

**FPT POLYTECHNIC**

## **DỰ ÁN 1**

### **Đề Tài: Tìm Trợ**

**Giáo Viên Hướng Dẫn:** Nguyễn Hữu Huy

**Sinh Viên Thực Hiện:**

- Đỗ Mạnh Tài - PH09785
- Nguyễn Thế Anh - PH09707
- Nguyễn Duy Tình - PH10128
- Hoàng Trung Kết - PH09802
- Hà Văn Xuân – PH09796

**Lớp:** PT 15353 – Lập Trình Mobile

Hà Nội – 2020

## This image shows a full page of white paper with horizontal dotted lines, typical of primary school writing paper. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

## Dự án 1 - Tìm Trọ

-Phần này bắt buộc với tất cả các đề tài:

This image shows a full page of primary-ruled paper. It features approximately 20 horizontal dotted lines spaced evenly down the page, providing a guide for handwriting practice. The paper is otherwise blank, with no margins, text, or other markings.

**NHẬN XÉT**  
(Của Hội Đồng Phản Biện)

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Hội đồng phản biện ký, ghi rõ họ tên

**-Phần này bắt buộc với tất cả các đề tài:**

[illegible]

# PHỤ LỤC

<b>Lời Nói Đầu.....</b>	<b>8</b>
<b>Giải Thích Thuật Ngữ: .....</b>	<b>9</b>
<b>I. Phần 1: Giới Thiệu Dự Án. ....</b>	<b>9</b>
<b>1.1 Thông Tin Chung Về Dự Án. ....</b>	<b>9</b>
<b>1.2 Lý Do Lựa Chọn Dự Án.....</b>	<b>9</b>
<b>1.3 Cơ Hội Thị Trường Của Dự Án.....</b>	<b>9</b>
<b>1.4 Môi Trường Cạnh Tranh Của Dự Án. ....</b>	<b>10</b>
<b>1.5 Thực Hiện Khảo Sát – Thu Nhập Thông Tin. ....</b>	<b>10</b>
1. Ứng dụng AirBnB .....	11
2. Ứng dụng Luxstay .....	11
<b>1.6 Các đối tượng tham gia sử dụng dự án. ....</b>	<b>12</b>
<b>1.7 Sơ đồ làm việc nhóm. ....</b>	<b>12</b>
<b>1.8 Các Công Nghệ và Công Cụ Sử Dụng. ....</b>	<b>12</b>
<b>1.9 Kế hoạch phát triển: .....</b>	<b>13</b>
<b>II. Phần 2. Phân Tích Nội Dung – Thiết Kế Hệ Thống. ....</b>	<b>18</b>
<b>2.1 Phân Tích Nội Dung.....</b>	<b>18</b>
<b>2.2 Mô Tả Nội Dung, Chức Năng.....</b>	<b>18</b>
<b>2.3 Đặc Tả User-Case .....</b>	<b>20</b>
User-Case Đăng Ký Người Dùng.....	20
Bảng 2.3.1: Bảng User-Case Đăng Ký Người Dùng.....	20
Hình 2.3.2: Thể Hiện Giao Diện Cho Màn Hình Đăng Ký Người Dùng.....	21
Hình 2.3.3: Mô Tả Về Luồng Xử Lý Cơ Bản Khi Đăng Ký Người Dùng.....	21
User-Case Đăng Nhập.....	22
Bảng 2.3.4: Bảng User-Case Đăng Nhập. ....	22
Hình 2.3.5: Thể Hiện Giao Diện Cho Màn Hình Đăng Nhập. ....	22
Hình 2.3.6: Mô Tả Về Luồng Xử Lý Cơ Bản Khi Đăng Nhập. ....	22
User- Case Cập Nhật Thông Tin Người Dùng:.....	23
Bảng 2.3.7: Bảng User-Case Cập Nhật Thông Tin Người Dùng .....	23
Hình 2.3.8: Thể Hiện Giao Diện Cho Màn Hình Cập Nhật Thông Tin Người Dùng .....	23
Hình 2.3.9: Mô Tả Về Luồng Xử Lý Cơ Bản Khi Cập Nhật Thông Tin Người Dùng. ....	23
User-Case Cập Nhật Password Người Dùng: .....	24
Bảng 2.3.10: Bảng User-Case Cập Nhật Password Người Dùng.....	24
Hình 2.3.11: Thể Hiện Giao Diện Cho Màn Hình Cập Nhật Password. ....	24

Hình 2.3.12: Mô Tả Về Luồng Xử Lý Cơ Bản Khi Cập Nhật Password. ....	24
User-Case Thêm Bài Đăng:.....	25
Bảng 2.3.13: Bảng User-Case thêm Bài Đăng. ....	25
Hình 2.3.14: Thể Hiện Giao Diện Cho Màn Hình Thêm Bài Đăng. ....	25
Hình 2.3.15: Mô Tả Về Luồng Xử Lý Cơ Bản Khi Thêm Bài Đăng. ....	25
User-Case Sửa Bài Đăng:.....	26
Hình 2.3.16: Bảng User-Case Sửa Bài Đăng. ....	26
Hình 2.3.17: Thể Hiện Giao Diện Cho Màn Hình Chỉnh Sửa Bài Đăng. ....	26
Hình 2.3.18: Mô Tả Về Luồng Xử Lý Cơ Bản Khi Chỉnh Sửa Bài Đăng. ....	26
User-Case Xóa Bài Đăng: .....	27
Bảng 2.3.19: Bảng User-Case Xóa Bài Đăng. ....	27
Hình 2.3.20: Thể Hiện Giao Diện Cho Màn Hình Xóa Bài Đăng.....	27
Hình 2.3.21: Mô Tả Về Luồng Xử Lý Cơ Bản Khi Xóa Bài Đăng.....	27
User-Case Bản Tin: .....	28
Bảng 2.3.22 : Bảng User-Case Bản Tin. ....	28
Hình 2.3.23: Thể Hiện Giao Diện Cho Màn Hình Bản Tin. ....	28
User-Case Thông Tin Chi Tiết Bài Đăng:.....	29
Bảng 2.3.22 : Bảng User-Case Bản Tin chi tiết. ....	29
Hình 2.3.26: Thể Hiện Giao Diện Cho Màn Hình Thông Tin Chi Tiết Của Bài Đăng.....	30
Hình 2.3.27: Mô Tả Về Luồng Xử Lý Cơ Bản Của Thông Tin Bài Đăng.....	30
User-Case Thông Tin Người Dùng. ....	31
Bảng 2.3.25: Bảng User-Case Thông Tin người dùng. ....	31
Hình 2.3.29: Thể Hiện Giao Diện Cho Màn Hình Thông Tin Người Dùng. ....	31
Bảng 2.3.30: Mô Tả Về Luồng Xử Lý Cơ Bản Của Thông Tin Bài Đăng. ....	31
<b>III. Phần 3. Xây Dựng Cơ Sở Dữ Liệu. ....</b>	<b>32</b>
<b>1. Bảng User:.....</b>	<b>32</b>
<b>2. Bảng Image: .....</b>	<b>32</b>
<b>3. Bảng Post: .....</b>	<b>32</b>
<b>IV. Phần 4. Giao Diện Người Dùng. ....</b>	<b>33</b>
- <b>Đăng kí. ....</b>	<b>33</b>
- <b>Đăng Nhập: .....</b>	<b>33</b>
- <b>Main: .....</b>	<b>34</b>
- <b>Lọc Bài Đăng: .....</b>	<b>35</b>
- <b>Người Dùng:.....</b>	<b>36</b>

- Thông Tin Người Dùng: .....	37
- Sửa Thông Tin Người Dùng: .....	38
- Cập Nhật Password: .....	39
- Đăng Bài: .....	40
- Bài Đăng Của Tôi: .....	41
- Xóa Bài Đăng: .....	42
- Sửa Bài Đăng: .....	43
- Bài Đăng Chi Tiết: .....	44
<b>V. Phần 5 Tổng Kết: .....</b>	<b>45</b>
<b>5.1 Thời gian hoàn thành dự án: .....</b>	<b>45</b>
<b>5.2 Mức độ hoàn thành dự án: .....</b>	<b>45</b>
<b>5.3 Khó khăn, rủi ro gặp phải và cách giải quyết. ....</b>	<b>45</b>
<b>5.4 Những bài học rút ra: .....</b>	<b>45</b>
<b>LỜI CẢM ƠN .....</b>	<b>46</b>

## Lời Nói Đầu

Tìm – Thuê nhà trọ luôn là một vấn đề nan giải ở các thành phố lớn đối với công nhân, nhân viên và đặc biệt là sinh viên. Mỗi năm có hàng ngàn công nhân, sinh viên từ các tỉnh thành trên cả nước theo vào làm ở các công ty, nhà máy, các trường đại học, cao đẳng. Khả năng đáp ứng nhu cầu còn rất hạn chế. Tiếp đến, phải đối mặt với việc tìm kiếm nhà trọ gần khu trường học, công ty phù hợp giá điều kiện, vị trí, giao thông và an ninh. Xuất phát từ thực tiễn trên, nhóm chúng em đã nhận thấy được đây là vấn đề cần giải quyết, đây tiền thực tiễn và tiềm năng nên đã chọn “ Tìm – Thuê Nhà Trọ” làm chủ đề cho dự án lần này với tên gọi “Tìm Trọ”.



## Giải Thích Thuật Ngữ:

Tên thuật ngữ	Giải thích thuật ngữ
Tester	Người kiểm tra sản phẩm mà lập trình viên làm ra
Designer	Người tạo ra giao diện 1 cách hoàn chỉnh
Backend Developer	Người xử lý mọi logic nghiệp vụ phức tạp ở ẩn phía sau, giúp hệ thống hoạt động trơn tru
QA	Người chịu trách nhiệm đảm bảo chất lượng sản phẩm, thông qua việc đưa ra quy trình làm việc giữa các bên liên quan.
ERD	Entity Relationship Diagram (mối quan hệ giữa các thực thể)
Thành viên	Người dùng trở thành, thành viên thông qua đã đăng nhập thành công, thành viên có thể bao gồm các đối tượng member, admin
SDT	Số điện thoại
Module	Chức năng

## I. Phần 1: Giới Thiệu Dự Án.

### 1.1 Thông Tin Chung Về Dự Án.

-Tên Dự Án: “Tìm Trợ”

### 1.2 Lý Do Lựa Chọn Dự Án.

Việt Nam hiện nay đang là một trong những nơi có người sử dụng Internet để tìm kiếm, giải trí hằng ngày rất đông đảo. Theo số liệu báo cáo từ tổ chức We Are Social, tính đến tháng 01 năm 2020, dân số Việt Nam có hơn 96,9 triệu người, có 68 triệu người tiếp cận với internet (tương đương với tỉ lệ thâm nhập là 70%). Đáng chú ý hơn là trong số những người được khảo sát, có đến 93% sử dụng smartphone. Do đó, lượng người tìm kiếm thông tin liên quan đến các nhu cầu cấp thiết là rất lớn, phổ biến nhất là nhu cầu về nhà trọ, đặc biệt với học sinh, sinh viên. Xuất phát từ những thực tế trên, nhóm em đã chọn đề tài “Tìm Trợ” để thực hiện dự án 1 của nhóm.

### 1.3 Cơ Hội Thị Trường Của Dự Án.

Dựa trên những thực tế mà nhóm đã nhìn nhận được để đưa ra lý do lựa chọn đề tài, chúng ta có thể thấy thị trường tìm kiếm nhà trọ đang rất tiềm năng ở Việt Nam. Hiện

nay, ứng dụng di động là hình thức trực tiếp để tìm kiếm người dùng một cách nhanh chóng nhất đặc biệt là giới trẻ. Các nhu cầu và thị hiếu người dùng ngày càng phát triển kèm theo đó là số lượng người truy cập rất lớn. Không những thế, thị trường này còn đang có số lượng khách hàng ngày càng gia tăng, đa dạng về độ tuổi,...đây chính là cơ hội giúp cho ứng dụng ngày càng có phát triển. **Tìm Trợ** đã nhận thấy được cơ hội thị trường của mình, nhóm đang nỗ lực để phát triển ứng dụng này phổ biến với người dùng Việt Nam, nhanh chóng bắt đầu thực hiện mô hình kinh doanh đã đề ra để tìm doanh thu phát triển.

#### 1.4 Môi Trường Cạnh Tranh Của Dự Án.

**Đối thủ trực tiếp** : **Tìm Trợ** đang phải gặp những đối thủ cực kì lớn ngay từ lúc lên ý tưởng cho dự án như: **AirBnB, Luxstay** ...những trang mạng trên cũng đang từng bước nâng cao và phát triển các chiến lược về dịch vụ cho khách hàng.

**Đối thủ gián tiếp** : Đó chính là các trang mạng giải trí khác như trang mạng trò chơi trực tuyến, mạng xã hội: **Facebook, Zalo, Lotus**,...và các trang báo điện tử: **Vnexpress, kenh14, Dantri**,...Đây là các trang mạng cung cấp cho người dùng trên mạng nhiều cách khác nhau để giải trí: có tìm kiếm, trò chơi giải trí, tin tức xã hội. Đây quả thật là các đối thủ cạnh tranh, nhất là các trang báo điện tử và mạng xã hội khi mà giới trẻ ngày nay đang rất quan tâm và thích sử dụng.

#### 1.5 Thực Hiện Khảo Sát – Thu Nhập Thông Tin.

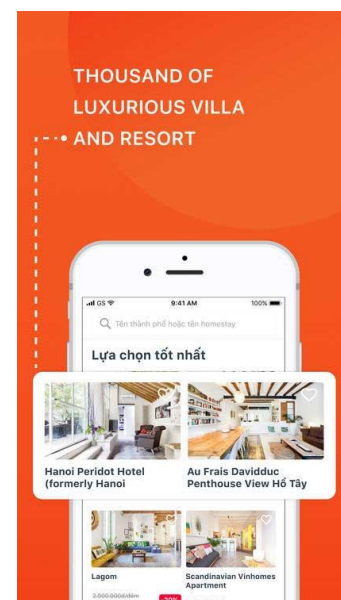
Nhóm đã tiến hành khảo sát người dùng, tiến hành tìm hiểu các hệ thống tương tự với dự án đang làm và đã đưa ra được những đánh giá như sau:

## 1. Ứng dụng AirBnB



- **Điểm mạnh:** Họ có một lượng dữ liệu lớn, đã được phát triển và đi vào hoạt động từ lâu, có một số lượng người dùng đông đảo, nhận được nhiều đánh giá tốt từ người dùng lẫn chủ nhà cho thuê.
- **Điểm yếu:** Khách hàng của họ đa số là người nước ngoài, số lượng homestay tại Việt Nam chưa nhiều.

## 2. Ứng dụng Luxstay



- **Điểm mạnh :** Là ứng dụng đầu tiên của người Việt về cho thuê và thuê Homestay được phát triển tốt nhất hiện nay. Có số lượng đánh giá tốt.

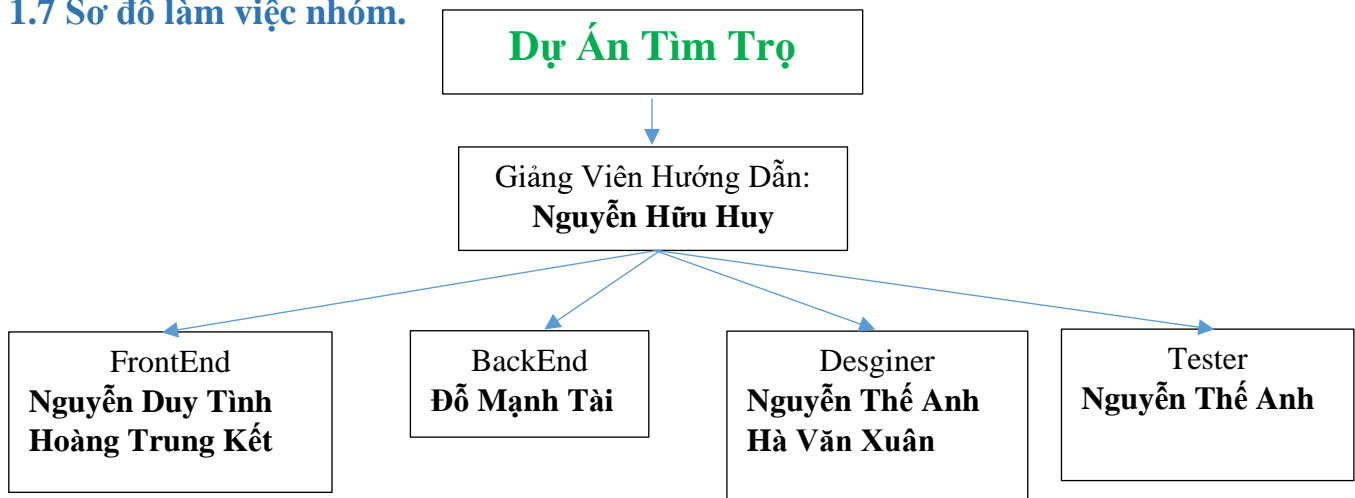
- **Điểm yếu:** Khách hàng cần phải hoàn tất đặt phòng trong vòng 24h nếu không sẽ bị huỷ đặt phòng

## 1.6 Các đối tượng tham gia sử dụng dự án.

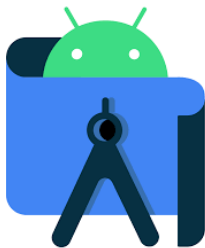
Các đối tượng tham gia sử dụng dự án của nhóm:

- Những người sử dụng Internet thường xuyên để tìm kiếm, cập nhật tin tức về nhà ở.
- Các chủ nhà trọ, chung cư, chung cư mini có nhu cầu tìm kiếm khách hàng.

## 1.7 Sơ đồ làm việc nhóm.



## 1.8 Các Công Nghệ và Công Cụ Sử Dụng.



Android Studio 4.1



FireBase



Github 2.29.2



Figma



Zalo

- Android studio:
  - File version: 4.1.0
  - Product version: 4.1.0.AI-201.8743.12.41.6858069
- Firebase:
  - [com.google.firebase:firebase-bom:26.1.1](https://firebase.google.com/)
  - Link: [Firebase \(google.com\)](https://firebase.google.com/)( <https://firebase.google.com/>)
- Github:
  - Link: [GitHub](https://github.com/) (<https://github.com/>)

- Figma:

- Link: [Recently viewed – Figma](https://www.figma.com/files/recent) (<https://www.figma.com/files/recent>)

- Zalo:

- Link: [Zalo PC - Tải Zalo PC để làm việc nhóm hiệu quả và gửi file nhanh](https://zalo.me/pc) (<https://zalo.me/pc>)

### 1.9 Kế hoạch phát triển:

STT	STT	Task	Finish	Person	Progress
1	Lên ý tưởng khảo sát thông tin		16/11/2020	Group	100%
2	Vẽ mockup		20/11/2020	AnhNT	100%
3	Vẽ prototype		22/11/2020	XuanHV	100%
4	Thiết kế giao diện, cơ sở dữ liệu		28/11/2020		100%
	1	Chức năng Đăng Ký		XuanHV	100%
	2	Chức năng Đăng Nhập		KetHT	100%
	3	Chức năng Đổi Mật Khẩu		TinhNV	100%
	4	Chức năng Bản Tin		AnhNT	100%
	5	Chức năng Lọc Bài Đăng		XuanHV	100%
	6	Chức năng Tìm Kiếm		KetHT	100%
	7	Chức năng Đăng Bài		TinhNV	100%
	8	Chức năng Quản Lý Bài Đăng		TinhNV	100%
	9	Chức năng Xem Thông Tin Người Dùng		TinhNV	100%
	10	Chức năng Chỉnh Sửa Thông Tin		KetHT	100%
	11	Chức năng Đăng Xuất		KetHT	100%
	12	Chức năng Xem Thông Tin Chi Tiết Bài Đăng		AnhNT	100%
	13	Chức năng Alo cho Chủ Bài Đăng		XuanHV	100%
	14	Chức năng Nhắc Tin cho Chủ Bài Đăng		AnhNT	100%
	15	Chức năng Xóa Bài Đăng của mình		XuanHV	100%
	16	Chức năng Sửa Bài đăng của mình		KetHT	100%
	17	Thiết kế database firebase		TaiDM	100%
5	Code		7/12/2020		
	1	Chức năng Đăng Ký		TaiDM	100%
	2	Chức năng Đăng Nhập		TaiDM	100%
	3	Chức năng Đổi Mật Khẩu		TaiDM	100%
	4	Chức năng Bản Tin		TaiDM	100%
	5	Chức năng Lọc Bài Đăng		TaiDM	100%

	6	Chức năng Hiển Thị Chế Độ Xem Bài Đăng		TaiDM	100%
	7	Chức năng Tìm Kiếm		TaiDM	100%
	8	Chức năng Đăng Bài		TaiDM	100%
	9	Chức năng Quản Lý Bài Đăng		TaiDM	100%
	10	Chức năng Xem Thông Tin Người Dùng		TaiDM	100%
	11	Chức năng Chỉnh Sửa Thông Tin		TaiDM	100%
	12	Chức năng Đăng Xuất		TaiDM	100%
	13	Chức năng Xem Thông Tin Chi Tiết Bài Đăng		TaiDM	100%
	14	Chức năng Alo cho Chủ Bài Đăng		TaiDM	100%
	15	Chức năng Nhắc Tin cho Chủ Bài Đăng		TaiDM	100%
	16	Chức năng Xóa Bài Đăng của mình		TaiDM	100%
	17	Chức năng Sửa Bài đăng của mình		TaiDM	100%
6	Kiểm thử và fix lỗi		10/12/2020	AnhNT	100%
STT	Màn hình	Test case		Kết quả	Fixed
1	Đăng kí	Tài khoản	Nhập thật dài	Báo nhập lại. OK	
			Không nhập	Báo chưa nhập. OK	
			Nhập đúng	OK	
			Nhập trùng	Nút Đăng kí không chọn được. OK	
		Giới tính	Không chọn	Báo chưa chọn giới tính. OK	
		Số điện thoại	Không nhập	Báo nhập sdt. OK	
			Nhập thật dài	Báo sai định dạng. OK	
			Nhập sai định dạng	Báo sai định dạng.OK	
			Nhập có nhiều số 0	Không pass được validate. Error	Fixed
		Họ tên	Nhập đúng sau fixed	OK	
			Nhập thật dài	Báo nhập lại. OK	
Không nhập	Báo chưa nhập. OK				
2	Đăng nhập	Tài khoản	Nhập đúng	OK	
			Nhập dài	Chương trình không crash không thông báo.Error	Fixed
			Không nhập		
		Mật khẩu	Nhập đúng tài khoản, nhập sai hoặc không nhập mật khẩu	Báo sai tài khoản mk. Error	Fixed
			Nếu tài khoản đúng. Nhập dài, không nhập	Thông báo lỗi.OK	

3	Xem bản tin		Nếu tài khoản sai. Nhập dài, không nhập	Không có thông báo. OK	
			Tài khoản đúng, mật khẩu sai	Có thông báo.OK	
			Tài khoản đúng mật khẩu đúng	OK	
		Nút lưu	Đăng nhập thành công	Lưu tên tk, mk. OK	
			Đăng nhập không thành công	Không lưu. OK	
		Tìm kiếm	Nhập chữ hoa	OK	
			Nhập chữ thường	OK	
			Nhập cả hoa, thường	OK	
		Đổi kiểu xem	Ấn	OK	
		Lọc	Ấn	Crash. Error	Fixed
4	Thêm bài viết	Chọn ảnh	Chọn nhiều	OK	
			Chọn một	OK	
			Không chọn	Báo chọn. OK	
		Tiêu đề	Nhập dài	Báo limited. OK	
			Không nhập	Báo nhập. OK	
			Nhập đúng fomat	OK	
		Giá	Nhập dài	Báo limited. OK	
			Không nhập	Báo nhập. OK	
			Nhập đúng fomat	OK	
			Nhập chữ	Không nhập được chữ. OK	
		Địa chỉ	Nhập dài	Báo limited. OK	
			Không nhập	Báo nhập. OK	
			Nhập đúng fomat	OK	
		Số điện thoại	Không nhập	Báo nhập sdt. OK	
			Nhập thật dài	Báo sai định dạng. OK	
			Nhập sai định dạng	Báo sai định dạng.OK	
			Nhập có nhiều số 0	OK	
			Nhập đúng sau fixed	OK	
		Diện tích	Nhập dài	Báo limited. OK	
			Không nhập	Báo nhập. OK	
			Nhập đúng fomat	OK	
			Nhập chữ	Không nhập được chữ. OK	
		Danh mục		OK	
		Mô tả	Nhập dài	Báo limited. OK	
			Không nhập	OK	
			Nhập đúng fomat	OK	
		Hủy		OK	

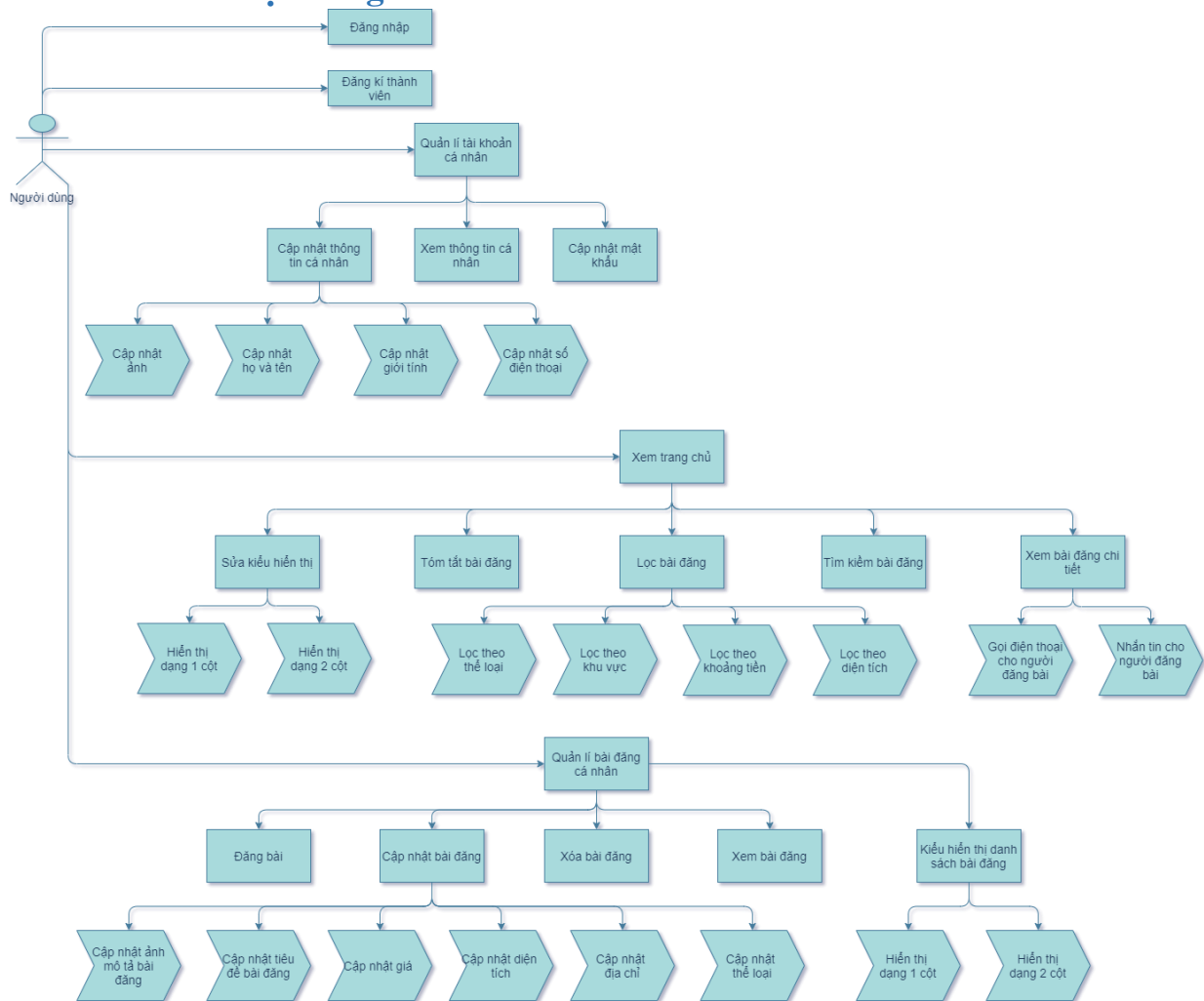
5	Người dùng	Chọn ảnh đại diện		OK	
		Xem ảnh đại diện to		OK	
		Đăng xuất		OK	
6	Bài đăng cá nhân	Chế độ xem	Đổi qua lại 2 chế độ	OK	
		Xóa		OK	
7	Sửa bài cá nhân	Chọn ảnh	Chưa chọn ảnh mới	Thông báo chọn. OK	
			Hiện ảnh cũ	OK	
			Chọn nhiều	OK	
			Chọn một	OK	
			Không chọn	Báo chọn. OK	
		Tiêu đề	Nhập dài	Báo limited. OK	
			Không nhập	Báo nhập. OK	
			Nhập đúng fomat	OK	
		Giá	Nhập dài	Báo limited. OK	
			Không nhập	Báo nhập. OK	
			Nhập đúng fomat	OK	
			Nhập chữ	Không nhập được chữ. OK	
		Địa chỉ	Nhập dài	Báo limited. OK	
			Không nhập	Báo nhập. OK	
			Nhập đúng fomat	OK	
		Số điện thoại	Không nhập	Báo nhập sdt. OK	
			Nhập thật dài	Báo sai định dạng. OK	
			Nhập sai định dạng	Báo sai định dạng. OK	
			Nhập có nhiều số 0	OK	
			Nhập đúng sau fixed	OK	
		Diện tích	Nhập dài	Báo limited. OK	
			Không nhập	Báo nhập. OK	
			Nhập đúng fomat	OK	
			Nhập chữ	Không nhập được chữ. OK	
		Danh mục		OK	
		Mô tả	Nhập dài	Báo limited. OK	
			Không nhập	OK	
			Nhập đúng fomat	OK	
		Hủy		OK	
8	Chỉnh sửa thông tin	Họ tên	Nhập thật dài	Báo nhập lại. OK	
			Không nhập	Báo chưa nhập. OK	
			Nhập đúng	OK	



		Tên tài khoản	Không sửa	OK	
		Giới tính		OK	
		Số điện thoại	Không nhập	Báo nhập sdt. OK	
			Nhập thật dài	Báo sai định dạng. OK	
			Nhập sai định dạng	Báo sai định dạng. OK	
			Nhập có nhiều số 0	OK	
			Nhập đúng sau fixed	OK	
		Nút Ok		OK	
<b>9</b>	Cập nhật mật khẩu	Mật khẩu cũ	Đúng mật khẩu cũ	OK	
			Sai mật khẩu cũ	Báo sai mk. OK	
			Nhập thật dài	Báo nhập lại. OK	
			Không nhập	Báo chưa nhập. OK	
		Mật khẩu mới	2 mật khẩu không khớp	Báo nhập không khớp	
			Nhập ngắn	Báo nhập ngắn. OK	
		Cập nhật	Đúng	OK	
			Sai	OK	
<b>10</b>	Lọc	Danh mục	Chọn 1 danh mục	OK	
			Chọn nhiều	OK	
			Không chọn	OK	
		Khu vực	Nhập chọn từ list	OK	
			Không chọn từ list	OK	Có thể tìm không ra
		Sắp xếp	Tin mới trước	OK	
			Giá thấp trước	OK	
		Khoảng tiền	Chọn từ đề xuất	OK	
			Chọn từ đề xuất rồi nhập	OK	
			Nhập 1 trường	OK	
			Nhập đủ 2 trường	OK	
		Diện tích	Chọn từ đề xuất	OK	
			Chọn từ đề xuất rồi nhập	OK	
			Nhập 1 trường	OK	
			Nhập đủ 2 trường	OK	
		Kết hợp nhiều điều kiện		OK	
		Xác nhận		OK	
		Hủy		OK	
<b>11</b>	Gọi			OK	
<b>12</b>	Nhắn tin			OK	
<b>7</b>	Hoàn thành dự án		12/12/2020		100%

## II. Phần 2. Phân Tích Nội Dung – Thiết Kế Hệ Thống.

### 2.1 Phân Tích Nội Dung.



### 2.2 Mô Tả Nội Dung, Chức Năng.

STT	Chức Năng	Nội Dung	Input	Ouput
1	Đăng Ký	Người dùng đăng ký tài khoản thành viên để có thể đăng bài, tìm kiếm xem nội các nội dung của bài đăng khác.	<ul style="list-style-type: none"> <li>Họ tên</li> <li>Tên tài khoản</li> <li>Giới tính</li> <li>Mật khẩu</li> <li>Số điện thoại</li> </ul>	Hiện thị kết quả đăng ký, sau đó chuyển về trang đăng nhập.

2	Đăng Nhập		Người dùng đăng nhập để sử dụng các chức năng của ứng dụng	User và Password	Đăng nhập thành công sẽ chuyển sang màn hình Bản Tin
3	Main	Fragment Bản Tin	Hiển thị các bài đăng	Ảnh, tên bài đăng, diện tích, số điện thoại, giá và địa chỉ.	<ul style="list-style-type: none"> <li>- Khi nhấn vào bài đăng sẽ chuyển sang màn hình của bài đăng chi tiết.</li> <li>- Khi nhấn button filter chuyển sang màn hình lọc</li> <li>- Nhấn vào thay đổi cách xem thì thay đổi kiểu xem</li> </ul>
		Fragment Thêm Bài Đăng	Người dùng thêm vào các nội dung của bài đăng.	Thêm ảnh, tên bài đăng, giá, địa chỉ, số điện thoại, diện tích và mô tả bài viết.	Khi thêm thành công màn hình sẽ chuyển sang Bản Tin.
		Fragment Người Dùng	Xem, chỉnh sửa quản lý bài đăng, thông tin.	Bao gồm Quản lý bài đăng, thông tin, cập nhật thông tin, cập nhật password, cập nhật ảnh đại diện	<ul style="list-style-type: none"> <li>- Khi nhấn vào Quản lý bài đăng sẽ chuyển sang màn hình Quản lý bài đăng.</li> <li>- Khi nhấn Thông Tin sẽ chuyển sang màn hình Thông Tin Người Dùng.</li> <li>- Khi Cập Nhật Thông Tin sẽ chuyển sang màn hình Cập Nhật .</li> <li>- Khi nhấn vào Cập Nhật Password sẽ chuyển sang màn hình cập nhật Password.</li> </ul>
4	Kiểm tra tài khoản, sự khả dụng Internet		Kiểm tra đã có tài khoản lưu hay không, kiểm tra máy có kết nối internet không		<ul style="list-style-type: none"> <li>- Nếu có internet thì bắt đầu kiểm tra tài khoản và khởi chạy app</li> <li>- Nếu có tài khoản thì không vào màn hình đăng nhập nữa</li> </ul>
5	Thông Tin Bài Đăng		Hiển thị nội dung chi tiết về bài đăng.	Gồm ảnh, tên bài, giá, diện tích, địa chỉ, số điện thoại và mô tả chi tiết về bài đăng.	<ul style="list-style-type: none"> <li>- Khi nhấn nút gọi thì chuyển sang màn hình gọi của máy điện thoại</li> <li>- Khi nhấn vào nút nhấn tin thì chuyển sang màn hình tin nhắn của máy</li> </ul>
6	Thông Tin Người Dùng		Hiển thị các nội dung về người dùng khi đăng ký tài khoản.		Hiển thị Ảnh, họ tên, tên tài khoản, giới tính, số điện thoại.
7	Quản lý bài đăng		Hiển thị các bài đăng của mình trước đó.		Cho phép xóa, sửa bài đăng của mình.

8	Cập Nhật Passwrod	Cập nhật mật khẩu mới.	Password cũ, password mới, nhập lại password.	-Khi nhấn nút cập nhật thì cập nhật mật khẩu đưa ra thông báo - Khi nhấn hủy thì sẽ quay trở về màn hình trước
9	Sửa Thông Tin	Sửa thông tin của người dùng.	Gồm họ tên, tên tài khoản, số điện thoại.	-Khi nhấn nút cập nhật thì cập nhật mật khẩu đưa ra thông báo - Khi nhấn hủy thì sẽ quay trở về màn hình trước
10	Sửa Bài Đăng	Sửa thông tin bài đăng.	Gồm ảnh, điện tích, địa chỉ, số điện thoại.	-Khi nhấn nút cập nhật thì cập nhật mật khẩu đưa ra thông báo - Khi nhấn hủy thì sẽ quay trở về màn hình trước
11	Tìm Kiếm Bài Đăng	Hiện thị ra bài đăng tìm kiếm	Tên của bài đăng	Trả về các bài đăng tìm kiếm được
12	Lọc Bài Đăng	Hiện thị ra bài đăng theo yêu cầu lọc	Danh mục, khu vực, sắp xếp (Tin mới nhất, giá thấp nhất), khoảng tiền, diện tích	Trả về các bài đăng thỏa mãn tiêu chí lọc
13	Chế Độ Xem	Có 2 chế độ xem		Chế độ xem dạng GridView và ListView.
14	Đăng Xuất	Đăng xuất tài khoản người dùng		Khi đăng xuất sẽ chuyển về màn hình Đăng Nhập.

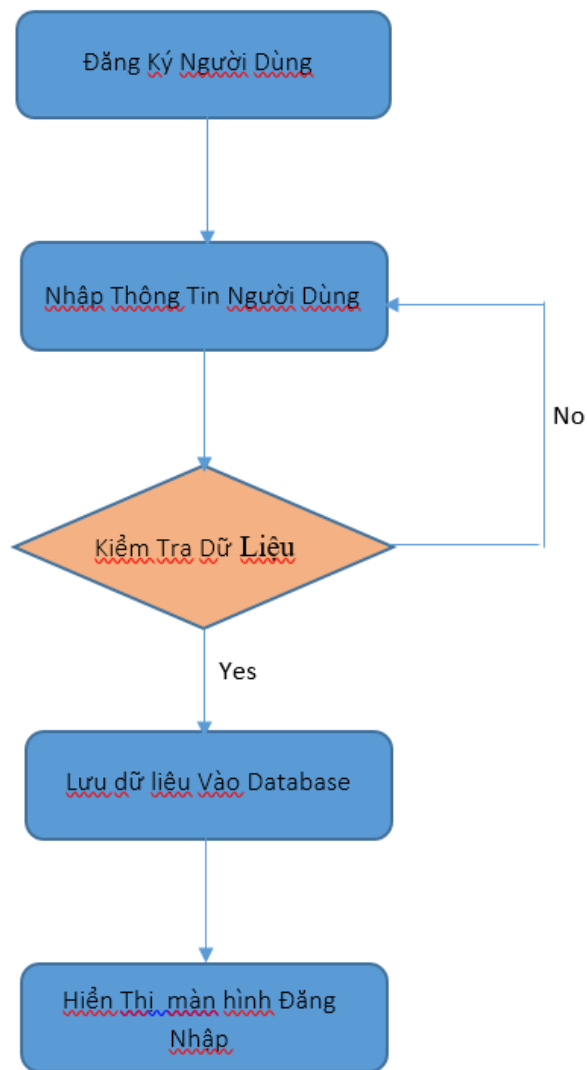
## 2.3 Đặc Tả User-Case

User-Case Đăng Ký Người Dùng.

**Bảng 2.3.1: Bảng User-Case Đăng Ký Người Dùng.**

<b>Bảng USE-CASE Đăng Ký Người Dùng</b>	
Tên Tài Khoản	Tên của Người Dùng đặt
Học Tên	Họ tên của Người Dùng
Giới Tính	Nam hoặc Nữ
Số Điện Thoại	Số điện thoại của Người Dùng
Mật Khẩu	Mật Khẩu do Người Dùng đặt
Nhập Lại Mật Khẩu	Phải nhập lại mật khẩu trên
Tiền Điều Kiện	Hệ thống sẽ hiển thị màn hình đăng ký Người Dùng
Hậu Điều Kiện	Nếu Use-Case thực hiện thành công sẽ chuyển sang màn hình Đăng Nhập, Ngược lại nếu thất bại sẽ báo lỗi

Hình 2.3.2: Thể Hiện Giao Diện Cho Màn Hình Đăng Ký Người Dùng.



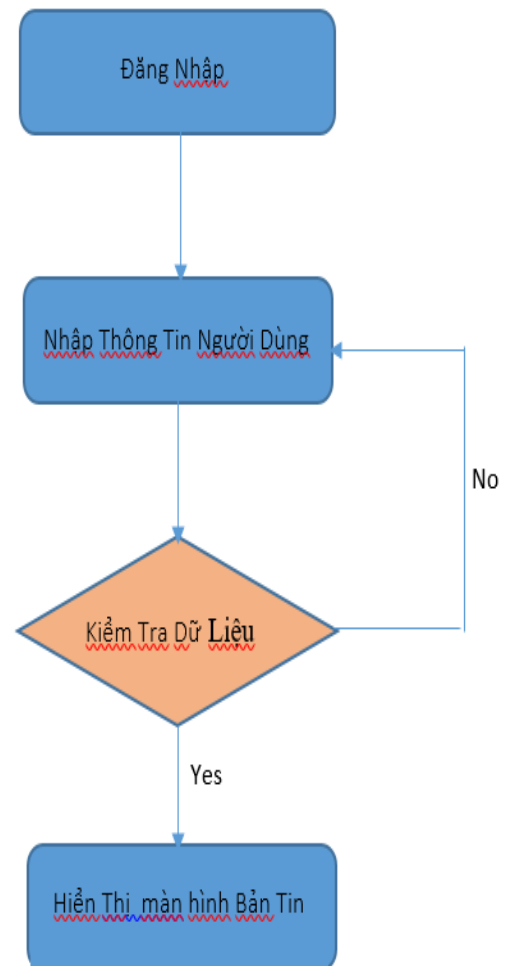
Hình 2.3.3: Mô Tả Về Luồng Xử Lý Cơ Bản Khi Đăng Ký Người Dùng.

Bảng 2.3.4: Bảng User-Case Đăng Nhập.

Bảng USE-CASE Đăng Nhập	
Học Tên	Họ tên của Người Dùng
Mật Khẩu	Mật Khẩu do Người Dùng đặt
Tiền Điều Kiện	Hệ thống sẽ hiển thị màn hình Đăng Nhập.
Hậu Điều Kiện	Nếu Use-Case thực hiện thành công sẽ chuyển sang màn hình Bản Tin, Ngược lại nếu thất bại sẽ báo lỗi



Hình 2.3.5: Thể Hiện Giao Diện Cho Màn Hình Đăng Nhập.

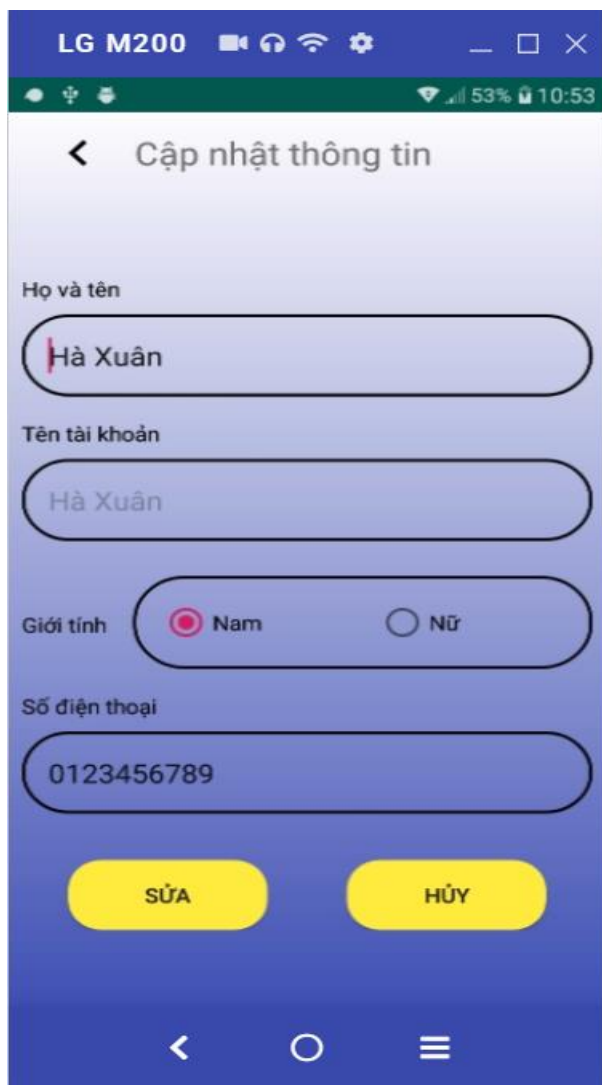


Hình 2.3.6: Mô Tả Về Luồng Xử Lý Cơ Bản Khi Đăng Nhập.

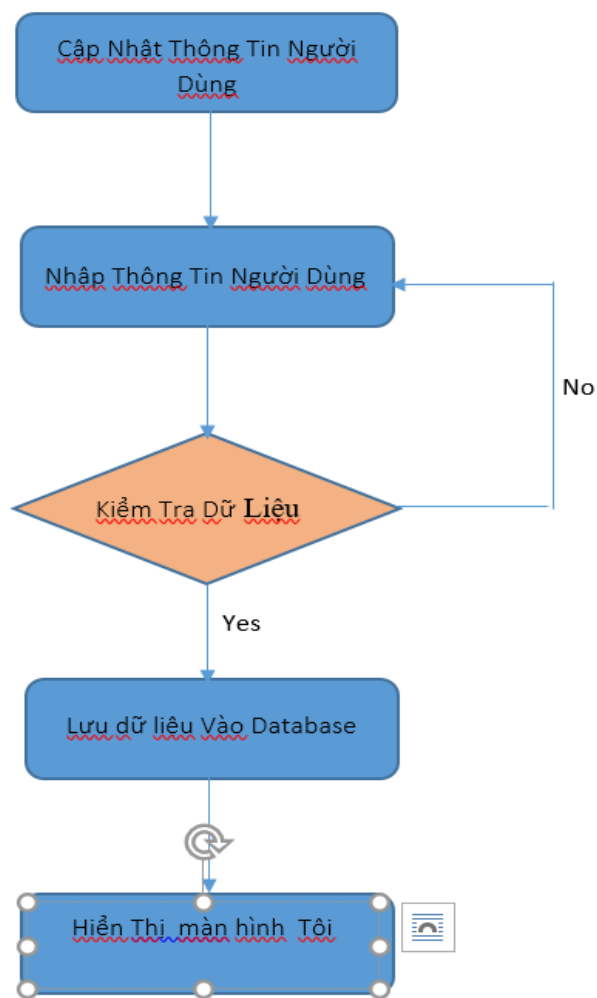
User- Case Cập Nhật Thông Tin Người Dùng:

Bảng 2.3.7: Bảng User-Case Cập Nhật Thông Tin Người Dùng.

Bảng USE-CASE Cập Nhật Thông Tin Người Dùng	
Học Tên	Họ tên của Người Dùng
Giới Tính	Nam hoặc Nữ
Số Điện Thoại	Số điện thoại của Người Dùng
Tiền Điều Kiện	Hệ thống sẽ hiển thị màn hình Cập Nhật Thông Tin.
Hậu Điều Kiện	Nếu Use-Case thực hiện thành công sẽ chuyển sang màn hình Người Dùng, Ngược lại nếu thất bại sẽ báo lỗi



Hình 2.3.8: Thể Hiện Giao Diện Cho Màn Hình Cập Nhật Thông Tin Người Dùng



Hình 2.3.9: Mô Tả Về Luồng Xử Lý Cơ Bản Khi Cập Nhật Thông Tin Người Dùng.

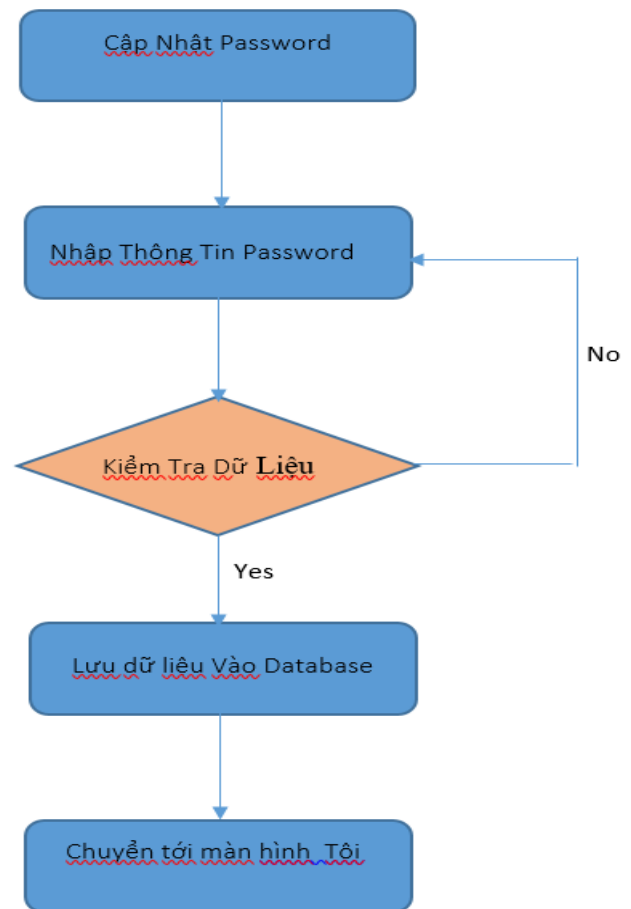
User-Case Cập Nhật Password Người Dùng:

Bảng 2.3.10: Bảng User-Case Cập Nhật Password Người Dùng.

<b>Bảng USE-CASE Cập Nhật Password Người Dùng</b>	
Mật Khẩu Cũ	Mật Khẩu do người dùng đặt lúc đăng ký
Mật Khẩu Mới	Nhập Mật khẩu mới do người dùng đặt
Nhập lại Mật Khẩu	Nhập lại mật khẩu mới trên
Tiền Điều Kiện	Hệ thống sẽ hiển thị màn hình Cập Nhật Password.
Hậu Điều Kiện	Nếu Use-Case thực hiện thành công sẽ chuyển sang màn hình Người Dùng, Ngược lại nếu thất bại sẽ báo lỗi



Hình 2.3.11: Thể Hiện Giao Diện Cho Màn Hình Cập Nhật Password.



Hình 2.3.12: Mô Tả Về Luồng Xử Lý Cơ Bản Khi Cập Nhật Password.

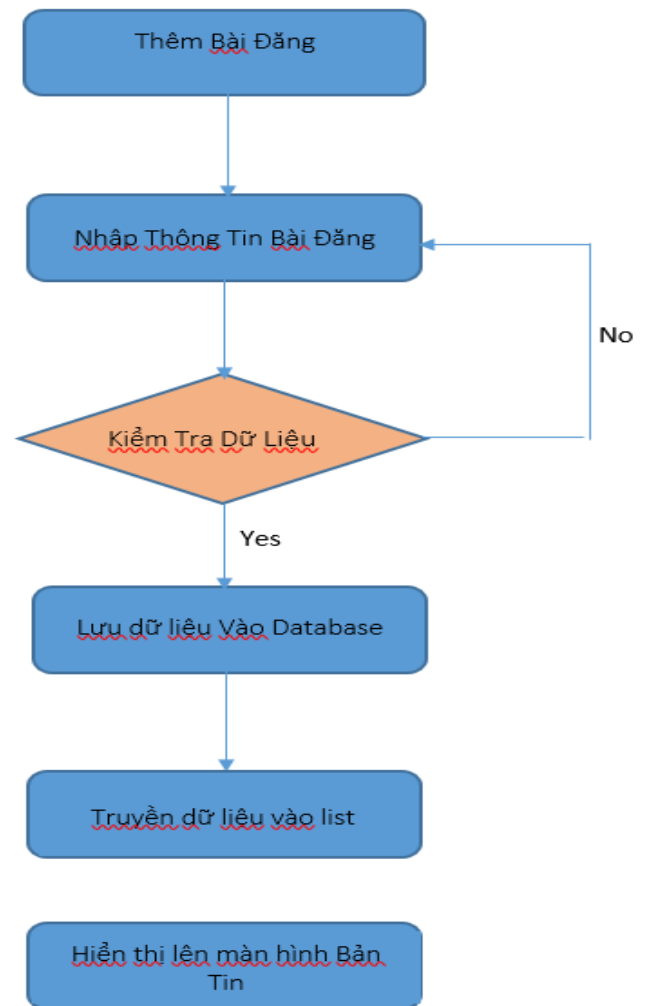


*Bảng 2.3.13: Bảng User-Case thêm Bài Đăng.*

<b>Bảng USE-CASE Thêm Bài Đăng</b>	
Ảnh	Ảnh của bài đăng
Tiêu Đề	Tên của bài đăng
Giá	Giá của bài đăng
Địa Chỉ	Địa chỉ bài đăng
Số Điện Thoại	Số điện thoại chủ của bài đăng
Diện Tích	Diện tích của bài đăng
Danh Mục	Chọn 6 danh mục trong spinner
Tiền Điều Kiện	Hệ thống sẽ hiển thị màn hình Đăng Bài
Hậu Điều Kiện	Nếu Use-Case thực hiện thành công sẽ chuyển sang màn hình Bản Tin, Ngược lại nếu thất bại sẽ báo lỗi



*Hình 2.3.14: Thể Hiện Giao Diện Cho Màn Hình Thêm Bài Đăng.*



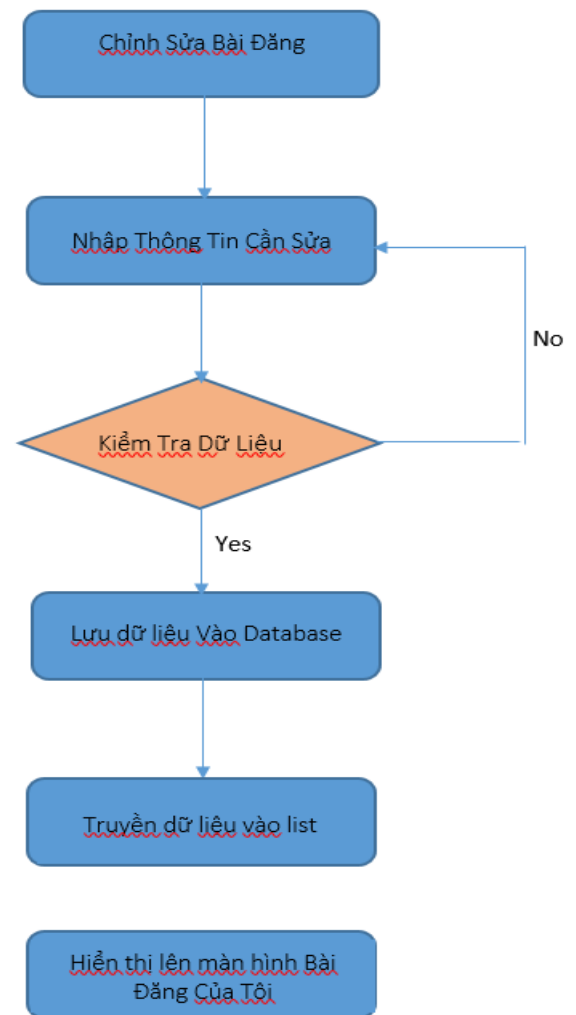
*Hình 2.3.15: Mô Tả Về Luồng Xử Lý Cơ Bản Khi Thêm Bài Đăng.*

Hình 2.3.16: Bảng User-Case Sửa Bài Đăng

<b>Bảng USE-CASE Thêm Bài Đăng</b>	
Ảnh	Ảnh của bài đăng
Tiêu Đề	Tên của bài đăng
Giá	Giá của bài đăng
Địa Chỉ	Địa chỉ bài đăng
Số Điện Thoại	Số điện thoại chủ của bài đăng
Diện Tích	Diện tích của bài đăng
Danh Mục	Chọn 6 danh mục trong spinner
Tiền Điều Kiện	Hệ thống sẽ hiển thị màn hình sửa bài đăng
Hậu Điều Kiện	Nếu Use-Case thực hiện thành công sẽ chuyển sang màn hình bài đăng của tôi, Ngược lại nếu thất bại sẽ báo lỗi



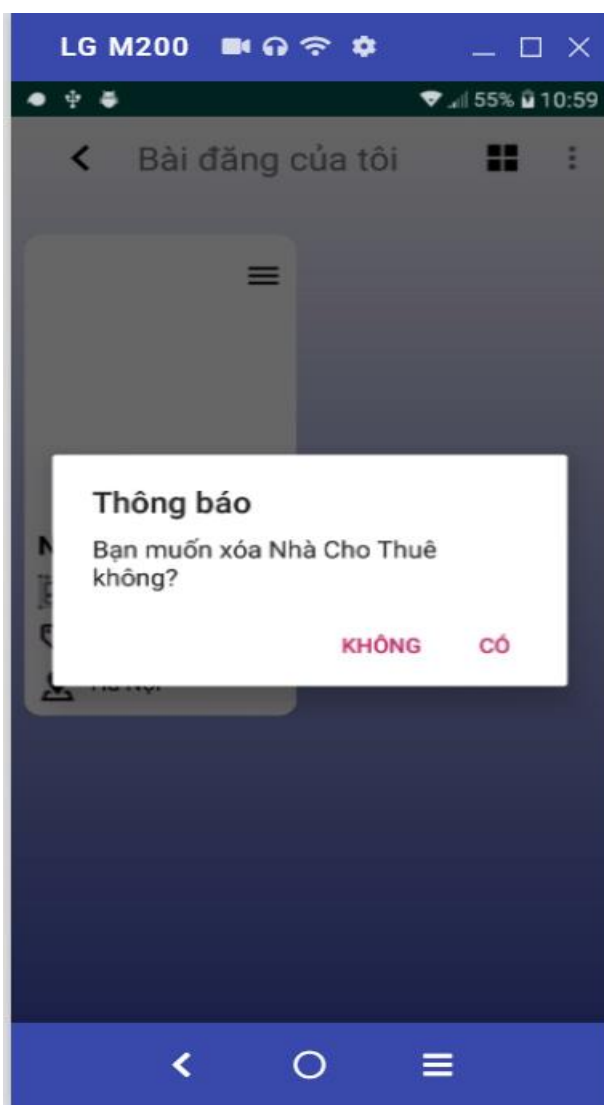
Hình 2.3.17: Thể Hiện Giao Diện Cho Màn Hình Chỉnh Sửa Bài Đăng



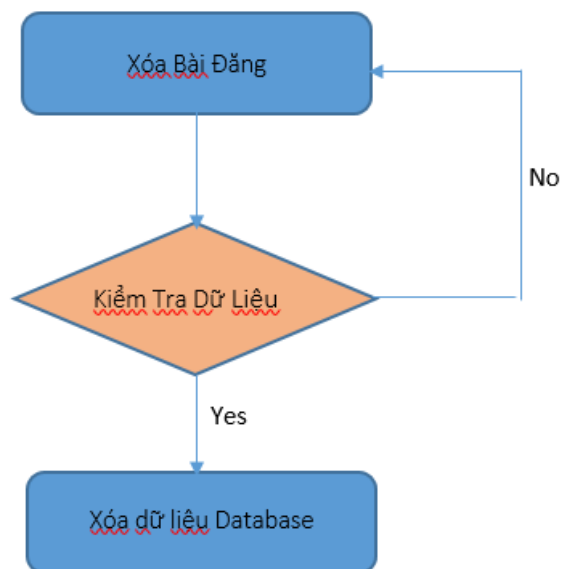
Hình 2.3.18: Mô Tả Về Luồng Xử Lý Cơ Bản Khi Chỉnh Sửa Bài Đăng.

*Bảng 2.3.19: Bảng User-Case Xóa Bài Đăng*

<b>Bảng USE-CASE Xóa Bài Đăng</b>	
Ảnh	Xóa Ảnh của bài đăng
Tiêu Đề	Xóa Tên của bài đăng
Giá	Xóa Giá của bài đăng
Địa Chỉ	Xóa Địa chỉ bài đăng
Số Điện Thoại	Xóa Số điện thoại chủ của bài đăng
Diện Tích	Xóa Diện tích của bài đăng
Danh Mục	Xóa Chọn 6 danh mục trong spinner
Tiền Điều Kiện	Hệ thống sẽ hiển thị màn hình xóa bài đăng
Hậu Điều Kiện	Nếu Use-Case thực hiện thành công sẽ chuyển sang màn hình bài đăng của tôi, Ngược lại nếu thất bại sẽ báo lỗi



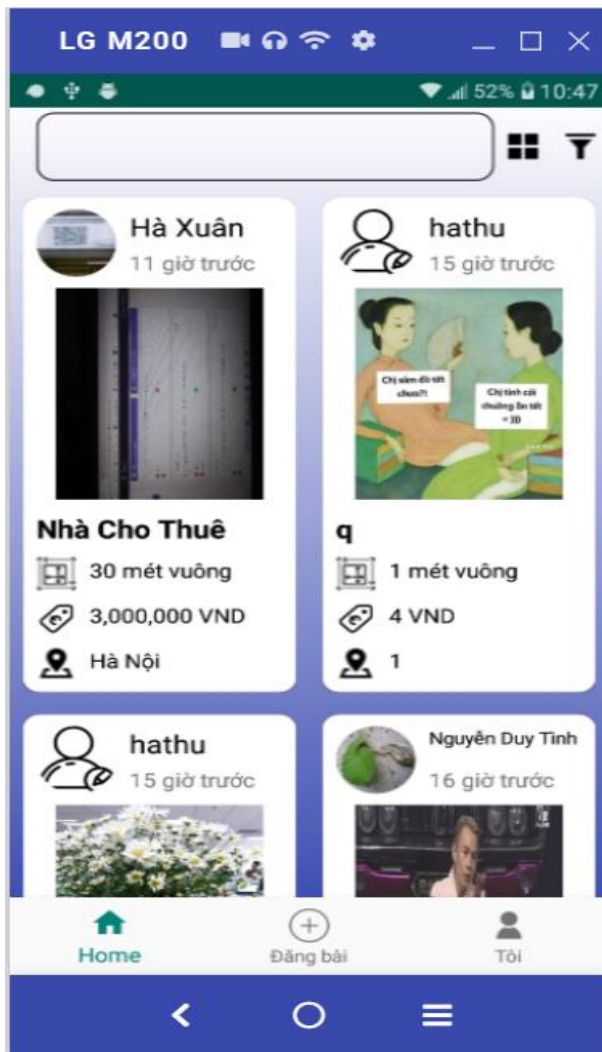
*Hình 2.3.20: Thể Hiện Giao Diện Cho Màn Hình Xóa Bài Đăng.*



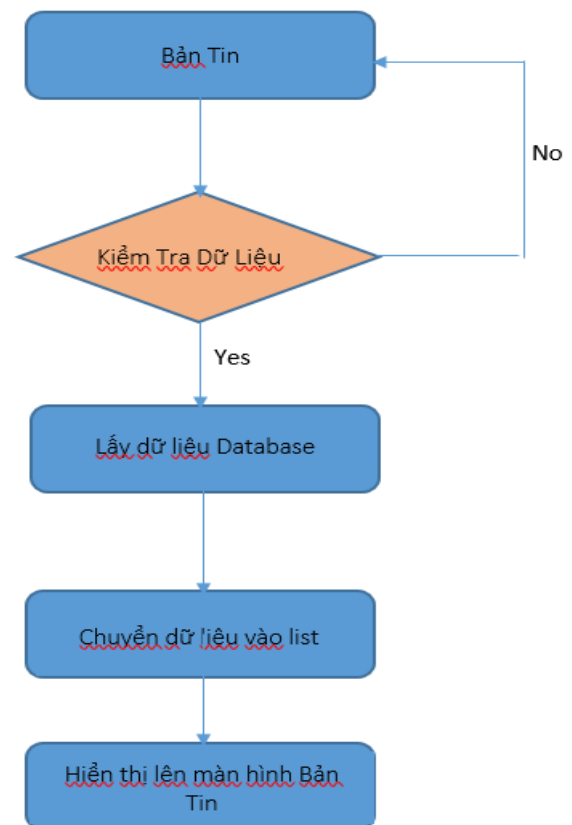
*Hình 2.3.21: Mô Tả Về Luồng Xử Lý Cơ Bản Khi Xóa Bài Đăng*

Bảng 2.3.22 : Bảng User-Case Bản Tin

<b>Bảng USE-CASE Bản Tin</b>	
Ảnh bài đăng	Hiển thị ảnh của các bài đăng
Tiêu Đề	Tên của bài đăng
Giá	Giá của bài đăng
Địa Chỉ	Địa chỉ bài đăng
Avatar người dùng	Hiển thị ảnh của người dùng
Tên của Người dùng	hiển thị tên của người dùng
Thời gian bài đăng	Thời gian đăng bài so với thời gian hiện tại
Diện Tích	Xóa Diện tích của bài đăng
Tiền Điều Kiện	Hệ thống sẽ hiển thị màn hình xóa bài đăng
Hậu Điều Kiện	Nếu Use-Case thực hiện thành công xem được các bài đăng của những người dùng khác.



Hình 2.3.23: Thể Hiện Giao Diện Cho Màn Hình Bản Tin

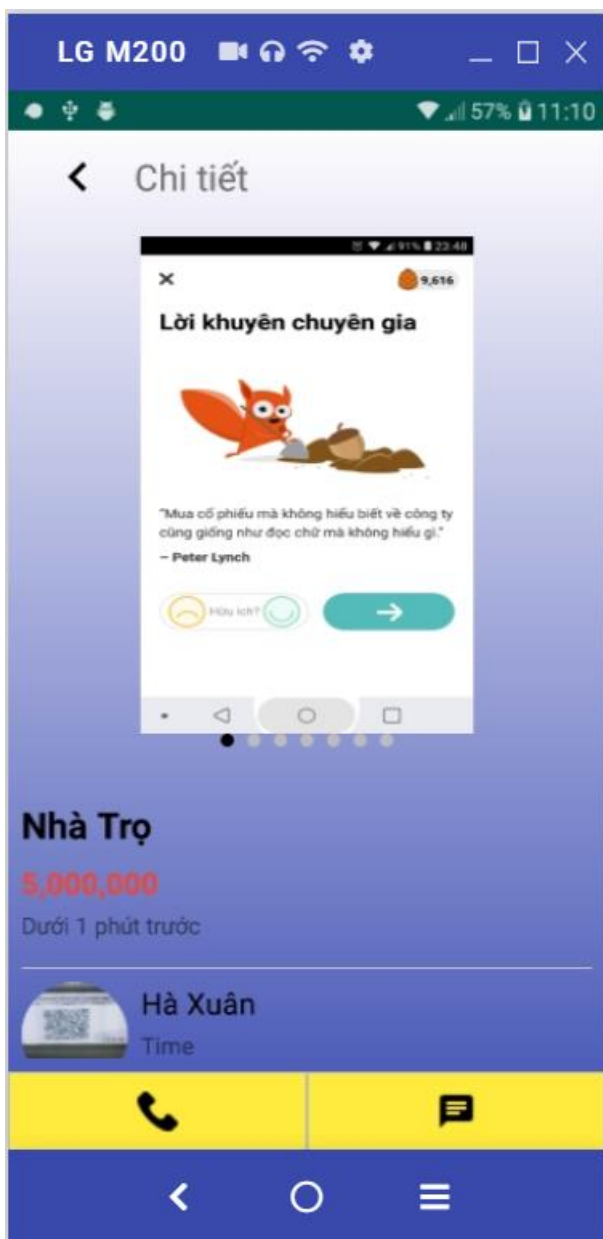


Hình 2.3.24: Mô Tả Về Luồng Xử Lý Cơ Bản Của Bản Tin.

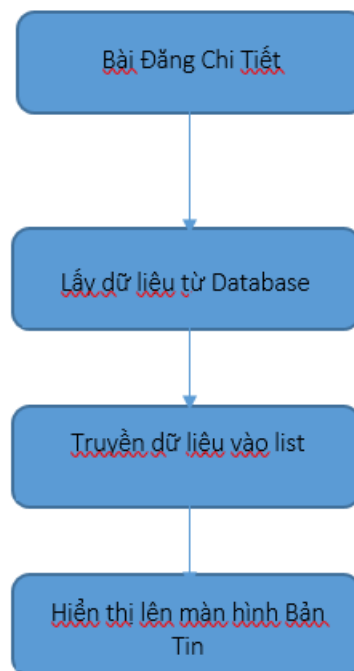
## User-Case Thông Tin Chi Tiết Bài Đăng:

*Bảng 2.3.22 : Bảng User-Case Bản Tin chi tiết*

Bảng USE-CASE Bản Tin chi tiết	
Ảnh bài đăng	Hiện thị ảnh của bài đăng
Tiêu Đề	Tên của bài đăng
Giá	Giá của bài đăng
Địa Chỉ	Địa chỉ bài đăng
Liên hệ ngay	Hiện thị số điện thoại của người cho thuê
Avatar người dùng	Hiện thị ảnh của người dùng
Khu Vực	Hiện thị khu vực của bài đăng
Tên của Người dùng	hiển thị tên của người dùng
Thời gian bài đăng	Thời gian đăng bài so với thời gian hiện tại
Diện Tích	Xóa Diện tích của bài đăng
Danh Mục cho thuê	Hiện thị danh mục cho thuê
Mô tả	Mô tả chi tiết về bài đăng
Tiền Điều Kiện	Hệ thống sẽ hiển thị màn hình xóa bài đăng
Gọi Điện	Hiện thị ra Dialog chuyển đến màn hình cuộc gọi trên điện thoại của người dùng
Tin Nhắn	Chuyển đến màn hình tin nhắn trên điện thoại của người dùng
Hậu Điều Kiện	Nếu Use-Case thực hiện thành công xem được các bài đăng của những người dùng khác.



Hình 2.3.26: Thể Hiện Giao Diện Cho Màn Hình Thông Tin Chi Tiết Của Bài Đăng.



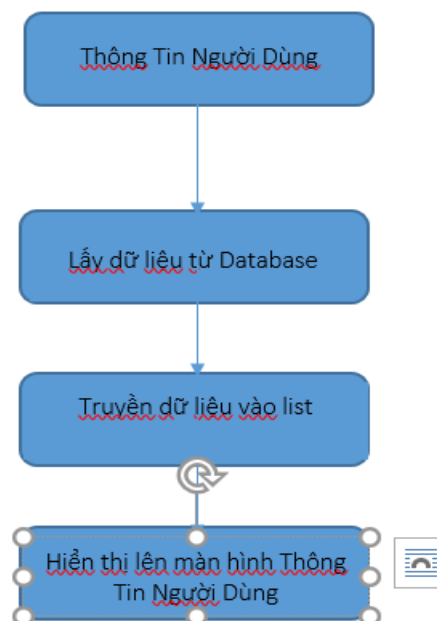
Hình 2.3.27: Mô Tả Về Luồng Xử Lý Cơ Bản Của Thông Tin Bài Đăng

*Bảng 2.3.25: Bảng User-Case Thông Tin người dùng*

Bảng USE-CASE Thông Tin Người Dùng	
Ảnh bài đăng	Hiển thị ảnh của người dùng
Họ Tên	Họ tên của người dùng
Tên Tài Khoản	Tên tài khoản của người dùng
Giới Tính	Giới tính của người dùng
Số Điện Thoại	Số điện thoại của người dùng
Tiền Điều Kiện	Hệ thống sẽ hiển thị màn hình thông tin người dùng.
Hậu Điều Kiện	Nếu Use-Case thực hiện thành công xem được thông tin của mình



*Hình 2.3.29: Thể Hiện Giao Diện Cho Màn Hình Thông Tin Người Dùng.*



*Bảng 2.3.30: Mô Tả Về Luồng Xử Lý Cơ Bản Của Thông Tin Bài Đăng.*

### III. Phần 3. Xây Dựng Cơ Sở Dữ Liệu.

#### 1. Bảng User:

Tên Trường	Kiểu dữ liệu	Độ dài	Khóa	Mô Tả
avatar	String			Là link ảnh đại diện
gender	String			Nam,nữ
keyId	String			Tự sinh
password	String	lớn hơn 6 ký tự		
phoneNumber	String	9-10 số		Số điện thoại
userName	String	Max 50 ký tự		không chữ ký tự đặc biệt.

#### 2. Bảng Image:

Tên Trường	Kiểu dữ liệu	Độ dài	Khóa	Mô Tả
idImage	String			Id của ảnh
idPost	String			Id của bài đăng
link	String			Link ảnh của mô tả nhà trọ

#### 3. Bảng Post:

Tên Trường	Kiểu dữ liệu	Độ dài	Khóa	Mô Tả
danhMuc	String			Loại nhà trọ
diaChi	String			Địa chỉ nhà trọ
dienTich	String	Max 5 số		Diện tích nhà trọ cho thuê
gia	String	Max 12 số		Giá nhà trọ
id	String			Id bài đăng
idUser	String			Id người dùng
moTa	String			Mô Tả nhà trọ
soDienThoai	String	9-10 số		Số điện thoại người đăng
tieuDe	String	Giới hạn 200 ký tự		Tiêu đề bài đăng
timePost	String			Thời gian bài đăng



## IV. Phần 4. Giao Diện Người Dùng.

### - Đăng kí

```
private void SignUp(final String s1) {
    if (validate()) {
        String keyUser = databaseReference.child(NODE_USER).child(s1).push().getKey();
        User u = new User(keyUser,
            edSignUpNhapUserName.getText().toString(),
            edSignUpHoTen.getText().toString(),
            gender,
            edSignUpSoDienThoai.getText().toString(),
            edSignUpNhapLaiPassword.getText().toString());
        final ProgressDialog progressDialog = new ProgressDialog(context: SignUpActivity.this);
        progressDialog.setMessage("Xin chờ ...");
        progressDialog.show();
        databaseReference
            .child(NODE_USER)
            .child(s1)
            .child(keyUser)
            .setValue(u) Task<Void> {
                .addOnSuccessListener((OnSuccessListener) (aVoid) -> {
                    progressDialog.dismiss();
                    Toast.makeText(context: SignUpActivity.this, text: "Đăng kí thành công", Toast.LENGTH_SHORT).show();
                    getSharedPreferences(name: "user", mode: MODE_PRIVATE).edit().putString(s: "user", edSignUpNhapUserName.getText().toString()).apply();
                    getSharedPreferences(name: "user", mode: MODE_PRIVATE).edit().putString(s: "pass", edSignUpNhapPassword.getText().toString()).apply();
                    getSharedPreferences(name: "user", mode: MODE_PRIVATE).edit().putBoolean(s: "checkbox", b: true).apply();
                    finish();
                })
                .addOnFailureListener((e) -> {
                    Toast.makeText(context: SignUpActivity.this, text: "Đăng kí không thành công", Toast.LENGTH_SHORT).show();
                });
            });
    }
}
```

+ Giải thích code: Khi nhấn nút đăng kí, client kiểm tra các trường dữ liệu được nhập đủ, đúng định dạng, tài khoản chưa tồn tại. Kiểm tra đúng, gọi firebase tạo 1 key → tạo mới đối tượng User → show dialog → thành công → lưu tài khoản database → lưu share preference → không thành công → thông báo đăng kí lại

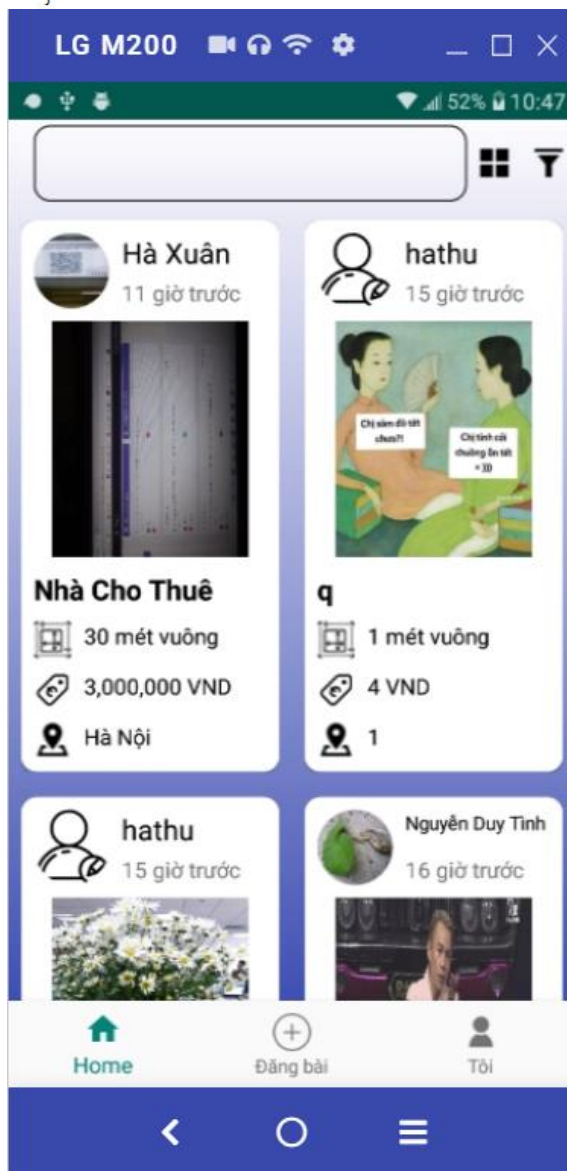
### - Đăng Nhập:

```
private void SignIn(final String s1, final String s2){
    final ProgressDialog progress = new ProgressDialog(context: LoginActivity.this);
    progress.setMessage("Xin chờ ...");
    progress.show();
    databaseReference.child("user").addChildEventListener(new ChildEventListener() {
        @Override
        public void onChildAdded(@NonNull DataSnapshot snapshot, @Nullable String previousChildName) {
            if(s1.trim().equals(snapshot.getKey())){
                databaseReference.child("user").child(snapshot.getKey()).addChildEventListener(new ChildEventListener() {
                    @Override
                    public void onChildAdded(@NonNull DataSnapshot snapshot, @Nullable String previousChildName) {
                        getSharedPreferences(name: "userLog", mode: 0).edit().putString(s: "idLog", snapshot.getKey()).apply();
                        User user = snapshot.getValue(User.class);
                        if(s2.trim().equals(user.getPassword())){
                            SharedPreferences sharedPreferences = getSharedPreferences(name: "user", mode: MODE_PRIVATE);
                            Editor editor = sharedPreferences.edit();
                            if(cbLoginLuuk.IsChecked()){
                                editor.putString(s: "user", s1);
                                editor.putString(s: "pass", s2);
                                editor.putBoolean(s: "checkbox", b: true);
                            }else{
                                editor.putString(s: "user", s1);
                                editor.putString(s: "pass", s1);
                                editor.putBoolean(s: "checkbox", b: false);
                            }
                            editor.apply();
                            getSharedPreferences(name: "userLog", mode: 0).edit().putString(s: "userName", s1.trim()).apply();
                            getSharedPreferences(name: "userLog", mode: 0).edit().putString(s: "password", s2.trim()).apply();
                            getSharedPreferences(name: "userLog", mode: 0).edit().putString(s: "keyUser", user.getKey()).apply();
                            startActivity(new Intent(context: LoginActivity.this, MainActivity.class));
                            userName = s1;
                            progress.dismiss();
                            finish();
                        }else{
                            Toast.makeText(context: LoginActivity.this, text: "Sai tài khoản hoặc mật khẩu", Toast.LENGTH_SHORT).show();
                        }
                    }
                });
            }
        }
    });
}
```

+ Giải thích code: Khi click nút đăng nhập sẽ show progress dialog → kiểm tra tài khoản mật khẩu người dùng nhập có tồn tại hay không. Nếu đã tồn tại → lưu share preference → đăng nhập, kết thúc dialog. Nếu không tồn tại thì thông báo sai tài khoản mật khẩu

### - Main:

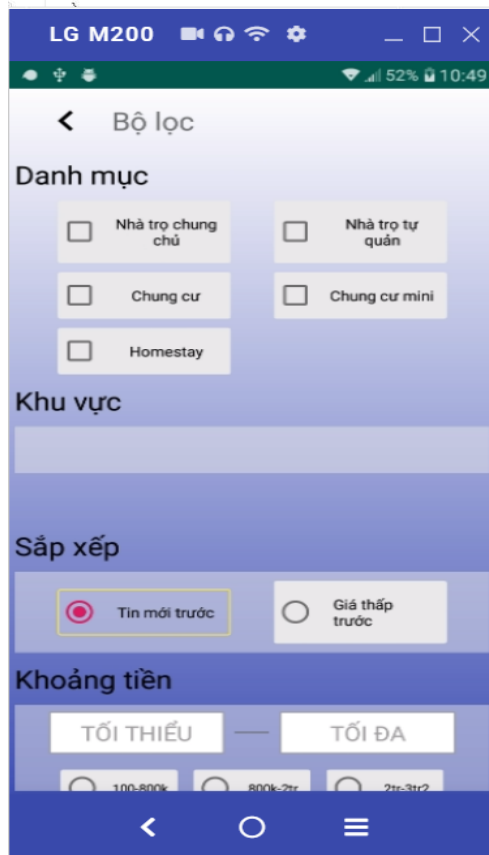
```
private void setBottomNavigation() {
    getSupportFragmentManager().beginTransaction().replace(R.id.frame_layout, HomeFragment.newInstance()).commit();
    navView.setOnNavigationItemSelectedListener((item) -> {
        Fragment fragment = null;
        if (item.getItemId() == R.id.navigation_home){
            fragment = HomeFragment.newInstance();
        }
        else if (item.getItemId() == R.id.navigation_post)
            fragment = AddPostFragment.newInstance();
        else if (item.getItemId() == R.id.navigation_profile)
            fragment = ProfileFragment.newInstance();
        getSupportFragmentManager().beginTransaction().replace(R.id.frame_layout, fragment).commit();
        return true;
    });
}
```



Giải thích code: Khi click vào item ở navigation sẽ hiện ra các fragment tương ứng với các chức năng khác nhau

## - Loại Bài Đăng:

```
final SharedPreferences sharedPreferences = getContext().getSharedPreferences( R.string.filter, Context.MODE_PRIVATE);
Log.d(TAG, "msg: " + "checkPriceAndAcreage: " + sharedPreferences.getString( R.string.place, s1: "123123"));
Log.d(TAG, "msg: " + "checkPriceAndAcreage: dia chi post" + post.getDiaChi());
if (post.getDiaChi().toLowerCase().equals(sharedPreferences.getString( R.string.place, s1: "").toLowerCase())) {
    postAllAdapter.notifyDataSetChanged();
    Log.d(TAG, "msg: " + "checkPriceAndAcreage: dia chi" + sharedPreferences.getString( R.string.place, s1: ""));
    if (!sharedPreferences.getString( R.string.moneyMin, s1: "").trim().equals("") ||
        sharedPreferences.getString( R.string.moneyMax, s1: "").trim().equals("")) {
        postAllAdapter.notifyDataSetChanged();
        if (Integer.parseInt(post.getGia()) >= Integer.parseInt(sharedPreferences.getString( R.string.moneyMin, s1: "-1")) &&
            Integer.parseInt(post.getGia()) <= Integer.parseInt(sharedPreferences.getString( R.string.moneyMax, s1: "-1"))) {
            if (!sharedPreferences.getString( R.string.acreageMin, s1: "-1").trim().equals("") ||
                sharedPreferences.getString( R.string.acreageMax, s1: "-1").trim().equals("")) {
                postAllAdapter.notifyDataSetChanged();
                if (Integer.parseInt(post.getDienTich()) >= Integer.parseInt(sharedPreferences.getString( R.string.acreageMin, s1: "0")) &&
                    Integer.parseInt(post.getDienTich()) <= Integer.parseInt(sharedPreferences.getString( R.string.acreageMax, s1: "0"))) {
                    sortPost(post);
                }
            } else {
                sortPost(post);
            }
        }
    }
} else {
    if (!sharedPreferences.getString( R.string.acreageMin, s1: "-1").trim().equals("") ||
        sharedPreferences.getString( R.string.acreageMax, s1: "-1").trim().equals("")) {
        postAllAdapter.notifyDataSetChanged();
        if (Integer.parseInt(post.getDienTich()) >= Integer.parseInt(sharedPreferences.getString( R.string.acreageMin, s1: "0")) &&
            Integer.parseInt(post.getDienTich()) <= Integer.parseInt(sharedPreferences.getString( R.string.acreageMax, s1: "0"))) {
            sortPost(post);
        }
    } else {
        sortPost(post);
    }
}
}
```



Giải thích code: Kiểm tra từ trên xuống dưới từng trường xem người dùng có chọn hay không → nếu điều kiện trên không chọn thì chuyển qua kiểm tra điều kiện tiếp theo, nếu điều kiện trên được chọn thì kiểm tra tiếp tục điều kiện tiếp cứ như vậy đến hết. Ta nhận được list bản tin phù hợp

## - Người Dùng:

```
ivPersonalMainPickImage.setOnClickListener((view) -> {
    Intent pickIntent = new Intent(Intent.ACTION_PICK, MediaStore.Images.Media.INTERNAL_CONTENT_URI);
    startActivityForResult(pickIntent, PICK_PHOTO_FOR_AVATAR);
    ivPersonalMainPickImage.setBackgroundResource(R.drawable.bg_image_select);
});

imgPersonalMainAvatar.setOnClickListener((view) -> {
    final PopupMenu menu = new PopupMenu(getContext(), view);
    MenuInflater inflater = menu.getMenuInflater();
    inflater.inflate(R.menu.menu_image_select, menu.getMenu());
    menu.setOnMenuItemClickListener((menuItem) -> {
        if (menuItem.getItemId() == R.id.menu_upload_image) {
            Intent pickIntent = new Intent(Intent.ACTION_PICK, MediaStore.Images.Media.INTERNAL_CONTENT_URI);
            startActivityForResult(pickIntent, PICK_PHOTO_FOR_AVATAR);
            ivPersonalMainPickImage.setBackgroundResource(R.drawable.bg_image_select);
            return true;
        } else if (menuItem.getItemId() == R.id.menu_seen_image) {
            startActivity(intent);
            return true;
        } else return false;
    });
    menu.show();
});
```



Giải thích code: khi người dùng click vào ảnh hay biểu tượng chọn ảnh thì chuyển sang bộ sưu tập ảnh của máy → khi người dùng chọn được ảnh thì set ảnh lên giao diện của app

## - Thông Tin Người Dùng:

```
databaseReference.child("user").child(getSharedPreferences( name: "userLog", mode: 0).getString( s: "userName", s1: null)).addChildEventListener(new ChildEventListener() {
    @Override
    public void onChildAdded(@NonNull DataSnapshot snapshot, @Nullable String previousChildName) {
        User user = snapshot.getValue(User.class);
        Picasso.get().load(user.getAvatar()).resize( targetWidth: 120, targetHeight: 120).placeholder(R.drawable.user_3).into(imgPersonalMainInfoAvatar);
        tvPersonalMainInfoName.setText(user.getName());
        tvPersonalMainInfoNumberPhone.setText(user.getPhoneNumber());
        tvPersonalMainInfoSex.setText(user.getGender());
        tvPersonalMainInfoTaikhoan.setText(user.getUserName());
    }

    @Override
    public void onChildChanged(@NonNull DataSnapshot snapshot, @Nullable String previousChildName) {
    }

    @Override
    public void onChildRemoved(@NonNull DataSnapshot snapshot) {
    }

    @Override
    public void onChildMoved(@NonNull DataSnapshot snapshot, @Nullable String previousChildName) {
    }

    @Override
    public void onCancelled(@NonNull DatabaseError error) {
    }
});
```

Giải thích code: Lấy dữ liệu từ firebase rồi set vào các trường tương ứng





## - Sửa Thông Tin Người Dùng:

```
private void changeDataUser() {
    final Map<String, Object> map = new HashMap<>();

    if (rbGtNam.isChecked()) {
        map.put( k: "gender", rbGtNam.getText().toString());
    }
    if (rbGtNu.isChecked()) {
        map.put( k: "gender", rbGtNu.getText().toString());
    }

    map.put( k: "name", edtHoTen.getText().toString().trim());
    map.put( k: "phoneNumber", edtSoDienThoai.getText().toString().trim());
    final ProgressDialog progressDialog = new ProgressDialog( context: ChangeInfoActivity.this);
    progressDialog.show();

    databaseReference
        .child("user") DatabaseReference
        .child(getSharedPreferences( name: "userLog", mode: 0).getString( s: "userName", s1: null))
        .child(getSharedPreferences( name: "userLog", mode: 0).getString( s: "idLog", s1: null))
        .updateChildren(map) Task<Void>
        .addOnSuccessListener((OnSuccessListener) (aVoid) → {
            progressDialog.dismiss();
            Toast.makeText( context: ChangeInfoActivity.this, text: "Cập nhật thành công", Toast.LENGTH_SHORT).show();
        })
        .addOnFailureListener((e) → {
            progressDialog.dismiss();
            Toast.makeText( context: ChangeInfoActivity.this, text: "Không thành công", Toast.LENGTH_SHORT).show();
        });
}
```

Giải thích code: lấy dữ liệu người dùng thay đổi đưa vào 1 map → dùng câu lệnh thích hợp update dữ liệu người dùng đã tồn tại trên firebase

## - Cập Nhật Password:

```

btnUpdatePasswordUpdate.setOnClickListener((view) -> {
    databaseReference
        .child("user")
        .child(getSharedPreferences( name: "userLog", mode: 0).getString( s: "userName", s1: null))
        .addChildEventListener(new ChildEventListener() {
            @Override
            public void onDataChange(@NonNull DataSnapshot snapshot, @Nullable String previousChildName) {
                User user = snapshot.getValue(User.class);
                if (user.getPassword().equals(edtUpdatePasswordOld.getText().toString().trim())) {
                    if (edtUpdatePasswordNew.getText().toString().length() == 0 || edtUpdatePasswordNew2.getText().toString().length() == 0) {
                        Toast.makeText( context: ChangePasswordActivity.this, text: "Chưa nhập mật khẩu mới", Toast.LENGTH_SHORT).show();
                    } else if (edtUpdatePasswordNew.getText().toString().trim().equals(edtUpdatePasswordNew2.getText().toString().trim())
                        && edtUpdatePasswordNew.getText().toString().length() >= 6) {
                        Map<String, Object> map = new HashMap<>();
                        map.put( k: "password", edtUpdatePasswordNew.getText().toString());
                        databaseReference.child("user")
                            .child(getSharedPreferences( name: "userLog", mode: 0).getString( s: "userName", s1: null))
                            .updateChildren(map) Task<Void>
                            .addOnSuccessListener((onSuccessListener) (aVoid) -> {
                                Toast.makeText( context: ChangePasswordActivity.this, text: "Mật khẩu cập nhật thành công", Toast.LENGTH_SHORT).show();
                                finish();
                            })
                            .addOnFailureListener((e) -> {
                                Toast.makeText( context: ChangePasswordActivity.this, text: "Cập nhật không thành công", Toast.LENGTH_SHORT).show();
                            });
                    } else {
                        Toast.makeText( context: ChangePasswordActivity.this, text: "Sai định dạng mật khẩu hoặc không khớp", Toast.LENGTH_SHORT).show();
                    } else if (edtUpdatePasswordOld.getText().toString().length() == 0) {
                        Toast.makeText( context: ChangePasswordActivity.this, text: "Chưa nhập mật khẩu cũ", Toast.LENGTH_SHORT).show();
                    }
                }
            }
            @Override
            public void onCancelled(@NonNull DataSnapshot snapshot, @Nullable String previousChildName) {
            }
        });
    }
}

```

Giải thích code: sau khi kiểm tra sơ bộ mật khẩu nhập mới giống nhau, và mật khẩu cũ người dùng nhập giống với mật khẩu của tài khoản đang đăng nhập → tiến hành update mật khẩu rồi đưa ra thông báo khi thành công hay thất bại

## - Đăng Bài:

```
private void uploadPost() {
    if (validate()) {
        final Date date = Calendar.getInstance().getTime();
        final String key = databaseReference.child("post").push().getKey();
        String idUser = databaseReference.child("user").child(getActivity().getSharedPreferences("userLog", mode: 0).getString("userName", "")).getKey();
        final Post post = new Post(key, idUser, edtAddPostTitle.getText().toString(), edtAddPostPrice.getText().toString(),
            edtAddPostAddress.getText().toString(), edtAddPostPhoneNumber.getText().toString(),
            edtAddPostAcreage.getText().toString(), spnAddPostBrand.getSelectedItem().toString(), edtAddPostDescription.getText().toString(), date.toString());
        databaseReference.child("post").child(key).setValue(post);
        final ProgressDialog dialog = new ProgressDialog(getContext());
        dialog.show();

        for (Uri uri : uris) {
            final String nameImage = UUID.randomUUID().toString();
            StorageReference
                .child("images/" + nameImage) StorageReference
                .putFile(uri) UploadTask
                .addOnSuccessListener((OnSuccessListener) (taskSnapshot) -> {
                    StorageReference
                        .child("images/" + nameImage)
                        .getDownloadUrl()
                        .addOnSuccessListener((OnSuccessListener) (uri) -> {
                            Log.d(TAG, "onSuccess: " + uri.toString());
                            String keyImage = databaseReference.child("post").child(key).child("linkAnh").push().getKey();
                            Image image = new Image(keyImage, key, uri.toString());
                            databaseReference
                                .child("post")
                                .child(key)
                                .child("linkAnh")
                                .child(keyImage)
                                .setValue(image);
                        });
                    dialog.dismiss();
                }) StorageTask<UploadTask.TaskSnapshot>
                .addOnFailureListener((e) -> {
                    Toast.makeText(getContext(), "Fail", Toast.LENGTH_SHORT).show();
                })
                .addOnProgressListener((OnProgressListener) (snapshot) -> {
                    double progress = (100.0 * snapshot.getBytesTransferred() / snapshot
                        .getTotalByteCount());
                });
        }
    }
}
```



Giải thích code: sau khi kiểm tra các trường nhập đúng format → tạo ra các trường cần thiết → Thêm dữ liệu vào firebase



## - Bài Đăng Của Tôi:

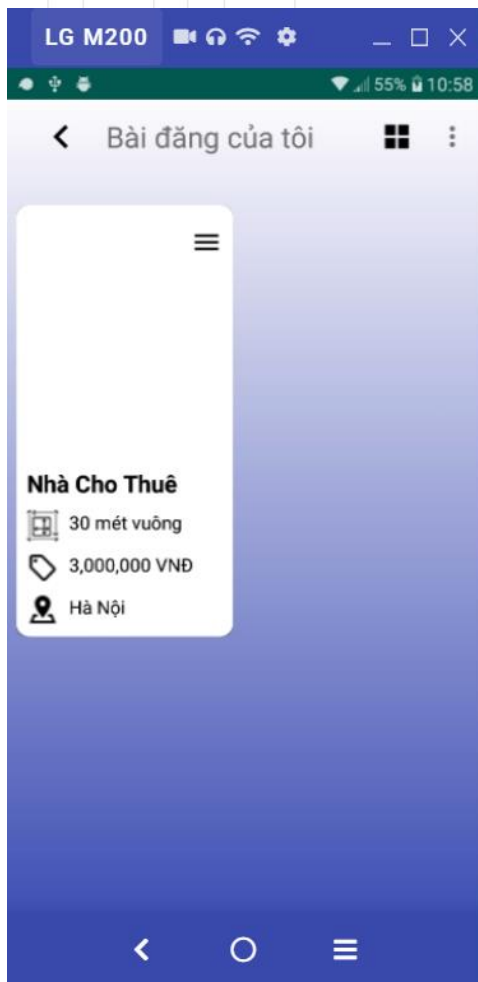
```

private void onDataChange() {
    databaseReference
        .child("post")
        .addChildEventListener(new ChildEventListener() {
            @Override
            public void onChildAdded(@NonNull DataSnapshot snapshot, @Nullable String previousChildName) {
                databaseReference
                    .child("post")
                    .child(snapshot.getKey())
                    .child("linkAnh")
                    .addChildEventListener(new ChildEventListener() {
                        @Override
                        public void onChildAdded(@NonNull DataSnapshot snapshot, @Nullable String previousChildName) {
                            images.add(snapshot.getValue(Image.class));
                            postPersonalAdapter.notifyDataSetChanged();
                        }
                    });
            }

            @Override
            public void onChildChanged(@NonNull DataSnapshot snapshot, @Nullable String previousChildName) {
            }

            @Override
            public void onChildRemoved(@NonNull DataSnapshot snapshot) {
            }
        });
}

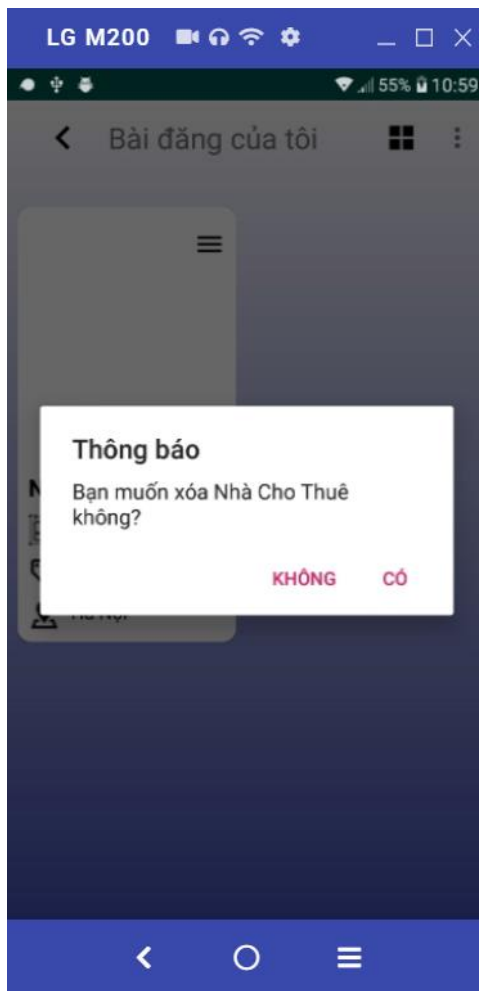
```



Giải thích code: lấy dữ liệu bài đăng từ firebase từ đúng tài khoản đăng nhập rồi thêm vào list → đổ dữ liệu ra recyclerview.

## - Xóa Bài Đăng:

```
private void deleteItem(final Post post) {
    AlertDialog.Builder builder = new AlertDialog.Builder(context);
    builder.setTitle("Thông báo");
    builder.setMessage("Bạn muốn xóa " + post.getTieuDe() + " không?");
    builder.setPositiveButton("Có", (dialogInterface, i) → {
        databaseReference.child("post").child(post.getId()).removeValue();
        ((Activity) context).finish();
        context.startActivity(((Activity) context).getIntent());
    });
    builder.setNegativeButton("Không", listener: null);
    builder.show();
}
```



Giải thích code: Khi ấn vào nút có ở alert dialog  
→ dùng interface của firebase → remove bài đăng đó

## - Sửa Bài Đăng:

```

databaseReference
    .child("post")
    .child(post.getId())
    .setValue(post1) Task<Void>
    .addOnSuccessListener((OnSuccessListener) (aVoid) -> {
        Toast.makeText( context: EditPostActivity.this, text: "Thành công", Toast.LENGTH_SHORT).show();
    })
    .addOnFailureListener((e) -> {
        Toast.makeText( context: EditPostActivity.this, text: "Gặp lỗi trong quá trình update", Toast.LENGTH_SHORT).show();
    });
final ProgressDialog dialog = new ProgressDialog( context: EditPostActivity.this);
dialog.show();
databaseReference
    .child("post")
    .child(post.getId())
    .child("linkAnh")
    .removeValue();
for (Uri uri : uris) {
    final String nameImage = UUID.randomUUID().toString();
    storageReference
        .child("images/" + nameImage) StorageReference
        .putFile(uri) UploadTask
        .addOnSuccessListener((OnSuccessListener) (taskSnapshot) -> {
            storageReference
                .child("images/" + nameImage)
                .getDownloadUrl()
                .addOnSuccessListener((OnSuccessListener) (uri) -> {
                    String keyImage = databaseReference.child("post").child(post.getId()).child("linkAnh").push().getKey();
                    Image image = new Image(keyImage, post.getId(), uri.toString());
                    databaseReference
                        .child("post")
                        .child(post.getId())
                        .child("linkAnh")
                        .child(keyImage)
                        .setValue(image)
                        .addOnSuccessListener((OnSuccessListener) (aVoid) -> {

```



Giải thích code: lấy ra đúng bài đăng của người dùng hiện tại rồi chỉnh sửa các trường → khi các trường được nhập đúng format → dùng interface của firebase update bài đăng

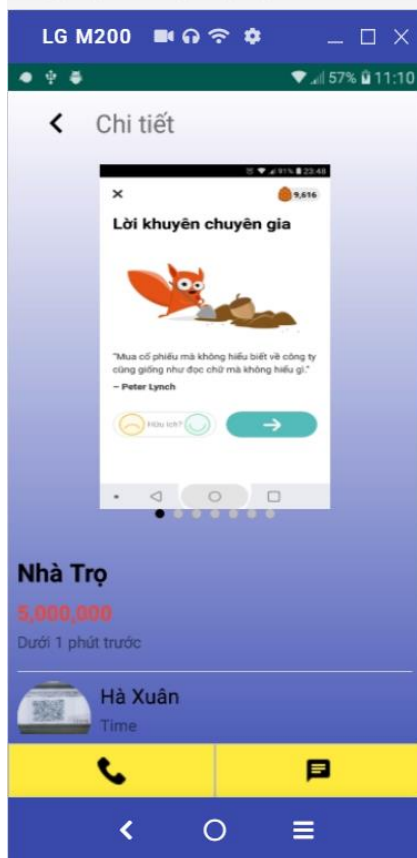
## - Bài Đăng Chi Tiết:

```

databaseReference
    .child("post")
    .addChildEventListener(new ChildEventListener() {
        @Override
        public void onChildAdded(@NonNull DataSnapshot snapshot, @Nullable String previousChildName) {
            Log.d(TAG, "onChildAdded: personal post" + snapshot.getValue(Post.class).getTieuDe());
            final Post post = snapshot.getValue(Post.class);
            if (keyPost.equals(post.getId())) {
                tvPostDetailTitle.setText(post.getTieuDe());
                tvPostDetailAcreage.setText("Diện tích: " + post.getDienTich() + " mét vuông");
                tvPostDetailDescription.setText(post.getMoTa());
                tvPostDetailPhoneNumber.setText("Liên hệ ngay: " + post.getSoDienThoai());
                tvPostDetailPlace.setText("Khu vực: " + post.getDiaChi());
                tvPostDetailBrand.setText("Danh mục: " + post.getDanhMuc());
                NumberFormat numberFormat = NumberFormat.getInstance();
                tvPostDetailPrice.setText(numberFormat.format(Double.parseDouble(post.getGia())));
                Date date = Calendar.getInstance().getTime();
                Date date1 = new Date(post.getTimePost());
                long diff = (date.getTime() - date1.getTime()) / 1000;
                int days = (int) (diff / 86400);
                int hours = (int) ((diff % 86400) / 3600);
                int minutes = (int) ((diff % 86400) % 3600) / 60;
                if (days >= 1) {
                    tvPostDetailTimePost.setText(days + " " + "ngày trước");
                } else if (hours >= 1) {
                    tvPostDetailTimePost.setText(hours + " " + "giờ trước");
                } else if (minutes >= 1) {
                    tvPostDetailTimePost.setText(minutes + " " + "phút trước");
                } else if (minutes < 1) {
                    tvPostDetailTimePost.setText("Dưới 1 phút trước");
                }
            }
        }
    });

```

Giải thích code: lấy ra toàn bộ chi tiết bài đăng ở firebase rồi set vào các trường tương ứng



## V. Phần 5 Tổng Kết:

### 5.1 Thời gian hoàn thành dự án:

- Dự Án thực hiện từ ngày 9/11/2020 – 12/12/2020 chi tiết như sau:
- + 9/11 – 16/11: Lên ý tưởng, thu nhập thông tin khảo sát về ý tưởng
- + 20/11: Hoàn thành mockup
- + 22/11: Hoàn thành Prototype thiết kế bằng Figma
- + 28/11: Hoàn thành giao diện FrontEnd
- + 2/12: Làm song các chức năng cơ bản
- + 7/12: Hoàn thành các chức năng của dự án
- + 12/12: Hoàn thành kiểm thử, sửa lỗi
- + 12/12: Hoàn thành song dự án.

### 5.2 Mức độ hoàn thành dự án:

- So với mục tiêu đặt ra đầu tiên của nhóm thì mức độ hoàn thành của nhóm chúng em đã đạt 100%.
- Đã hoàn thành tất cả mục tiêu đề ra ban đầu.

### 5.3 Khó khăn, rủi ro gặp phải và cách giải quyết.

Khó Khăn	Cách Giải Quyết
Các thành viên chưa đồng nhất với ý tưởng.	Tổ chức họp đầu mỗi hàng tuần để thống nhất ý tưởng.
Các bạn thành viên trong nhóm còn bận đi làm hoặc việc học.	Thành viên cố gắng sắp xếp thời gian hợp lý để không làm ảnh hưởng tới nhóm.
Khó khăn trong việc liên lạc với nhau	Chúng em sử dụng hình thức liên lạc chính là gọi điện thoại, facebook và zalo.

### 5.4 Những bài học rút ra:

Trong quá trình làm dự án với nhau chúng em rút ra được những kinh nghiệm như sau:

- Chủ động lắng nghe: Lắng nghe những ý kiến của các thành viên khác,
- Tìm hiểu và hoàn thiện những thiếu sót của bản thân
- Tôn trọng ý kiến của người khác
- Đặt tinh thần trách nhiệm vào công việc: Khi được bàn giao công việc,

- luôn cố gắng hoàn thành công việc sớm nhất có thể
- Kiểm soát tiến trình công việc
- Hiểu được các phân chia công việc theo khả năng của các thành viên.

### **LỜI CẢM ƠN**

Trước tiên chúng em xin gửi lời cảm ơn chân thành sâu sắc tới các thầy cô giáo trong trường FPT Polytechnic nói chung và các thầy cô giáo trong khoa Công nghệ Thông tin, chuyên ngành ứng dụng phần mềm nói riêng đã tận tình giảng dạy, truyền đạt cho chúng em những kiến thức, kinh nghiệm quý báu trong suốt thời gian qua.

Đặc biệt em xin gửi lời cảm ơn đến thầy Nguyễn Hữu Duy, thầy đã tận tình giúp đỡ, trực tiếp chỉ bảo, hướng dẫn em trong suốt quá trình làm đồ án. Trong thời gian làm việc với thầy, em không ngừng tiếp thu thêm nhiều kiến thức bổ ích mà còn học tập được tinh thần làm việc, thái độ nghiên làm việc nghiêm túc, hiệu quả, đây là những điều rất cần thiết cho em trong quá trình học tập và công tác sau này.

Đồng thời xin chân thành cảm ơn thầy Lã Ngọc Quang và thầy Vũ Chí Thành, đã tạo mọi điều kiện về cơ sở vật chất giúp chúng có một môi trường tốt để học tập trong suốt thời gian học tập qua.

Sau cùng xin gửi lời cảm ơn chân thành tới gia đình, bạn bè đã động viên, đóng góp ý kiến và giúp đỡ trong quá trình học tập, nghiên cứu và hoàn thành đồ án.