

## LỜI CAM ĐOAN

*Tôi xin cam đoan:*

- 1 Những nội dung trong luận văn này là do tôi thực hiện dưới sự hướng dẫn trực tiếp của thầy Nguyễn Kim Hưng*
- 2 Mọi tham khảo dùng trong luận văn đều được trích dẫn rõ ràng tên tác giả, tên công trình, thời gian, địa điểm công bố.*
- 3 Mọi sao chép không hợp lệ, vi phạm quy chế đào tạo, hay gian trá, tôi xin chịu hoàn toàn trách nhiệm.*

*Sinh viên thực hiện*

*Đỗ Thị Mỹ Nhi*

# MỤC LỤC

TRANG PHỤ BÌA

LỜI CAM ĐOAN

MỤC LỤC .....	1
DANH MỤC TỪ VIẾT TẮT .....	3
DANH MỤC HÌNH ẢNH.....	4
Chương 1. TỔNG QUAN .....	5
1.1 Giới thiệu đề tài: .....	5
1.1.1 Sự cần thiết của đề tài: .....	5
1.1.2 Mục tiêu của đề tài .....	6
Chương 2. CƠ SỞ LÝ THUYẾT.....	10
2.1 Tổng quan về hệ điều hành Android:.....	10
2.1.1 Giới thiệu về hệ điều hành Android .....	10
2.1.2 Kiến trúc cơ bản của hệ điều hành Android.....	11
2.2 Giới thiệu về ngôn ngữ lập trình Java:.....	12
2.3 Các khái niệm cơ bản trong lập trình ứng dụng Android: .....	13
2.3.1 Activity.....	13
2.3.2 Service .....	14
2.3.3 Content Provider .....	14
2.3.4 Broadcast Receive .....	15
2.3.5 View .....	15
2.3.6 Intent.....	15
2.4 Firebase: .....	15
2.4.1 Giới thiệu.....	15
2.4.2 Tính năng cơ bản.....	16
2.4.3 Realtime Database.....	16
2.4.4 Firebase Authentication.....	16
2.4.5 Firebase Hosting.....	17
2.4.6 Cloud Messaging.....	17
2.4.7 Firebase Storage .....	17
2.4.8 Firebase Test Lab và Crash Reporting .....	18
2.5 Phân tích thiết kế hệ thống:.....	18

2.5.1	Phân tích và đặt tả yêu cầu .....	18
2.5.1.1	Mô tả bài toán: .....	18
2.5.1.2	Phân tích và đặc tả yêu cầu .....	18
2.5.2	Phân tích thiết kế hệ thống: .....	19
2.5.2.1	Biểu đồ use case tổng quát.....	19
2.5.2.2	Sơ đồ luồng dữ liệu (DFD) .....	20
2.5.2.3	Biểu đồ lớp (ERD):.....	21
Chương 3. KẾT QUẢ THỰC NGHIỆM .....		22
3.1	Kết quả triển khai và đánh giá: .....	22
3.1.1	Kết quả triển khai một số chức năng chính của ứng dụng Re-Heaven .....	22
3.1.2	Kết quả triển khai một số chức năng chính của ứng dụng Admin (Re-Heaven Administer):.....	26
3.2	Đánh giá kết quả thực hiện: .....	29
3.2.1	Kết quả đạt được: .....	29
3.2.2	Kết quả chưa đạt được: .....	30
Chương 4. KẾT LUẬN VÀ KIẾN NGHỊ.....		31
4.1	Kết quả đạt được: .....	31
4.2	Ưu điểm của đề tài: .....	31
4.3	Hạn chế còn tồn tại: .....	31
4.4	Kết luận: .....	31
4.5	Hướng phát triển: .....	32
TÀI LIỆU THAM KHẢO .....		33

## DANH MỤC TỪ VIẾT TẮT

Từ viết tắt	Diễn giải
API	Application Programming Interface
BQL	Ban Quản Lý của tòa nhà
JVM	Dalvik Virtual Machine
JSON	JavaScript Object Noattion
URL	Uniform Resource Locator
OOP	Object Oriented Programming
SSL	Secure Sockets Layer
XML	Extensible Markup Language

## DANH MỤC HÌNH ẢNH

Hình 1.1 Cư dân xem bảng thông báo của cư dân.....	6
Hình 1.2- Logo của ứng dụng.....	7
Hình 2.1 -Tổng quan về hệ điều hành Android .....	11
Hình 2.2 – Kiến trúc hệ điều hành Android .....	12
Hình 2.3 - Vòng đời của một Activity .....	13
Hình 2.4 – Các tính năng cơ bản của Firebase .....	16
Hình 2.5 – Mô tả về tính năng Firebase Mesaging .....	17
Hình 2.6 – Biểu đồ use case tổng quát chức năng của người dùng.....	19
Hình 2.7 - Biểu đồ use case tổng quát chức năng của admin.....	20
Hình 2.8 – Biểu đồ use case phân rã .....	20
Hình 2.9 - Biểu đồ hoạt động của ứng dụng.....	21
Hình 2.10 – Sơ đồ ERD quản lý cư dân .....	21
Hình 3.1 – Màn hình hiển thị giao diện đăng nhập của app admin.....	22
Hình 3.2 – Màn hình hiển thị giao diện đăng nhập của app user .....	22
Hình 3.3 – Màn hình hiển thị danh sách các tin tức .....	23
Hình 3.4 – Màn hình hiển thị thanh công cụ và chức năng tìm kiếm .....	23
Hình 3.5 - Màn hình hiển thị giao diện nhắn tin của user với admin.....	24
Hình 3.6 – Màn hình hiển thị danh sách các thông báo .....	25
Hình 3.7 – Màn hình hiển thị hóa đơn dịch vụ hàng tháng .....	25
Hình 3.8 – Màn hình hiển thị giao diện hình thức thanh toán của thanh toán hóa đơn	26
Hình 3.9 - Màn hình hiển thị giao diện admin có thể xem và chỉnh sửa hoặc xóa thông báo đã được đăng.....	27
Hình 3.10 – Màn hình hiển thị thanh công cụ tìm kiếm của admin .....	27
Hình 3.11 - Màn hình hiển thị giao diện chứa danh sách các tài khoản user đã nhắn tin cho admin .....	28
Hình 3.12 – Màn hình hiển thị khung chat riêng gcủa admin với 1 tài khoản user .....	28
Hình 3.13 – Màn hình hiển thị menu quản lý của admin .....	29
Hình 3.14 – Màn hình hiển thị giao diện đăng tin tức của admin .....	29

# Chương 1. TỔNG QUAN

## 1.1 Giới thiệu đề tài:

Trong những năm qua, căn hộ chung cư luôn là phân khúc quan trọng dẫn dắt thị trường bất động sản trên cả nước nói chung và tại TP. Hồ Chí Minh nói riêng. Là một trung tâm kinh tế lớn và là thành phố đông dân cư nhất cả nước, TP. Hồ Chí Minh đang đứng trước một thực tế là nhu cầu mua căn hộ chung cư tăng cao. Để giữ ổn định cho phân khúc thị trường căn hộ chung cư, thời gian tới, các doanh nghiệp bất động sản cần khắc phục những tồn tại, hạn chế cũng như tìm hiểu rõ hơn về nhu cầu, thị hiếu của hộ gia đình để có phương án.

Hầu hết những khách hàng lựa chọn mua căn hộ chung cư hiện tại đa phần đều là những khách hàng trẻ, trong đó số lượng khách hàng ở độ tuổi 8x và 9x chiếm tới 35%, là lực lượng lao động chính có khả năng tự chủ tài chính, có nhu cầu cao về việc sở hữu căn hộ riêng cho bản thân hoặc gia đình của mình.

Thực tế cuộc sống hiện nay cũng cho thấy hầu hết những thế hệ trẻ đều ưa chuộng các căn hộ thông minh có khả năng tiện ích cao, diện tích vừa và nhỏ để chọn lựa làm nơi an cư lâu dài.

Để hòa nhập và phát triển trên nền công nghệ 4.0, cũng như bắt kịp xu thế căn hộ chung cư hiện nay, em quyết định xây dựng “ứng dụng quản lý cư dân trong chung cư”.

Sử dụng ứng dụng quản lý cư dân là giải pháp quản lý toàn diện và hiệu quả cho các tòa nhà chung cư. Hệ thống phần mềm giúp cả cư dân lẫn ban quản lý tòa nhà có thể dễ dàng tra cứu thông tin, quản lý người dùng, xử lý yêu cầu của cư dân và kết nối với cư dân nhanh chóng tiện dụng mọi lúc, mọi nơi chỉ thông qua điện thoại thông minh.

### 1.1.1 Sự cần thiết của đề tài:

Hiện nay, vẫn còn khá nhiều tòa nhà chung cư sử dụng phương pháp quản lý thông thường. Để quản lý các cư dân của chung cư, BQL vẫn phải soạn thảo văn bản, thông báo dán ở bảng thông báo hoặc trước cửa thang máy.



Hình 1.1 Cư dân xem bảng thông báo của cư dân

Hàng tháng, chi phiếu tiền điện, tiền phí dịch vụ vẫn được nhân viên quản lý cất công gửi đến từng nhà.

Căn hộ xảy ra vấn đề cần bảo trì, sửa chữa thì cư dân phải di chuyển đến phòng ban quản lý thông báo hoặc viết đơn đề BQL lưu lại thông tin khách hàng.

#### *1.1.2 Mục tiêu của đề tài*

- Quản lý thông tin khách hàng.
- Quản lý công nợ khách hàng hiệu quả
- Quản lý dịch vụ, bảo trì, sửa chữa trong tòa nhà.
- Báo cáo hoạt động của tòa nhà theo định kỳ.
- Cư dân liên hệ với BQL một cách nhanh chóng, tiện lợi.

### **1.2 Giới thiệu dự án “Ứng dụng quản lý”:**

- Tên ứng dụng: Re-Heaven (Heaven Residents)
  - + Heaven: Tên của Chung cư mà dự án đang thực hiện
  - + Residents: Cư dân

- Logo ứng dụng:



Hình 1.2- Logo của ứng dụng

- Các tính năng:

- + Quản lý thông tin khách hàng (cư dân), căn hộ
- + Chatbox trực tiếp cho BQL, nhân viên chăm sóc khách hàng
- + Quản lý, nhắc nhở khách hàng (cư dân) khi còn tồn tại công nợ
- + Thông báo phí tiền điện, phí dịch vụ đến khách hàng (cư dân)
- + Gửi các thông báo từ BQL đến khách hàng (cư dân) thông qua tính năng newfeed

- Tiện ích mang đến:

+ Tương tác dễ dàng, hiệu quả: Thay vì phải in và dán thông báo trên bảng tin, lúc này ban quản lý có thể gửi trực tiếp các thông tin tòa nhà, sự kiện họp ... đến app quản lý trên điện thoại của cư dân. Cư dân tiếp nhận thông báo của ban quản lý, phản hồi ý kiến về các thông tin, căn cứ vào đó ban quản lý sẽ điều chỉnh phù hợp với tình hình thực tế. Không chỉ vậy, quá trình tiếp nhận và xử lý kiến nghị của ban quản lý sẽ đơn giản hơn khi cư dân phản hồi trực tiếp trên app di động bằng tính năng kiến nghị, góp phần nâng cao chất lượng dịch vụ quản lý tòa nhà đồng thời gắn kết cộng đồng cư dân, giảm tải các mâu thuẫn không đáng có.

+ Sự chăm sóc đặc biệt: Sự chăm sóc 1-1 giữa 1 nhân viên trong đội ngũ nhân viên chăm sóc khách hàng với 1 khách hàng. Trả lời nhanh chóng với khách hàng đảm bảo sự hài lòng và tin tưởng.

+ Thanh toán tiện lợi: Trong những năm gần đây, thanh toán điện tử đang trở thành một xu hướng tương lai vì tính ưu việt trong phương thức thanh toán. Cư dân quá bận rộn, cư dân đi công tác, cư dân làm việc giờ hành chính không có thời gian đi đóng



tiền điện, tiền nước, internet... Tất cả đều được **Ứng dụng quản lý** hỗ trợ thanh toán trực tuyến thông qua các cổng thanh toán uy tín, an toàn hàng đầu hiện nay như VNPays, Momo, AirPay...

+ Trải nghiệm thông minh, miễn phí: Ứng dụng đang được coi là phương tiện hiện đại và hiệu quả nhất hiện nay. Đây là lựa chọn thông minh cho khách hàng. Trải nghiệm này sẽ được miễn phí trên CH Play dành cho smartphone chạy trên nền tảng hệ điều hành Android.

- Ý nghĩa: Mang một xu hướng mới cho hệ thống quản lý tòa nhà chung cư, an toàn, tiện lợi, hiện đại,... Cho khách hàng cái nhìn mới khu chung cư thời đại 4.0

- Đối tượng sử dụng & nghiên cứu: Toàn thể cư dân trong chung cư Heaven Riverview và đội ngũ nhân viên trong bộ phận quản lý bao gồm chăm sóc khách hàng, lễ tân, bảo vệ,...

- Phạm vi giới hạn: không gian, thời gian, lĩnh vực thực hiện dự án

Ví dụ: Là cư dân, nhân viên Heaven Riverview,... có tên trong danh sách Quản lý cư dân của bộ phận BQL.

### **1.3. Phương pháp thực hiện:**

#### *1.3.1 Kỹ thuật và phương pháp triển khai*

Kỹ thuật thuật và phương pháp triển khai để xây dựng ứng dụng quản lý cư dân trong chung cư trên nền tảng Android:

- Phân tích thực trạng thực tế để lựa chọn đề tài, lựa chọn hệ điều hành, ngôn ngữ phát triển.
- Tìm hiểu về ngôn ngữ lập trình trên nền tảng Android.
- Phân tích, thiết kế hệ thống.
- Thiết kế, phác thảo giao diện, thiết kế Cơ sở dữ liệu cho ứng dụng.
- Xây dựng chương trình Android.
- Kiểm thử chức năng.

#### *1.3.2 Ngôn ngữ sử dụng*

Ứng dụng quản lý cư dân online trên hệ điều hành Android sử dụng ngôn ngữ Java-Android để xây dựng ứng dụng.

#### *1.3.3 Môi trường phát triển*

Sử dụng phần mềm Android Studio để phát triển ứng dụng.

#### **1.4. Dự kiến kết quả đạt được:**

Dự kiến sẽ đạt được kết quả như sau:

- Phân tích hệ thống ứng dụng Quản lý cư dân trong chung cư trên nền tảng Android.
- Xây dựng ứng dụng với các chức năng thanh toán online và các chức năng quản lý cho người quản lý.

#### **1.5. Cấu trúc của dự án:**

**Chương 1. Tổng quan:** Giới thiệu về đề tài; lý do chọn đề tài; mục đích, ý nghĩa chọn đề tài và giới thiệu về các phương pháp triển khai, ngôn ngữ, môi trường sử dụng.

**Chương 2. Cơ sở lý thuyết:** Giới thiệu về hệ điều hành Android, cơ sở lý thuyết của ngôn ngữ thực hiện (Java-Android). Phân tích hệ thống bao gồm các phương pháp đặc tả người dùng, xây dựng các sơ đồ mô tả hệ thống, qua đó có cách nhìn tổng quan về hệ thống hơn.

**Chương 3. Kết quả thực nghiệm:** Trình bày quá trình thực hiện, triển khai, kết quả demo và đánh giá kết quả.

**Chương 4. Kết luận và kiến nghị:** Trình bày kết quả đạt được khi xây dựng chương trình và các hạn chế còn tồn tại, qua đó đánh giá lại sản phẩm và nêu các hướng phát triển.

**Tài liệu tham khảo.**

## **Chương 2. CƠ SỞ LÝ THUYẾT**

### **2.1 Tổng quan về hệ điều hành Android:**

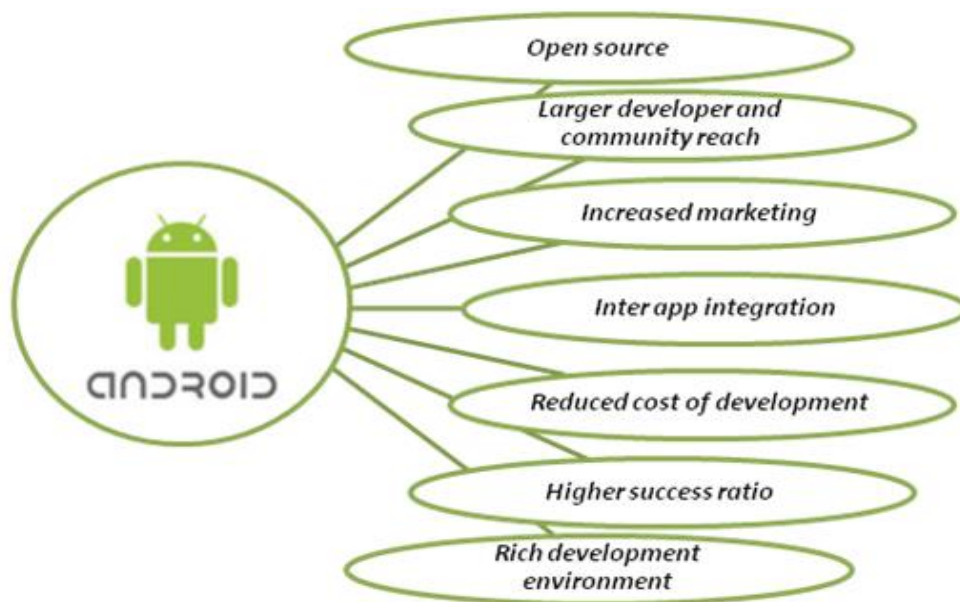
#### *2.1.1 Giới thiệu về hệ điều hành Android*

Android là một hệ điều hành dựa trên nền tảng Linux được thiết kế dành cho các thiết bị di động có màn hình cảm ứng như điện thoại thông minh và các thiết bị máy tính bảng. Ban đầu, Android được phát triển bởi tổng công ty Android, với sự hỗ trợ tài chính từ Google và sau này được chính Google mua lại vào năm 2005. Android ra mắt năm 2007 cùng với tuyên bố thành lập liên minh thiết bị cảm tay mở cho các thiết bị di động.

Android có mã nguồn mở và Google phát hành mã nguồn theo Giấy phép Apache. Chính mã nguồn mở cùng với một giấy phép không có nhiều ràng buộc đã cho phép các nhà phát triển thiết bị, mạng di động và các lập trình viên nhiệt huyết được điều chỉnh và phân phối Android một cách tự do.

Các ứng dụng cho Android được phát triển bằng ngôn ngữ Java sử dụng bộ phát triển phần mềm Android (SDK). SDK bao gồm một bộ đầy đủ các công cụ dùng để phát triển gồm có công cụ gỡ lỗi, thư viện phần mềm, bộ giả lập điện thoại dựa trên QEMU, tài liệu hướng dẫn, mã nguồn mẫu, và hướng dẫn từng bước. Môi trường phát triển tích hợp (IDE) được hỗ trợ chính thức là Eclipse sử dụng phần bổ sung Android Development Tools (ADT). Các công cụ phát triển khác cũng có sẵn, gồm có Bộ phát triển gốc dành cho các ứng dụng hoặc phần mở rộng viết bằng C hoặc C++, Google App Inventor, một môi trường đồ họa cho những nhà lập trình mới bắt đầu, và nhiều nền tảng ứng dụng web di động đa nền tảng phong phú.

Android được Google tự phát triển riêng cho đến khi những thay đổi và cập nhật đã hoàn thiện, khi đó mã nguồn mới được công khai. Mã nguồn này, nếu không sửa đổi, chỉ chạy trên một số thiết bị, thường là thiết bị thuộc dòng Nexus. Có nhiều thiết bị có chứa những thành phần được giữ bản quyền do nhà sản xuất đặt vào thiết bị Android của họ.



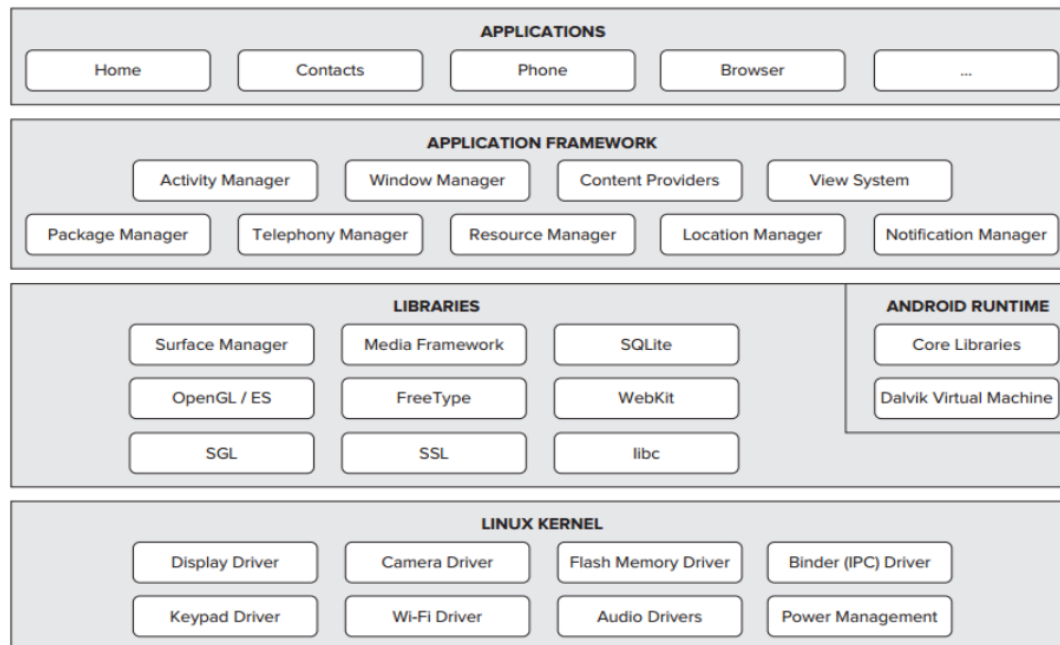
Hình 2.1 -Tổng quan về hệ điều hành Android

### 2.1.2 Kiến trúc cơ bản của hệ điều hành Android

Android gồm 5 phần chính sau được chứa trong 4 lớp:

- **Nhân Linux:** Đây là nhân nền tảng mà hệ điều hành Android dựa vào nó để phát triển. Đây là lớp chứa tất cả các thiết bị giao tiếp ở mức thấp dùng để điều khiển các phần cứng khác trên thiết bị Android.
- **Thư viện:** Chứa tất cả các mã cái mà cung cấp các tính năng chính của hệ điều hành Android, đôi với ví dụ này thì SQLite là thư viện cung cấp việc hỗ trợ làm việc với database dùng để chứa dữ liệu. Hoặc Webkit là thư viện cung cấp những tính năng cho trình duyệt Web.
- **Android runtime:** Là tầng cùng với lớp thư viện Android runtime cung cấp một tập các thư viện cốt lõi để cho phép các lập trình viên phát triển viết ứng dụng bằng việc sử dụng ngôn ngữ lập trình Java. Android Runtime bao gồm máy ảo Dalvik (ở các version < 4.4, hiện tại là phiên bản máy ảo ART được cho là mạnh mẽ hơn trong việc xử lý biên dịch). Là cái để điều khiển mọi hoạt động của ứng dụng Android chạy trên nó(máy ảo Dalvik sẽ biên dịch ứng dụng để nó có thể chạy(thực thi) được , tương tự như các ứng dụng được biên dịch trên máy ảo Java vậy). Ngoài ra máy ảo còn giúp tối ưu năng lượng pin cũng như CPU của thiết bị Android

- **Android framework:** Là phần thể hiện các khả năng khác nhau của Android(kết nối, thông báo, truy xuất dữ liệu) cho nhà phát triển ứng dụng, chúng có thể được tạo ra để sử dụng trong các ứng dụng của họ.
- **Application:** Tầng ứng dụng là tầng bạn có thể tìm thấy chuyển các thiết bị Android như Contact, trình duyệt... Và mọi ứng dụng bạn viết đều nằm trên tầng này.



Hình 2.2 – Kiến trúc hệ điều hành Android

## 2.2 Giới thiệu về ngôn ngữ lập trình Java:

Java là một ngôn ngữ lập trình cao cấp, ban đầu được phát triển bởi Sun Microsystems và được phát hành vào năm 1995. Java chạy trên nhiều nền tảng, chẳng hạn như Windows, Mac OS và các phiên bản khác nhau của UNIX. Là ngôn ngữ kế thừa trực tiếp từ C/C++ và là một ngôn ngữ lập trình hướng đối tượng.

Java như là một công nghệ hay một nền tảng phát triển. Nó bao gồm các bộ phận:

- Máy ảo Java: JVM
- Bộ công cụ phát triển: J2SDK
- Các đặc tả chi tiết kỹ thuật (specifications)
- Ngôn ngữ lập trình (programming language)

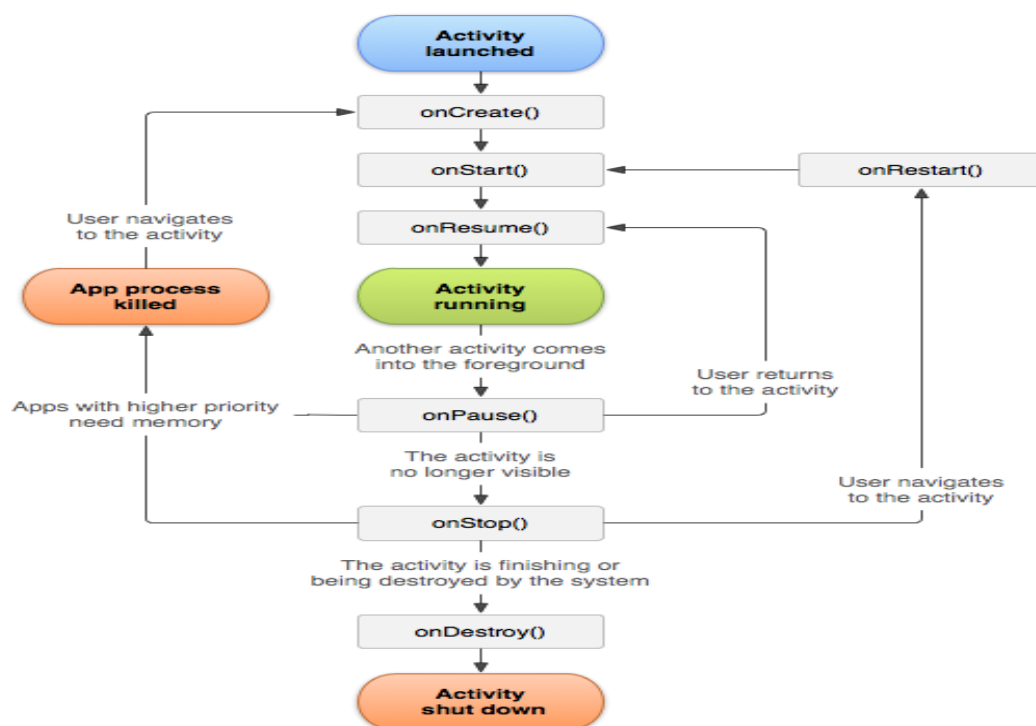
## 2.3 Các khái niệm cơ bản trong lập trình ứng dụng Android:

### 2.3.1 Activity

Một Activity là một thể hiện của ứng dụng. Một ứng dụng có thể gồm chỉ 1 hoặc nhiều Activity. Activity chính phải được hiển thị đầu tiên khi khởi động chương trình. Chuyển từ một Activity này sang một Activity khác bằng cách cho Activity hiện thời khởi động Activity kế tiếp.

Một cây phân cấp View được đặt trong một cửa sổ của Activity bằng phương thức `Activity setContentView()`, Content view là đối tượng View ở gốc của cây phân cấp.

Class cơ sở Activity định nghĩa một loạt các sự kiện mà điều chỉnh vòng đời của một hoạt động.



Hình 2.3 - Vòng đời của một Activity

Class Activity định nghĩa các sự kiện sau đây :

- `onCreate()`: Được gọi khi hoạt động được tạo ra lần đầu tiên.

- onStart(): Được gọi khi hoạt động trở nên hữu hình so với người dùng.
- onResume(): Được gọi khi hoạt động bắt đầu tương tác với người sử dụng.
- onPause(): Được gọi để dừng các hoạt động hiện tại và nối lại các hoạt động trước đó.
- onStop(): Được gọi khi hoạt động không còn hiển thị với người dùng.
- onDestroy(): Được gọi trước khi hoạt động bị phá hủy bởi hệ thống (bằng tay hoặc bằng hệ thống để bảo tồn bộ nhớ).
- onRestart(): Được gọi khi hệ thống đã được dừng lại và khởi động lại một lần nữa.

### 2.3.2 Service

Một Service không có giao diện trực quan, nó chạy nền trong một khoảng thời gian không xác định.

Trong khi kết nối, người dùng có thể giao tiếp với Service thông qua giao diện mà Service đó trung qua.

Service chạy trong Thread chính của ứng dụng, vì thế chúng không ngăn chặn các thành phần khác hay giao diện của người sử dụng, chúng thường tạo ra các Thread khác cho các nhiệm vụ hao tốn thời gian.

### 2.3.3 Content Provider

Một Content Provider tạo ra một tập cụ thể các dữ liệu của ứng dụng khả dụng cho các ứng dụng khác. Dữ liệu có thể được lưu trữ trong hệ thống file, trong một cơ sở dữ liệu SQLite, hay trong một cách khác nào đó. Content Provider mở rộng của lớp cơ sở ContentProvider để cài đặt một tập các chuẩn các phương thức cho phép các ứng dụng khác đạt được và lưu trữ dữ liệu của kiểu mà nó điều khiển. Tuy nhiên, các ứng dụng không gọi trực tiếp các phương thức này, chúng sử dụng một đối tượng ContentResolver và gọi các phương thức của nó. Một ContentResolver có thể nói chuyện với bất cứ Content Provider nào, chúng cộng tác với provider để quản lý giao tiếp liên tiến trình.

### 2.3.4 Broadcast Receive

Một Broadcast Receiver là một thành phần không làm gì ngoài việc nhận và đáp lại các thông báo Broadcast. Nhiều Broadcast khởi đầu trong mã hệ thống. Các ứng dụng có thể tạo ra các Broadcast, chẳng hạn để ứng dụng khác biết được một số dữ liệu đã được tải về thiết bị và sẵn sàng cho việc sử dụng chúng.

Broadcast Receiver không hiển thị một giao diện người dùng. Tuy nhiên, chúng có thể bắt đầu một Activity để đáp lại thông tin mà chúng nhận, hay chúng có thể sử dụng NotificationManager để cảnh báo người dùng. Notifications có thể gây sự chú ý của người dùng bằng nhiều cách.

### 2.3.5 View

Trong một ứng dụng Android, giao diện người dùng được xây dựng từ các đối tượng View và ViewGroup. Có nhiều kiểu View và ViewGroup. Mỗi một kiểu là một con của Class View và tất cả các kiểu đó gọi là các Widget.

Tất cả các Widget đều có chung các thuộc tính cơ bản như là trình bày vị trí, background, kích thước, lề... Tất cả những thuộc tính này được thể hiện hết ở trong đối tượng View.

Để thể hiện một màn hình thì trong hàm onCreate của mỗi Activity cần phải được gọi một hàm là setContentView (R.layout.my\_layout ). Hàm này sẽ load giao diện từ file XML lên để phân tích thành mã bytecode.

### 2.3.6 Intent

Là cầu nối giữa các Activity: ứng dụng Android thường bao gồm nhiều Activity, mỗi Activity hoạt động độc lập với nhau và thực hiện những công việc khác nhau. Intent chính là người đưa thư, giúp các Activity có thể triệu gọi cũng như truyền các dữ liệu cần thiết tới một Activity khác.

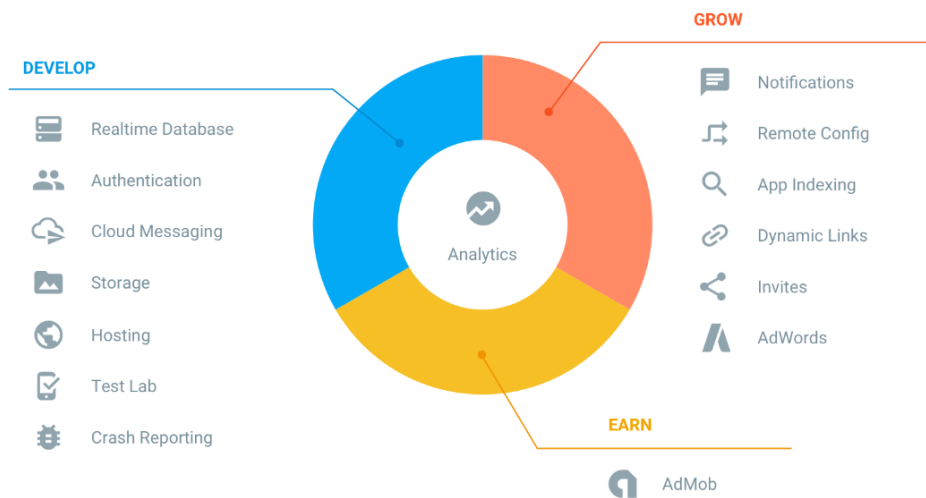
## 2.4 Firebase:

### 2.4.1 Giới thiệu

Firebase là một nền tảng ứng dụng di động và web với các công cụ và hạ tầng được thiết kế để giúp các lập trình viên xây dựng các ứng dụng chất lượng cao.



### 2.4.2 Tính năng cơ bản



Hình 2.4 – Các tính năng cơ bản của Firebase

### 2.4.3 Realtime Database

Firebase lưu trữ dữ liệu database dưới dạng JSON và thực hiện đồng bộ database tới tất cả các client theo thời gian thực. Cụ thể hơn là bạn có thể xây dựng được client đa nền tảng (cross-platform client) và tất cả các client này sẽ cùng sử dụng chung 1 database đến từ Firebase và có thể tự động cập nhật mỗi khi dữ liệu trong database được thêm mới hoặc sửa đổi. Ngoài ra Firebase còn cho phép bạn phân quyền một cách đơn giản bằng cú pháp tương tự như Javascript.

- Tự động tính toán quy mô ứng dụng của bạn
- Các tính năng bảo mật lớp đầu
- Làm việc offline

### 2.4.4 Firebase Authentication

Với Firebase, bạn có thể dễ dàng xác thực người dùng từ ứng dụng của bạn trên Android, IOS và JavaScript SDKs chỉ với một vài đoạn mã. Firebase đã xây dựng chức năng cho việc xác thực người dùng với Email, Facebook, Twitter, GitHub, Google, và xác thực nặc danh. Các ứng dụng sử dụng chức năng xác thực của FireBase có thể giải quyết được vấn đề khi người dùng đăng nhập, nó sẽ tiết kiệm thời gian và rất nhiều các

vấn đề phức tạp về phần backend. Hơn nữa bạn có thể tích hợp xác thực người dùng với các chức năng backend đã có sẵn sử dụng.

#### 2.4.5 *Firebase Hosting*

Phát triển ứng dụng web của bạn trong thời gian ngắn với các hosting tĩnh đã được cung cấp sẵn. Tất cả các kết nối được phân phối qua SSL từ CDN trên toàn thế giới của Firebase.

Triều khai siêu tốc: Việc triển khai sử dụng các công cụ dòng lệnh Firebase và có thể quay trở lại với phiên bản trước chỉ với một cú click chuột. Tất cả các ứng dụng sẽ có đường dẫn mặc định ở sau `firebaseapp.com` và nếu trả phí thì có thể triển khai một tên miền tùy chỉnh.

SSL bởi default: Mọi ứng dụng được xử lý thông qua một kết nối an toàn, và Firebase đã cẩn thận cung cấp SSL cert cho bạn.

#### 2.4.6 *Cloud Messaging*



Hình 2.5 – Mô tả về tính năng Firebase Messaging

#### 2.4.7 *Firebase Storage*

Firebase cũng sẽ ra mắt Firebase Storage để giúp lập trình viên upload và download file dễ dàng. Firebase Storage được Google Cloud Storage hỗ trợ, mang đến khả năng lưu trữ khổng lồ và cho phép Google Cloud project truy cập file dễ dàng.

Firebase Storage client SDKs còn có logic nâng cao có thể dễ dàng xử lý điều kiện mạng kém.

#### 2.4.8 *Firebase Test Lab và Crash Reporting*

Firebase Crash Reporting là công cụ báo cáo crash có khả năng ưu tiên, hành động nhanh giúp bạn chuẩn đoán và xử lý ngay các vấn đề của ứng dụng iOS và Android. Crash Reporting còn có thể kết nối đến nhóm Audiences trong Firebase Analytics, từ đó bạn có thể biết được người dùng bị crash đang dùng thiết bị nào, khu vực địa lý nào,...

### 2.5 Phân tích thiết kế hệ thống:

#### 2.5.1 *Phân tích và đặt tả yêu cầu*

##### 2.5.1.1 Mô tả bài toán:

Chức năng cho người dùng:

- + Thông báo: nhận thông báo chung và riêng của ban quản lý. Khi có thông tin mới, người dùng sẽ được admin thông báo đến trang chủ của ứng dụng. Người dùng có khả năng tìm kiếm lại thông tin của các thông báo cũ.

- + Nhắn tin: tính năng hỗ trợ 24/7 phục vụ khách hàng mọi lúc mọi nơi. Người dùng có thể gửi tin nhắn với bất cứ thắc mắc gì, kiến nghị hay phản ánh.

- + Thanh toán: thông báo riêng các khoản phí cho từng căn hộ, người dùng có thể thanh toán trực tiếp trên app.

Chức năng cho ban quản lý:

- + Quản lý cư dân: Quản lý thông tin tất cả các cư dân, căn hộ. Gửi tài khoản cố định cho cư dân.

- + Chức năng thông báo: Cho phép admin chỉnh sửa, đăng bài thông báo.

- + Chức năng nhắn tin: Cho phép admin trả lời tất cả các thắc mắc của cư dân.

##### 2.5.1.2 Phân tích và đặc tả yêu cầu

Chức năng cho người dùng:

- + Đối với chức năng thông báo: Cho phép người dùng nhận tất cả các thông báo được đăng trên trang thông báo của app. Người dùng không được phép xóa, sửa các thông tin trên thông báo. Ngoài ra, người dùng có thể tìm kiếm các bài thông báo cũ bằng thanh công cụ tìm kiếm.

+ Đối với chức năng nhắn tin: Cho phép người dùng soạn văn bản tin nhắn và gửi trực tiếp cho admin.

+ Đối với chức năng thanh toán: Người dùng xem thông báo hóa đơn chi phí dịch vụ hàng tháng được gửi vào trang thông báo của ứng dụng. Khi người dùng chọn vào 1 item thông báo, thì hóa đơn của căn hộ sẽ được hiển thị kèm theo button thanh toán. Người dùng chọn button thanh toán sẽ tiến hành bước hình thức thanh toán.

Chức năng cho ban quản lý:

+ Với chức năng quản lý cư dân, ban quản lý có thể lựa chọn cấp thêm hoặc hủy tư cách đăng nhập của cư dân theo điều khoản của công ty bằng thao tác chọn button tương ứng trên danh sách cư dân.

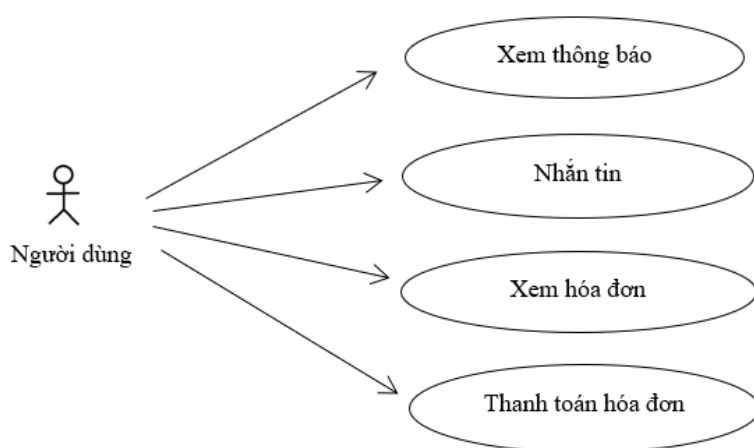
+ Đối với chức năng thông báo:

+ Đối với chức năng nhắn tin:

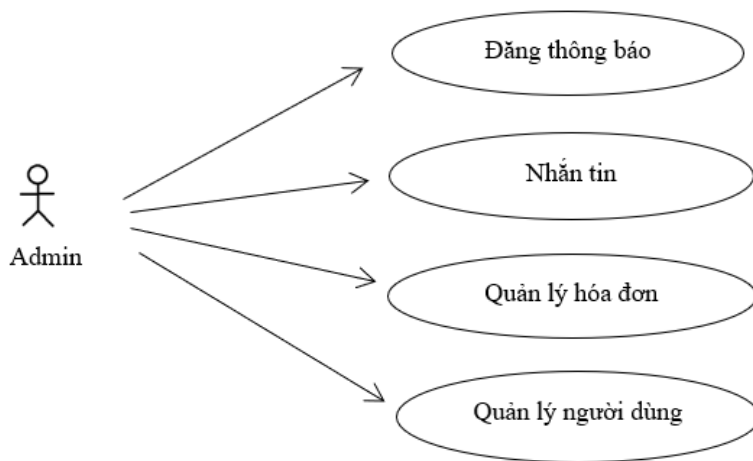
## 2.5.2 Phân tích thiết kế hệ thống:

### 2.5.2.1 Biểu đồ use case tổng quát

Dựa vào yêu cầu của bài toán, ta có thể xác định chức năng chính của Ứng dụng quản lý cư dân trong chung cư được thực hiện với khách hàng và nhà quản lý trong đó nhà quản lý sẽ bao gồm các chức năng quản lý và khách hàng sẽ bao gồm các chức năng xem, nhắn tin. Dựa vào yêu cầu của bài toán ta có sơ đồ use case như sau:

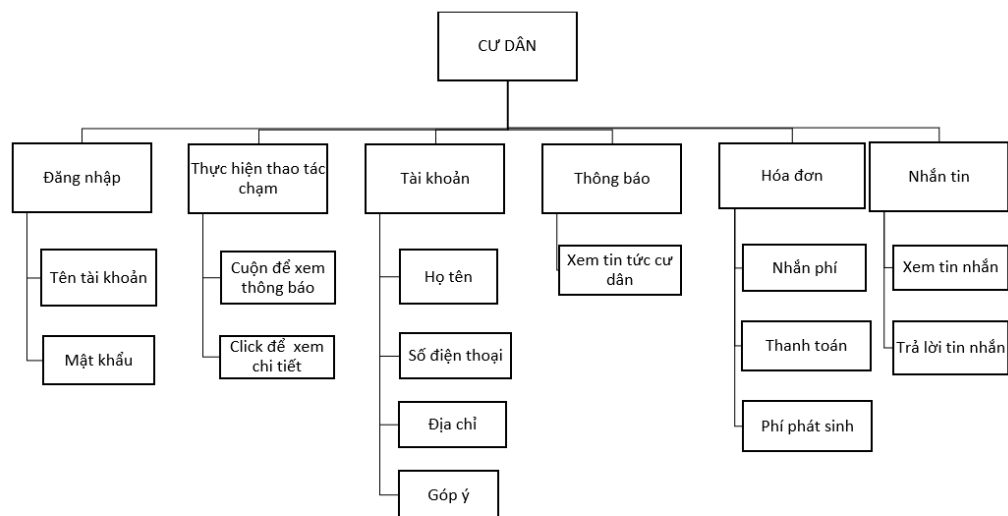


Hình 2.6 – Biểu đồ use case tổng quát chức năng của người dùng



Hình 2.7 - Biểu đồ use case tổng quát chức năng của admin

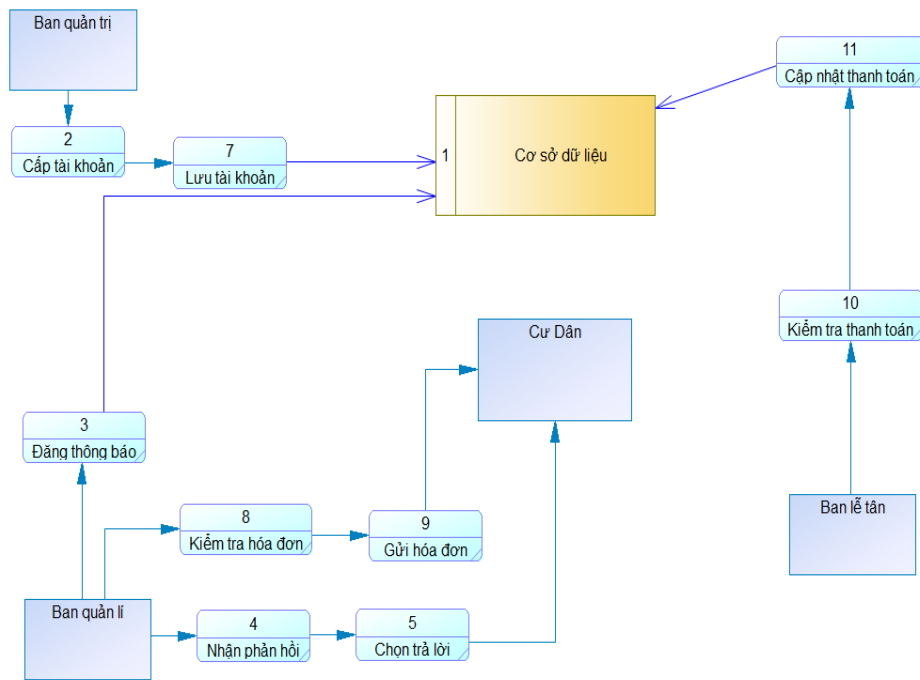
➤ Biểu đồ use case phân rã: Mô hình phân rã chức năng cho ta một cái nhìn chủ quan về hệ thống nên cần tạo ra mô hình tốt và đạt được sự thống nhất với người sử dụng.



Hình 2.8 – Biểu đồ use case phân rã

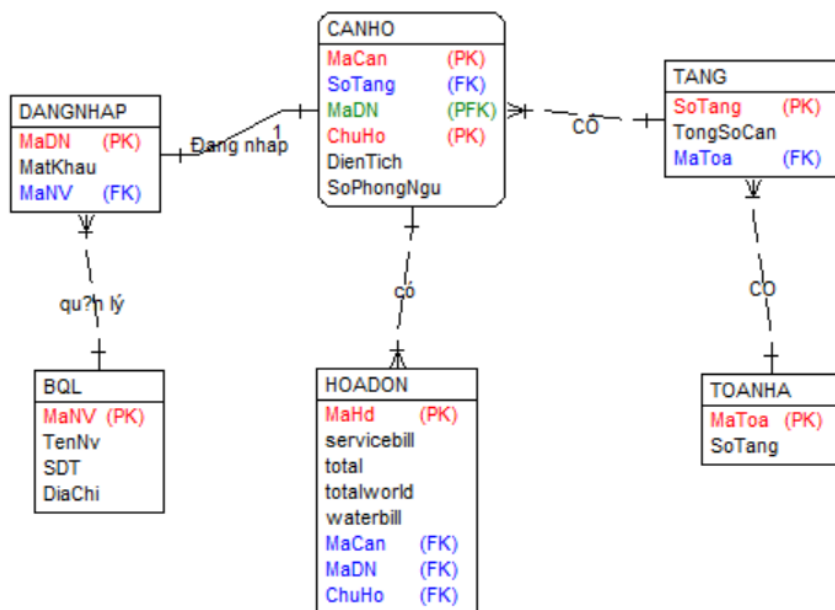
#### 2.5.2.2 Sơ đồ luồng dữ liệu (DFD)

Sơ đồ luồng dữ liệu (DFD) là một mô hình hệ thống cân xứng cả dữ liệu và tiến trình (progress). Nó chỉ ra cách thông tin chuyển vận từ một tiến trình hoặc từ chức năng này trong hệ thống sang một tiến trình hoặc chức năng khác.



Hình 2.9 - Biểu đồ hoạt động của ứng dụng.

### 2.5.2.3 Biểu đồ lớp (ERD):

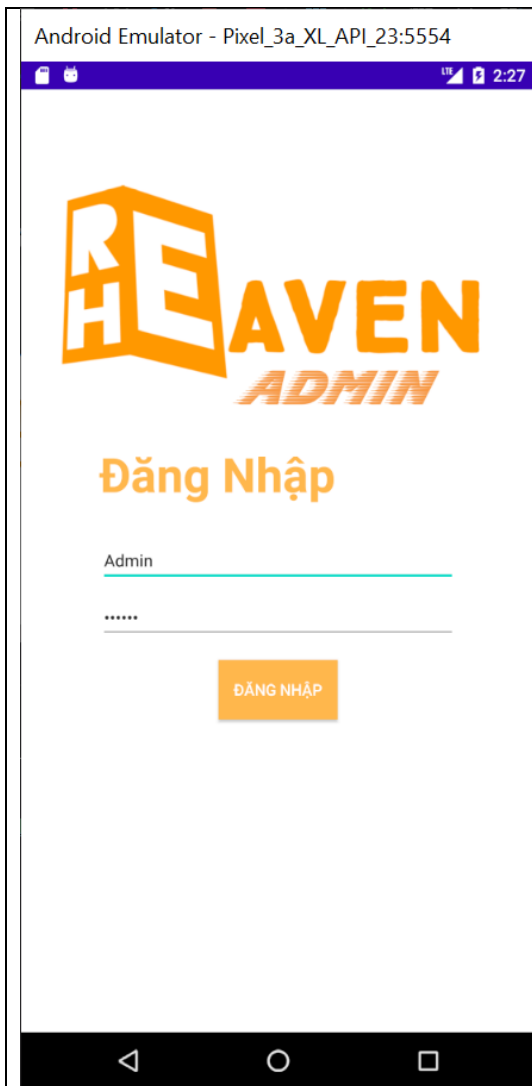


Hình 2.10 – Sơ đồ ERD quản lý cư dân

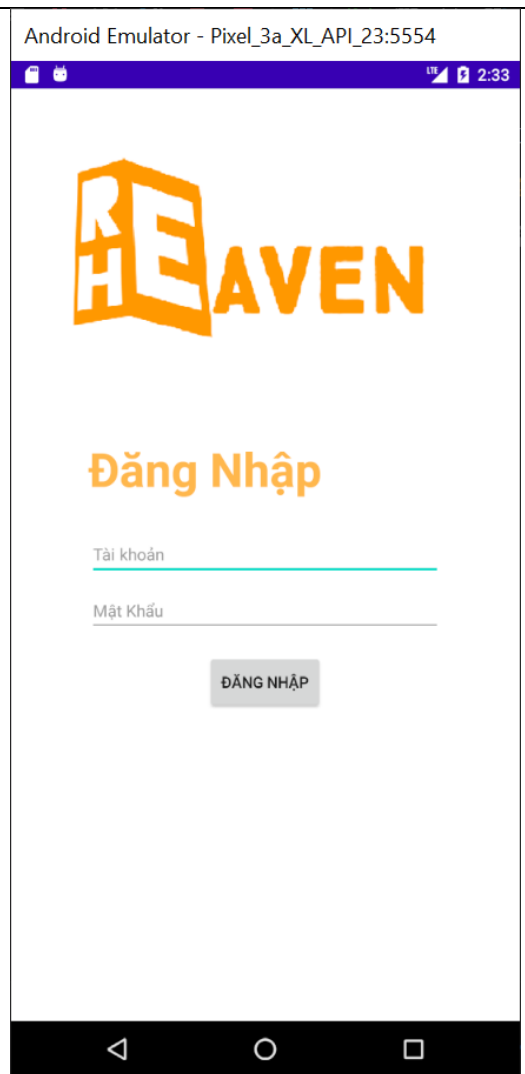
## Chương 3. KẾT QUẢ THỰC NGHIỆM

### 3.1 Kết quả triển khai và đánh giá:

- Tính năng chung của admin và user:



Hình 3.1 – Màn hình hiển thị giao diện đăng nhập của app admin

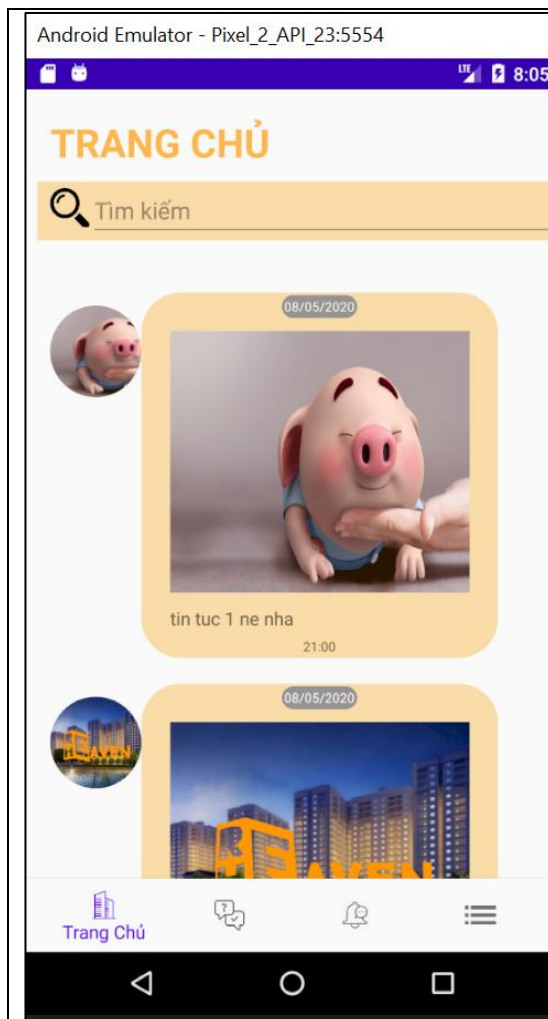


Hình 3.2 – Màn hình hiển thị giao diện đăng nhập của app user

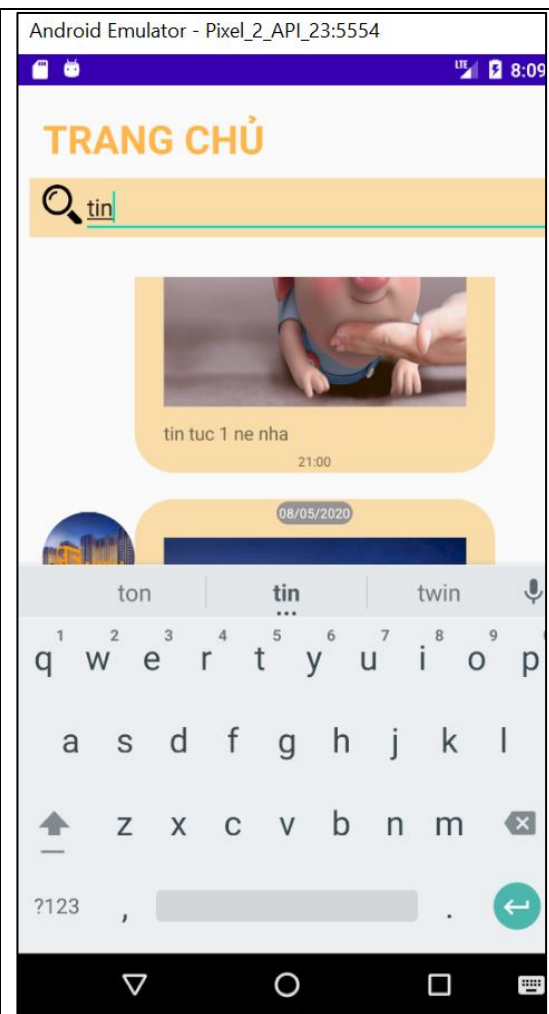
#### 3.1.1 Kết quả triển khai một số chức năng chính của ứng dụng Re-Heaven

- Chức năng hiển thị danh sách các tin tức và tìm kiếm các tin cũ:

- Chức năng nhắn tin của user đến admin:

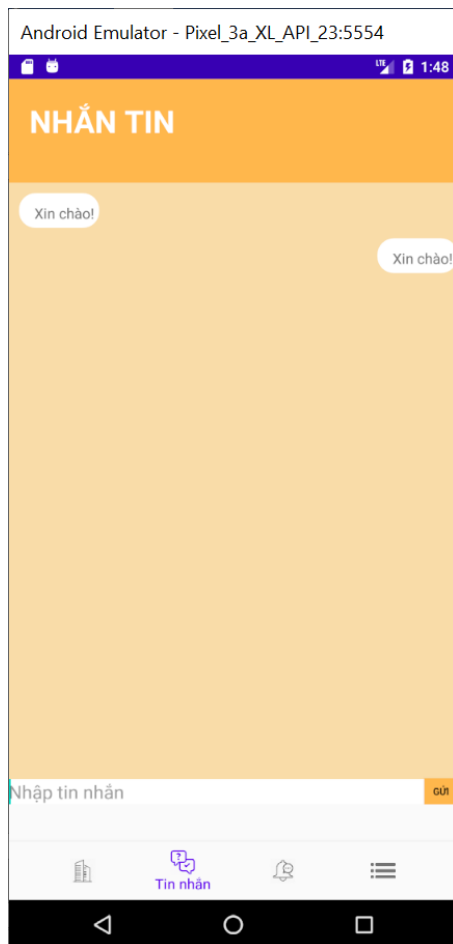


Hình 3.3 – Màn hình hiển thị danh sách các tin tức



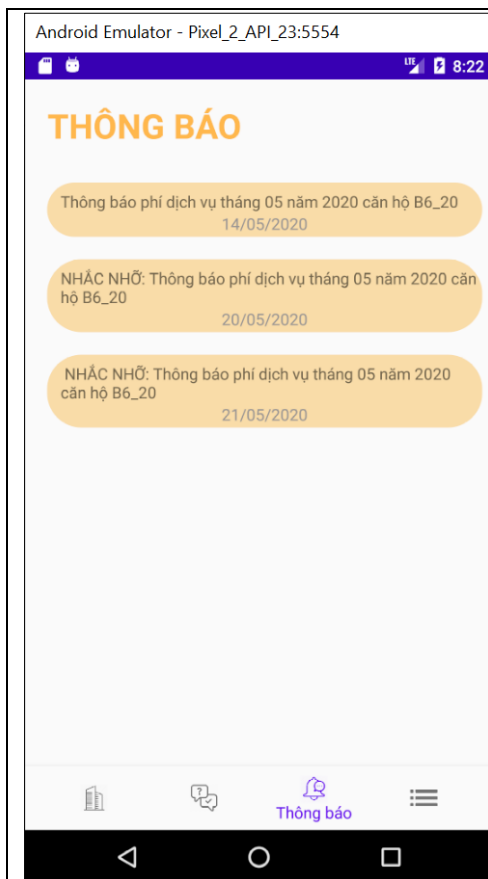
Hình 3.4 – Màn hình hiển thị thanh công cụ và chức năng tìm kiếm



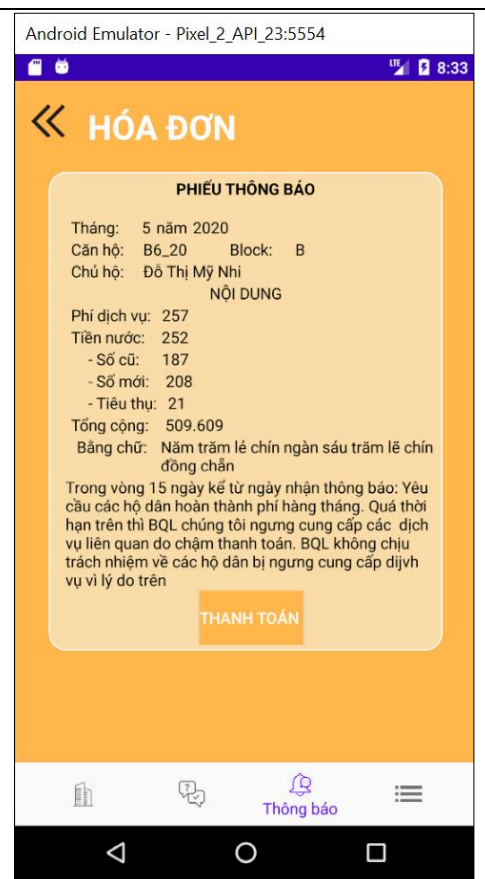


Hình 3.5 - Màn hình hiển thị giao diện nhắn tin của user với admin

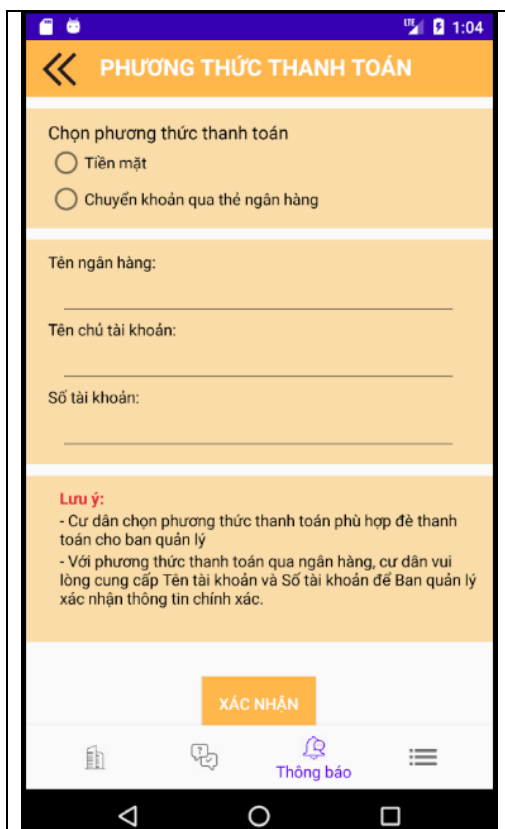
- Chức năng thanh toán hóa đơn của user:



Hình 3.6 – Màn hình hiển thị danh sách các thông báo



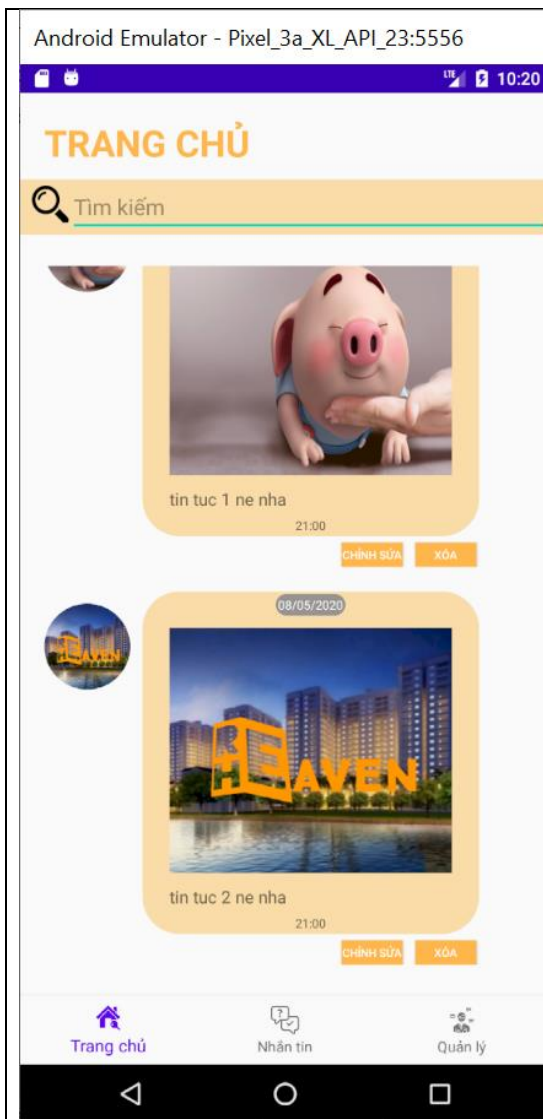
Hình 3.7 – Màn hình hiển thị hóa đơn dịch vụ hàng tháng



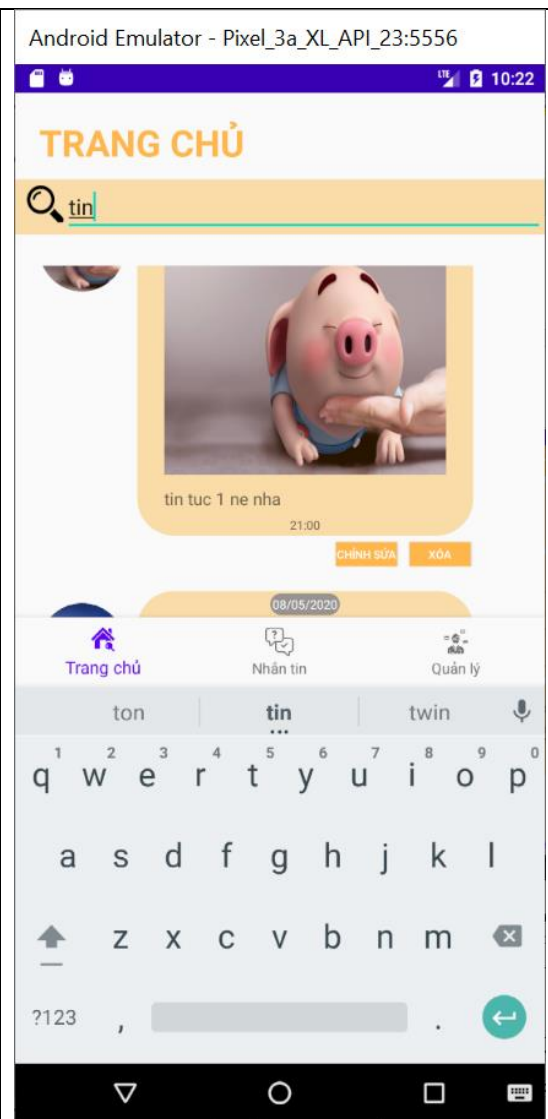
Hình 3.8 – Màn hình hiển thị giao diện hình thức thanh toán của thanh toán hóa đơn

### 3.1.2 Kết quả triển khai một số chức năng chính của ứng dụng Admin (Re-Heaven Administer):

- Chức năng chỉnh sửa và xóa thông báo, xem thông báo và tìm kiếm thông báo cũ của admin:

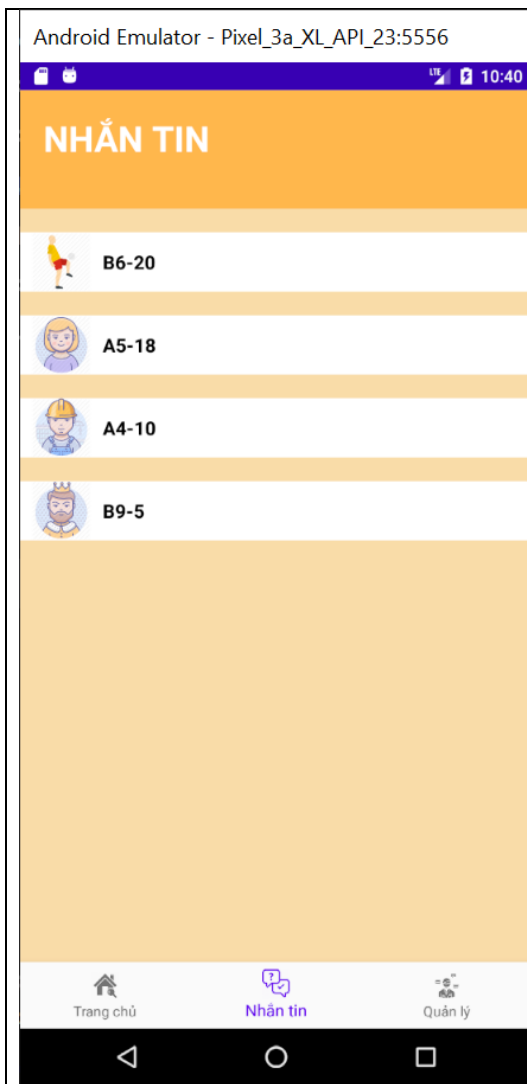


Hình 3.9 - Màn hình hiển thị giao diện admin có thể xem và chỉnh sửa hoặc xóa thông báo đã được đăng

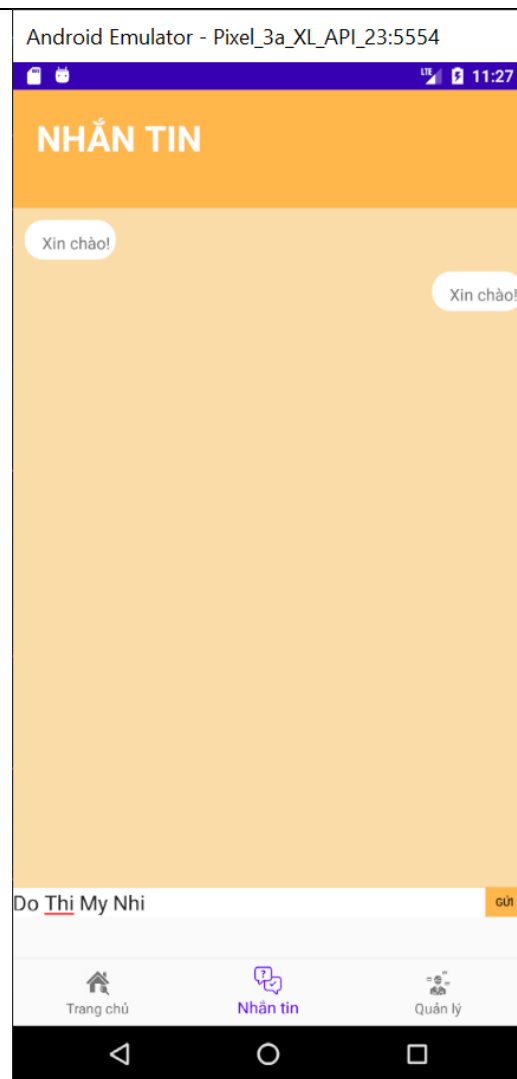


Hình 3.10 – Màn hình hiển thị thanh công cụ tìm kiếm của admin

- Chức năng nhắn tin của admin với user:

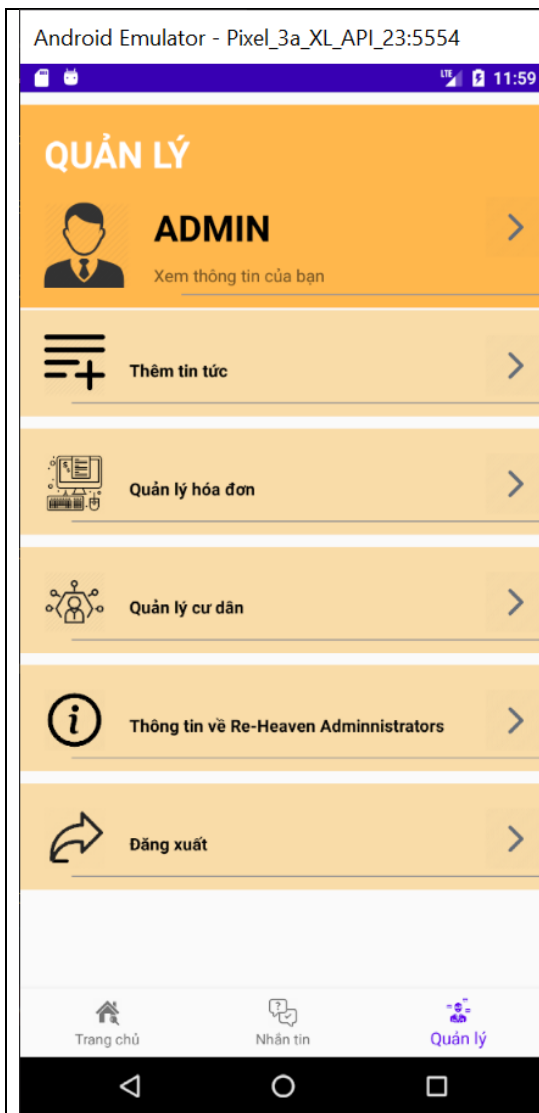


Hình 3.11 - Màn hình hiển thị giao diện chứa danh sách các tài khoản user đã nhắn tin cho admin

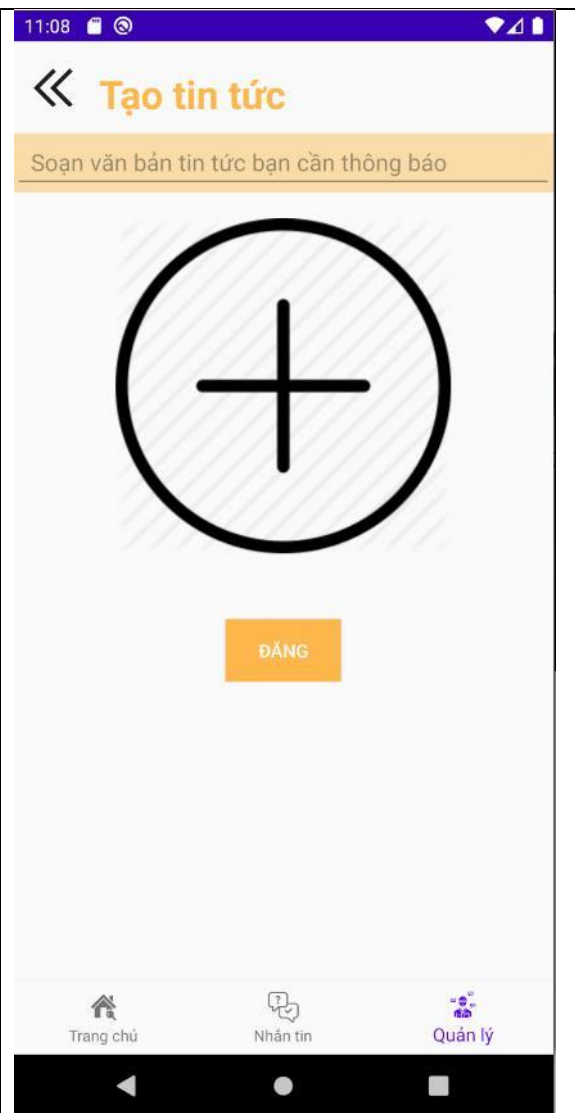


Hình 3.12 – Màn hình hiển thị khung chat riêng giữa admin với 1 tài khoản user

- Chức năng thêm tin tức, thông báo của admin:



Hình 3.13 – Màn hình hiển thị menu quản lý của admin



Hình 3.14 – Màn hình hiển thị giao diện đăng tin tức của admin

## 3.2 Đánh giá kết quả thực hiện:

### 3.2.1 Kết quả đạt được:

Sau quá trình tìm hiểu về nghiệp vụ quản lý cư dân trong chung cư những trang web quản lý, bảng thông báo ở chung cư hiện tại em sinh sống, em đã xây dựng được ứng dụng quản lý cư dân với các chức năng ban đầu đáp ứng được những yêu cầu:

- + Xây dựng được ứng dụng mà người dùng có thể xem thông báo một cách dễ dàng, trực quan và bảo mật.

- + Cho phép cư dân xem các hóa đơn phí hàng tháng trong thời gian bất kỳ, tìm kiếm, thanh toán tiện lợi.

- + Cho phép người dùng đăng nhập.

- + Cho phép admin quản lý cư dân của mình, quản lý các chi phí, quản lý các thông báo xem các tin tức.

### 3.2.2 *Kết quả chưa đạt được:*

Tuy nhiên, trong quá trình phát triển ứng dụng vẫn có một số kết quả chưa đạt được như:

- + Chưa tích hợp được thanh toán online của ngân hàng.

- + Chưa tích hợp được các thông báo bằng tin nhắn cho người sử dụng.

## **Chương 4. KẾT LUẬN VÀ KIẾN NGHỊ**

### **4.1 Kết quả đạt được:**

Sau quá trình tìm hiểu về nghiệp vụ quản lý và thông báo tin tức cùng hóa đơn phí dịch vụ hàng tháng của chung cư, em đã xây dựng được ứng dụng Quản lý cư dân trong chung cư với các chức năng ban đầu đáp ứng được những yêu cầu:

- Xây dựng được ứng dụng mà người dùng có thể xem tin tức thông báo dễ dàng, trực quan, bảo mật.
- Cho phép người dùng xem các hóa đơn một cách bảo mật.
- Cho phép người dùng quản lý xem hồ sơ cá nhân của mình.
- Thêm, sửa, xóa các loại dịch vụ.

### **4.2 Ưu điểm của đề tài:**

- Cho phép người dùng, người quản lý có thể thao tác dễ dàng với các chức năng.
- Đối với chức năng thêm một thông báo, tin tức, hóa đơn có thể dễ dàng truy xuất, chức năng nhắc nhở trực tiếp qua giao diện thông báo của ứng dụng một cách trực quan và tiện lợi.
- Đối với khách hàng, họ có thể dễ dàng xem các thông báo, tin tức và hóa đơn đã đặt trên giao diện để xem và nhận biết thuận tiện.
- Đảm bảo sự tương tác giữa người sử dụng và ứng dụng.
- Giao diện đơn giản, tông màu nhẹ giúp chế độ trực quan tốt hơn.
- Rút ngắn được thời gian chờ đợi của cư dân.

### **4.3 Hạn chế còn tồn tại:**

- Chưa tích hợp thanh toán online vào ứng dụng.
- Mới chỉ chạy trên hệ điều hành Android.
- Một số chức năng chưa thật tốt.

### **4.4 Kết luận:**



Sau khi thực hiện đồ án với đề tài Quản lý cư dân online trên hệ điều hành Android đã cho em được nhiều kiến thức sâu về hệ điều hành Android, đặc biệt là các ngôn ngữ dùng để phát triển ứng dụng như Java. Bên cạnh đó, sau khi thực hiện đồ án đã cung cấp cho em những kinh nghiệm về việc phân tích, thiết kế, triển khai phát triển một ứng dụng.

Tuy em đã cố gắng nhưng còn một số vấn đề hạn chế vẫn tồn tại như bảo mật, đa nền tảng, kích thước dữ liệu, giao diện vẫn chưa thật sự hoàn chỉnh.

#### **4.5 Hướng phát triển:**

- Hoàn thiện các chức năng còn thiếu sót.
- Hoàn thiện tốt nhất, tối ưu các chức năng đã thực hiện.
- Tích hợp các phương pháp thanh toán online vào ứng dụng.
- Phát triển ứng dụng trên các nền tảng hệ điều hành khác nhau.

# TÀI LIỆU THAM KHẢO

## Tài liệu:

- [1] Dương Anh Đức, 2000, "Bài giảng Phân tích thiết kế hướng đối tượng sử dụng UML", Đại học KHTN-Đại học Quốc gia TP.HCM.
- [2] Đặng Văn Đức, 2001, "Phân tích thiết kế hướng đối tượng bằng UML", Nhà xuất bản Giáo dục Việt Nam.
- [3] Jason Ostrander, 2012, "Android UI Fundamentals: Develop & Design", Peachpit Press.
- [4] Ths. Dương Thành Phết. Slide bài giảng Công nghệ phần mềm.

## Link Internet:

- [1] <https://androidcoban.com/gioi-thieu-ve-dieu-hanh-android.html>
- [2] <http://tapchitaichinh.vn/thi-truong-tai-chinh/phat-trien-thi-truong-can-ho-chung-cu-tai-tp-ho-chi-minh-313039.html>
- [3] [https://vi.wikipedia.org/wiki/Java\\_\(c%C3%B4ng\\_ngh%E1%BB%87\)](https://vi.wikipedia.org/wiki/Java_(c%C3%B4ng_ngh%E1%BB%87))
- [4] <https://viblo.asia/>
- [5] <https://developer.android.com/index.html>
- [6] <https://stackoverflow.com/>