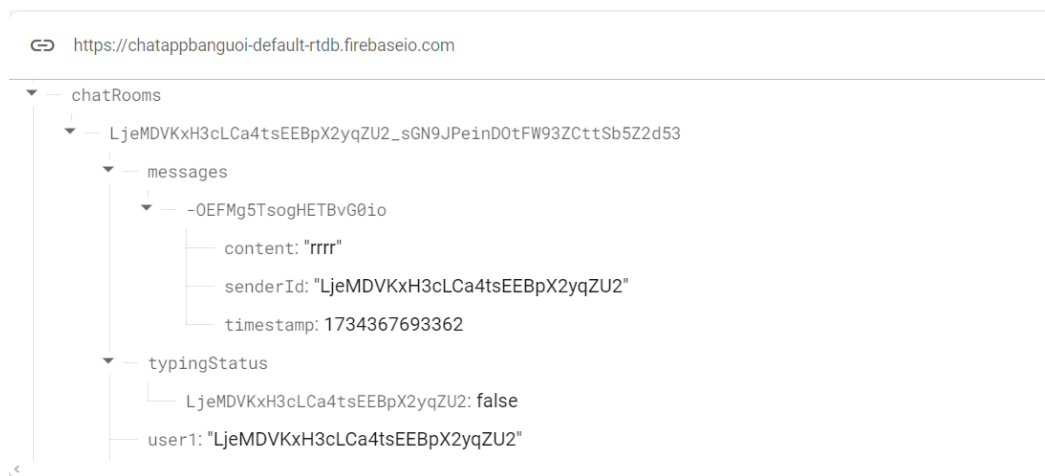


Hình 6. Cấu trúc các thông tin dùng trong ứng dụng đang lưu ở firestore

2.2.3 RealtimeDatabase

Realtime Database là một dịch vụ cơ sở dữ liệu NoSQL khác của Firebase, được thiết kế đặc biệt để đồng bộ dữ liệu theo thời gian thực.

- **Đặc điểm chính:**
 - Dữ liệu được lưu trữ dưới dạng cây JSON, dễ dàng truy cập qua các nút (nodes).
 - Cập nhật dữ liệu ngay lập tức trên tất cả thiết bị kết nối.
 - Tích hợp các quy tắc bảo mật dựa trên quyền truy cập người dùng (Firebase Security Rules).
- **Ứng dụng:**
 - Xây dựng ứng dụng chat thời gian thực.
 - Đồng bộ hóa trạng thái kết nối, trạng thái đang nhập



Hình 7. Cấu trúc phòng nhắn tin và các tin nhắn trong RealtimeDatabase

2.3. Các công nghệ khác

2.3.1 Glide

Glide là thư viện mã nguồn mở được sử dụng để tải và hiển thị hình ảnh trong ứng dụng Android. Các tính năng chính:

- Tải và hiển thị hình ảnh từ URL, tệp, hoặc tài nguyên.
- Tích hợp caching để tăng tốc độ hiển thị.
- Hỗ trợ hiệu ứng chuyển tiếp và tải ảnh động (GIF).

```
import com.bumptech.glide.Glide;|
Glide.with( activity: this) RequestManager
    .load(image) RequestBuilder<Drawable>
    .placeholder(R.drawable.default_avatar)
    .into(imageButton);
```

Hình 8. Dùng glide để hiển thị hình ảnh trong ứng dụng

2.3.2 Cloudinary

Cloudinary là dịch vụ đám mây hỗ trợ quản lý và xử lý hình ảnh, video. Nó cho phép:

- Lưu trữ hình ảnh và video trên đám mây.
- Tối ưu hóa hình ảnh và thay đổi kích thước theo yêu cầu.
- Tích hợp API dễ dàng để quản lý tài nguyên từ ứng dụng.

```
1 package com.example.chatapp;
2
3 import android.content.Context;
4 import com.cloudinary.android.MediaManager;
5 import java.util.HashMap;
6 import java.util.Map;
7
8 public class CloudinaryManager { 4 usages
9
10     private static boolean isInitialized = false; 2 usages
11
12     public static void initialize(Context context) { 2 usages
13         if (!isInitialized) {
14             Map<String, String> config = new HashMap<>();
15             config.put("cloud_name", "ddskv3qix");
16             config.put("api_key", "237429289958929");
17             config.put("api_secret", "72Fe5rWNVv0_3E8fAHa9lvZ2z6K");
18             MediaManager.init(context, config);
19             isInitialized = true; // Đánh dấu đã được khởi tạo
20         }
21     }
22 }
23 |
```

Hình 9. Cấu hình để truy cập được vào cloudinary

CHƯƠNG 3: ĐỀ TÀI

3.1. Các bước thực hiện đề tài

3.1.1. Xác định yêu cầu và mục tiêu của đề tài:

Trước khi bắt đầu, nhóm đã thống nhất về mục tiêu của ứng dụng chat, bao gồm việc xây dựng một nền tảng giúp người dùng đăng ký, đăng nhập, kết bạn, gửi tin nhắn, và tạo nhóm chat.

Các yêu cầu chức năng của ứng dụng được xác định rõ ràng và phân chia thành các phần nhỏ như quản lý tài khoản, kết bạn, nhắn tin, và quản lý quyền người dùng.

Đảm bảo các yêu cầu đề tài được đáp ứng đầy đủ, từ việc hỗ trợ nhiều loại tính năng (đăng nhập, gửi tin nhắn, tìm kiếm bạn bè) đến khả năng mở rộng cho các tính năng mới như quản lý quyền và tạo nhóm chat.

3.1.2. Lập kế hoạch phát triển:

Nhóm phân công công việc cho từng thành viên, bao gồm việc phát triển giao diện người dùng (UI), lập trình backend (Firebase), và tích hợp các chức năng cần thiết.

Mỗi thành viên sẽ chịu trách nhiệm về một phần công việc cụ thể, từ việc đăng ký tài khoản cho đến tính năng nhắn tin và quản lý bạn bè, ...

Các tính năng được phát triển tuần tự và phù hợp với yêu cầu của đề tài, đảm bảo tiến độ và chất lượng công việc.

3.1.3. Thiết kế giao diện và trải nghiệm người dùng:

Nhóm bắt đầu với việc thiết kế các màn hình giao diện chính như màn hình đăng nhập, đăng ký, tìm kiếm bạn bè, và màn hình chat.

Giao diện được tối ưu hóa để người dùng dễ dàng thực hiện các thao tác, đảm bảo đáp ứng yêu cầu về tính trực quan và dễ sử dụng của ứng dụng.

3.1.4. Lập trình và tích hợp các chức năng:

Cài đặt Firebase để lưu trữ dữ liệu người dùng và các chức năng liên quan đến quản lý tài khoản, như đăng nhập và đăng ký.

Tích hợp tính năng gửi và nhận tin nhắn, bao gồm cả văn bản và hình ảnh. Thiết lập hệ thống thông báo khi có yêu cầu kết bạn mới.

Đảm bảo các chức năng của ứng dụng hoạt động ổn định và đáp ứng đủ yêu cầu về tính bảo mật, hiệu suất và khả năng mở rộng của hệ thống.

3.1.5. Kiểm thử và sửa lỗi:

Sau khi phát triển, nhóm tiến hành kiểm thử các chức năng để đảm bảo hoạt động chính xác, từ đăng nhập, gửi tin nhắn cho đến quản lý bạn bè. Các lỗi và vấn đề phát sinh trong quá trình sử dụng sẽ được sửa chữa và cải tiến.

Kiểm thử được thực hiện trên nhiều thiết bị khác nhau để đảm bảo ứng dụng hoạt động ổn định và đáp ứng tốt yêu cầu đề tài.

Các kết quả và bước phát triển được ghi lại và chuẩn bị cho báo cáo, trong đó trình bày cách thức thực hiện từng phần của dự án. Đảm bảo báo cáo phản ánh đúng các yêu cầu của đề tài, với các chức năng được thực hiện đầy đủ và hoạt động hiệu quả.

3.2. Mô tả chức năng của đề tài

3.2.1. Chức năng của người dùng:

3.2.1.1. Đăng Ký thành viên (bao gồm kiểm tra dữ liệu đăng ký)

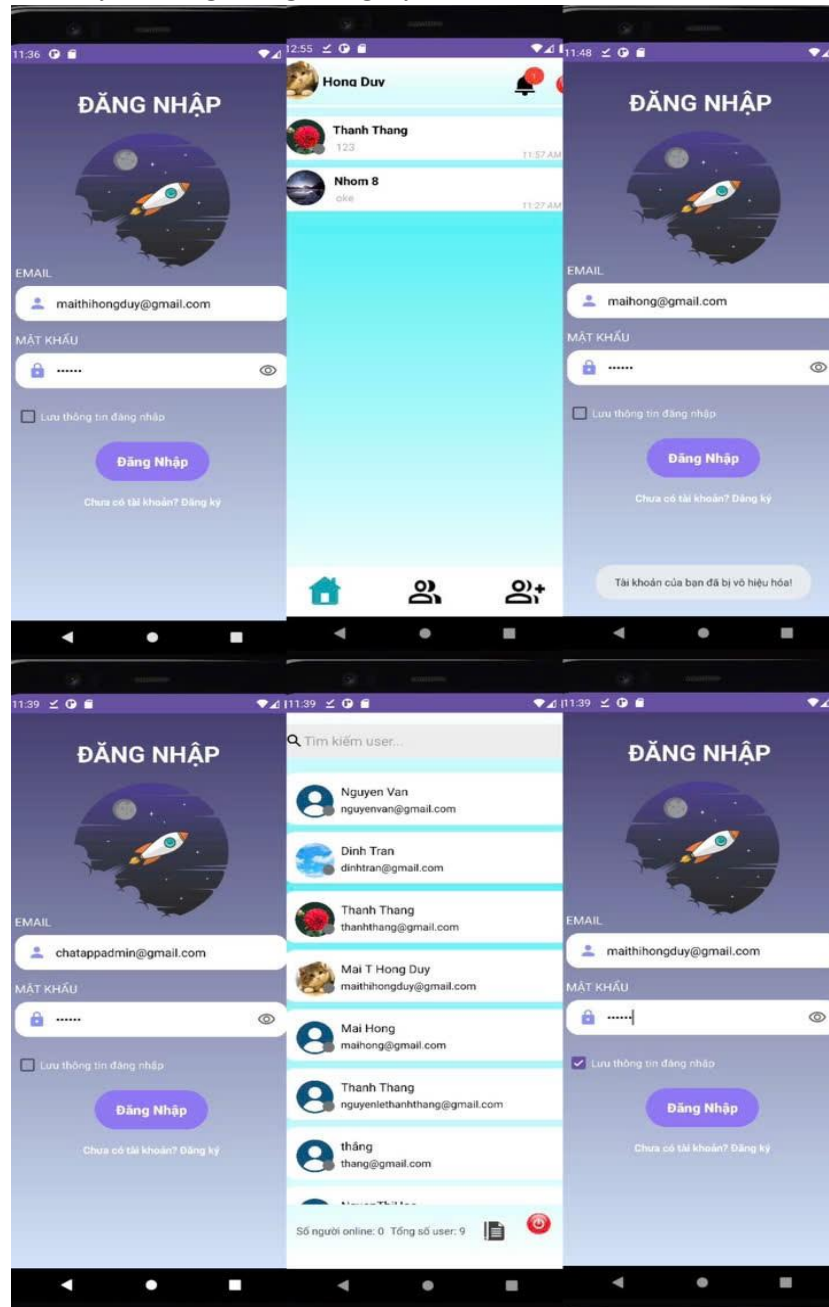
- Nếu người dùng bỏ trống một trong các ô nhập liệu bắt buộc nhập thì sẽ hiện thông báo yêu cầu nhập vào ô nhập liệu bỏ trống.
- Email phải nhập đúng định dạng có đuôi là @gmail.com. Nếu không đúng sẽ có cảnh báo yêu cầu nhập đúng dạng.
- Ô nhập liệu số điện thoại chỉ cho phép nhập số, mật khẩu phải nhập tối thiểu 6 ký tự, ẩn hiện mật khẩu khi đăng ký, mật khẩu và nhập lại mật khẩu phải khớp nhau mới thực hiện đăng ký thành công.
- Email đăng ký không được phép trùng với email của người quản trị, nếu trùng sẽ không thể đăng ký

The screenshots illustrate the registration process with the following details:

- Screenshot 1:** 'TẠO TÀI KHOẢN' form with fields for TÊN (Hong Duy), EMAIL (maithongduy@gmail.com), SỐ ĐIỆN THOẠI (0353549555), and MẬT KHẨU (6 asterisks). A 'Đăng Ký' button is at the bottom.
- Screenshot 2:** Similar to Screenshot 1, but with a red error message: 'Email phải có đuôi @gmail.com'.
- Screenshot 3:** Similar to Screenshot 1, but with a red error message: 'Số điện thoại của bạn'.
- Screenshot 4:** Similar to Screenshot 1, but with a red error message: 'Mật khẩu'.
- Screenshot 5:** Similar to Screenshot 1, but with a red error message: 'Mật khẩu phải có ít nhất 6 ký tự'.
- Screenshot 6:** Similar to Screenshot 1, but with a red error message: 'Mật khẩu không khớp'.
- Screenshot 7:** Similar to Screenshot 1, but with a red error message: 'Email này không thể đăng ký'.
- Screenshot 8:** Similar to Screenshot 1, but with a green success message: 'Tạo tài khoản thành công'.
- Screenshot 9:** 'ĐĂNG NHẬP' screen with fields for EMAIL (Nhập email của bạn) and MẬT KHẨU (Nhập mật khẩu của bạn), and a 'Đăng Nhập' button.

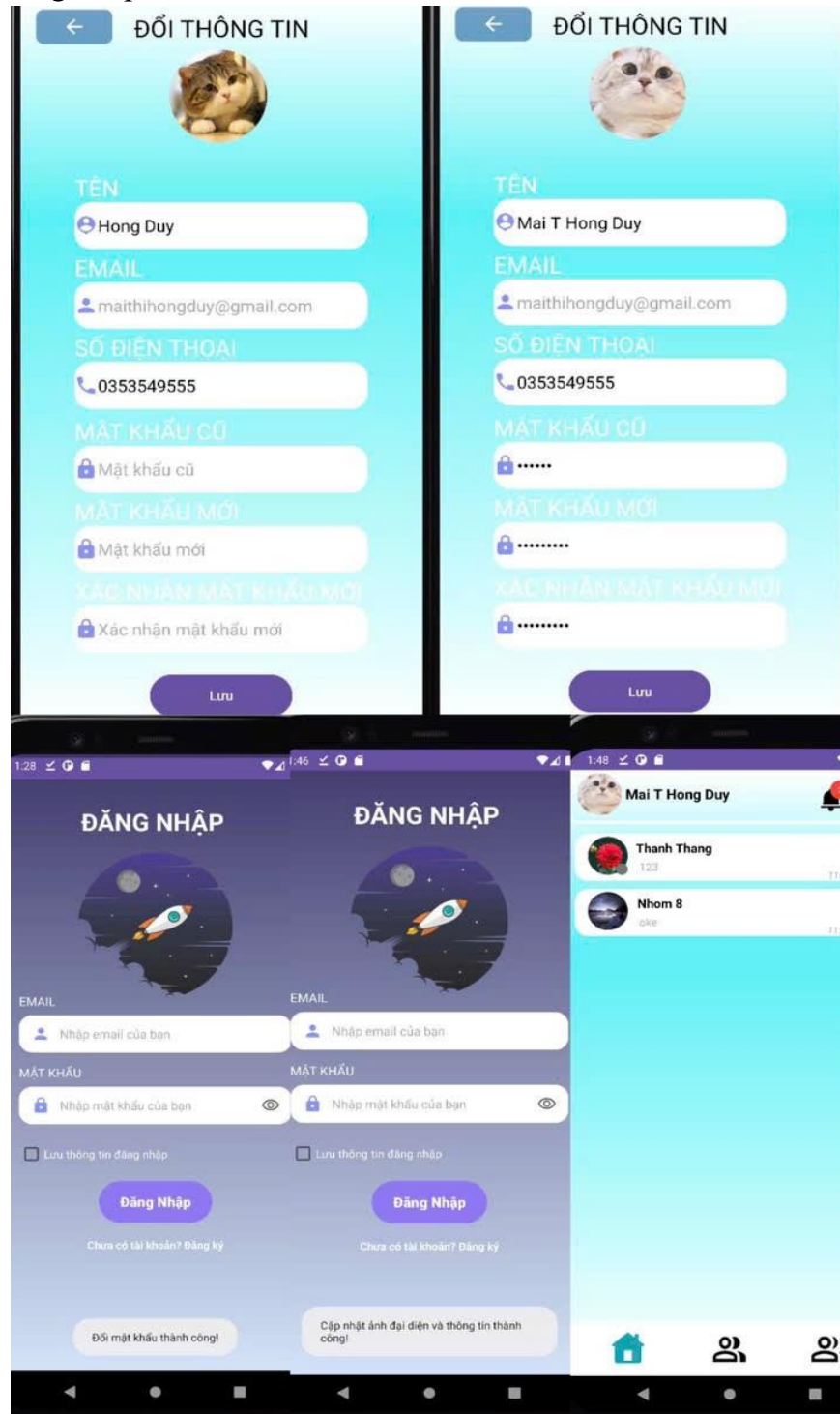
3.2.1.2. Đăng nhập

Chức năng đăng nhập cho phép người dùng truy cập vào ứng dụng bằng email và mật khẩu. Hệ thống xác định vai trò từ **Firestore**: nếu là **Admin**, chuyển đến màn hình quản lý; nếu là **User**, kiểm tra tài khoản có bị vô hiệu hóa không trước khi vào giao diện chính. Tính năng **ghi nhớ mật khẩu** giúp tự động đăng nhập lần sau nếu được chọn. Chuyển sang trang đăng ký nếu chưa có tài khoản.



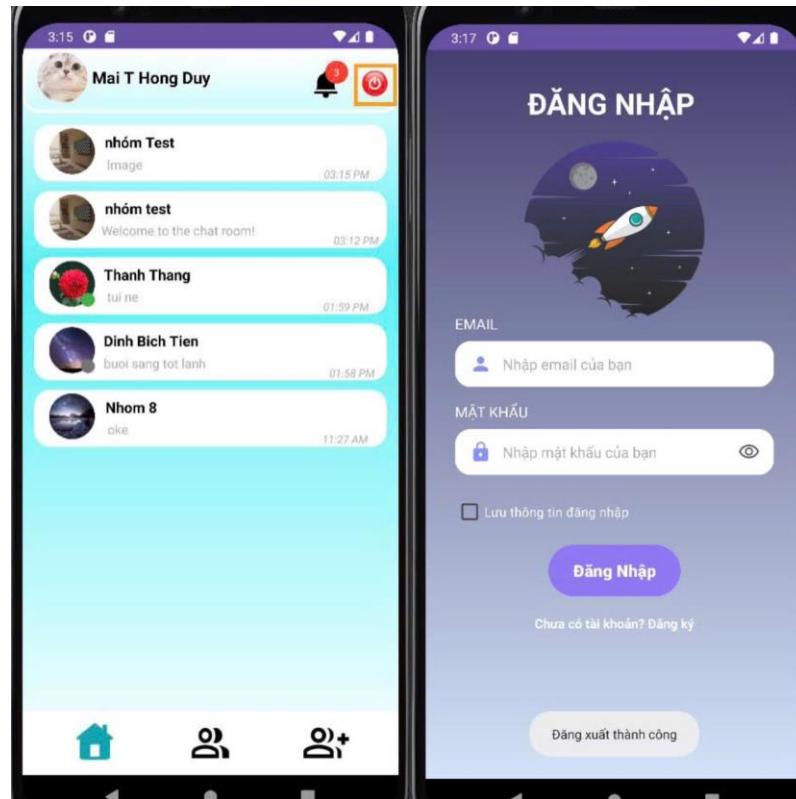
3.2.1.3. Cài đặt người dùng

Chức năng này cho phép người dùng thay đổi thông tin cá nhân như ảnh đại diện, tên, số điện thoại và mật khẩu. Người dùng có thể chọn ảnh mới, cập nhật tên và số điện thoại, hoặc thay đổi mật khẩu. Thông tin được lưu vào Firestore và ảnh được tải lên Cloudinary. Sau khi thay đổi mật khẩu, người dùng sẽ phải đăng nhập lại.



3.2.1.4. Đăng xuất

Chức năng đăng xuất đưa người dùng ra khỏi phiên làm việc, xóa thông tin đăng nhập đã lưu, cập nhật trạng thái "offline" trên Firebase và chuyển về màn hình đăng nhập.



3.2.1.5. Tìm kiếm bạn bè

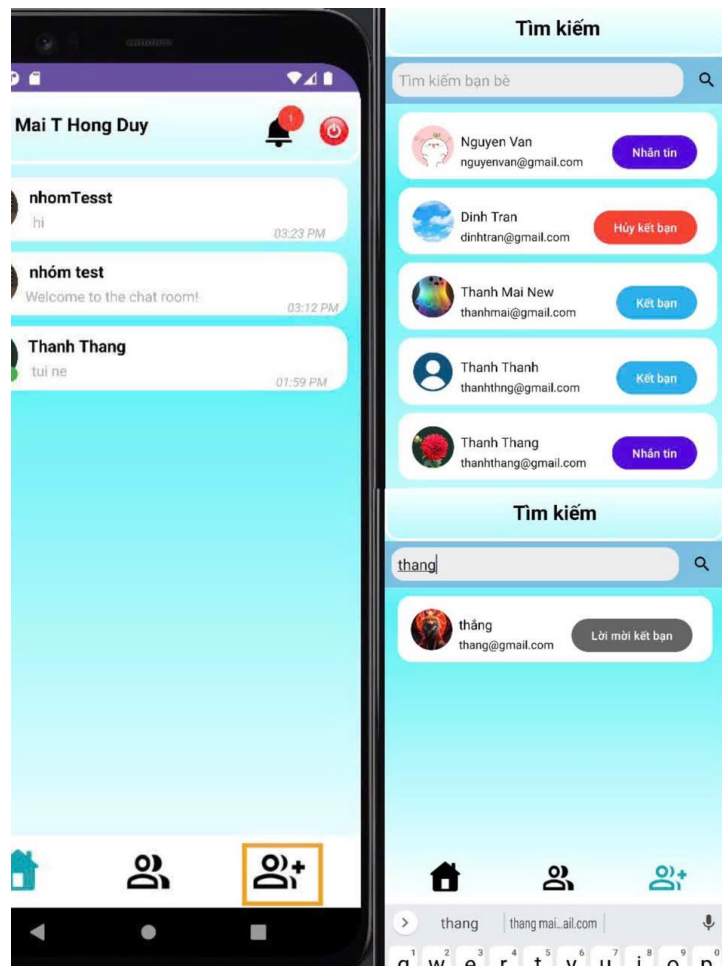
Chức năng Tìm kiếm bạn bè trong ứng dụng cho phép người dùng tìm kiếm và tương tác với những người dùng khác dựa trên tên. Khi người dùng nhập tên vào ô tìm kiếm, kết quả sẽ hiển thị danh sách các người dùng phù hợp. Mỗi người dùng trong danh sách sẽ được gắn trạng thái rõ ràng:

- + Nếu người đó đã là bạn bè của người dùng, hệ thống sẽ hiển thị "Đã kết bạn".

- + Nếu người đó chưa phải là bạn bè, người dùng có thể gửi yêu cầu kết bạn.

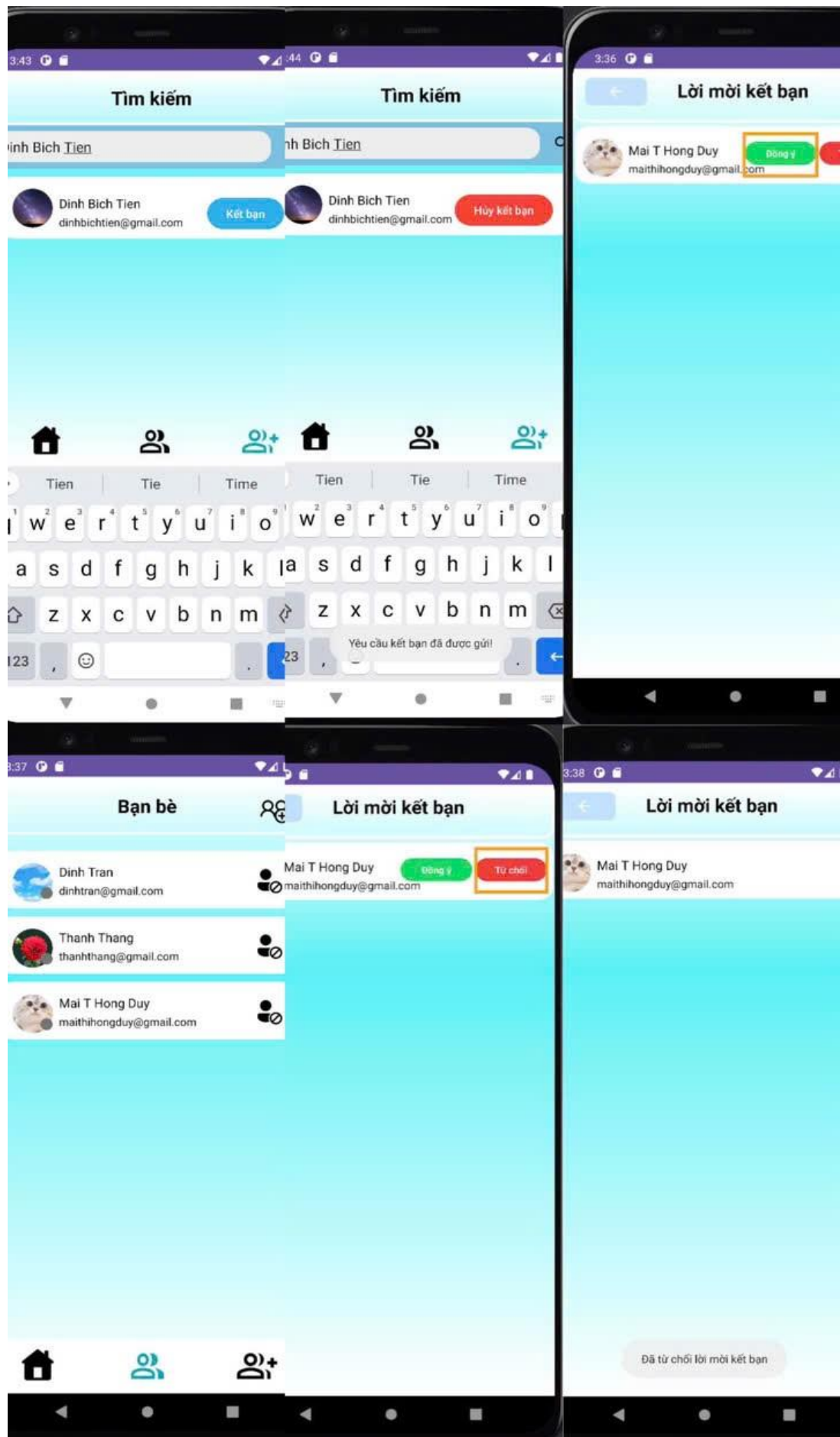
- + Nếu người dùng đã gửi lời mời kết bạn, hệ thống sẽ hiển thị trạng thái "Đã gửi lời mời kết bạn".

Bên cạnh đó, người dùng có thể quản lý các yêu cầu kết bạn của mình, bao gồm gửi yêu cầu kết bạn, nhận yêu cầu từ người khác, hoặc hủy yêu cầu kết bạn đã gửi.



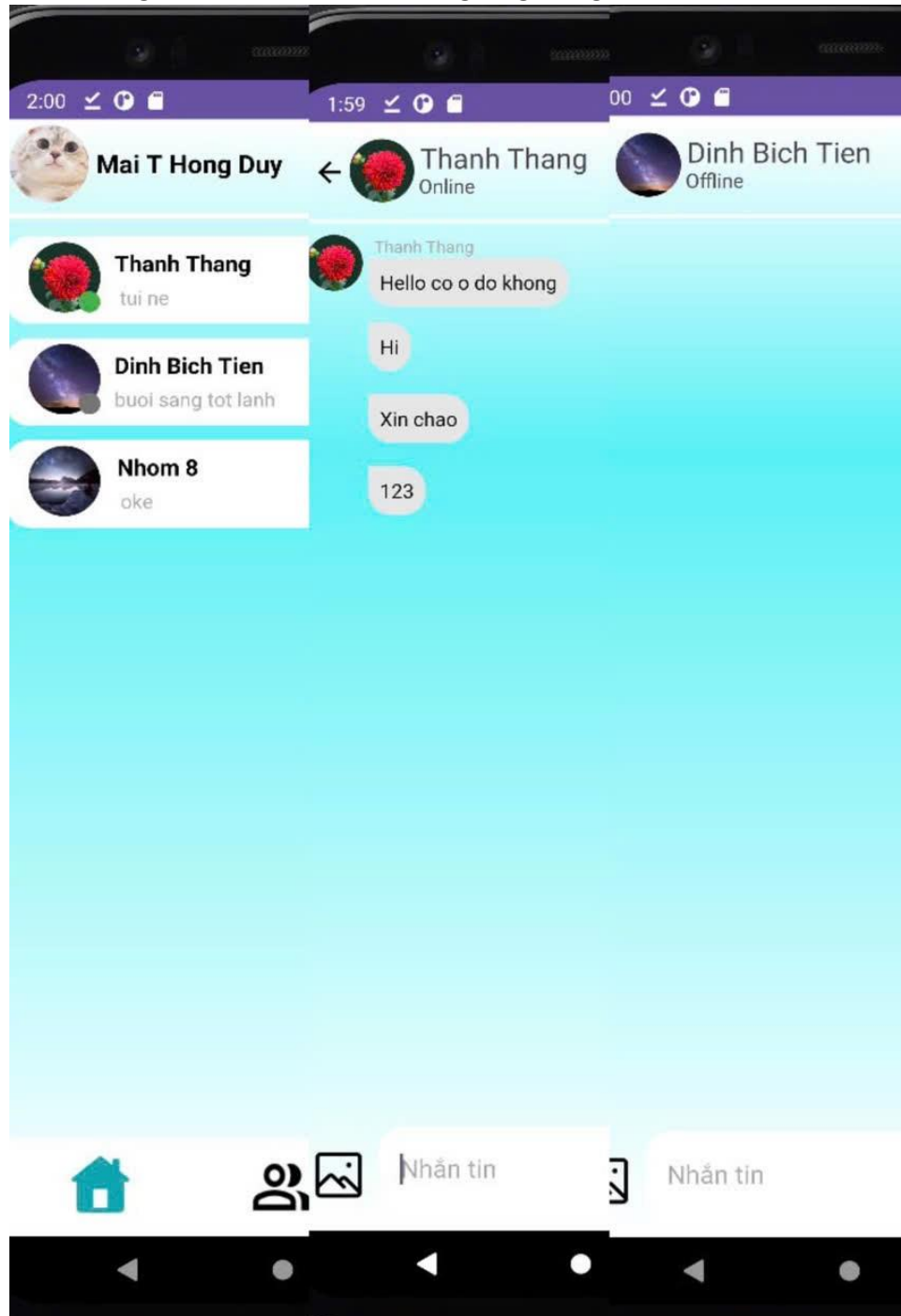
3.2.1.6. Kết bạn giữa user

Chức năng kết bạn giữa người dùng cho phép người dùng gửi và nhận lời mời kết bạn. Khi một người dùng nhận được lời mời kết bạn, họ có thể lựa chọn chấp nhận hoặc từ chối. Nếu chấp nhận, hệ thống sẽ thêm người đó vào danh sách bạn bè của cả hai bên và xóa yêu cầu kết bạn. Ngược lại, nếu từ chối, lời mời sẽ bị xóa và không có quan hệ bạn bè giữa hai người.



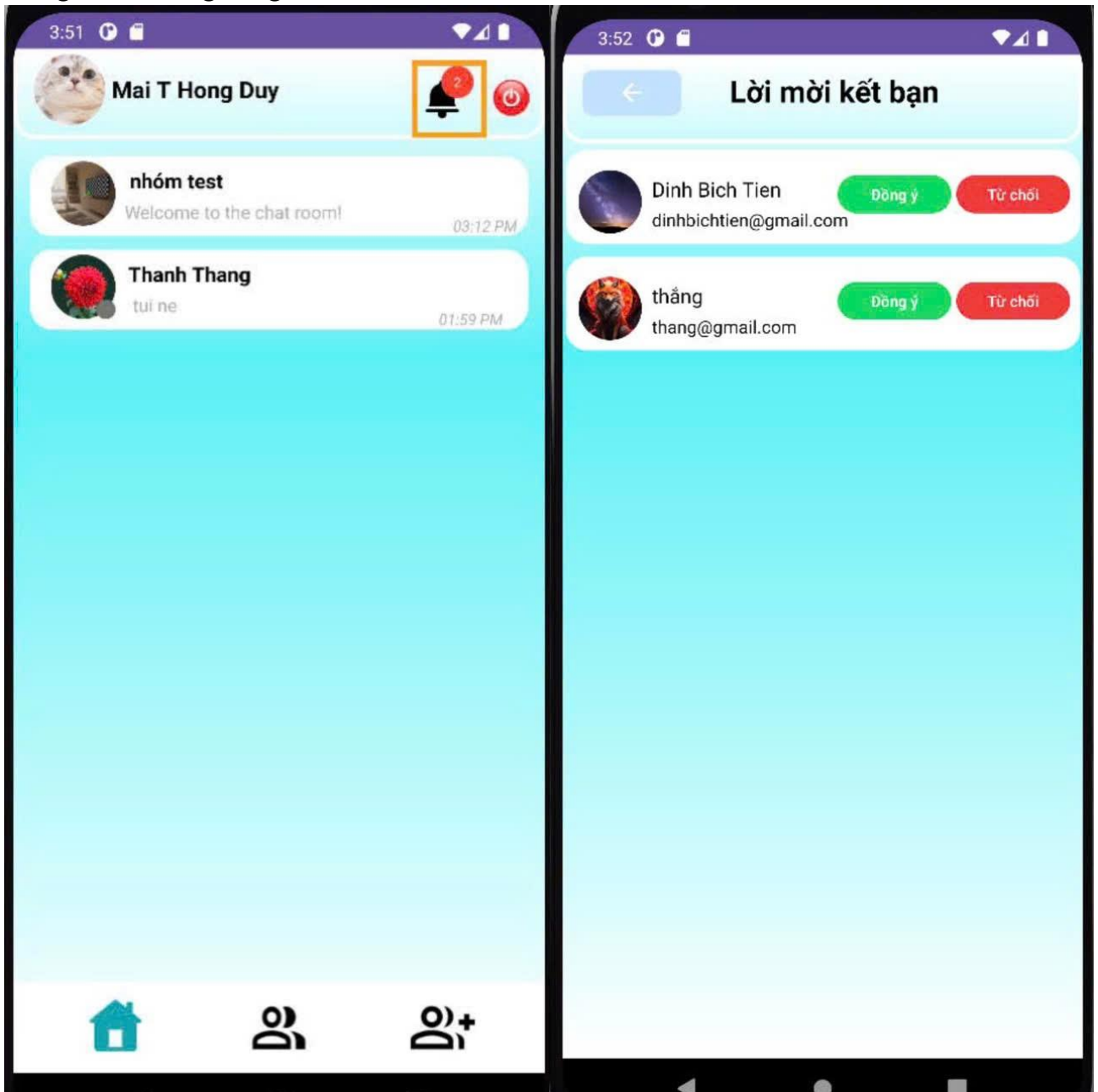
3.2.1.7. Báo tình trạng online (online/offline)

Chức năng báo cáo tình trạng online trong ứng dụng này giúp theo dõi trạng thái kết nối của người dùng với Firebase. Khi người dùng mở ứng dụng, trạng thái của họ được cập nhật thành "online" trong cơ sở dữ liệu Firebase, và khi ứng dụng bị tắt hoặc mất kết nối, trạng thái tự động chuyển thành "offline". Điều này giúp hiển thị tình trạng hoạt động của người dùng, đồng thời thông báo cho các người dùng khác về tình trạng kết nối của bạn bè trong ứng dụng.



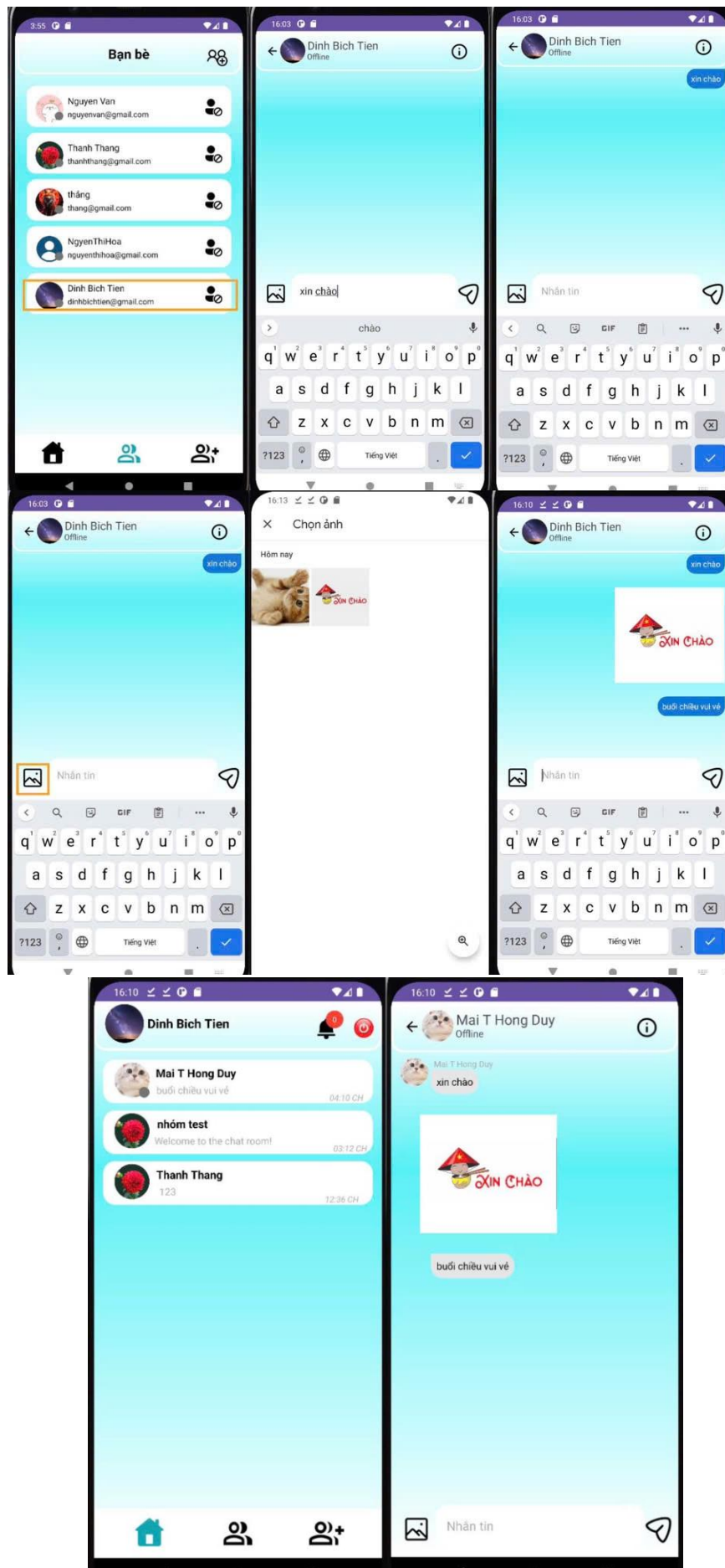
3.2.1.8. Thông báo

Chức năng thông báo lời mời kết bạn giúp người dùng nhận và xem số lượng lời mời kết bạn chưa được xử lý. Số lượng lời mời sẽ được hiển thị trong giao diện ứng dụng.



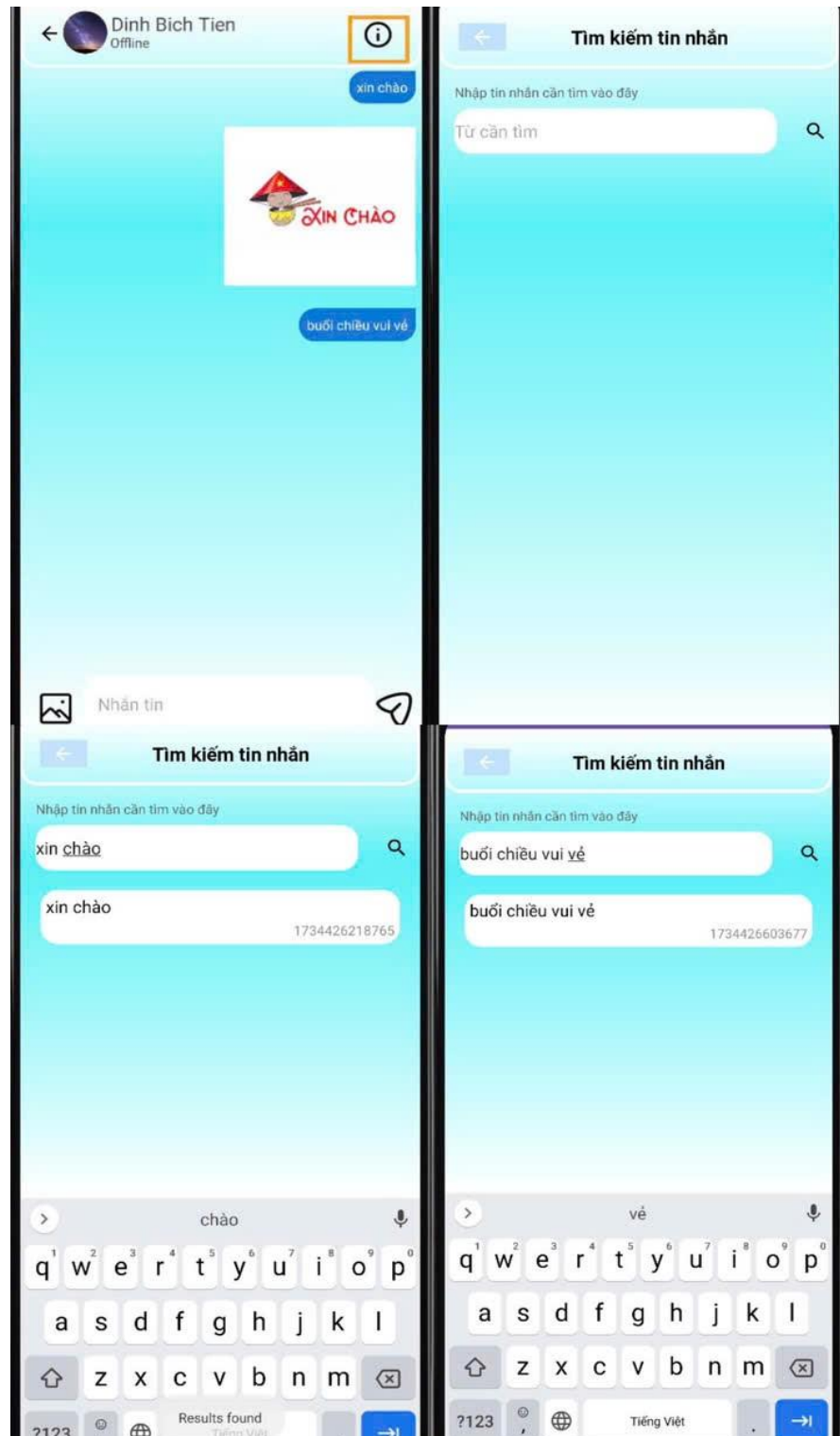
3.2.1.9. Gửi và nhận tin nhắn giữa các user đã kết bạn

Chức năng gửi và nhận tin nhắn cho phép người dùng gửi và nhận tin nhắn (bao gồm cả văn bản và hình ảnh) giữa những người dùng đã kết bạn. Tin nhắn được hiển thị trong các phòng chat riêng biệt giữa các người dùng đã kết nối.



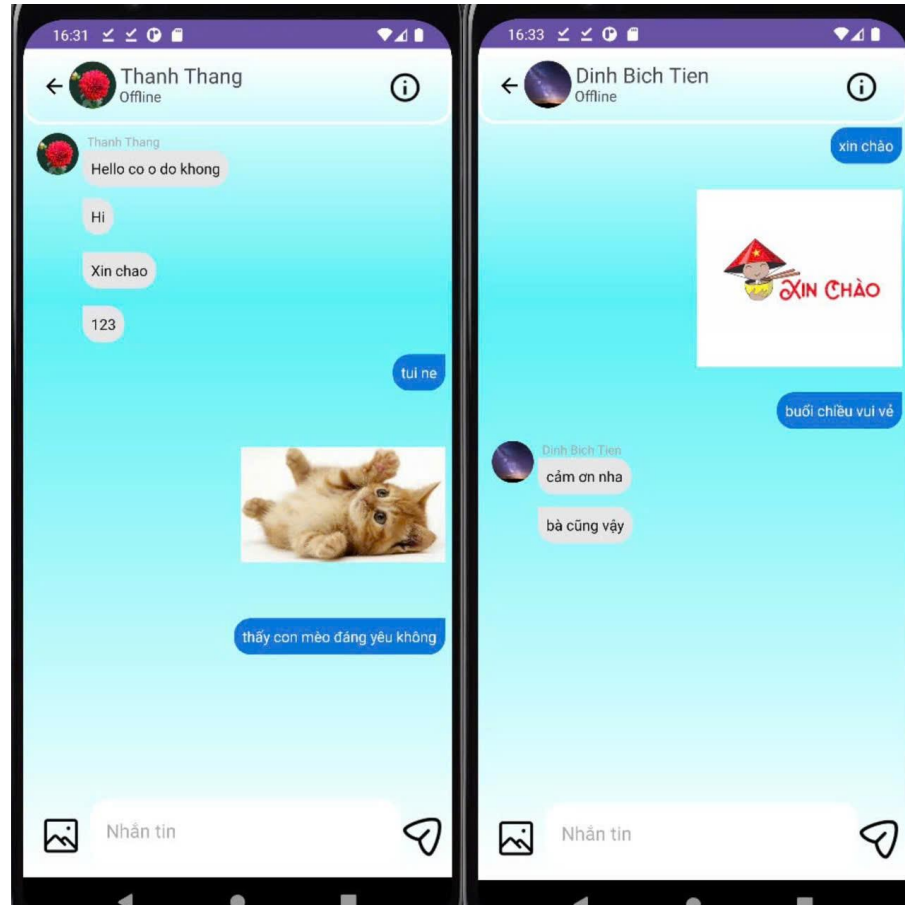
3.2.1.10. Tìm kiếm các đoạn chat có chứa thông tin text cần tìm

Chức năng "Tìm kiếm các đoạn chat có chứa thông tin text cần tìm" cho phép người dùng nhập từ khóa để tìm nhanh các tin nhắn trong các cuộc trò chuyện. Hệ thống sẽ tìm kiếm qua tất cả các tin nhắn và hiển thị kết quả phù hợp, giúp người dùng dễ dàng truy xuất lại thông tin cần thiết.



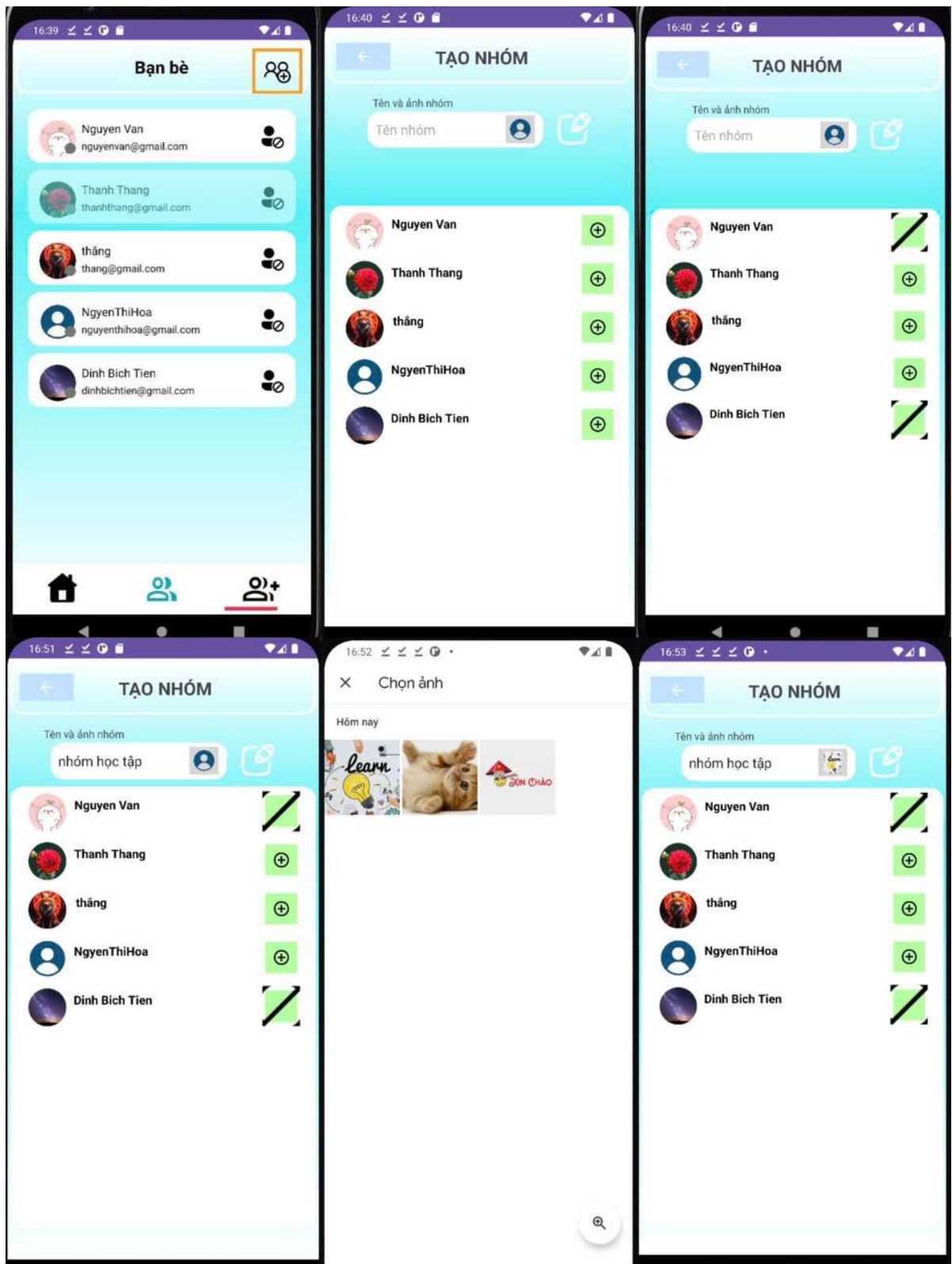
3.2.1.11. Xem lịch sử tin nhắn giữa các user đã kết bạn

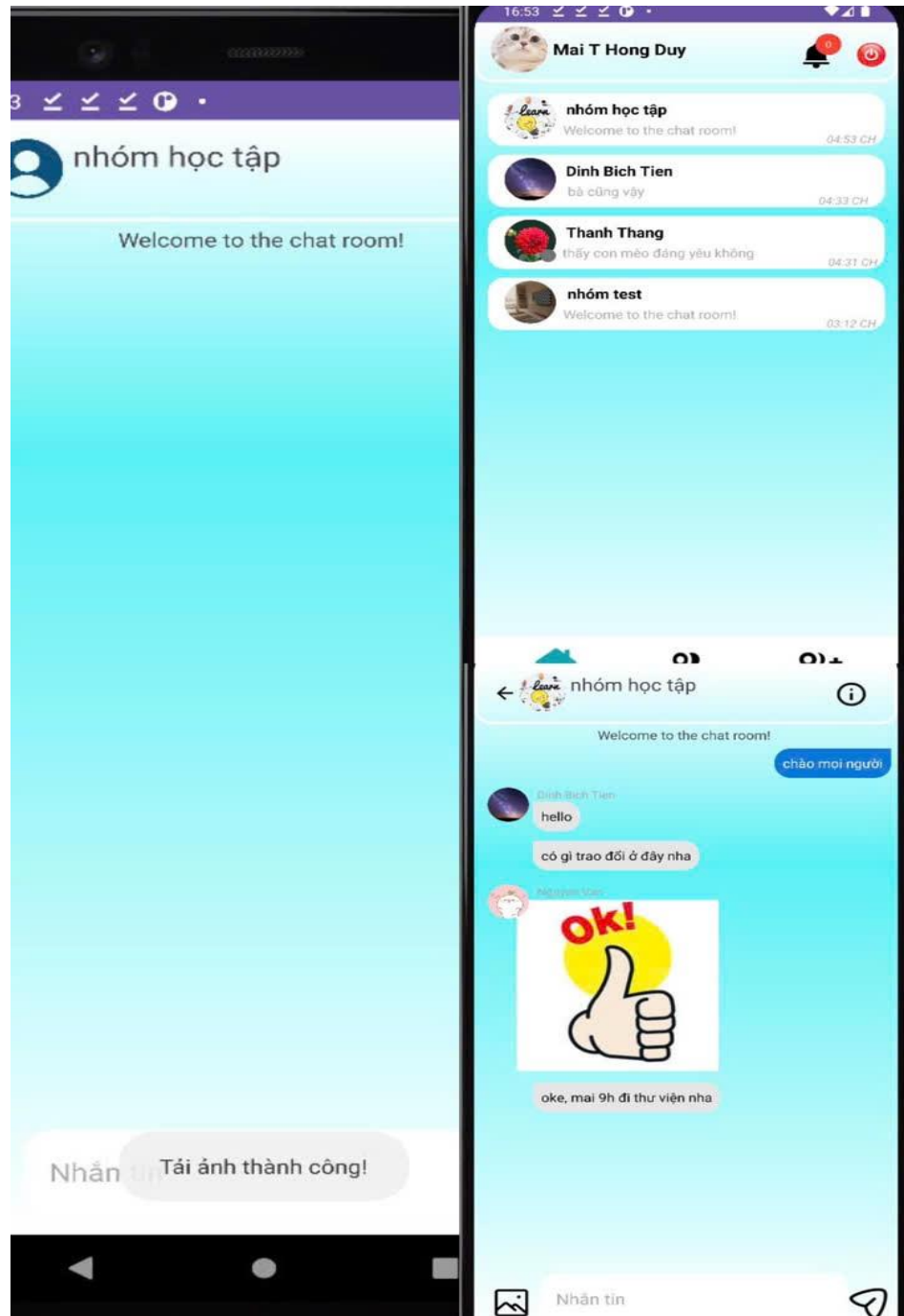
Chức năng "Xem lịch sử tin nhắn giữa các user đã kết bạn" cho phép người dùng xem lại các tin nhắn đã gửi và nhận trong cuộc trò chuyện với bạn bè đã kết bạn. Người dùng có thể mở cuộc trò chuyện, xem các tin nhắn trước đó, bao gồm cả văn bản và hình ảnh đã gửi.



3.2.1.12. Tạo nhóm chat, nhắn tin nhóm chat

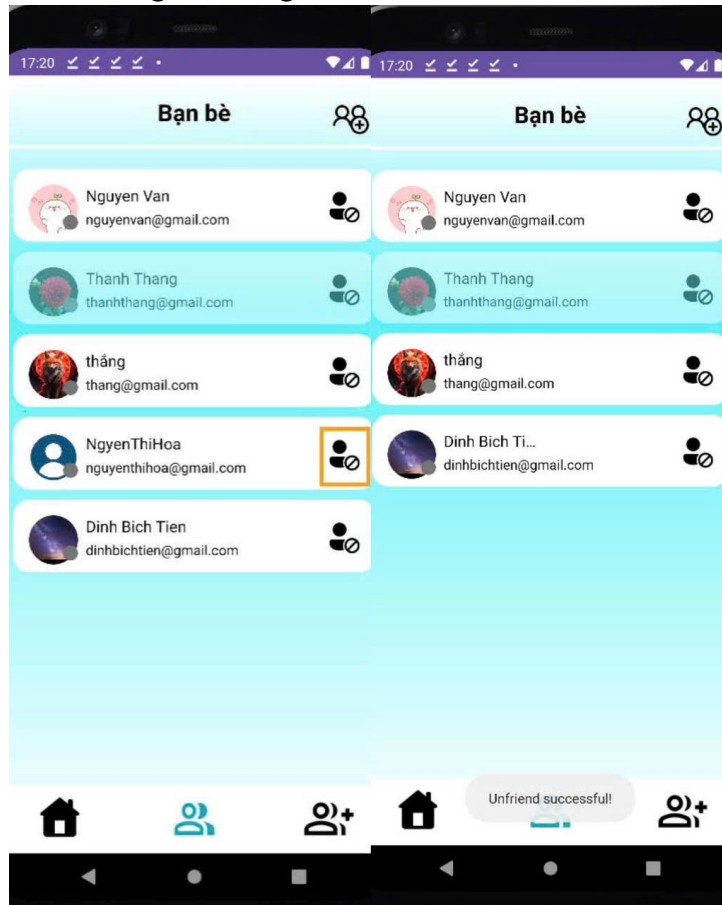
Chức năng tạo nhóm chat cho phép người dùng chọn bạn bè và tạo nhóm với ảnh đại diện. Sau khi nhóm được tạo, người dùng có thể nhắn tin nhóm, gửi văn bản và hình ảnh. Tin nhắn được lưu và hiển thị qua Firebase, và ảnh đại diện nhóm được lưu trên Cloudinary.





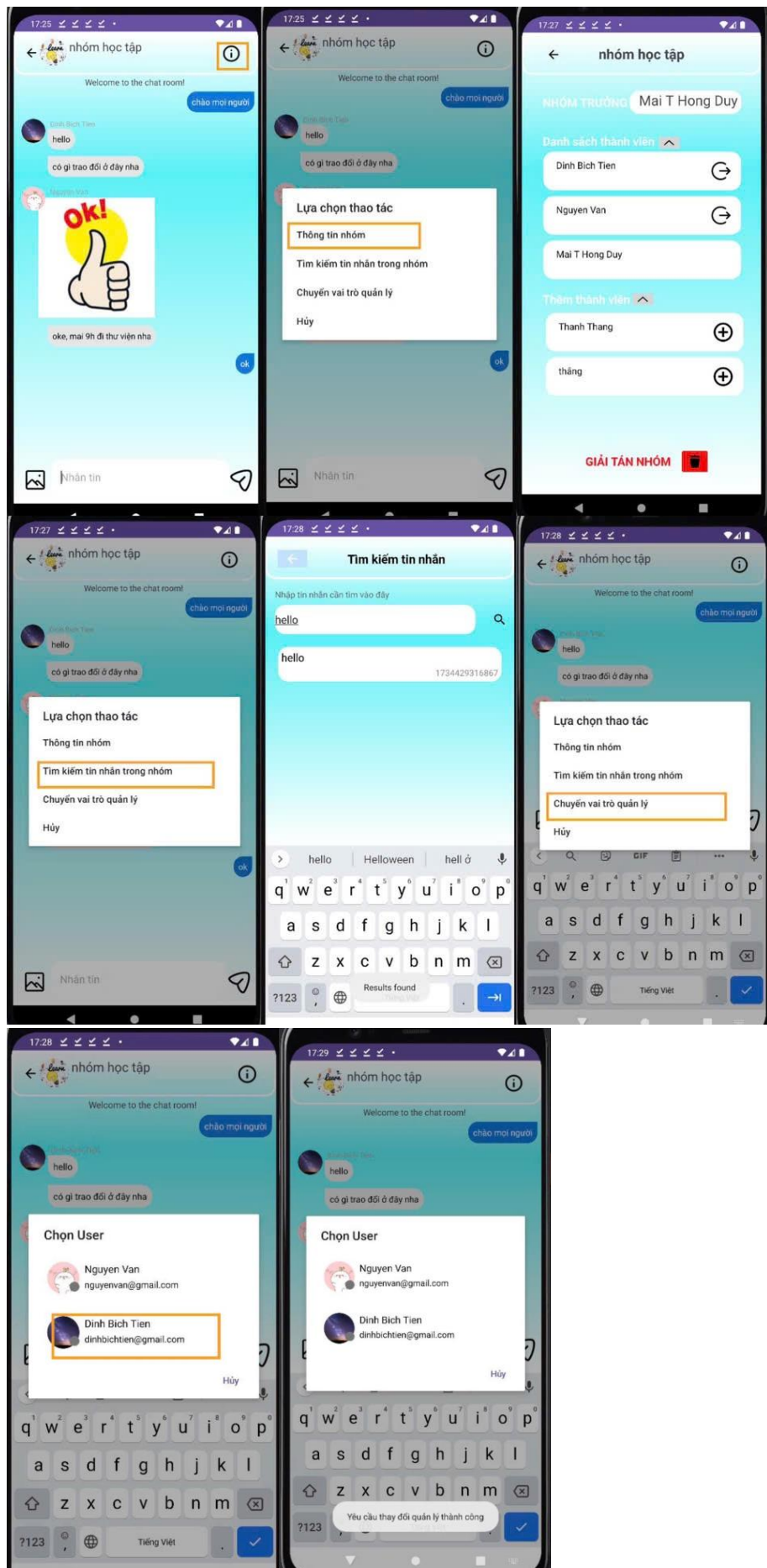
3.2.1.13. Hủy kết bạn

Chức năng này cho phép bạn hủy bỏ mối quan hệ bạn bè với người dùng, xóa kết nối và không còn là bạn bè trong hệ thống.



3.2.1.14. Gửi yêu cầu chuyển vai trò lên admin

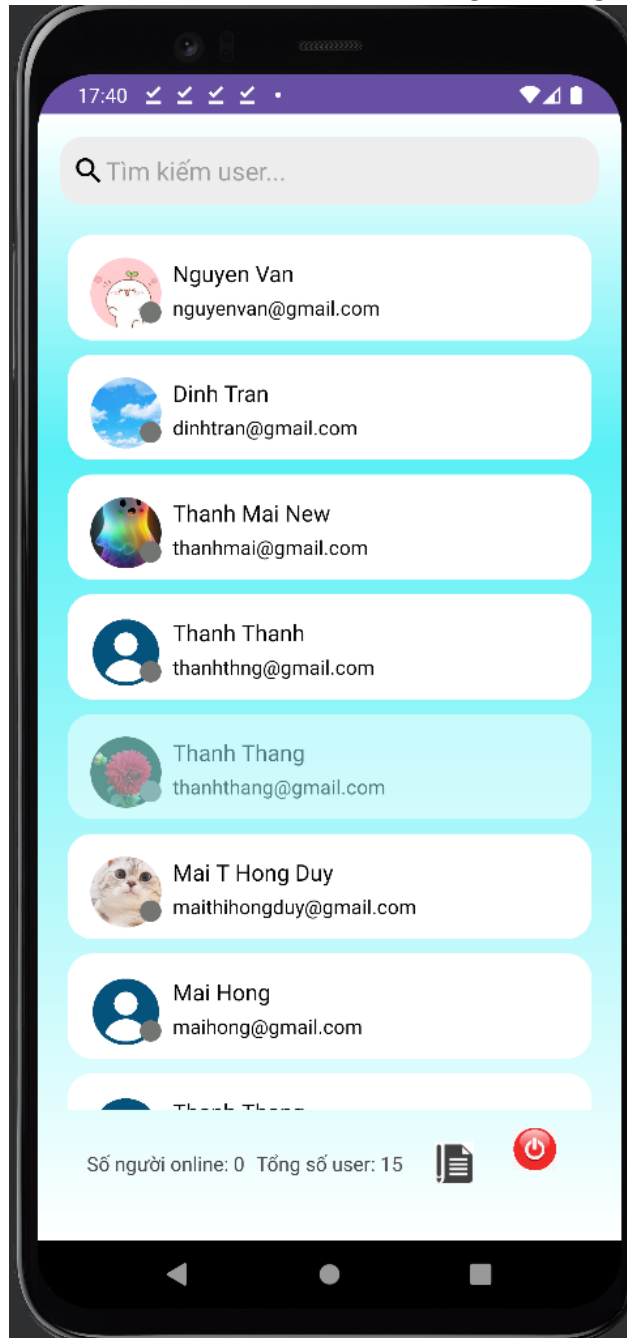
Chức năng "Gửi yêu cầu chuyển vai trò lên admin" cho phép xem thông tin nhóm, tìm tin nhắn trong nhóm, người tạo nhóm gửi yêu cầu thay đổi quyền sở hữu nhóm chat. Khi yêu cầu được gửi, thông tin sẽ được lưu trữ trong Firestore, và quản trị viên có thể chấp nhận hoặc từ chối yêu cầu này.



3.2.2. Chức năng của người quản trị:

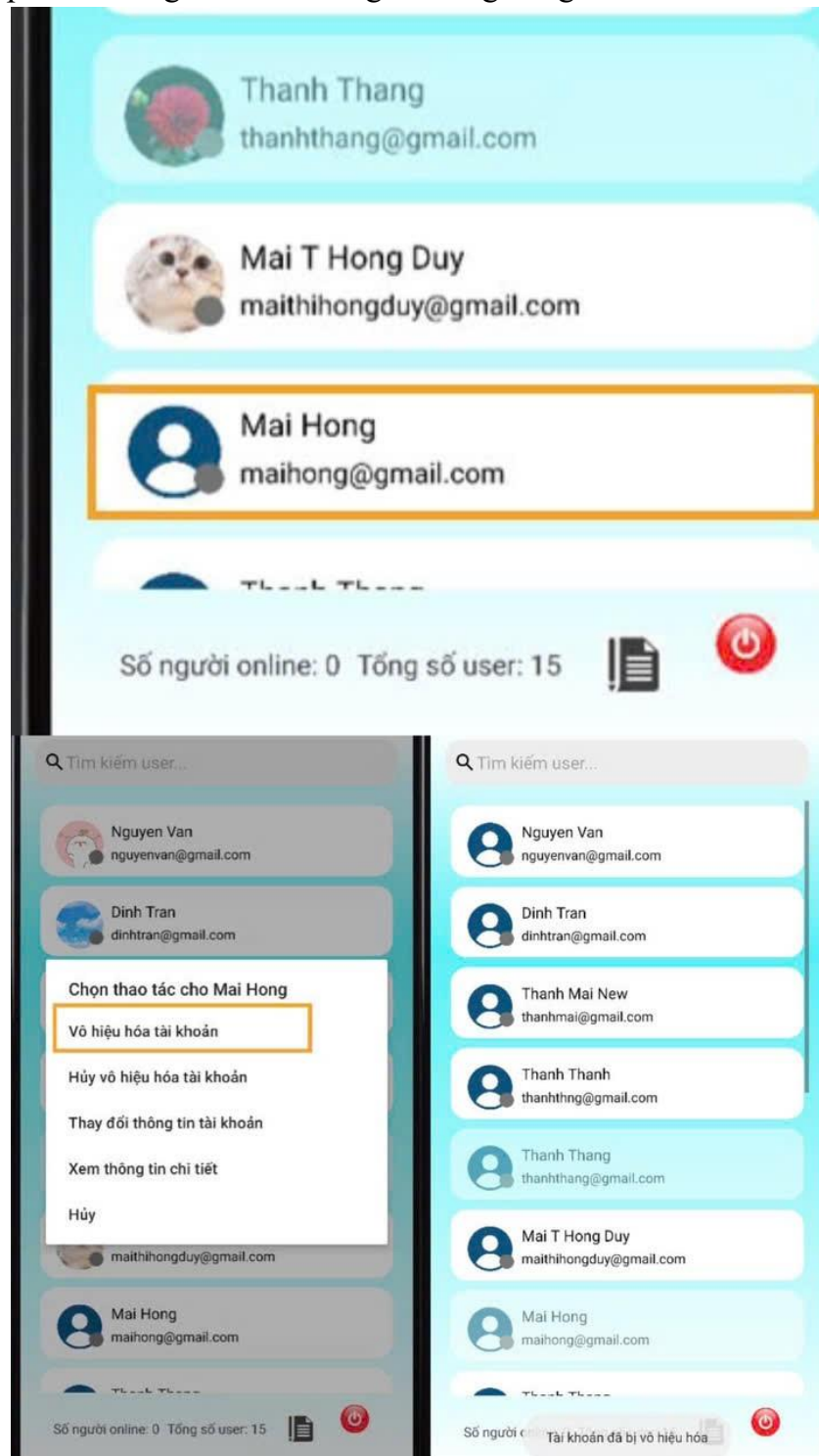
3.2.2.1. Xem danh sách người dùng

Hiển thị danh sách tất cả người dùng từ cơ sở dữ liệu, bao gồm tên, trạng thái và thông tin cơ bản. Có thể tìm kiếm và lọc theo tên người dùng.



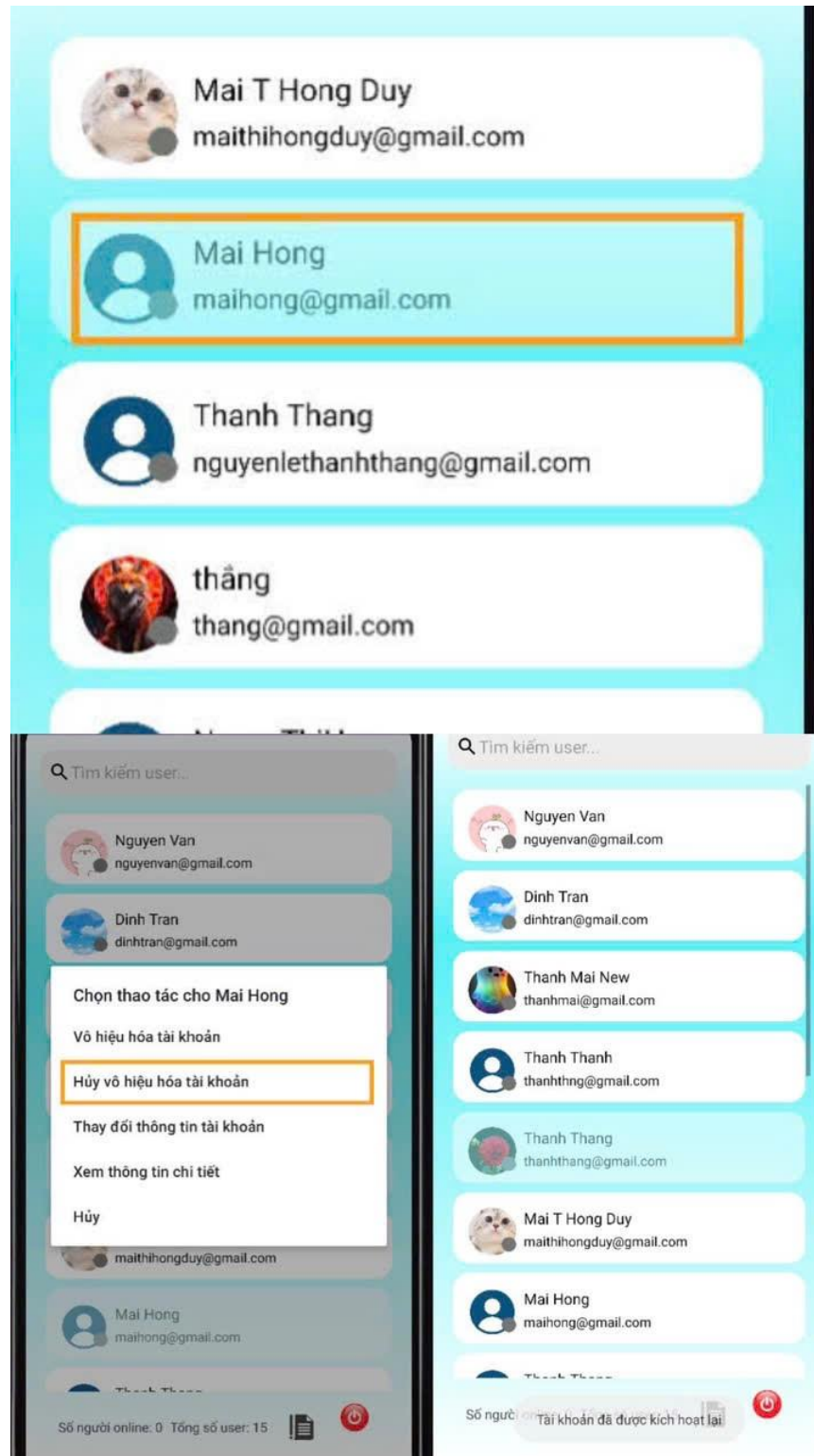
3.2.2.2. Vô hiệu hóa tài khoản người dùng

Tính năng cho phép quản trị viên vô hiệu hóa tài khoản của người dùng, ngừng quyền truy cập và sử dụng các chức năng của ứng dụng.



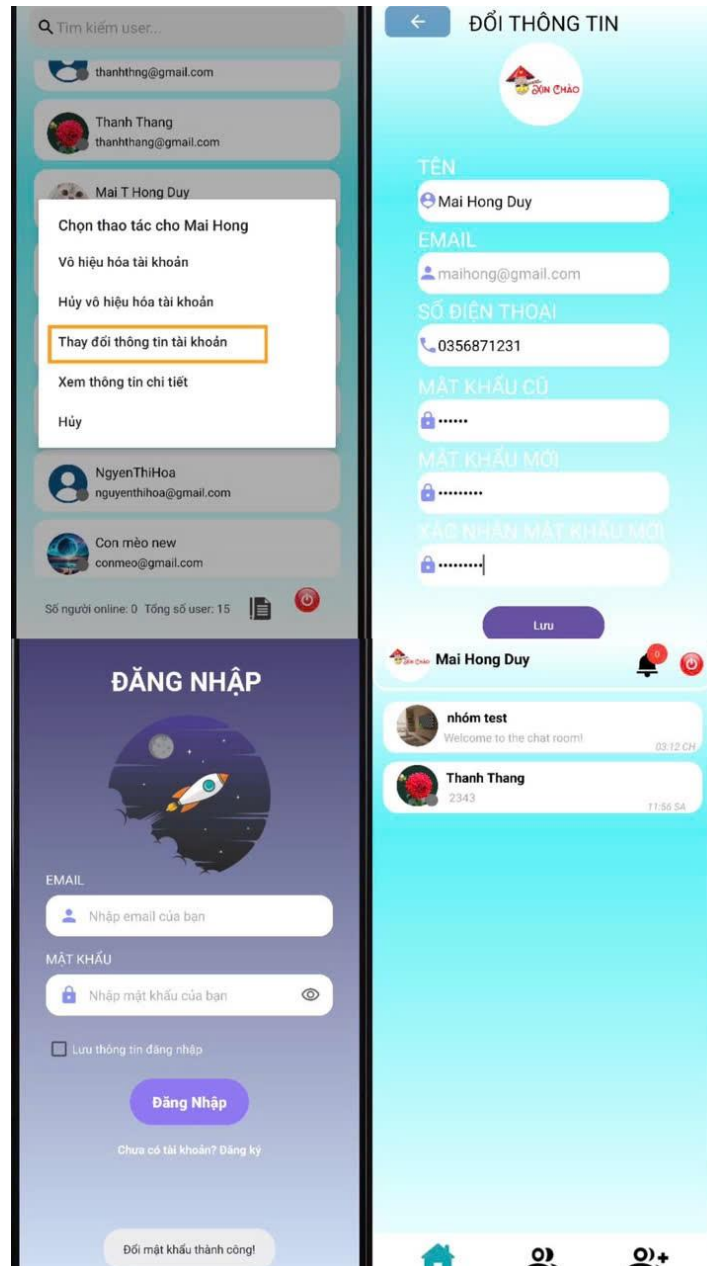
3.2.2.3. Hủy vô hiệu hóa tài khoản người dùng

Kích hoạt lại tài khoản người dùng đã bị vô hiệu hóa, cho phép người dùng tiếp tục sử dụng ứng dụng.



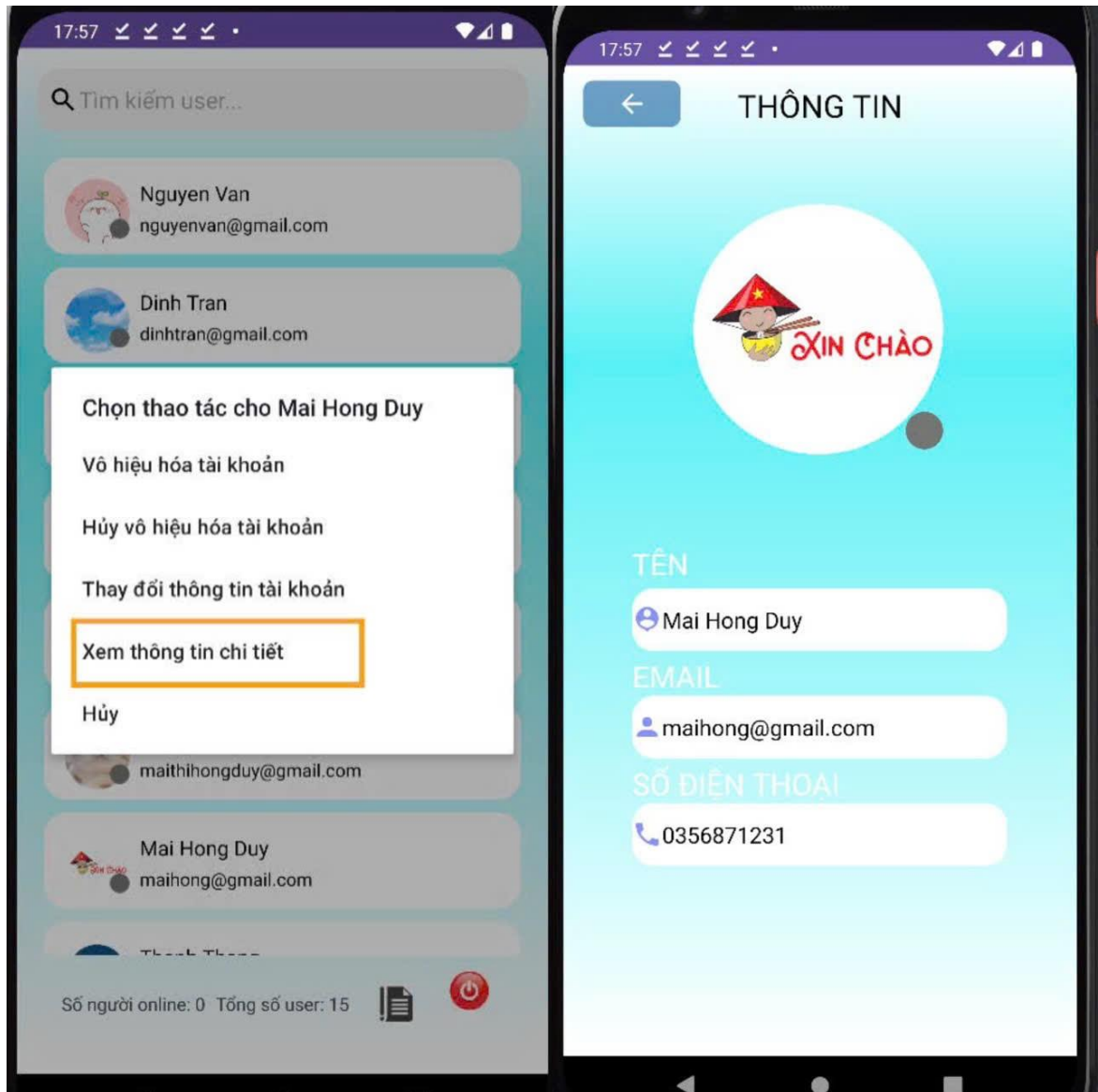
3.2.2.4. Chỉnh sửa thông tin người dùng

Cho phép quản trị viên thay đổi các thông tin cá nhân của người dùng như tên, ảnh đại diện, số điện thoại, email, v.v.



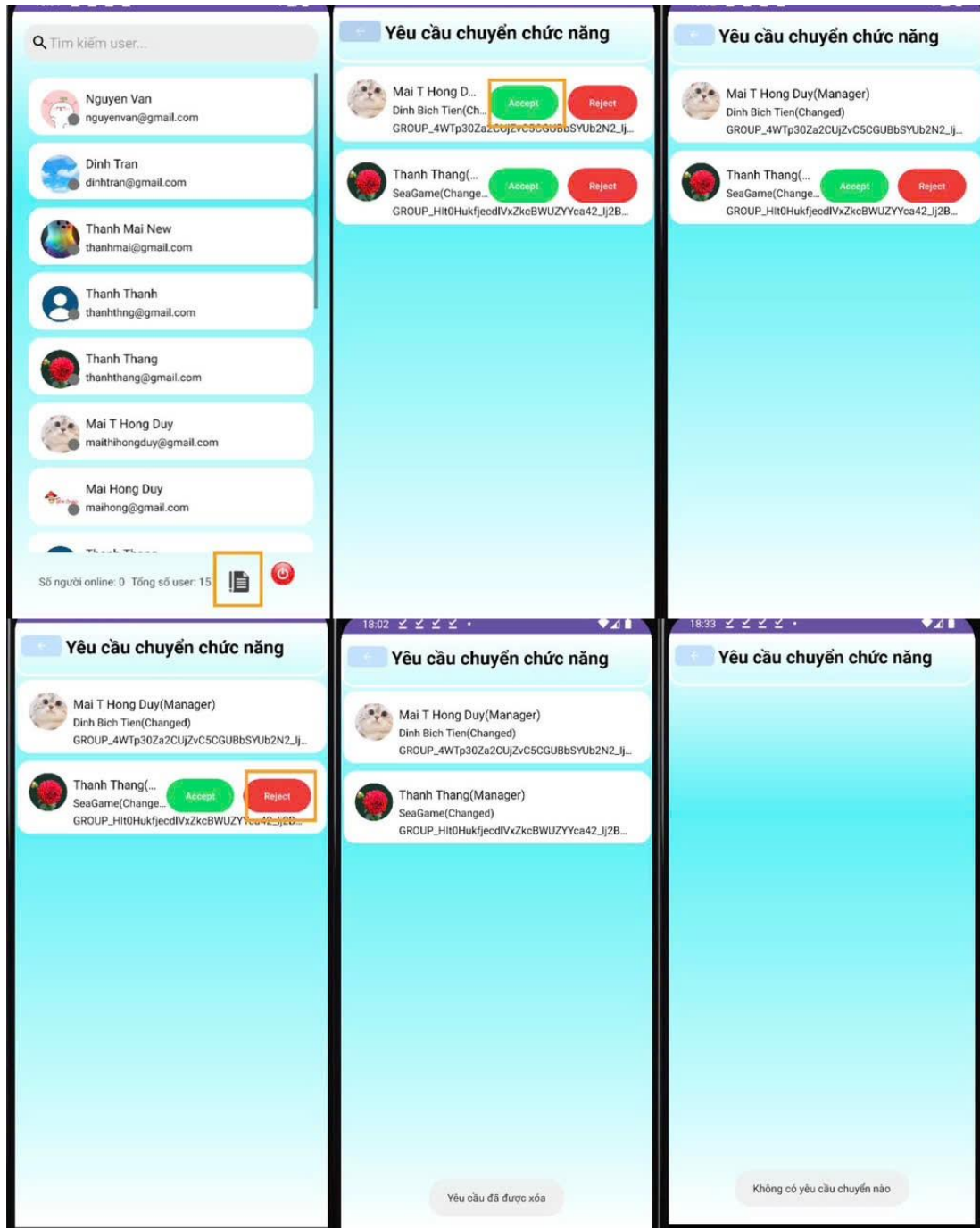
3.2.2.5. Xem thông tin chi tiết của người dùng

Hiển thị các thông tin chi tiết của người dùng như tên, ảnh đại diện, email, số điện thoại và các thông tin liên quan khác.



3.2.2.6. Xác nhận yêu cầu chuyển vai trò

Chức năng này cho phép quản trị viên xác nhận hoặc từ chối yêu cầu thay đổi vai trò của người dùng



3.3. Mô tả dữ liệu

Ứng dụng chat này sử dụng **Firestore** làm nền tảng chính để lưu trữ và quản lý dữ liệu người dùng cùng các tương tác trong ứng dụng. Dữ liệu được tổ chức một cách rõ ràng, đảm bảo hiệu suất và khả năng truy xuất nhanh chóng.

Dữ liệu người dùng được quản lý thông qua **Firestore Authentication** và **Firestore**. Mỗi người dùng sẽ có một tài khoản với các thông tin như email, mật khẩu, tên người dùng và ảnh đại diện. Mỗi tài khoản người dùng được gắn với một **ID duy nhất**, giúp hệ thống dễ dàng nhận diện và truy cập thông tin của từng cá nhân.

Mối quan hệ bạn bè giữa các người dùng được lưu trữ trong Firestore. Danh sách bạn bè của mỗi người dùng sẽ bao gồm các thông tin như **ID của người bạn**, trạng thái kết bạn và các hành động liên quan như gửi hoặc nhận lời mời. Trạng thái của mỗi quan hệ có thể là "đã kết bạn", "chờ xác nhận" hoặc "đã từ chối". Khi một người dùng gửi lời mời kết bạn, dữ liệu này cũng được lưu lại với ID người gửi, ID người nhận.

Tin nhắn giữa các người dùng hoặc trong nhóm chat được lưu trữ trong **Firestore** theo một cấu trúc riêng biệt. Mỗi tin nhắn sẽ chứa **ID người gửi**, **ID người nhận** và nội dung tin nhắn. Tin nhắn có thể là dạng **văn bản** hoặc **hình ảnh** (được lưu dưới dạng URL). Các cuộc trò chuyện giữa người dùng sẽ được sắp xếp theo cặp ID của hai người và hiển thị theo thứ tự thời gian. Điều này giúp người dùng dễ dàng theo dõi lịch sử trò chuyện và tìm kiếm thông tin khi cần.

Đối với tính năng nhóm chat, dữ liệu nhóm được tổ chức với một **ID nhóm duy nhất**. Mỗi nhóm chat sẽ chứa danh sách thành viên là các ID người dùng tham gia, cùng với lịch sử tin nhắn trong nhóm. Tin nhắn trong nhóm cũng có đầy đủ thông tin như ID người gửi và nội dung tin nhắn. Các nhóm chat đảm bảo người dùng có thể giao tiếp linh hoạt trong các nhóm nhỏ hoặc lớn tùy nhu cầu.

Để hỗ trợ quản lý vai trò trong nhóm, ứng dụng sử dụng **Firestore** để lưu thông tin liên quan đến quyền hạn của người dùng. Người dùng có thể gửi yêu cầu chuyển vai trò admin, và thông tin này sẽ được lưu trữ với các trường như **email**, **tên người dùng**, **vai trò** và **trạng thái** (ví dụ: "active"). Điều này giúp quản lý và xử lý các yêu cầu một cách hệ thống và minh bạch.

Toàn bộ dữ liệu trong ứng dụng được tổ chức và bảo mật trên Firestore, giúp đảm bảo an toàn thông tin người dùng và duy trì hiệu suất hoạt động của hệ thống. Cấu trúc dữ liệu đơn giản nhưng hiệu quả này cho phép mở rộng tính năng trong tương lai mà không làm ảnh hưởng đến hiệu năng của ứng dụng.

3.4. Mô tả sản phẩm

Sản phẩm của dự án là một ứng dụng chat được phát triển nhằm mang đến cho người dùng trải nghiệm giao tiếp nhanh chóng, mượt mà và thuận tiện. Ứng dụng cho phép người dùng dễ dàng kết nối, trò chuyện và quản lý các mối quan hệ bạn bè trong môi trường hiện đại và thân thiện.

Người dùng có thể bắt đầu sử dụng ứng dụng bằng cách đăng ký tài khoản với email và mật khẩu thông qua Firebase Authentication. Quá trình đăng ký đảm bảo tính bảo mật cao và đơn giản, giúp người dùng dễ dàng truy cập vào ứng dụng. Sau khi đăng nhập, người dùng sẽ được chuyển đến giao diện chính với đầy đủ tính năng như trò chuyện, tìm kiếm bạn bè và gửi lời mời kết bạn.

Tính năng quản lý bạn bè được thiết kế trực quan. Người dùng có thể tìm kiếm bạn bè thông qua tên người dùng và gửi lời mời kết bạn chỉ với vài thao tác đơn giản. Khi nhận được lời mời, người dùng có thể chấp nhận hoặc từ chối tùy ý. Danh sách bạn bè sẽ hiển thị thông tin về các mối quan hệ đã kết nối, kèm theo trạng thái online hoặc offline của từng người, giúp người dùng dễ dàng theo dõi hoạt động của bạn bè.

Chức năng nhắn tin là một trong những tính năng trọng tâm của ứng dụng. Người dùng có thể gửi tin nhắn văn bản hoặc hình ảnh đến bạn bè trong danh sách kết nối. Các tin nhắn này sẽ được lưu trữ trên Firebase Realtime Database, đảm bảo tính liên tục và dễ dàng truy xuất lịch sử trò chuyện. Tin nhắn được hiển thị theo thứ tự thời gian, giúp người dùng có thể theo dõi và quản lý cuộc trò chuyện một cách thuận tiện.

Ngoài ra, ứng dụng còn hỗ trợ tính năng nhóm chat, cho phép người dùng tạo nhóm và mời bạn bè cùng tham gia. Trong nhóm chat, các thành viên có thể gửi tin nhắn văn bản và hình ảnh tương tự như trong cuộc trò chuyện cá nhân. Dữ liệu tin nhắn của nhóm cũng được lưu trữ riêng biệt, đảm bảo tính rõ ràng và dễ dàng quản lý thông tin.

Một tính năng đặc biệt trong ứng dụng là gửi yêu cầu chuyển vai trò admin. Người dùng có thể gửi yêu cầu này lên hệ thống để xem xét và cấp quyền admin nếu được chấp thuận. Vai trò admin sẽ giúp người dùng có thêm quyền hạn trong việc quản lý nhóm chat và thực hiện các chức năng nâng cao khác.

Ứng dụng được xây dựng trên nền tảng Firebase, nơi mọi dữ liệu như tài khoản người dùng, thông tin bạn bè, tin nhắn và nhóm chat đều được lưu trữ và bảo mật chặt chẽ. Với thiết kế đơn giản nhưng đầy đủ tính năng, ứng dụng mang đến trải nghiệm giao tiếp tiện lợi và hiện đại cho người dùng.

CHƯƠNG 4: KẾT LUẬN

4.1. Kết quả đạt được

Sau quá trình phát triển ứng dụng chat trực tuyến đã mang lại nhiều bài học và kinh nghiệm quý giá cho nhóm. Đầu tiên, nhóm đã nắm được các kiến thức cơ bản và một ít kiến thức mở rộng về lập trình ứng dụng di động, đặc biệt là cách sử dụng Android Studio và tích hợp Firebase để quản lý cơ sở dữ liệu. Việc xây dựng các tính năng như đăng ký, đăng nhập, quản lý bạn bè, và nhắn tin đã giúp nhóm hiểu rõ hơn về thiết kế luồng hoạt động của ứng dụng và tổ chức logic chương trình một cách hiệu quả.

Quá trình kiểm thử ứng dụng đã giúp nhóm rèn luyện khả năng phát hiện và xử lý lỗi, từ đó nâng cao kỹ năng giải quyết vấn đề và làm việc nhóm hiệu quả. Nhìn chung, dự án không chỉ giúp nhóm thực hành và áp dụng những kiến thức đã học mà còn mở ra cơ hội khám phá các công nghệ khác, tạo nền tảng kiến thức và kinh nghiệm vững vàng cho các đề tài phát triển sau này..

4.2. Hạn chế của đề tài

Hạn chế của đề tài là vẫn còn nhiều điểm chưa được hoàn thiện để đáp ứng tốt hơn nhu cầu sử dụng của người dùng. Cụ thể, hệ thống thông báo chưa hoạt động hiệu quả, gây khó khăn trong việc cung cấp thông tin kịp thời. Việc quản lý dữ liệu vẫn còn hạn chế, dẫn đến khó khăn trong việc tổ chức và truy xuất thông tin. Bên cạnh đó, tính năng bảo mật dữ liệu chưa được tối ưu, làm giảm độ tin cậy trong việc bảo vệ thông tin cá nhân của người dùng. Đây là những vấn đề cần được khắc phục để ứng dụng trở nên hoàn thiện và hữu ích hơn.

4.3. Phát triển đề tài trong tương lai

Trong tương lai, đề tài sẽ được phát triển thêm để hoàn thiện các tính năng hiện có và mở rộng khả năng ứng dụng. Trước tiên, cần cải thiện hệ thống thông báo để đảm bảo việc cung cấp thông tin kịp thời, chính xác, và thuận tiện cho người dùng. Đồng thời, việc quản lý dữ liệu sẽ được tối ưu hóa nhằm tăng tốc độ truy xuất, cải thiện khả năng xử lý dữ liệu lớn, và nâng cao tính ổn định của hệ thống. Ngoài ra, nhóm sẽ chú trọng vấn đề bảo mật bằng cách tìm cách áp dụng các giao thức mã hóa tiên tiến và cơ chế xác thực mạnh mẽ hơn, nhằm bảo vệ thông tin người dùng khỏi các nguy cơ an ninh mạng.

Chúng em rất mong nhận được sự góp ý từ thầy qua bản báo cáo này, để có thể tìm ra những thiếu sót và định hướng phát triển thêm cho đề tài trong thời gian tới, giúp sản phẩm ngày càng hoàn thiện và hữu ích hơn.

TÀI LIỆU THAM KHẢO

Slide môn lập trình trên thiết bị di động

BẢNG PHÂN CÔNG CÔNG VIỆC

Họ và tên	Công việc	Mức độ hoàn thành
Nguyễn Lê Thanh Thắng 2251012127	Phát triển giao diện và chức năng báo tình trạng online của người dùng, xây dựng chức năng chat giữa các user đã kết bạn (gồm văn bản và hình ảnh), xem lịch sử tin nhắn giữa các user đã kết bạn và tìm kiếm các đoạn chat có chứa thông tin cần tìm. Xây dựng chức năng tạo nhóm chat, hỗ trợ chat trong nhóm, cùng với quyền thêm, xóa thành viên trong nhóm và quyền xóa nhóm chat cho manager. Chính sửa và trang trí toàn bộ giao diện.	100%
Mai Thị Hồng Duy 2251012047	Phát triển giao diện và chức năng đăng nhập, đăng ký tài khoản cho người dùng, cùng với chức năng đăng nhập cho admin và sử dụng Firebase Authentication để quản lý và xử lý tài khoản người dùng. Hỗ trợ giao diện admin, xây dựng chức năng cho admin để vô hiệu hóa và bỏ vô hiệu hóa tài khoản người dùng. Thực hiện kiểm thử ứng dụng, kiểm tra các chức năng trong ứng dụng.	100%
Đinh Bích Tiên 2251012132	Phát triển giao diện và chức năng kết bạn giữa các user, tìm kiếm bạn bè bằng tên user, hiển thị danh sách bạn bè và lời mời kết bạn, cùng với chức năng cho phép người dùng thay đổi thông tin cá nhân. Xây dựng chức năng gửi yêu cầu đổi người quản lý nhóm chat. Giao diện admin và chức năng thay đổi, xem thông tin tài khoản user và quyền xử lý các yêu cầu chuyển người quản lý trong các nhóm chat của admin.	100%