

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC PHENIKAA**



**BÀI TẬP LỚN**

**TÊN HỌC PHẦN: LẬP TRÌNH CHO THIẾT BỊ DI ĐỘNG**

**LỚP: N05**

**HỌC KỲ II NĂM HỌC 2023-2024**

**Đề tài: Xây dựng Ứng dụng tra cứu thông tin**

**về các loài Mèo**

**Hà Nội, năm 2024**

# LỜI NÓI ĐẦU

Hiện nay, việc dùng thiết bị di động để tìm kiếm thông tin và tương tác với nhiều loài động vật không những mang lại niềm vui mà còn giúp người sử dụng hiểu sâu hơn về môi trường sống. Trong chủ đề hôm nay, nhóm bọn em sẽ tập trung vào một loài động vật đặc biệt: mèo. Mèo là một trong những loài thú cưng phổ biến nhất trên trái đất đã thu hút sự chú ý của hàng triệu người.

Ứng dụng của nhóm em sẽ không chỉ cung cấp thông tin về các giống mèo, mà còn giúp người dùng tra cứu, xem chi tiết và lưu trữ thông tin yêu thích của mình. Để đảm bảo tính an toàn và tính bảo mật trong việc lưu trữ thông tin, chúng ta sẽ sử dụng dịch vụ Firebase để xác thực người dùng, lưu trữ thông tin yêu thích trên cloud và cung cấp một giao diện đẹp mắt, dễ dàng sử dụng.

Nhóm em sẽ khám phá và tích hợp các API từ nhiều nguồn khác nhau để cung cấp thông tin đa dạng về từng giống mèo, kèm theo hình ảnh giúp người dùng dễ dàng thưởng thức. Đồng thời, ứng dụng sẽ cung cấp tính năng tìm kiếm dựa trên nhiều thông tin khác nhau giúp người dùng có thể dễ dàng tìm kiếm thông tin mèo theo sở thích của họ.

Tóm lại, mục tiêu của chúng ta là phát triển một ứng dụng di động cho các loài mèo, với các tính năng như xác thực người dùng, đăng nhập thông qua Firebase, tìm kiếm và hiển thị thông tin chi tiết cho từng loài mèo, cộng với khả năng lưu trữ thông tin yêu thích trên cloud hoặc local. Đây sẽ là một trải nghiệm thú vị và hữu ích đối với những người yêu thích mèo và động vật.

# MỤC LỤC

[**LỜI NÓI ĐẦU 0**](#_fhflvlyu78qq)

[**MỤC LỤC 1**](#_2wnlcuqpyzpf)

[**MỤC LỤC HÌNH ẢNH 2**](#_xj440iq3rn34)

[**BẢNG CÁC TỪ VIẾT TẮT 3**](#_qjtl1zkonx4w)

[**Chương 1. GIỚI THIỆU CÔNG NGHỆ 4**](#_ihgxet4015t0)

[1.1. Giới thiệu về công nghệ Flutter 4](#_wwn5695v1nt5)

[1.1.1. Giới thiệu Flutter: 4](#_j6vjctjdl62r)

[1.1.2. Lợi ích của việc sử dụng Flutter: 4](#_qehbapr2kiem)

[1.1.2.1. Phát triển đa nền tảng: 4](#_tdufsip6ty2j)

[1.1.2.2. Hiệu suất cao: 4](#_ux3e4744ck1d)

[1.1.2.3. Giao diện người dùng đẹp và linh hoạt: 4](#_v4201e1n1zwy)

[1.1.2.4. Hot Reload: 5](#_d49skc4kg4z5)

[1.1.2.5. Cộng đồng và tài liệu phong phú: 5](#_ve73ik34wx5c)

[1.1.3. Tổng quan về kiến trúc Flutter 5](#_cwe49z4shlxd)

[1.1.3.1. Flutter Framework: 5](#_4dgy28er44ty)

[1.1.3.2. Flutter Engine: 6](#_8nlcwzelv5qq)

[1.2. Giới thiệu về firebase 7](#_mioaropfynu9)

[1.2.1. Firebase là gì 7](#_cvrwj66ar3ot)

[1.2.2. Các dịch vụ Firebase 7](#_f94tietegtks)

[1.2.3. Sử dụng Firebase Authentication để xác thực người dùng 8](#_16ns67a3m9ao)

[1.3. Giới thiệu về IFCAT 10](#_ex33q64j1xas)

[1.3.1. IFCAT là gì? 10](#_es8dawwuw18q)

[1.3.2. Khả năng tích hợp IFCAT vào ứng dụng di động 10](#_phfhaomn92vy)

[**Chương 2. Xây dựng ứng dụng di động 1**](#_d93cbo5slpmd)**2**

[2.1. Cài đặt môi trường 1](#_wptvuheena9d)2

[2.1.1. Cài đặt Firebase 1](#_mng2lrkxrcfu)2

[2.2. Xây dựng ứng dụng](#_gzq58pbm56gi) 14

[2.2.1. Chức năng đăng nhập & đăng ký](#_c1ufzqz82mr) 15

[2.2.2.](#_vhz0men78020)Giao diện trang chủ19

[2.2.2.](#_vhz0men78020)Giao diện tìm kiếm19

**Kết luận****20**

**Danh mục tham khảo****22**

# 

# MỤC LỤC HÌNH ẢNH

[Hình 1:](#_l1j9pgw3sys1) Màn hình khởi tạo project trên Firebase [1](#_l1j9pgw3sys1)2

[Hình 2:](#_h3q2txetecsm) Màn hình Authentication của Firebase…………………………………...…[1](#_h3q2txetecsm)3

[Hình 3:](#_a76p2bnftu7k) Màn hình Cloud Firestore của Firebase……………………………………...[15](#_a76p2bnftu7k)

Hinh 4: Màn hình demo trang Log In [16](#_3216ylocfdz9)

Hình 5:*Màn hình demo trang Sign Up* [17](#_itopbgw00wd4)

[Hình 6:](#_5ucbf6li484n) *Màn hình demo trang chủ* [1](#_5ucbf6li484n)8

[Hình 7:](#_iulnyl5zjjhh) *Màn hình demo xem chi tiết*19

[Hinh 8:](#_ktjf9octmnkg) *Màn hình demo tìm kiếm*20

# BẢNG CÁC TỪ VIẾT TẮT

| **STT** | **TỪ VIẾT TẮT** | **VIẾT ĐẦY ĐỦ** |
| --- | --- | --- |
| **1** | **CSDL** | **Cơ sở dữ liệu** |
| **2** | **UI** | **User Interface (giao diện người dùng)** |
| **3** | **API** | **Application Programming Interface (Giao diện lập trình ứng dụng)** |
| **4** | **VM** | **Virtual Machine**  **(máy ảo)** |
| **5** | **FCM** | **Firebase Cloud Messaging** |
| **6** | **GPS** | **Global Positioning System**  **(Hệ thống định vị toàn cầu)** |

# GIỚI THIỆU CÔNG NGHỆ

## Giới thiệu về công nghệ Flutter

### Giới thiệu Flutter:

Flutter là một framework mã nguồn mở phát triển bởi Google, được sử dụng để xây dựng ứng dụng di động đa nền tảng. Nó cho phép nhà phát triển tạo ra các ứng dụng chất lượng cao với giao diện người dùng hấp dẫn và hiệu suất tốt trên cả hệ điều hành Android và iOS từ một mã nguồn duy nhất.

Cách Flutter hoạt động: Khi ứng dụng Flutter được chạy, mã nguồn Flutter được biên dịch thành mã máy gốc thông qua Flutter Engine. Mã máy gốc này được chạy trực tiếp trên hệ điều hành của thiết bị mà không cần thông qua bất kỳ lớp trung gian nào. Điều này cho phép Flutter đạt được hiệu suất cao và giao diện người dùng mượt mà.

### Lợi ích của việc sử dụng Flutter:

#### Phát triển đa nền tảng:

Flutter cho phép nhà phát triển tạo ra một ứng dụng di động đa nền tảng từ một mã nguồn duy nhất. Điều này giúp tiết kiệm thời gian và công sức, vì không cần viết lại toàn bộ ứng dụng cho từng nền tảng riêng biệt. Flutter hỗ trợ phát triển cho cả Android và iOS, giúp giảm bớt công việc phát triển và duy trì ứng dụng trên nhiều nền tảng.

#### Hiệu suất cao:

Flutter sử dụng Flutter Engine để biên dịch mã nguồn thành mã máy gốc, cho phép ứng dụng chạy mượt mà và đạt hiệu suất cao. Flutter sử dụng Skia, một thư viện đồ họa mạnh mẽ, để vẽ giao diện người dùng, giúp tạo ra các thành phần giao diện đẹp và tương tác mượt mà.

#### Giao diện người dùng đẹp và linh hoạt:

Flutter cung cấp một bộ công cụ giao diện người dùng (UI toolkit) mạnh mẽ gọi là "Widget" để xây dựng giao diện người dùng. Widget trong Flutter là tương tác và có thể được kết hợp với nhau để tạo ra các giao diện phức tạp. Flutter cũng cung cấp các widget phong phú và hỗ trợ tùy chỉnh giao diện linh hoạt, giúp nhà phát triển tạo ra các ứng dụng di động đẹp và đồng nhất trên các nền tảng khác nhau.

#### Hot Reload:

Flutter cung cấp tính năng Hot Reload, cho phép nhà phát triển thực hiện thay đổi và xem kết quả ngay lập tức trên máy ảo hoặc thiết bị thật. Điều này giúp tăng tốc quá trình phát triển và thử nghiệm ứng dụng, giúp nhà phát triển điều chỉnh và cải thiện giao diện người dùng một cách nhanh chóng.

#### Cộng đồng và tài liệu phong phú:

Flutter có một cộng đồng lớn và năng động, cung cấp hỗ trợ và tài liệu phong phú cho nhà phát triển. Cộng đồng này cung cấp các gói thư viện mở rộng và công cụ hữu ích, giúp giảm thiểu công việc lập trình và tăng tốc quá trình phát triển ứng dụng.

Tóm lại, Flutter là một framework mạnh mẽ cho phát triển ứng dụng di động đa nền tảng, mang lại nhiều lợi ích như phát triển đa nền tảng, hiệu suất cao, giao diện đẹp và linh hoạt, hot reload và hỗ trợ từ cộng đồng lớn. Việc sử dụng Flutter giúp nhà phát triển tiết kiệm thời gian và công sức trong quá trình phát triển ứng dụng di động.

### Tổng quan về kiến trúc Flutter

Kiến trúc của Flutter bao gồm hai thành phần chính là Flutter Framework và Flutter Engine.

#### Flutter Framework:

Flutter Framework là một bộ công cụ phát triển ứng dụng di động đa nền tảng. Nó cung cấp các thành phần và API để xây dựng giao diện người dùng và xử lý logic ứng dụng. Flutter Framework bao gồm:

* Widget: Flutter sử dụng một cấu trúc gọi là Widget để xây dựng giao diện người dùng. Widget trong Flutter là các thành phần đơn giản, có thể kết hợp và lồng nhau để tạo thành các giao diện phức tạp. Flutter cung cấp một số lượng lớn các widget sẵn có và cho phép tùy chỉnh widget theo nhu cầu.
* Rendering Engine: Flutter sử dụng một rendering engine riêng gọi là Skia để vẽ giao diện người dùng trên màn hình. Skia là một thư viện đồ họa mạnh mẽ, đa nền tảng và hiệu quả, giúp Flutter tạo ra các giao diện đẹp và mượt mà trên cả Android và iOS.
* Gestures và Animations: Flutter cung cấp hỗ trợ tốt cho các tương tác và hiệu ứng chuyển động trong ứng dụng. Nó cung cấp một số lượng lớn các gestures sẵn có (như nhấn, vuốt, kéo) và hỗ trợ tạo ra các hiệu ứng chuyển động mượt mà và hấp dẫn.

#### Flutter Engine:

Flutter Engine là một công cụ đa nền tảng (cross-platform engine), chịu trách nhiệm biên dịch và chạy mã nguồn Flutter. Flutter Engine sử dụng Dart VM (Virtual Machine) để chạy mã nguồn Flutter và sử dụng Skia (Skia là một thư viện đồ họa mã nguồn mở được phát triển bởi Google. Nó cung cấp các công cụ và API để vẽ giao diện người dùng và thực hiện các thao tác đồ họa trên nền tảng đa nền tảng. Skia được sử dụng rộng rãi trong các dự án của Google như Chrome, Android và cả Flutter.) để vẽ giao diện người dùng. Nó cũng cung cấp các khả năng như quản lý tài nguyên, xử lý sự kiện và tương tác với hệ điều hành.

## Giới thiệu về firebase

### Firebase là gì

Firebase là một nền tảng phát triển ứng dụng di động và web của Google. Nó cung cấp một tập hợp các dịch vụ đám mây (cloud services) để giúp nhà phát triển xây dựng, triển khai và quản lý ứng dụng một cách dễ dàng và hiệu quả. Firebase giúp giảm bớt công việc phát triển phía máy chủ (backend) và cung cấp các tính năng như lưu trữ dữ liệu, xác thực người dùng, thông báo đẩy và phân tích.

### Các dịch vụ Firebase

Các dịch vụ Firebase cung cấp và vai trò của chúng trong phát triển ứng dụng:

* Firebase Authentication: Firebase Authentication cung cấp các tính năng xác thực người dùng trong ứng dụng. Nó hỗ trợ các phương thức xác thực phổ biến như email/mật khẩu, xác thực xã hội (Google, Facebook, Twitter), xác thực số điện thoại và nhiều hơn nữa. Firebase Authentication giúp xác thực và quản lý người dùng trong ứng dụng một cách dễ dàng và an toàn.
* Firebase Realtime Database: Firebase Realtime Database là một cơ sở dữ liệu thời gian thực, lưu trữ dữ liệu dạng JSON. Nó cho phép đồng bộ dữ liệu trực tiếp và tức thì giữa các thiết bị và nền tảng khác nhau. Firebase Realtime Database giúp lưu trữ và đồng bộ dữ liệu của ứng dụng một cách đơn giản và linh hoạt.
* Firebase Cloud Firestore: Firebase Cloud Firestore là một cơ sở dữ liệu linh hoạt, mở rộng, và thời gian thực. Firestore sử dụng cấu trúc tài liệu (document-oriented structure) và cung cấp tính năng truy vấn mạnh mẽ để lưu trữ và truy xuất dữ liệu. Firestore hỗ trợ đồng bộ dữ liệu trực tiếp và có tính năng offline, giúp đảm bảo dữ liệu luôn được cập nhật và sẵn sàng cho các thiết bị khác nhau.
* Firebase Cloud Messaging: Firebase Cloud Messaging (FCM) là một dịch vụ thông báo đẩy (push notification) cho phép gửi tin nhắn từ máy chủ đến ứng dụng di động và web. FCM giúp gửi thông báo đẩy tới người dùng để thông báo về các cập nhật, tin tức hoặc sự kiện quan trọng.
* Firebase Analytics: Firebase Analytics cung cấp các công cụ phân tích để hiểu và theo dõi hành vi người dùng trong ứng dụng. Nó thu thập thông tin về lượt sử dụng, sự tương tác và các sự kiện trong ứng dụng, giúp nhà phát triển đánh giá hiệu suất và tối ưu hóa ứng dụng.

Và nhiều dịch vụ khác như Firebase Storage, Firebase Hosting, Firebase Remote Config, Firebase Crashlytics, ...

### Sử dụng Firebase Authentication để xác thực người dùng

Firebase Authentication là một dịch vụ của nền tảng Firebase của Google, cung cấp các tính năng và công cụ để xác thực người dùng trong ứng dụng di động và web. Firebase Authentication giúp nhà phát triển xây dựng hệ thống xác thực mạnh mẽ, dễ dàng tích hợp và quản lý người dùng trong ứng dụng của họ.

Với Firebase Authentication, nhà phát triển có thể triển khai các phương thức xác thực người dùng phổ biến như:

* Xác thực email/mật khẩu: Người dùng có thể đăng ký và đăng nhập bằng cách cung cấp địa chỉ email và mật khẩu.
* Xác thực xã hội: Người dùng có thể đăng nhập và đăng ký thông qua các tài khoản xã hội như Google, Facebook, Twitter, GitHub và nhiều nền tảng khác.
* Xác thực số điện thoại: Người dùng có thể xác thực danh tính của họ bằng cách xác minh số điện thoại di động qua SMS.

Firebase Authentication cung cấp các tính năng quản lý người dùng như đặt lại mật khẩu, xác thực lại email, cung cấp thông tin người dùng (tên, ảnh đại diện, v.v.), phân quyền và kiểm soát truy cập.

Firebase Authentication tích hợp chặt chẽ với các dịch vụ khác của Firebase, cho phép nhà phát triển kết hợp xác thực người dùng với lưu trữ dữ liệu, phân tích, thông báo đẩy và các tính năng khác trong ứng dụng.

Sử dụng Firebase Authentication giúp nhà phát triển tiết kiệm thời gian và công sức trong việc xây dựng và quản lý hệ thống xác thực người dùng, đồng thời đảm bảo tính bảo mật và an toàn cho ứng dụng của họ.

## Giới thiệu về IFCAT

### IFCAT là gì?

IFCAT là một ứng dụng di động đầy thú vị và hữu ích dành cho những người yêu mèo và muốn tìm hiểu về các loài mèo khác nhau. Với MeowInfo, người dùng có thể khám phá và tìm hiểu về hàng trăm loài mèo khác nhau từ nhiều nguồn thông tin đáng tin cậy trên Internet. Ứng dụng không chỉ cung cấp thông tin chi tiết về mỗi loài mèo, mà còn cho phép người dùng xem hình ảnh đẹp và thú vị của chúng.

Một trong những tính năng nổi bật của IFCAT là khả năng tìm kiếm linh hoạt. Người dùng có thể tìm kiếm các loài mèo dựa trên các tiêu chí khác nhau như tên loài, màu sắc, kích thước, tính cách, và nhiều thông tin khác. Điều này giúp người dùng dễ dàng tìm ra thông tin về loài mèo mình quan tâm.

Ngoài ra, IFCAT cũng cung cấp chức năng lưu giữ thông tin yêu thích. Người dùng có thể lưu lại thông tin về các loài mèo mà họ yêu thích và truy cập nhanh chóng vào những thông tin này bất kỳ khi nào họ muốn. Dữ liệu yêu thích này được lưu trữ một cách an toàn trên cloud thông qua dịch vụ Firebase, đảm bảo rằng người dùng có thể truy cập thông tin của mình từ bất kỳ thiết bị nào.

Với giao diện đẹp mắt, dễ sử dụng và tính năng đa dạng, IFCAT sẽ là một người bạn đồng hành lý tưởng cho những người yêu mèo, giúp họ tìm hiểu và khám phá về thế giới đa dạng và đáng yêu của loài mèo.

### Khả năng tích hợp IFCAT vào ứng dụng di động

IFCAT có khả năng tích hợp hoàn hảo vào ứng dụng di động thông qua việc sử dụng các công nghệ và dịch vụ hiện đại. Dưới đây là một số điểm nổi bật về khả năng tích hợp của IFCAT:

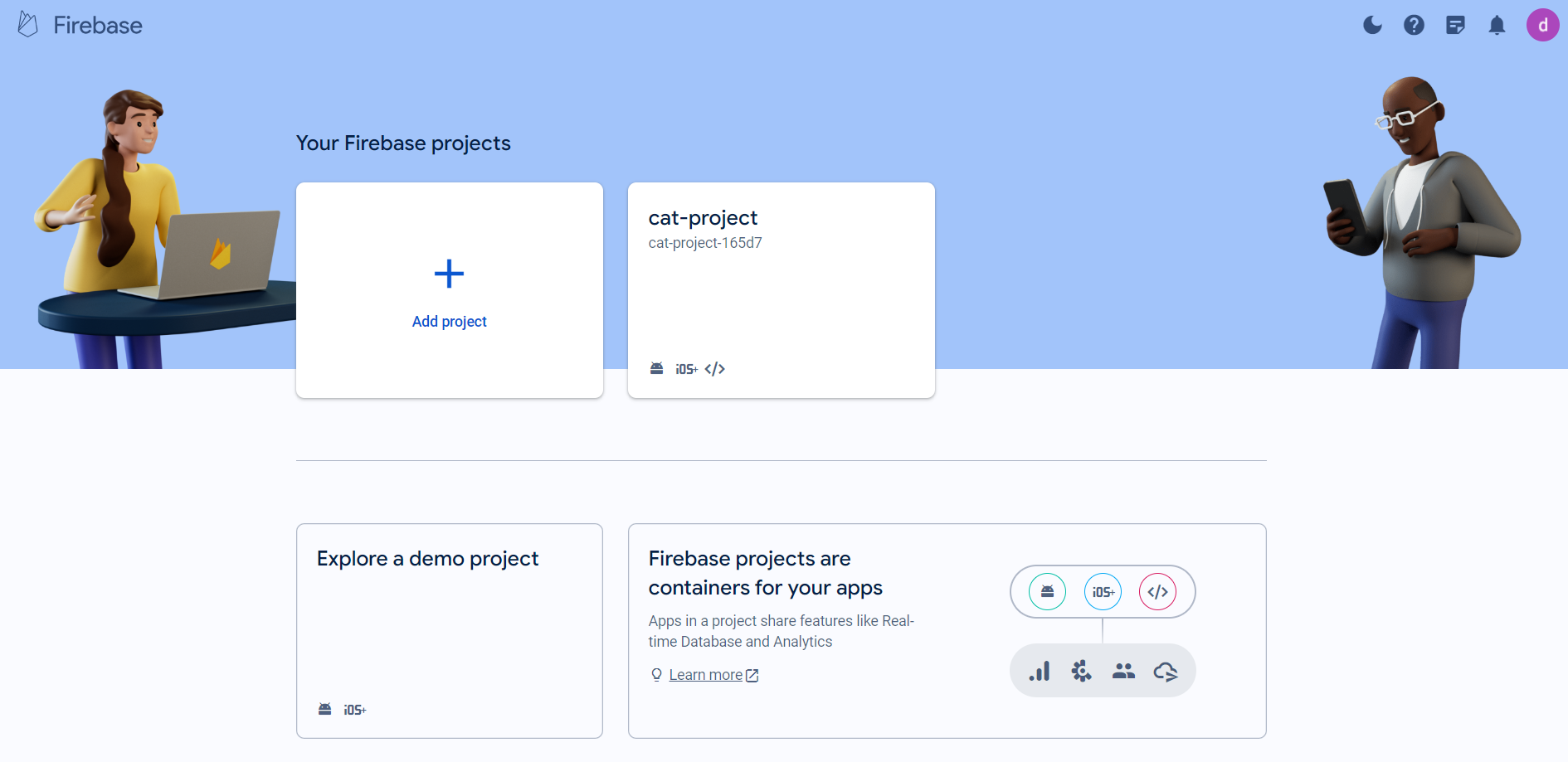
* Firebase Authentication: IFCAT sử dụng Firebase Authentication để xác thực người dùng. Điều này giúp bảo vệ thông tin cá nhân của người dùng và đảm bảo rằng chỉ những người được phép mới có thể truy cập vào các tính năng của ứng dụng.
* Firebase Realtime Database hoặc Firestore: Thông tin về các loài mèo và danh sách yêu thích của người dùng được lưu trữ trên Firebase Realtime Database hoặc Firestore. Điều này giúp dữ liệu được đồng bộ hóa trên nhiều thiết bị và người dùng có thể truy cập vào thông tin của mình bất kỳ lúc nào.
* API Integration: IFCAT tích hợp các API từ nhiều nguồn khác nhau để cung cấp thông tin phong phú về các loài mèo. Các API này được tích hợp một cách linh hoạt và hiệu quả, giúp ứng dụng cung cấp thông tin đáng tin cậy và đầy đủ.
* Search Functionality: Khả năng tìm kiếm trong IFCAT được xây dựng dựa trên dữ liệu từ các API đã tích hợp. Điều này đảm bảo rằng người dùng có thể tìm kiếm thông tin về các loài mèo một cách dễ dàng và nhanh chóng.
* Cloud Storage Integration (Firebase Storage): Nếu IFCAT cho phép người dùng tải lên hình ảnh mèo yêu thích của họ, ứng dụng có thể tích hợp với Firebase Storage để lưu trữ các hình ảnh này một cách an toàn và dễ dàng quản lý.

Những tính năng và khả năng tích hợp này giúp IFCAT trở thành một ứng dụng di động mạnh mẽ và linh hoạt, cung cấp cho người dùng trải nghiệm tốt nhất khi khám phá và tìm hiểu về thế giới đáng yêu của các loài mèo

# Xây dựng ứng dụng di động

## Cài đặt môi trường

### Cài đặt Firebase

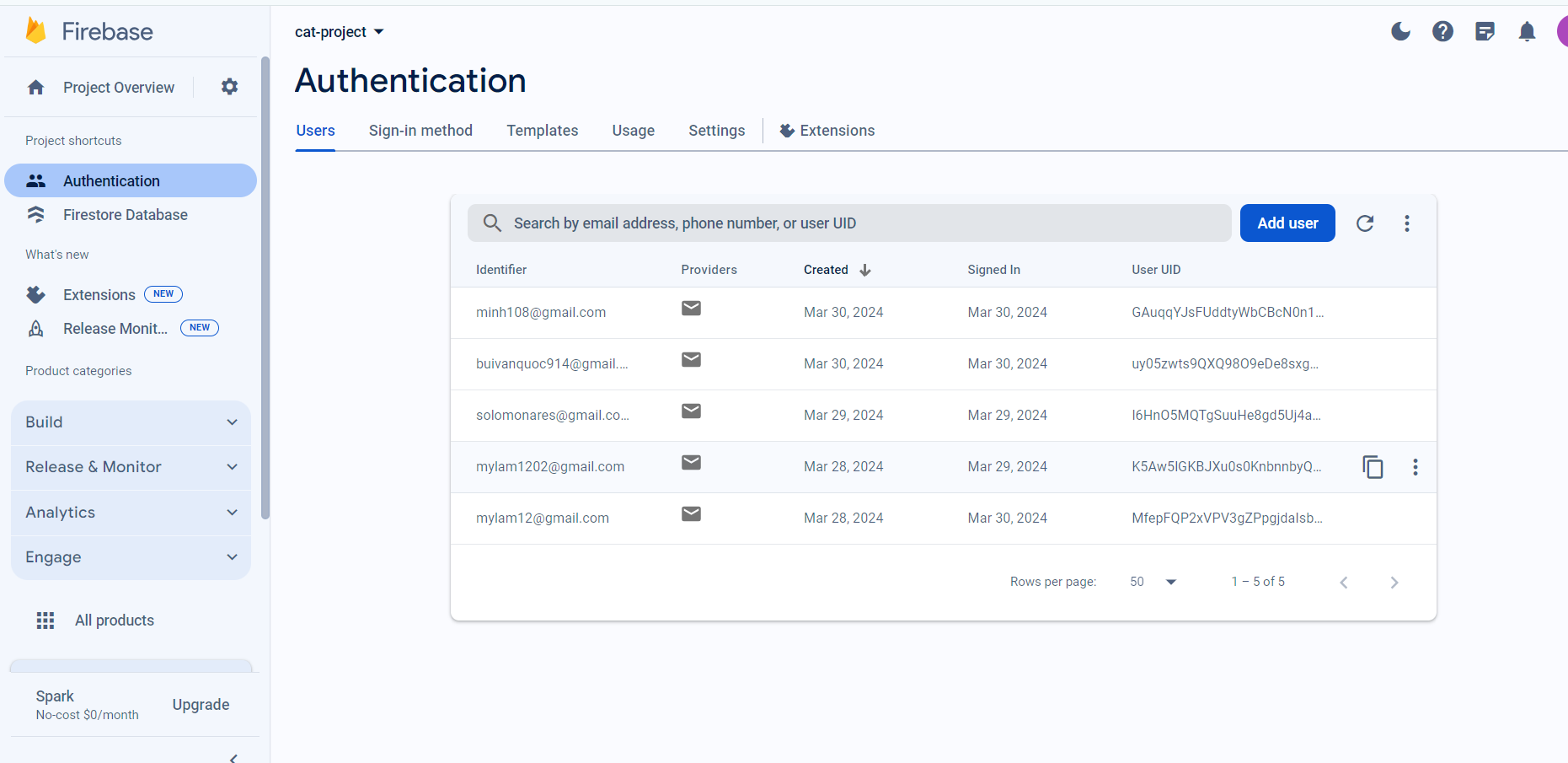


##### Hinh 1: Màn hình khởi tạo project trên Firebase

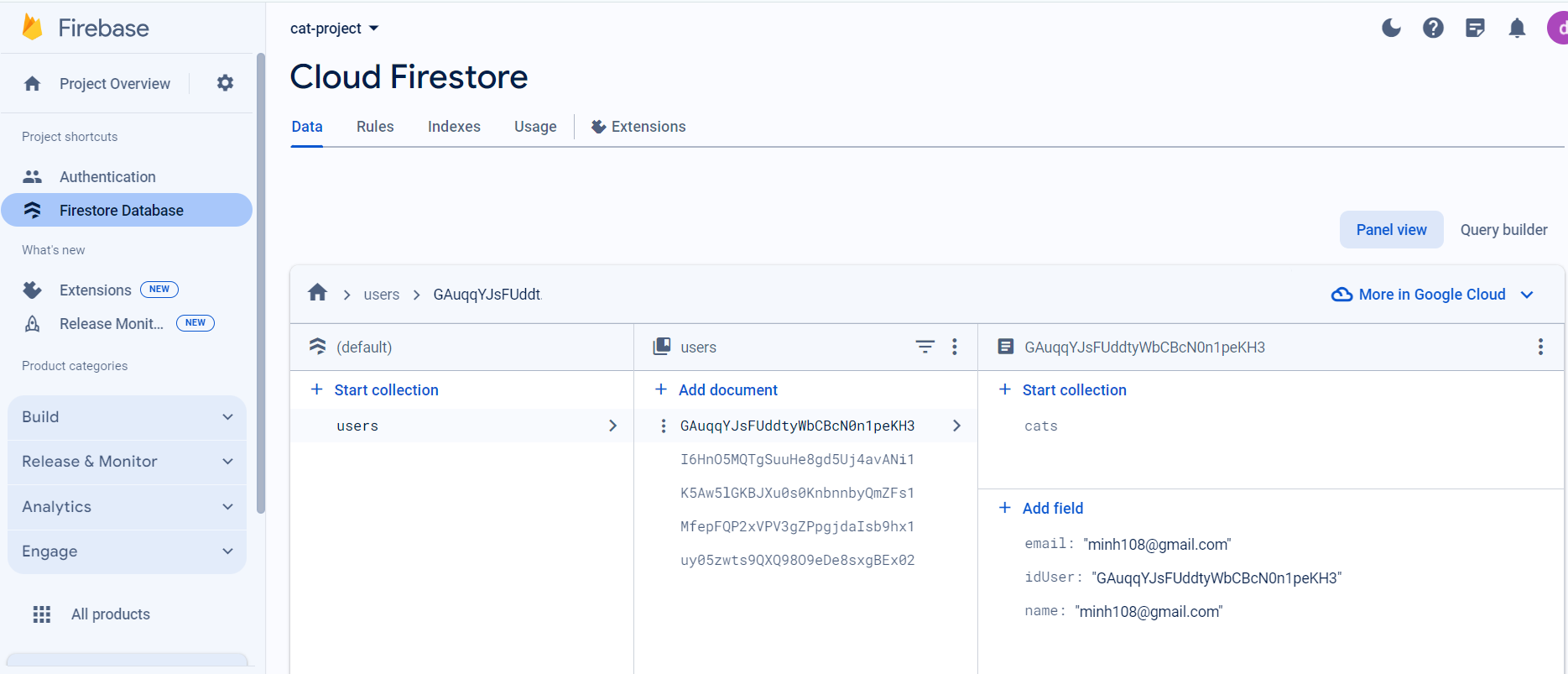
Sau khi tạo dự án thì thêm câu lệnh



để cài đặt firebase vào dự án



##### Hinh 2: Màn hình Authentication của Firebase



##### Hinh 3: Màn hình Cloud Firestore của Firebase

## Xây dựng ứng dụng

Trong ứng dụng này chúng tôi thiết kế có chức năng bao gồm:

* Đăng nhập và xác thực người dùng:
  + Cho phép người dùng đăng nhập bằng tài khoản Firebase Authentication.
  + Xác thực người dùng để đảm bảo chỉ người dùng đã đăng nhập mới có thể truy cập vào các chức năng khác của ứng dụng.
* Danh sách các loài mèo:
  + Hiển thị một danh sách các loài mèo với hình ảnh và một số thông tin cơ bản.
  + Cho phép người dùng nhấp vào một loài mèo để xem thông tin chi tiết về loài đó.
* Chi tiết từng loài mèo:
  + Hiển thị thông tin chi tiết về một loài mèo cụ thể, bao gồm hình ảnh và các mô tả về loài đó.
  + Cung cấp các tính năng như chia sẻ thông tin, thêm vào danh sách yêu thích, v.v.
* Tìm kiếm loài mèo:
  + Cung cấp chức năng tìm kiếm để người dùng có thể tìm kiếm loài mèo dựa trên một hoặc nhiều thông tin đầu vào như tên loài, màu sắc, kích thước, v.v.
* Lưu giữ thông tin yêu thích:
  + Cho phép người dùng lưu trữ thông tin về các loài mèo yêu thích của họ.
  + Dữ liệu yêu thích có thể được lưu trữ trên Firebase Realtime Database hoặc Cloud Firestore để đồng bộ hóa trên nhiều thiết bị.
* Quản lý thông tin yêu thích:
  + Cho phép người dùng xem danh sách các loài mèo họ đã thêm vào danh sách yêu thích.
  + Cung cấp chức năng để người dùng có thể xóa hoặc chỉnh sửa các mục trong danh sách yêu thích của họ.
* Đồng bộ hóa dữ liệu:
  + Đảm bảo rằng thông tin yêu thích của người dùng được đồng bộ hóa đúng cách giữa các thiết bị của họ, nếu họ sử dụng cùng một tài khoản đăng nhập trên nhiều thiết bị.
* Giao diện người dùng thân thiện:
  + Thiết kế giao diện người dùng dễ sử dụng và thân thiện với người dùng, bao gồm các nút và thanh điều hướng rõ ràng và dễ nhìn.

Bằng cách tích hợp các chức năng này vào ứng dụng của bạn, người dùng sẽ có trải nghiệm tốt khi sử dụng và tận hưởng các tính năng và thông tin về các loài mèo.

### Chức năng đăng nhập & đăng ký

Hai chức năng đăng nhập và đăng ký được chúng tôi thức hiện thông qua hai trang Sign In và Sign Up

Đầu tiên là việc xây dựng trang Sign In trong dự án Flutter. Trang này được sử dụng để hiển thị giao diện đăng nhập cho người dùng và xử lý việc xác thực sử dụng Firebase Authentication.

Trang Sign In có các mục đích chính sau:

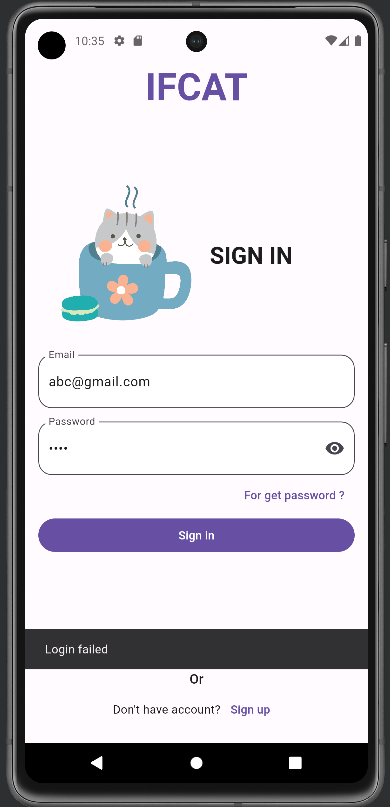
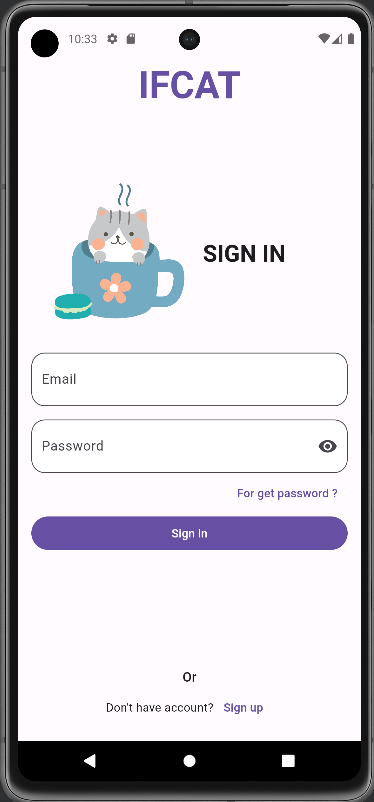
* Hiển thị giao diện đăng nhập, bao gồm các trường nhập email và password.
* Xử lý việc đăng nhập sử dụng Firebase Authentication.
* Điều hướng người dùng đến màn hình chính khi đăng nhập thành công.
* Cung cấp lựa chọn để người dùng chuyển đến màn hình sign\_up\_screen nếu chưa có tài khoản.

Trang Log In được xây dựng dựa trên StatefulWidget và có các thành phần chính như sau:

* GlobalKey<FormState>: Dùng để xác định và quản lý trạng thái của Form.
* TextEditingController: Dùng để quản lý và lấy giá trị từ các trường nhập liệu (email và password).
* String \_errorMessage: Biến lưu trữ thông báo lỗi (nếu có).
* Hàm \_signIn(BuildContext context): Xử lý việc đăng nhập sử dụng Firebase Authentication.
* Hàm \_goToSignUp(BuildContext context): Điều hướng người dùng đến màn hình sign\_up\_screen.

Giao diện của trang Log In gồm các thành phần sau:

* Tiêu đề "Sign In" và tên app “IFCAT”
* Hai trường nhập liệu: email và password.
* Nút "Sign In" để xác nhận đăng nhập.
* Hiển thị thông báo lỗi (nếu có).

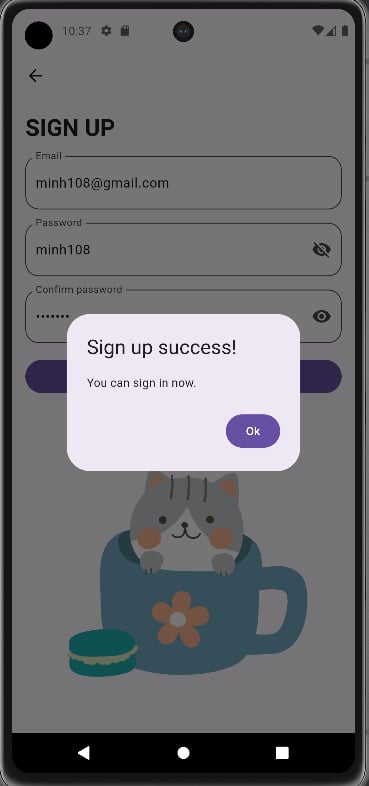
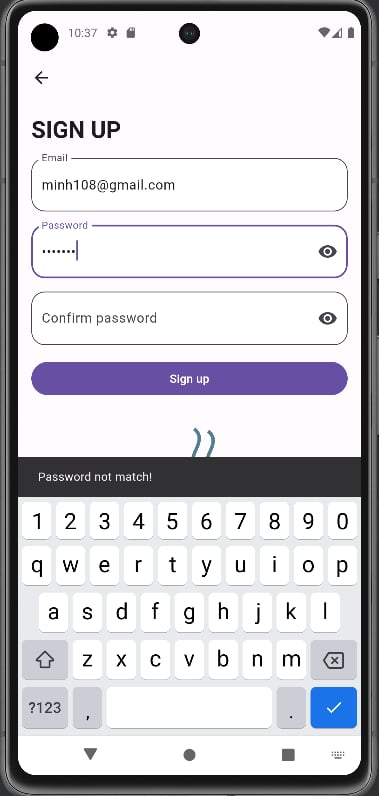
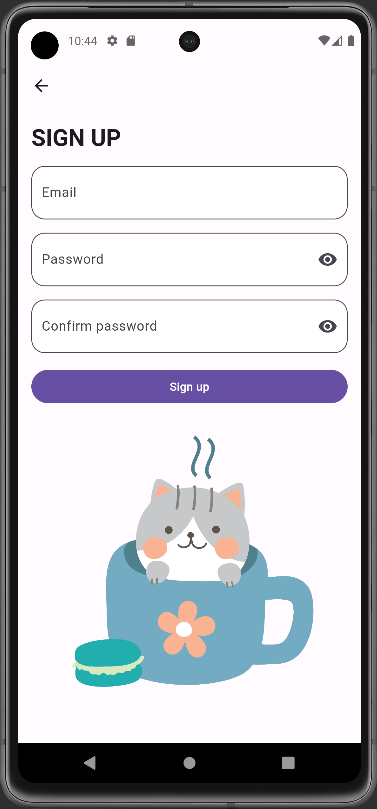


##### Hinh 4: Màn hình demo trang Log In

Tiếp theo là việc xây dựng trang Sign Up trong dự án Flutter. trang này được sử dụng để hiển thị giao diện đăng ký tài khoản mới cho người dùng và xử lý việc đăng ký sử dụng Firebase Authentication.

Trang Sign Up có các mục đích chính sau:

* Hiển thị giao diện đăng ký tài khoản, bao gồm các trường nhập email, password và confirm password.
* Xử lý việc đăng ký tài khoản sử dụng Firebase Authentication.
* Điều hướng người dùng đến màn hình chính khi đăng ký thành công.



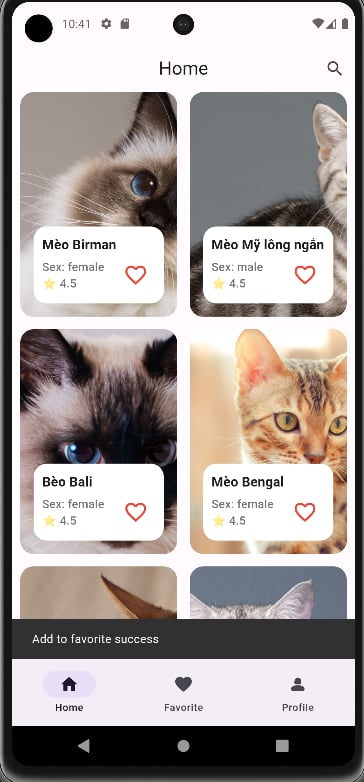
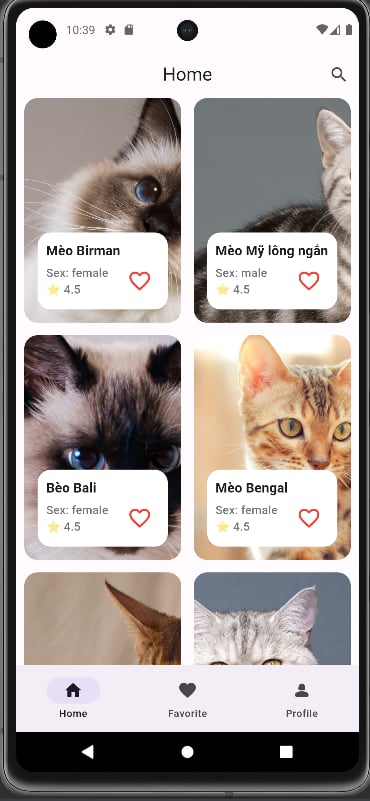
*Hình 5: Màn hình demo trang Sign Up*

Giao diện của trang sign\_up\_screen gồm các thành phần sau:

* Tiêu đề "Sign Up".
* Biểu tượng hình khóa.
* Ba trường nhập liệu: email, password và confirm password.
* Nút "Sign Up" để xác nhận đăng ký.
* Hiển thị thông báo lỗi (nếu có).

### Giao diện trang chủ.

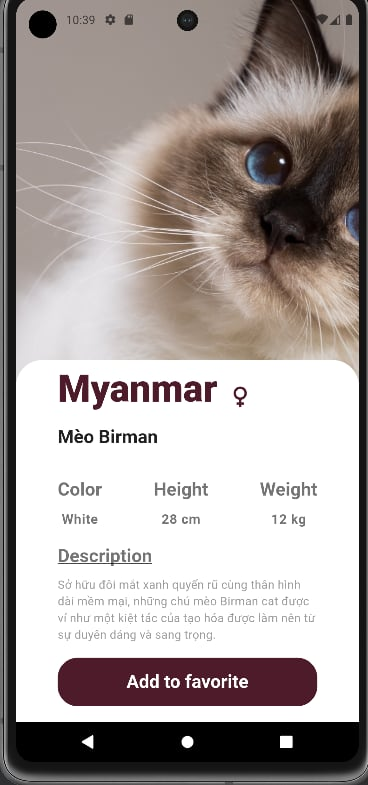
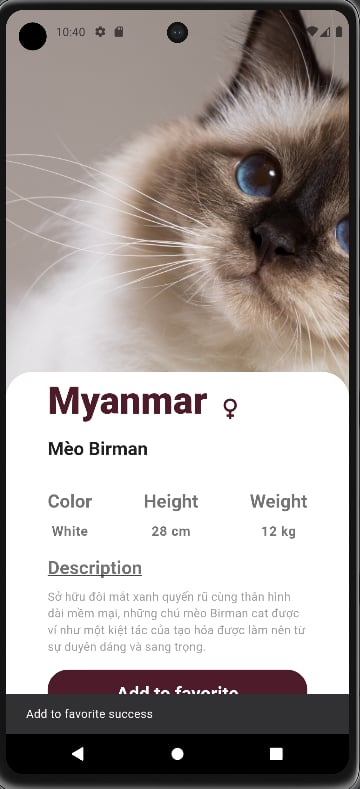
Sau khi đăng nhập người dùng sẽ thấy danh sách thông tin về các loài mèo bao gồm: hình ảnh, tên, đánh giá. Sau khi người dùng click vào icon trái tim sẽ thêm loại mèo vào mục Favorite và hiển thị trên màn hình “ Add to favorite success ”



##### Hình 6: Màn hình demo trang chủ

* + 1. ***Chức năng xem chi tiết.***

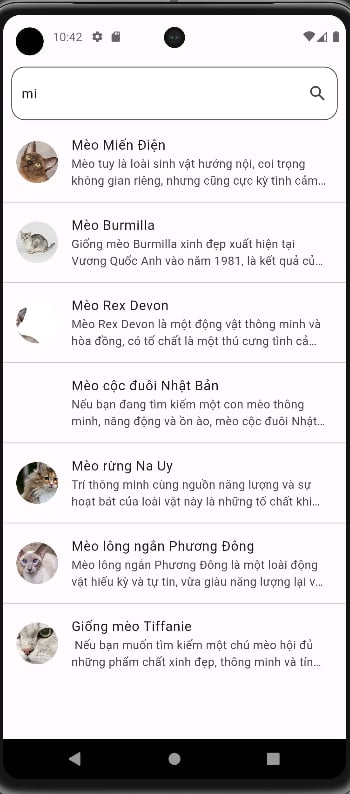
Khi người dùng click vào từng thẻ loại mèo sẽ hiện ra thêm chức năng xem thông tin chi tiết. Người dùng cũng có thể thêm loại mèo vào danh sách yêu thích thông qua mục “ Add to favorite ”.

##### Hình 7: Màn hình demo xem chi tiết

### Giao diện tìm kiếm.

Người dùng có thể tìm kiếm danh sách mèo thông qua tên hay các ký tự trong phần description.



##### Hình 8: Màn hình demo tìm kiếm

**KẾT LUẬN**

IFCAT là một ứng dụng di động đầy hứa hẹn và hữu ích cho những người yêu mèo. Với khả năng tích hợp vào ứng dụng di động thông qua các công nghệ và dịch vụ tiên tiến như Firebase Authentication, Firebase Realtime Database hoặc Firestore, API Integration và Cloud Storage Integration, IFCAT không chỉ cung cấp thông tin đáng tin cậy và chi tiết về các loài mèo mà còn đảm bảo tính bảo mật và tính linh hoạt trong việc quản lý dữ liệu của người dùng.

Với giao diện thân thiện, dễ sử dụng và tính năng đa dạng như tìm kiếm linh hoạt và lưu trữ thông tin yêu thích trên cloud hoặc local, IFCAT hứa hẹn mang lại trải nghiệm tuyệt vời cho những người yêu mèo, giúp họ khám phá và tìm hiểu về thế giới đáng yêu của các loài mèo một cách dễ dàng và thú vị.

### Ưu điểm:

* Đa dạng thông tin: IFCAT cung cấp thông tin phong phú về các loài mèo từ nhiều nguồn khác nhau, giúp người dùng hiểu rõ hơn về đa dạng của thế giới mèo.
* Tích hợp Firebase: Sử dụng Firebase Authentication, Realtime Database hoặc Firestore và Firebase Storage giúp ứng dụng có tính bảo mật cao và dữ liệu được đồng bộ hóa một cách linh hoạt.
* Giao diện thân thiện: Giao diện đẹp mắt và dễ sử dụng giúp người dùng dễ dàng tương tác với ứng dụng.
* Tính năng tìm kiếm linh hoạt: Khả năng tìm kiếm dựa trên nhiều tiêu chí giúp người dùng dễ dàng tìm kiếm thông tin về các loài mèo theo nhu cầu của mình.

### Nhược điểm:

* Phụ thuộc vào nguồn dữ liệu bên ngoài: Nếu các nguồn dữ liệu bên ngoài gặp vấn đề, có thể ứng dụng sẽ gặp khó khăn trong việc cập nhật thông tin mới nhất về các loài mèo.
* Cần cải thiện tính tương tác: Mặc dù giao diện dễ sử dụng, nhưng có thể cần thêm tính năng tương tác như đánh giá, bình luận từ người dùng để tăng tính thú vị và sự tương tác của ứng dụng.

### Hướng phát triển:

* Nâng cao tính tương tác: Thêm tính năng cho phép người dùng đánh giá, bình luận và chia sẻ thông tin về các loài mèo để tạo ra cộng đồng mèo đa dạng và sôi động.
* Mở rộng nguồn dữ liệu: Thích hợp thêm nguồn dữ liệu mới và đa dạng để cung cấp thông tin phong phú và đáng tin cậy hơn về các loài mèo.
* Phát triển tính năng xã hội: Xây dựng các tính năng mạng xã hội như nhóm, tin nhắn, hoặc diễn đàn để tạo ra một cộng đồng mèo trực tuyến sôi động.
* Tối ưu hóa hiệu suất và tương thích: Liên tục cải thiện hiệu suất và tương thích của ứng dụng trên các nền tảng di động khác nhau để đảm bảo trải nghiệm người dùng tốt nhất.

Với các phát triển và cải thiện này, IFCAT sẽ tiếp tục phát triển và trở thành một ứng dụng di động đa dạng và hữu ích cho cộng đồng yêu mèo

DANH MỤC TÀI LIỆU THAM KHẢO

1. https://mockapi.io/
2. https://viblo.asia/p/lien-ket-firebase-den-app-flutter-cho-android-va-ios-RnB5p6zdZPG