

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA KHOA HỌC VÀ KỸ THUẬT THÔNG TIN



**BÁO CÁO ĐỒ ÁN MÔN HỌC
CÔNG NGHỆ LẬP TRÌNH ĐA NỀN TẢNG
CHO ỨNG DỤNG DI ĐỘNG**

Đề tài:

Ứng dụng đọc báo online

News App

GVHD: ThS. Võ Ngọc Tân

Sinh viên thực hiện:

- | | |
|---------------------|----------------|
| 1. Lê Hoàng Huy | MSSV: 20521392 |
| 2. Nguyễn Huy Hoàng | MSSV: 20521343 |

Thành phố Hồ Chí Minh, tháng 1 năm 2024

[illegible]

(Ký tên và ghi rõ họ tên)

BẢNG PHÂN CÔNG, ĐÁNH GIÁ THÀNH VIÊN*Bảng phân công công việc, đánh giá thành viên*

Họ và tên	MSSV	Công việc thực hiện	Đánh giá
Lê Hoàng Huy	20521392	<ul style="list-style-type: none"> - Viết source code xây dựng API bài báo và thực hiện sử dụng Firebase làm Back-end cho ứng dụng tin tức - Hiệu chỉnh các chi tiết các màn ứng dụng và hoàn thiện ứng dụng - Viết Word bài thu hoạch báo cáo kết quả thực hiện đồ án - Thực hiện file PowerPoint thuyết trình - Thuyết trình giới thiệu ứng dụng và trình bày cấu trúc hệ thống, các công nghệ đã áp dụng trong ứng dụng - Xây dựng video demo ứng dụng 	Hoàn thành tốt
Nguyễn Huy Hoàng	20521343	<ul style="list-style-type: none"> - Viết source code xây dựng Fronts-end của ứng dụng , tìm hiểu cách sử dụng Firebase và hướng thực hiện Back-end cho ứng dụng tin tức - Định hướng nội dung, phác hoạ dàn bài - Hiệu chỉnh và bổ sung, hoàn thiện file Word và PowerPoint báo cáo đồ án - Thiết kế và hoàn thiện các chức năng , bố cục của ứng dụng - Thuyết trình demo sản phẩm ứng dụng 	Hoàn thành tốt

LỜI CẢM ƠN

Lời đầu tiên, chúng em xin gửi lời cảm ơn chân thành đến Trường Đại học Công nghệ Thông tin – ĐHQG TP. Hồ Chí Minh đã tạo cơ hội cho chúng em được học tập tại một ngôi trường có cơ sở vật chất hàng đầu, với chất lượng giảng dạy vô cùng chuyên nghiệp và chiều sâu kiến thức chuyên ngành, và quan trọng nhất là trên tinh thần giảng dạy kiến thức thực tiễn cho chúng em.

Chúng em gửi lời cảm ơn đặc biệt đến thầy ThS. Võ Ngọc Tân là giảng viên hướng dẫn đề tài chuyên ngành của chúng em trong quá trình thực hiện đồ án bộ môn học Công nghệ lập trình đa nền tảng cho ứng dụng di động. Nhờ sự giảng dạy nhiệt tình, nhẫn nại dành thời gian hướng dẫn, chỉ bảo, giúp đỡ và trả lời những thắc mắc trong suốt quá trình học tập là điều kiện quý báu nhất để chúng em có thể đạt được kết quả thực hiện đồ án tốt nhất.

Tuy nhiên, vì còn nhiều hạn chế về quỹ thời gian và kinh nghiệm thực hành nên kết quả báo cáo đồ án này sẽ không thể tránh được những thiếu sót. Chúng em rất mong nhận được sự cảm thông, nhận xét đóng góp từ các Quý Thầy Cô để chúng em có điều kiện bổ sung, điều chỉnh và nâng cao kiến thức chuyên môn để phục vụ tốt hơn cho công tác thực tế sau này.

Chúng em xin chân thành cảm ơn!

Sau đây, nhóm chúng em sẽ trình bày tổng quan báo cáo kết quả thực hiện đồ án kết thúc môn học Công nghệ lập trình đa nền tảng cho ứng dụng di động với đề tài Xây dựng ứng dụng di đọc báo online News App khi điểu qua các phần trong bố cục các chương nội dung sau:

- **Chương 1:** Giới thiệu và minh họa ứng dụng
- **Chương 2:** Các công nghệ đã áp dụng
- **Chương 3:** Đánh giá và kết luận

MỤC LỤC

Chương 1 GIỚI THIỆU VÀ MINH HOẠ ỨNG DỤNG ĐỌC BÁO ONLINE NEWS APP	5
1. Giới thiệu ứng dụng.....	5
2. Sitemap và workflow	6
3. Mô tả tương tác từng trang chức năng.....	7
Chương 2 CÁC CÔNG NGHỆ ÁP DỤNG	12
1. React Native	12
1.1. Tổng quan về React Native.....	12
1.2. Các nền tảng và thư viện sử dụng trong đồ án.....	13
a. BACK END	13
b. FRONT END	15
2. Firebase.....	19
2.1. Tổng quan	19
2.2. Công cụ và dịch vụ chính.....	20
2.3. Cách đăng ký và cài đặt Firebase.....	22
3. Postman	29
3.1. Tổng quan	29
3.2. Cách sử dụng.....	30
Chương 3 ĐÁNH GIÁ VÀ KẾT LUẬN	32
1. Những điều làm được và chưa được	32
2. Hướng phát triển và mở rộng của đồ án	33
3. Kết luận.....	33
DANH MỤC NGUỒN THAM KHẢO	34

Chương 1 GIỚI THIỆU VÀ MINH HOẠ ỨNG DỤNG ĐỌC BÁO ONLINE NEWS APP

1. Giới thiệu ứng dụng

News App là một ứng dụng không thể thiếu cho những người yêu thích cập nhật tin tức hàng ngày. Với một giao diện thân thiện và dễ sử dụng, News App mang đến cho người dùng trải nghiệm đọc báo trực tuyến tuyệt vời. Bằng cách kết hợp tin tức từ các nguồn uy tín trên toàn thế giới, từ báo chí chính thống đến các trang tin tức độc lập, ứng dụng này cung cấp một cách tiếp cận toàn diện đối với thông tin. Đồng thời, tính năng cá nhân hóa giúp người dùng tùy chỉnh trải nghiệm đọc báo của họ, từ việc theo dõi các chủ đề yêu thích đến lưu trữ và chia sẻ các bài viết.

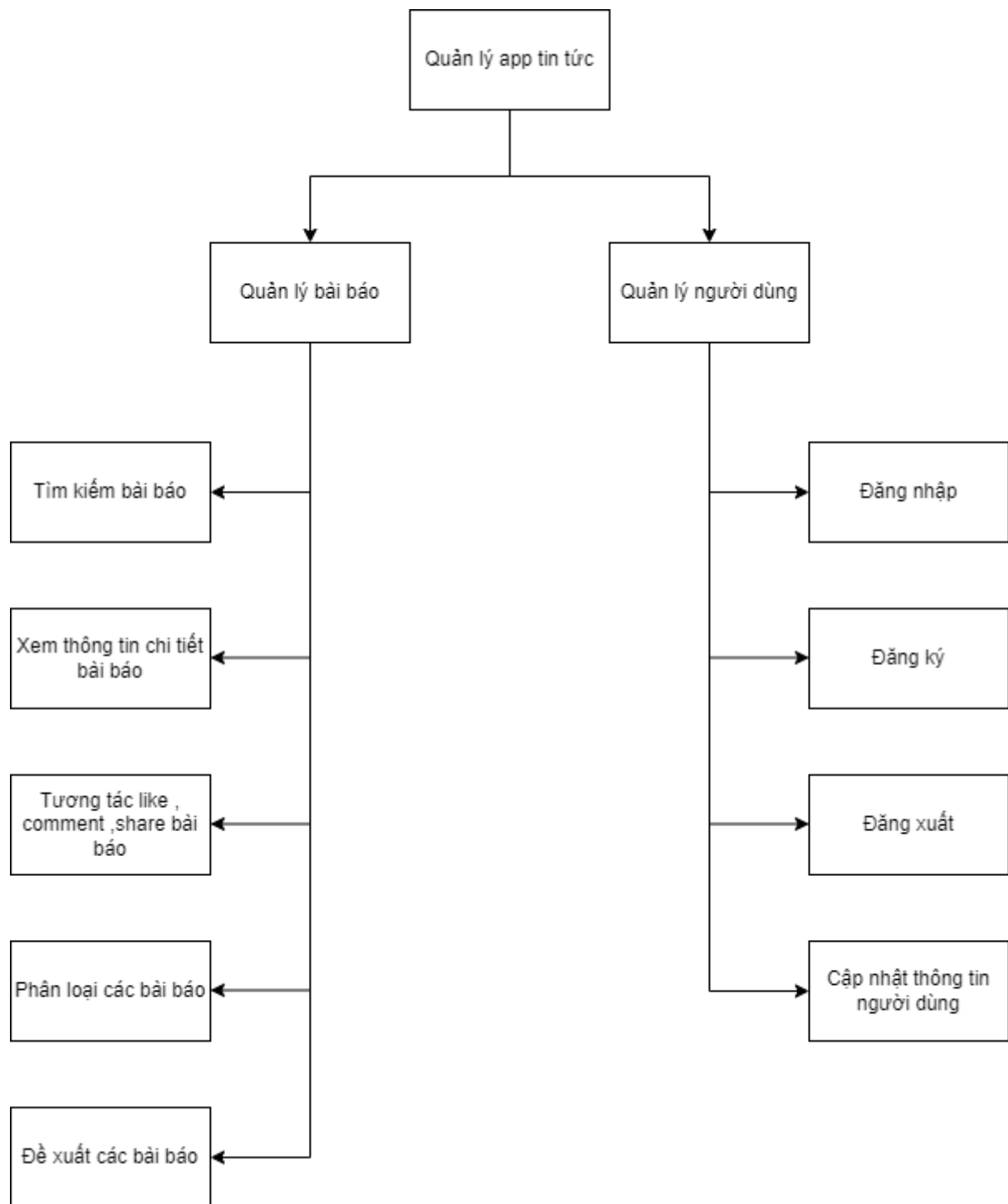
Đồng thời News App cung cấp cho người dùng trải nghiệm đọc tin tức đa dạng và tiện lợi với hàng ngàn nguồn tin tức từ nhiều nguồn khác nhau, News App cung cấp thông tin nhanh chóng về các sự kiện mới nhất từ khắp nơi trên thế giới. Ứng dụng này cũng cho phép người dùng cá nhân hóa trải nghiệm đọc báo bằng cách chọn các chủ đề yêu thích, lưu trữ các bài báo để đọc offline và nhận thông báo về các tin tức quan trọng. Điều này giúp người dùng tiếp cận thông tin theo cách thuận tiện nhất cho họ.

Với giao diện của News App được thiết kế đơn giản và hiện đại, tạo điều kiện thuận lợi cho người dùng để dễ dàng tìm kiếm và đọc các bài báo một cách nhanh chóng. Tính năng tối ưu hóa cho các thiết bị di động giúp người dùng trải nghiệm mượt mà mọi lúc, mọi nơi. Đồng thời, khả năng tùy chỉnh giao diện giúp người dùng có thể điều chỉnh cỡ chữ, chế độ đọc ban đêm và các tùy chọn cá nhân hóa khác để đáp ứng nhu cầu riêng của họ. Tất cả những tính năng này kết hợp với nhau tạo nên một trải nghiệm đọc báo trực tuyến thoải mái và tiện lợi.

Các đối tượng mà ứng dụng hướng đến quản lý bao gồm:

- Danh sách quản lý người dùng: Admin và Khách hàng:
 - + Danh mục màn hình user : đăng nhập , đăng ký
 - + Chức năng chi tiết : cập nhật thông tin người dùng , đăng xuất
- Danh sách quản lý bài báo:
 - + Danh mục màn hình phân loại các bài báo: tin tức xu hướng, tin tức giải trí, tin tức khoa học, tin tức sức khỏe , tin tức kinh doanh , tin tức thể thao...
 - + Thông tin chi tiết các bài báo, tìm kiếm bài báo , tương tác và đề xuất...

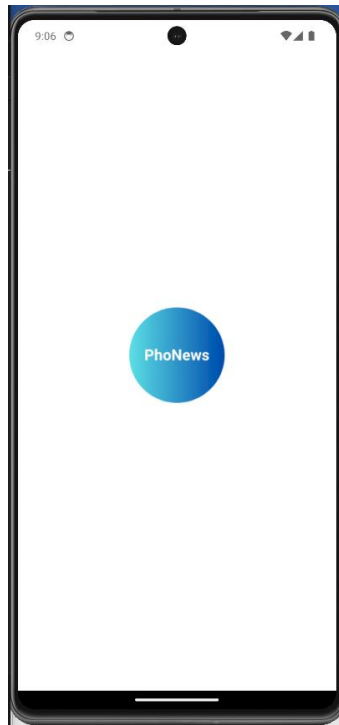
2. Sitemap và workflow



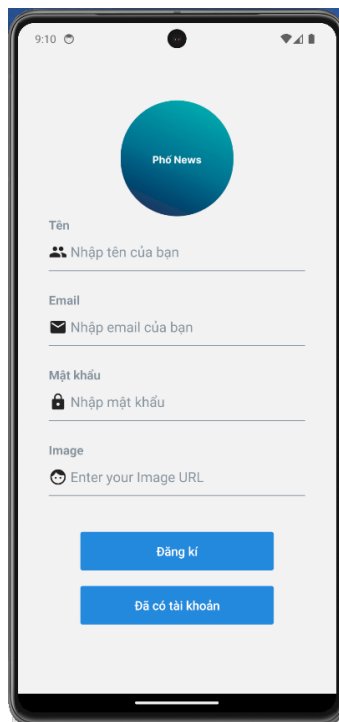
3. Mô tả tương tác từng trang chức năng

Các trang tương tác dành cho người dùng

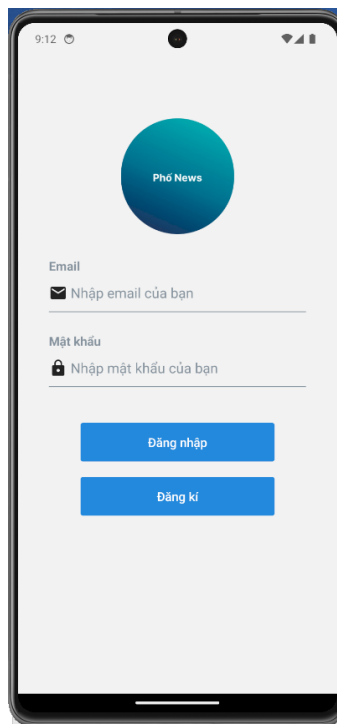
a. Màn hình giới thiệu ứng dụng (Splash + Introduction)



b. Trang đăng ký tài khoản



c. Trang đăng nhập tài khoản

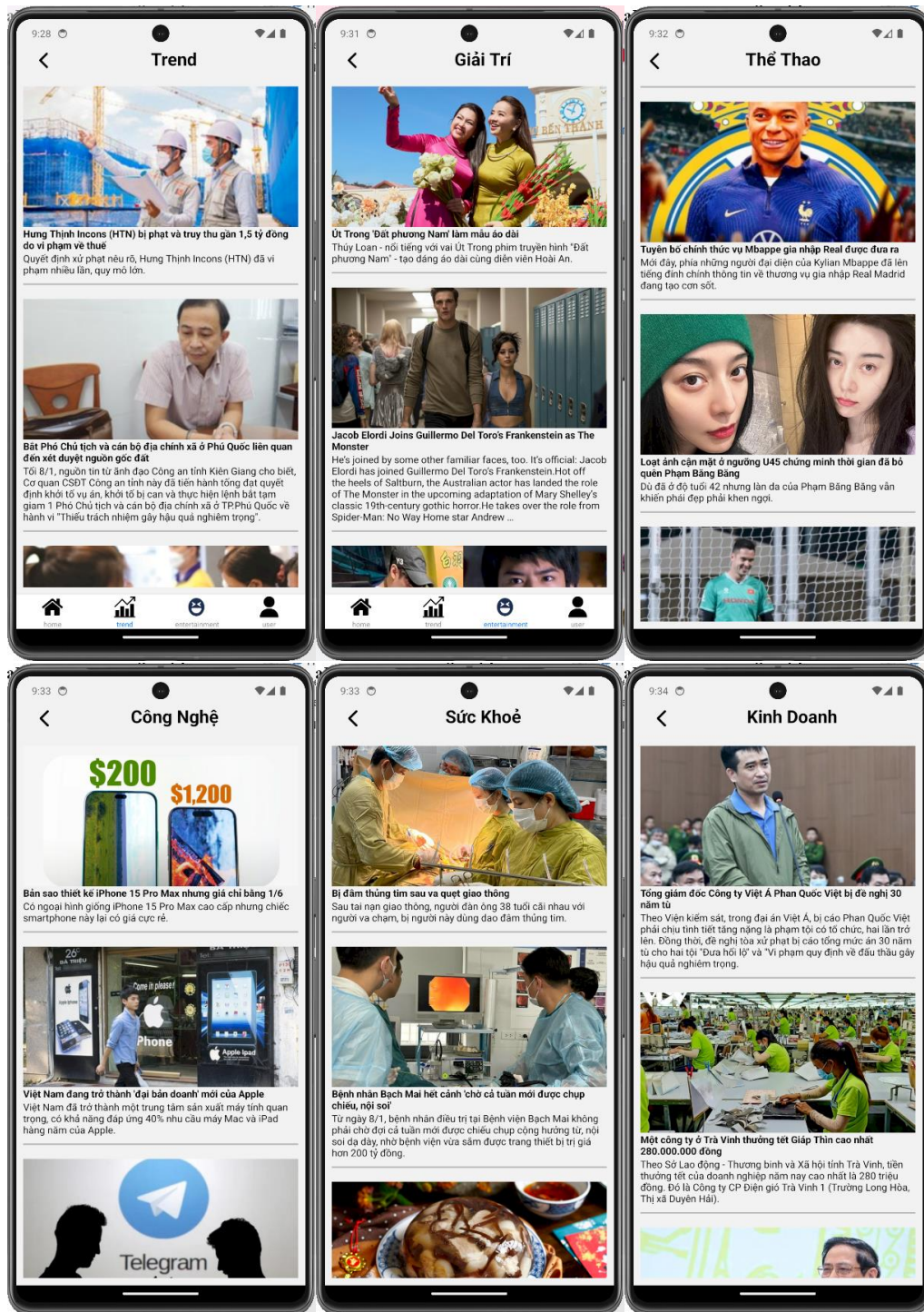


d. Trang chủ



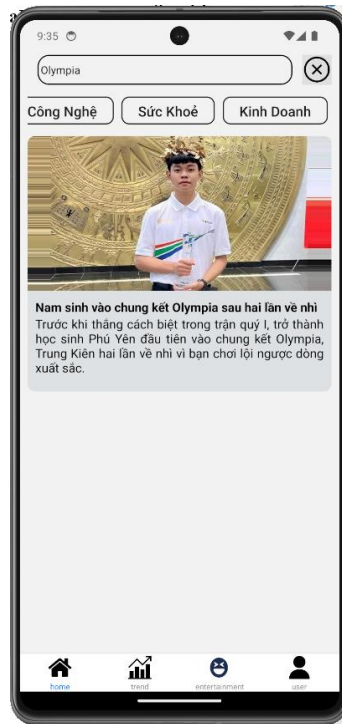


f. Trang danh sách theo từng loại bài báo

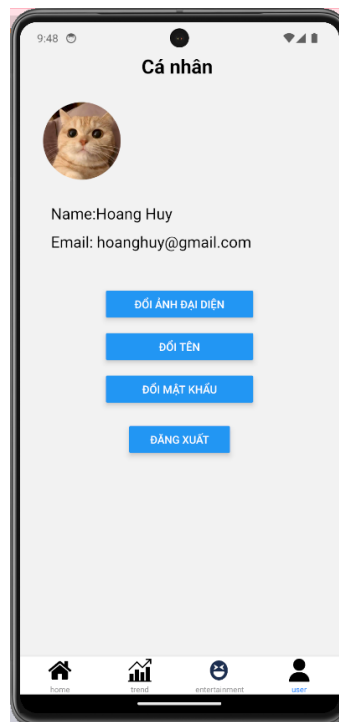


IE307 – Cross-Platform Programming

g. Trang tìm kiếm sản phẩm



h. Trang thông tin tài khoản



Chương 2 CÁC CÔNG NGHỆ ÁP DỤNG

1. React Native

1.1. Tổng quan về React Native

React Native là một framework được tạo bởi Facebook, cho phép các lập trình viên xây dựng các ứng dụng di động chạy trên cả hệ điều hành Android và iOS, còn gọi là lập trình đa nền tảng bằng việc sử dụng JavaScript và React, thư viện phổ biến cho người dùng là Javascript.

Lịch sử phát triển của React Native:

- Tháng 1 năm 2015: Facebook công bố React Native thông qua một bài viết trên blog công ty. Họ giới thiệu React Native là một cách để xây dựng ứng dụng di động sử dụng React và JavaScript.

- Tháng 3 năm 2015: Facebook phát hành phiên bản đầu tiên của React Native cho iOS, cho phép nhà phát triển xây dựng ứng dụng di động iOS bằng React Native.

- Tháng 9 năm 2015: Facebook phát hành phiên bản React Native cho Android, mở rộng khả năng phát triển ứng dụng đa nền tảng.

- Tháng 4 năm 2016: Facebook công bố việc chuyển React Native vào mã nguồn mở hoàn toàn, cho phép cộng đồng tham gia phát triển và cải tiến framework.

- Năm 2017: React Native trở nên phổ biến và được sử dụng bởi nhiều công ty và dự án lớn, bao gồm Airbnb, Instagram, và nhiều ứng dụng khác.

- Năm 2018: Microsoft công bố dự án "React Native for Windows" để hỗ trợ phát triển ứng dụng Windows 10 bằng React Native.

- Năm 2019: Facebook công bố phiên bản alpha của React Native Fabric, một phiên bản cải tiến của React Native với hiệu suất tốt hơn.

- Năm 2020: Microsoft thông báo hợp tác với Facebook để hỗ trợ React Native trên nền tảng Windows và công bố "React Native for macOS" để phát triển ứng dụng máy tính Mac.

- Năm 2021 - nay: Cộng đồng React Native tiếp tục phát triển và đóng góp cho framework, với nhiều phiên bản và cải tiến liên tục được công bố.

React Native đã trải qua nhiều sự phát triển và cải tiến trong suốt quá trình lịch sử của nó, và nó vẫn là một lựa chọn phổ biến cho việc phát triển ứng dụng di động đa nền tảng vào thời điểm hiện tại.

1.2. Các nền tảng và thư viện sử dụng trong đồ án

a. BACK END

a.1. Backend platform: Node.js

Node.js là một nền tảng phát triển dựa trên JavaScript runtime có thể chạy trên máy chủ, được xây dựng trên mã nguồn mở Chrome V8 JavaScript engine, và thường được sử dụng cho các ứng dụng web, APIs, và các loại ứng dụng server-side khác.

Một số đặc điểm và tính năng nổi bật của Node.js:

- *JavaScript Everywhere*: Node.js cho phép sử dụng JavaScript để phát triển cả phía máy chủ và phía người dùng, giúp thống nhất mã nguồn và kiến trúc ứng dụng.
- *Asynchronous and Event-Driven*: Node.js sử dụng mô hình xử lý không đồng bộ (asynchronous) và sự kiện (event-driven) để xử lý nhiều yêu cầu mà không gây blocking. Điều này làm cho Node.js phù hợp cho các ứng dụng có thể mở rộng tốt.
- *Single-Threaded, Non-Blocking I/O Model*: Node.js là mô hình singlethreaded, nhưng có khả năng xử lý hàng nghìn kết nối cùng một lúc bằng cách sử dụng event loop và không gian làm việc non-blocking I/O.
- *Nền tảng mã nguồn mở (Open source)*: Node.js là một dự án mã nguồn mở, có nghĩa là mã nguồn của nó có thể được xem và sửa đổi theo nhu cầu, và cộng đồng phát triển nó rộng lớn.
- *Module Ecosystem (npm)*: Node.js sử dụng npm (Node Package Manager) để quản lý các thư viện và module, giúp dễ dàng chia sẻ và tái sử dụng mã nguồn.
- *Cross-Platform*: Node.js có sẵn trên nhiều hệ điều hành khác nhau như Windows, macOS, và Linux, giúp phát triển và triển khai trên nhiều môi trường.
- *Real-Time Applications*: Node.js được sử dụng rộng rãi cho việc phát triển ứng dụng thời gian thực như ứng dụng trò chơi, chat, và streaming.
- *Active Community*: Node.js có một cộng đồng lớn và tích cực, cung cấp nguồn lực, thư viện, và hỗ trợ liên tục cho các nhà phát triển.
- *Scalability*: Nhờ vào mô hình xử lý không đồng bộ và khả năng xử lý đa luồng thông qua clustering, Node.js có thể mở rộng hiệu quả để đáp ứng yêu cầu tăng lên.

a.2. Backend dependencies**a.2.1. Babel**

Babel là một công cụ biên dịch (transpiler) trong môi trường JavaScript, được sử dụng để chuyển đổi mã nguồn JavaScript từ phiên bản mới (hoặc đang phát triển) về phiên bản cũ hơn có thể chạy trên các trình duyệt hoặc môi trường thực thi không hỗ trợ các tính năng mới.

Trong môi trường Node.js và phát triển ứng dụng web JavaScript hiện đại, Babel giúp đảm bảo rằng mã nguồn JavaScript của bạn có thể chạy trên nhiều trình duyệt và môi trường thực thi khác nhau mà không cần lo lắng về sự không tương thích của các tính năng JavaScript mới.

Babel cho phép bạn sử dụng các tính năng mới nhất của JavaScript như arrow functions, async/await, class syntax, và nhiều tính năng khác mà không cần phải lo lắng về khả năng tương thích với trình duyệt hoặc môi trường thực thi cụ thể. Babel cũng hỗ trợ các plugin và preset để tùy chỉnh quá trình biên dịch theo nhu cầu cụ thể của dự án.

a.2.2. Expo

Expo là một nền tảng phát triển ứng dụng di động mã nguồn mở, được sử dụng để xây dựng ứng dụng cho cả iOS và Android. Expo cung cấp một loạt các tính năng và công cụ giúp việc phát triển ứng dụng di động trở nên dễ dàng và nhanh chóng hơn, bao gồm:

- Trình giả lập và trình mô phỏng tích hợp sẵn, cho phép phát triển và kiểm tra ứng dụng trên cả hai nền tảng mà không cần phải cài đặt thiết bị vật lý.

- Hỗ trợ cho các công nghệ mới nhất, bao gồm React Native, ReactJS, và TypeScript.

- Một thư viện khổng lồ các thành phần và plugin, có thể được sử dụng để xây dựng ứng dụng nhanh chóng và hiệu quả.

Một số điểm quan trọng về Expo:

- Mã nguồn mở: Expo là mã nguồn mở, nghĩa là có thể tùy chỉnh và sửa đổi nó tùy theo nhu cầu của người sử dụng.

- Trình giả lập và trình mô phỏng: Trình giả lập và trình mô phỏng tích hợp sẵn của Expo giúp việc phát triển và kiểm tra ứng dụng trở nên dễ dàng và nhanh chóng hơn.

- Hỗ trợ cho các công nghệ mới nhất: Expo hỗ trợ cho các công nghệ mới nhất, bao gồm React Native, ReactJS, và TypeScript. Điều này cho phép phát triển ứng dụng di động hiện đại và hiệu quả.

- Một thư viện khổng lồ các thành phần và plugin: Thư viện thành phần và plugin khổng lồ của Expo cung cấp một nền tảng vững chắc để xây dựng ứng dụng di động.

Tóm lại, Expo là một lựa chọn phổ biến cho việc phát triển ứng dụng di động vì nó cung cấp một loạt các tính năng và công cụ giúp việc phát triển ứng dụng di động trở nên dễ dàng và nhanh chóng hơn.

b. FRONT END

b.1. Navigation

React Native Navigation là một thư viện khác cung cấp các cách tiếp cận điều hướng trong React Native. Nó thường được sử dụng khi cần tối ưu hiệu suất cho navigation trong ứng dụng lớn.

Cách cài đặt các gói dependencies cho Navigation:

```
npm install @react-navigation/native
npx expo install react-native-screens react-native-safe-area-context
npm install @react-navigation/native-stack npm install @react-
navigation/bottom-tabs
```

b.2. Axios

Trong React Native, Axios là một thư viện được sử dụng để thực hiện các yêu cầu HTTP (như GET, POST, PUT, DELETE) từ ứng dụng tới một server hoặc một API khác. Axios cung cấp một cách tiện lợi và linh hoạt để tạo và xử lý các yêu cầu HTTP. Nó hoạt động cả trên trình duyệt web và trong môi trường React Native.

Cài đặt: npm

```
install axios
```

b.3. Async Storage

Async Storage là một cơ chế lưu trữ dữ liệu đơn giản và bền vững trong React Native. Nó giúp lưu trữ dữ liệu local trên thiết bị di động, cho phép ứng dụng lưu trữ thông tin như cài đặt người dùng, token xác thực, thông tin cache, và nhiều loại dữ liệu khác một cách dễ dàng.

Async Storage cung cấp một giao diện lập trình cơ bản gồm các phương thức như `setItem`, `getItem`, `removeItem`, và `clear` để thực hiện các hoạt động cơ bản như lưu trữ, truy xuất, xóa và xóa sạch dữ liệu.

Cài đặt: `npm install @react-native-async-storage/async-storage`

b.4. Elements

React Native Elements là một thư viện UI mã nguồn mở cho React Native, cung cấp một bộ các thành phần giao diện người dùng được thiết kế sẵn. Thư viện này bao gồm hơn 30 thành phần, bao gồm các thành phần phổ biến như `Button`, `TextInput`, `Card`, và `Image`.

Các thành phần của React Native Elements được thiết kế để sử dụng dễ dàng và hiệu quả. Các thành phần này có giao diện và hành vi được tiêu chuẩn hóa, giúp bạn dễ dàng tạo giao diện người dùng nhất quán cho ứng dụng của mình.

Một trong những thành phần nổi bật của React Native Elements là `Image Slider Box`. Thành phần này cho phép bạn dễ dàng hiển thị danh sách các hình ảnh theo dạng slideshow hoặc carousel.

Các thành phần của React Native Elements trong ứng dụng:

```
import { Button, TextInput, Card, Image } from '@rneui/base';;
```

Để sử dụng React Native Elements, Cài đặt thư viện.

Cài `npm install react-native-elements`

b.5. Gesture Handler

React Native Gesture Handler là một thư viện mã nguồn mở cho React Native, cung cấp một API để xử lý các cử chỉ trên ứng dụng. Thư viện này thay thế cho hệ thống xử lý cử chỉ mặc định của React Native, được gọi là `Gesture Responder System`.

`Gesture Responder System` có một số hạn chế, bao gồm:

- Không thể xử lý các cử chỉ phức tạp, chẳng hạn như cử chỉ vuốt nhiều ngón tay.
- Không thể xử lý các cử chỉ một cách đáng tin cậy, đặc biệt là trên các thiết bị cũ hoặc có cấu hình phần cứng thấp.

React Native Gesture Handler giải quyết các hạn chế này bằng cách cung cấp một API mạnh mẽ và linh hoạt hơn. Thư viện này cho phép bạn xử lý các cử chỉ phức tạp một cách đáng tin cậy.

Để sử dụng React Native Gesture Handler, bạn cần cài đặt thư viện. Có thể thực hiện việc này bằng cách chạy lệnh sau trong terminal:

Cài đặt: `npm install react-native-gesture-handler`

b.6. Redux

Redux là một thư viện JavaScript giúp quản lý trạng thái của ứng dụng. Redux được xây dựng dựa trên kiến trúc Flux, trong đó trạng thái của ứng dụng được lưu trữ trong một đối tượng duy nhất, được gọi là store. Các thay đổi đối với trạng thái chỉ có thể được thực hiện thông qua các action, là các đối tượng mô tả những gì đã xảy ra.

Một số điểm quan trọng về Redux:

- Mã nguồn mở: Redux là mã nguồn mở, nghĩa là có thể tùy chỉnh và sửa đổi nó tùy theo nhu cầu của người sử dụng.
- Trạng thái duy nhất: Redux yêu cầu trạng thái của ứng dụng phải được lưu trữ trong một đối tượng duy nhất, được gọi là store. Điều này giúp đảm bảo rằng trạng thái của ứng dụng luôn nhất quán và dễ hiểu.
- Action: Các thay đổi đối với trạng thái chỉ có thể được thực hiện thông qua các action. Các action là các đối tượng mô tả những gì đã xảy ra.
- Reducer: Reducer là một hàm được sử dụng để cập nhật trạng thái của ứng dụng dựa trên các action.

Tóm lại, Redux là một lựa chọn phổ biến cho việc quản lý trạng thái của ứng dụng vì nó cung cấp một số lợi ích sau:

- Trạng thái duy nhất: Đảm bảo rằng trạng thái của ứng dụng luôn nhất quán và dễ hiểu.
- Action: Cho phép kiểm soát chặt chẽ các thay đổi đối với trạng thái.
- Reducer: Cho phép trạng thái của ứng dụng được cập nhật một cách nhất quán.

Cài đặt:

```
npm install @reduxjs/toolkit npm
install react-redux
```

b.7. Reanimated

React Native Reanimated là một thư viện mã nguồn mở cho React Native, cung cấp một API để tạo các hiệu ứng hoạt hình phức tạp. Thư viện này được xây dựng dựa trên Reanimated v1, nhưng đã được cải tiến đáng kể về hiệu suất và tính năng.

Reanimated v1 sử dụng một hệ thống dựa trên callback để tạo các hiệu ứng hoạt hình. Hệ thống này có thể khó sử dụng và không hiệu quả.

React Native Reanimated sử dụng một hệ thống dựa trên state để tạo các hiệu ứng hoạt hình. Hệ thống này dễ sử dụng hơn và hiệu quả hơn.

Để sử dụng React Native Reanimated, có thể thực hiện việc này bằng cách chạy lệnh sau trong terminal.

Cài đặt `npm install react-native-reanimated`

b.8. Safe Area Context

React Native Safe Area Context là một thư viện mã nguồn mở cho React Native, cung cấp một API để truy cập vào thông tin về vùng an toàn của thiết bị. Vùng an toàn là phần màn hình mà không bị chiếm bởi các thanh trạng thái, thanh điều hướng, hoặc các thành phần giao diện người dùng khác.

Vùng an toàn là một tính năng quan trọng của thiết kế ứng dụng di động. Nó giúp đảm bảo rằng nội dung của ứng dụng được hiển thị chính xác, ngay cả khi các thanh trạng thái hoặc thanh điều hướng thay đổi kích thước.

Trước khi React Native Safe Area Context ra mắt, các nhà phát triển React Native phải tự tính toán vùng an toàn của thiết bị. Điều này có thể gây khó khăn và tốn thời gian.

React Native Safe Area Context giải quyết vấn đề này bằng cách cung cấp một API đơn giản để truy cập vào thông tin về vùng an toàn của thiết bị.

Để sử dụng React Native Safe Area Context, có thể thực hiện việc này bằng cách chạy lệnh sau trong terminal:

Cài đặt: `npm install react-native-safe-area-context`

b.9. Screens

React Native Screens là một thư viện mã nguồn mở cho React Native, cung cấp một API để quản lý màn hình trong ứng dụng. Thư viện này thay thế cho hệ thống quản lý màn hình mặc định của React Native, được gọi là Stack Navigator.

Stack Navigator có một số hạn chế, bao gồm:

Không thể quản lý các màn hình phức tạp, chẳng hạn như màn hình có nhiều điều khiển hoặc màn hình được chia thành nhiều phần.

Không thể quản lý các màn hình có trạng thái riêng biệt.

React Native Screens giải quyết các hạn chế này bằng cách cung cấp một API mạnh mẽ và linh hoạt hơn. Thư viện này cho phép bạn quản lý các màn hình phức tạp và quản lý các màn hình có trạng thái riêng biệt.

Để sử dụng React Native Screens, có thể thực hiện việc này bằng cách chạy lệnh sau trong terminal: *Cài đặt:* `npm install react-native-screens`

b.10. Router DOM

- React Router DOM là một thư viện mã nguồn mở cho React, cung cấp một API để quản lý điều hướng trong ứng dụng. Thư viện này cho phép bạn tạo các ứng dụng có nhiều màn hình, với khả năng di chuyển giữa các màn hình một cách dễ dàng.

-

- Để sử dụng React Router DOM, có thể thực hiện việc này bằng cách chạy lệnh sau trong terminal:

Cài đặt:

`npm install react-router-dom`

2. Firebase**2.1. Tổng quan**

Firebase là một nền tảng phát triển ứng dụng của Google, cung cấp một loạt các dịch vụ backend và công cụ phát triển để xây dựng ứng dụng di động và web. Firebase cung cấp các tính năng như cơ sở dữ liệu thời gian thực, xác thực người dùng, lưu trữ dữ liệu, thông báo và nhiều tính năng khác, tạo nền tảng mạnh mẽ cho việc phát triển ứng dụng.

Một số đặc điểm quan trọng của Firebase bao gồm:

- *Cơ sở dữ liệu*: Firebase cung cấp một cơ sở dữ liệu NoSQL có thể được sử dụng để lưu trữ dữ liệu của ứng dụng. Cơ sở dữ liệu Firebase có thể được truy cập từ bất kỳ nền tảng nào, bao gồm web, di động và máy tính để bàn.

- *Xác thực*: Firebase cung cấp các tính năng xác thực để giúp người dùng đăng nhập vào ứng dụng của bạn. Firebase hỗ trợ nhiều loại xác thực, bao gồm xác thực email/mật khẩu, xác thực mạng xã hội và xác thực không cần mật khẩu.

- *Giao tiếp*: Firebase cung cấp các tính năng giao tiếp để giúp bạn gửi thông báo và dữ liệu cho người dùng của mình. Firebase hỗ trợ các loại giao tiếp khác nhau, bao gồm thông báo đẩy, tin nhắn tức thì và nhắn tin thời gian thực.

- *Lưu trữ*: Firebase cung cấp các tính năng lưu trữ để giúp bạn lưu trữ các tệp, chẳng hạn như hình ảnh, video và âm thanh. Firebase cung cấp nhiều loại lưu trữ, bao gồm lưu trữ đám mây, lưu trữ cục bộ và lưu trữ di động.

- *Máy học*: Firebase cung cấp các tính năng máy học để giúp bạn phân tích dữ liệu và tạo các ứng dụng thông minh hơn. Firebase hỗ trợ các loại máy học khác nhau, bao gồm phân tích, thị giác máy tính và xử lý ngôn ngữ tự nhiên.

Firestore thường được sử dụng trong các ứng dụng web, di động và các hệ thống khác đòi hỏi tính mở rộng và linh hoạt trong cách lưu trữ và truy xuất dữ liệu.

2.2. Công cụ và dịch vụ chính

Firestore Console:

Firestore Console là một giao diện web cho phép bạn quản lý tất cả các dự án Firestore của mình. Bạn có thể sử dụng Firestore Console để:

Tạo và quản lý các dự án Firestore: Khi bạn tạo một dự án Firestore mới, bạn sẽ được cung cấp một ID dự án và một chìa API. Bạn cần sử dụng ID dự án và chìa API này để tích hợp các tính năng của Firestore vào ứng dụng của mình.

Tạo và quản lý các cơ sở dữ liệu: Firestore cung cấp một cơ sở dữ liệu NoSQL có thể được sử dụng để lưu trữ dữ liệu của ứng dụng. Bạn có thể sử dụng Firestore Console để tạo và quản lý các cơ sở dữ liệu Firestore.

Xác thực người dùng: Firestore cung cấp các tính năng xác thực để giúp người dùng đăng nhập vào ứng dụng của bạn. Bạn có thể sử dụng Firestore Console để thiết lập các phương thức xác thực, chẳng hạn như xác thực email/mật khẩu, xác thực mạng xã hội và xác thực không cần mật khẩu.

Gửi thông báo: Firebase cung cấp các tính năng giao tiếp để giúp bạn gửi thông báo và dữ liệu cho người dùng của mình. Bạn có thể sử dụng Firebase Console để tạo và gửi thông báo đẩy, tin nhắn tức thì và nhắn tin thời gian thực.

Lưu trữ: Firebase cung cấp các tính năng lưu trữ để giúp bạn lưu trữ các tệp, chẳng hạn như hình ảnh, video và âm thanh. Bạn có thể sử dụng Firebase Console để tạo và quản lý các bucket lưu trữ Firebase.

Máy học: Firebase cung cấp các tính năng máy học để giúp bạn phân tích dữ liệu và tạo các ứng dụng thông minh hơn. Bạn có thể sử dụng Firebase Console để tạo và sử dụng các mô hình máy học.

Firebase Console là một công cụ hữu ích cho các nhà phát triển muốn quản lý các dự án Firebase của họ. Nó cung cấp một giao diện trực quan và dễ sử dụng cho phép bạn dễ dàng thực hiện các tác vụ quản trị khác nhau.

Firebase CLI

Firebase CLI là một bộ công cụ dòng lệnh cho phép bạn tương tác với Firebase từ dòng lệnh. Bạn có thể sử dụng Firebase CLI để:

Tạo và quản lý các dự án Firebase: Bạn có thể sử dụng Firebase CLI để tạo các dự án Firebase mới, tạo và quản lý các cơ sở dữ liệu, xác thực người dùng, gửi thông báo và hơn thế nữa.

Triển khai ứng dụng của bạn: Bạn có thể sử dụng Firebase CLI để triển khai ứng dụng của mình lên Firebase Hosting.

Tạo và quản lý các ứng dụng web có thể mở rộng: Bạn có thể sử dụng Firebase CLI để tạo và quản lý các ứng dụng web có thể mở rộng với Firebase Functions.

Firebase CLI là một công cụ mạnh mẽ cho các nhà phát triển muốn tự động hóa các tác vụ quản trị và triển khai ứng dụng. Nó có thể được sử dụng để thực hiện các tác vụ phức tạp một cách nhanh chóng và dễ dàng.

Firebase SDK

Firebase SDK cung cấp các thư viện cho phép bạn tích hợp các tính năng của Firebase vào ứng dụng của mình. Firebase SDK có sẵn cho nhiều ngôn ngữ và nền tảng khác nhau, bao gồm:

Web: Javascript, Typescript

Android: Java, Kotlin

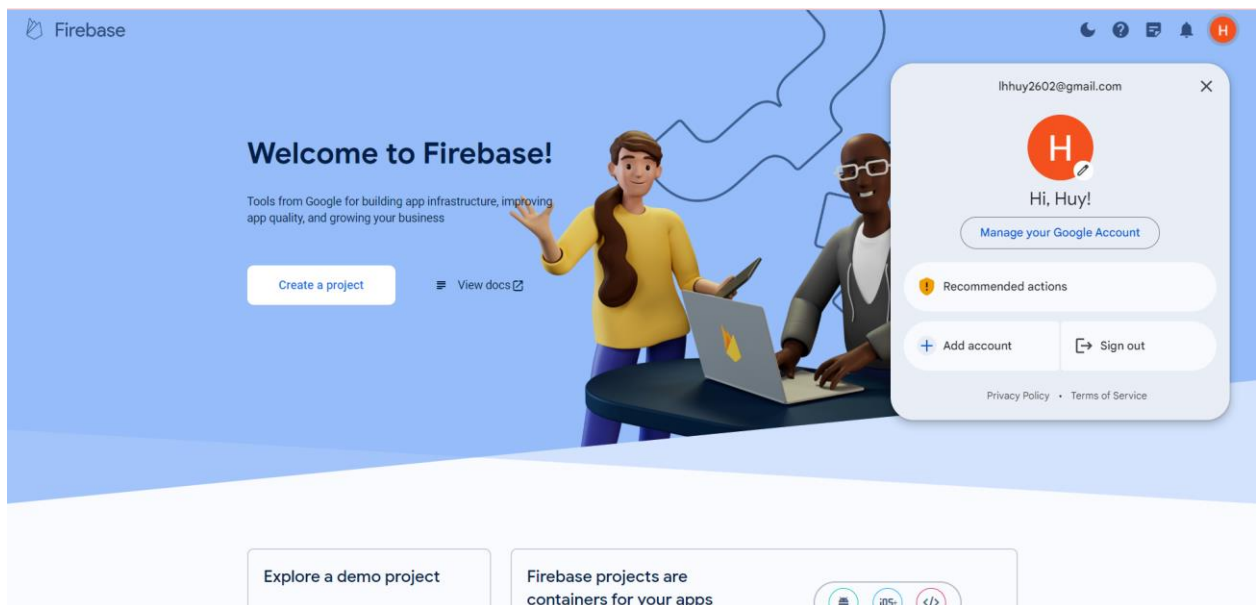
Firebase SDK cung cấp một API đơn giản và dễ sử dụng cho phép bạn truy cập các tính năng của Firebase từ ứng dụng của mình. Ví dụ: nếu bạn muốn sử dụng cơ sở dữ liệu Firebase trong ứng dụng web của mình, bạn có thể sử dụng thư viện Firebase SDK cho Javascript để truy cập cơ sở dữ liệu.

Firebase SDK là một công cụ quan trọng cho các nhà phát triển muốn tích hợp các tính năng của Firebase vào ứng dụng của mình. Nó cung cấp một cách dễ dàng để truy cập các tính năng của Firebase từ bất kỳ ngôn ngữ hoặc nền tảng nào

2.3. Cách đăng ký và cài đặt Firebase

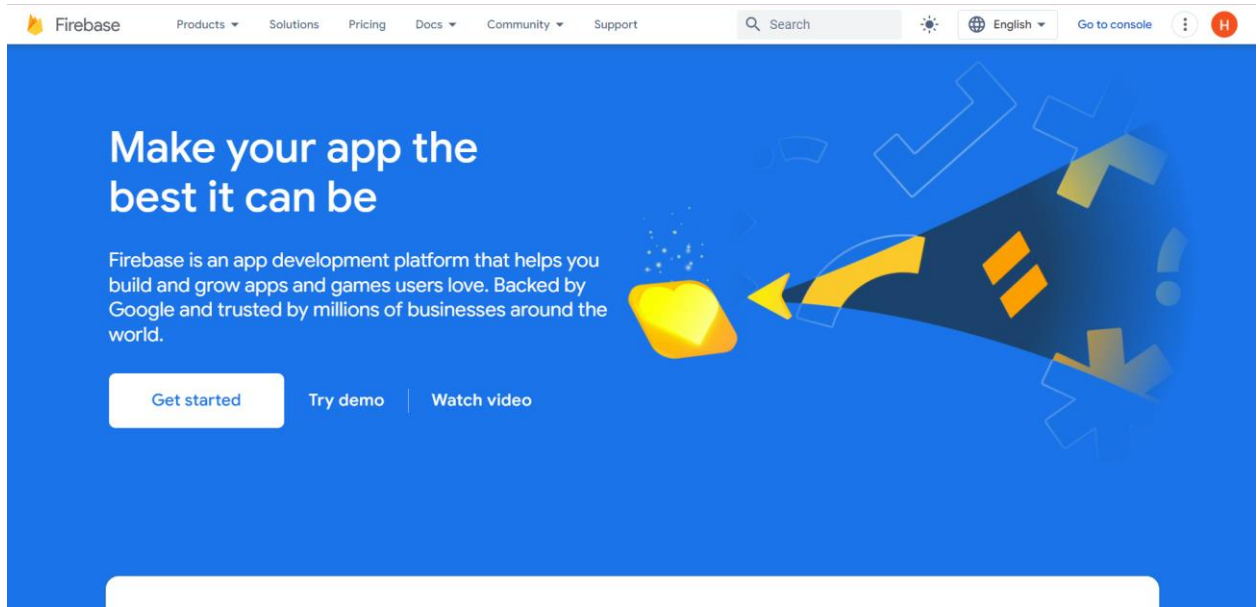
a. Đăng ký tài khoản Firebase

Truy cập: <https://firebase.google.com/> và thực hiện theo trình tự.

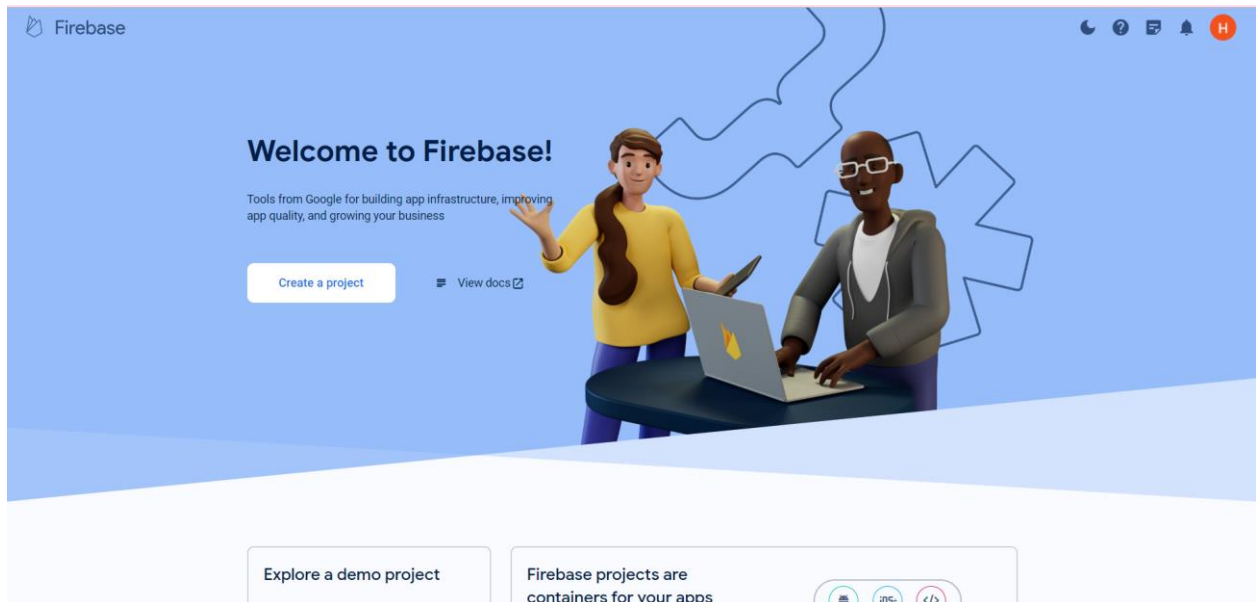


Bước 1. Nhập thông tin đăng ký tài khoản Firebase bằng tài khoản Google.

b. Bắt đầu và tạo project với Firebase



Bước 2. Chọn Get started hoặc Go to console.



Bước 3. Tạo project với nút Create a project.

×

Create a project (Step 1 of 3)

Let's start with a name for your project[Ⓜ]

Project name

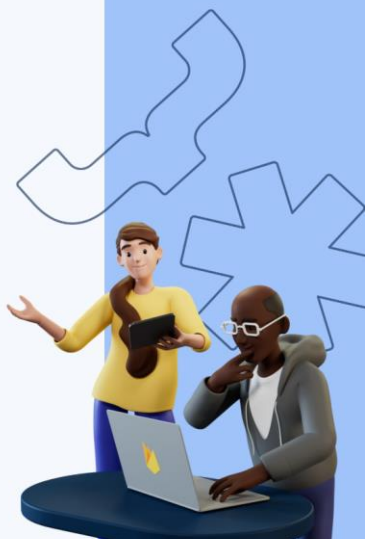
News App

news-app-2204a

☒ I accept the [Firebase terms](#) Ⓜ

☒ I confirm that I will use Firebase exclusively for purposes relating to my trade, business, craft, or profession.

Continue



Bước 4. Đặt tên cho project và chấp nhận các điều khoản rồi chọn vào nút Continue

×

Create a project (Step 2 of 3)

Google Analytics for your Firebase project

Google Analytics is a free and unlimited analytics solution that enables targeting, reporting, and more in Firebase Crashlytics, Cloud Messaging, In-App Messaging, Remote Config, A/B Testing, and Cloud Functions.

Google Analytics enables:

A/B testing Ⓜ

Crash-free users Ⓜ

User segmentation & targeting across Firebase products Ⓜ

Event-based Cloud Functions triggers Ⓜ

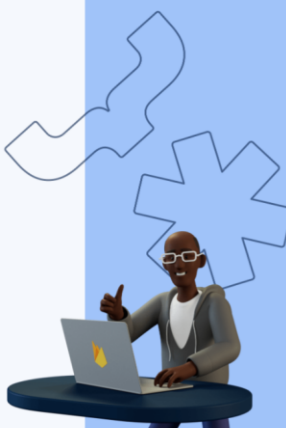
Free unlimited reporting Ⓜ

☒ Enable Google Analytics for this project

Recommended

Previous

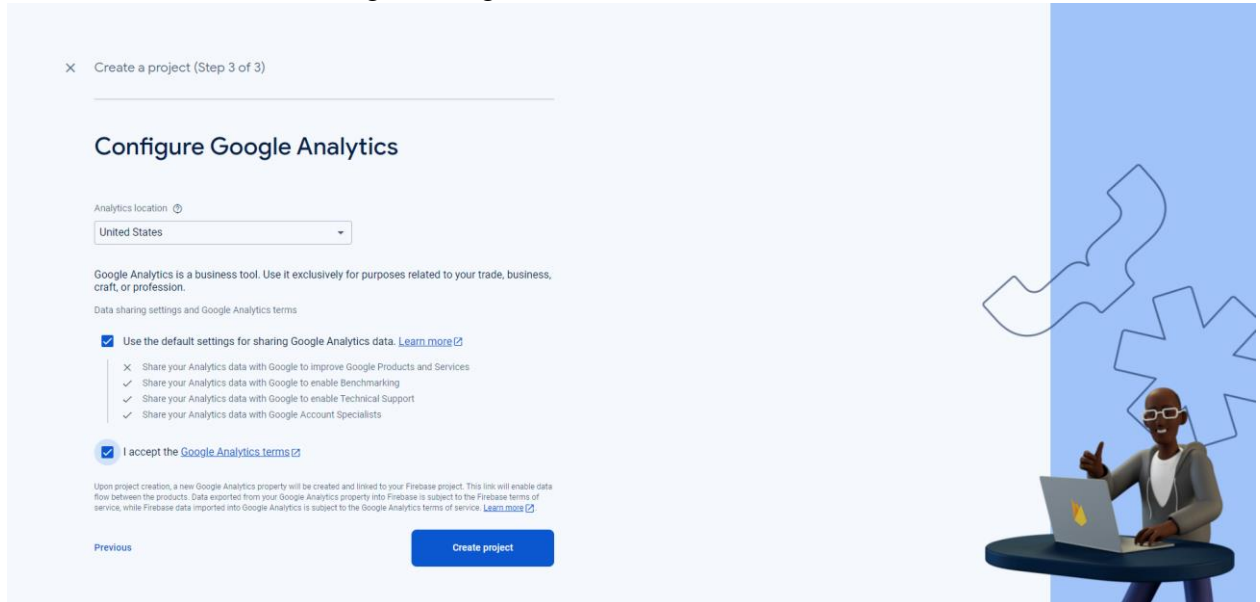
Continue



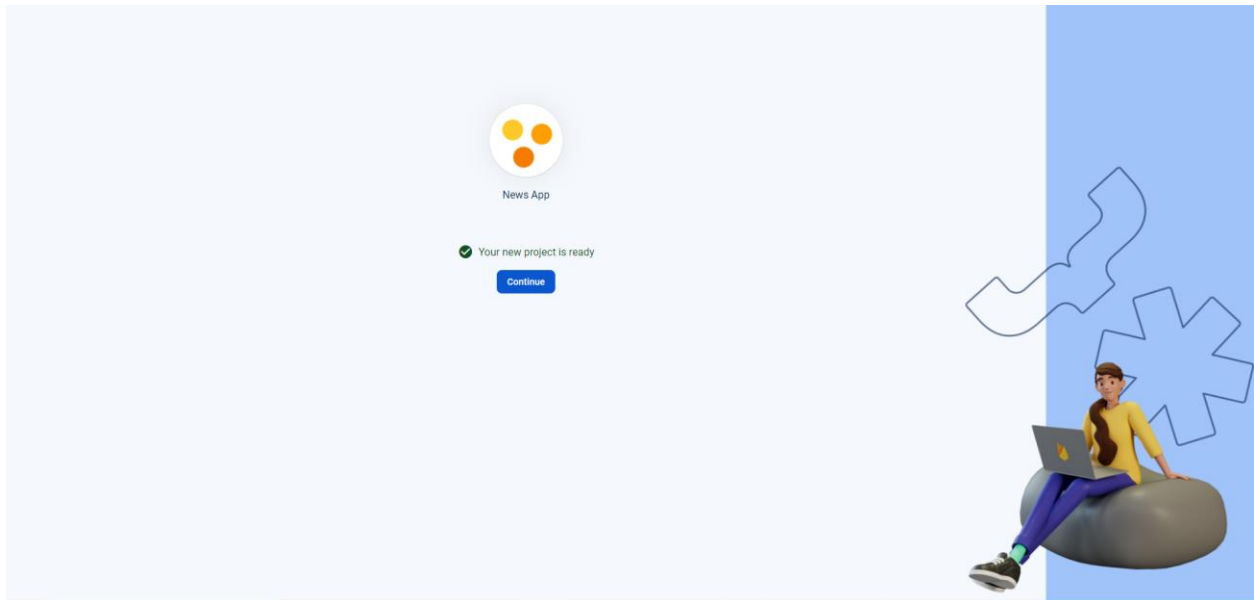
Bước 5. Chọn Continue với Google Analytics.

Phổ News – News App

24

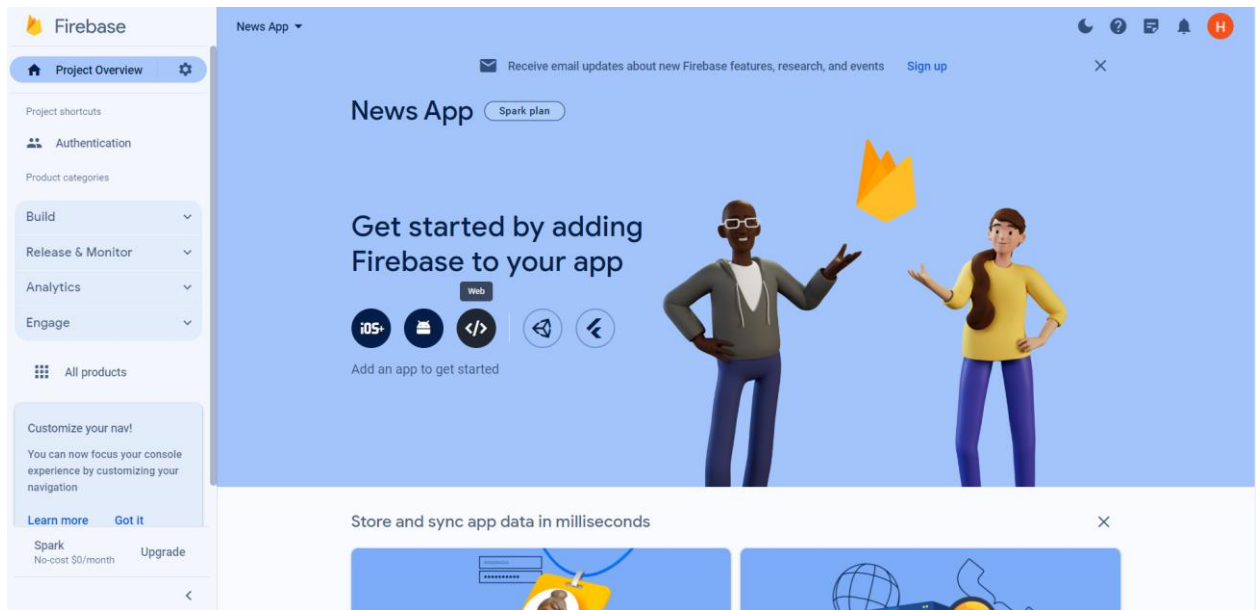


Bước 6. Chọn I accept the Google Analytics terms rồi chọn vào nút Create project.

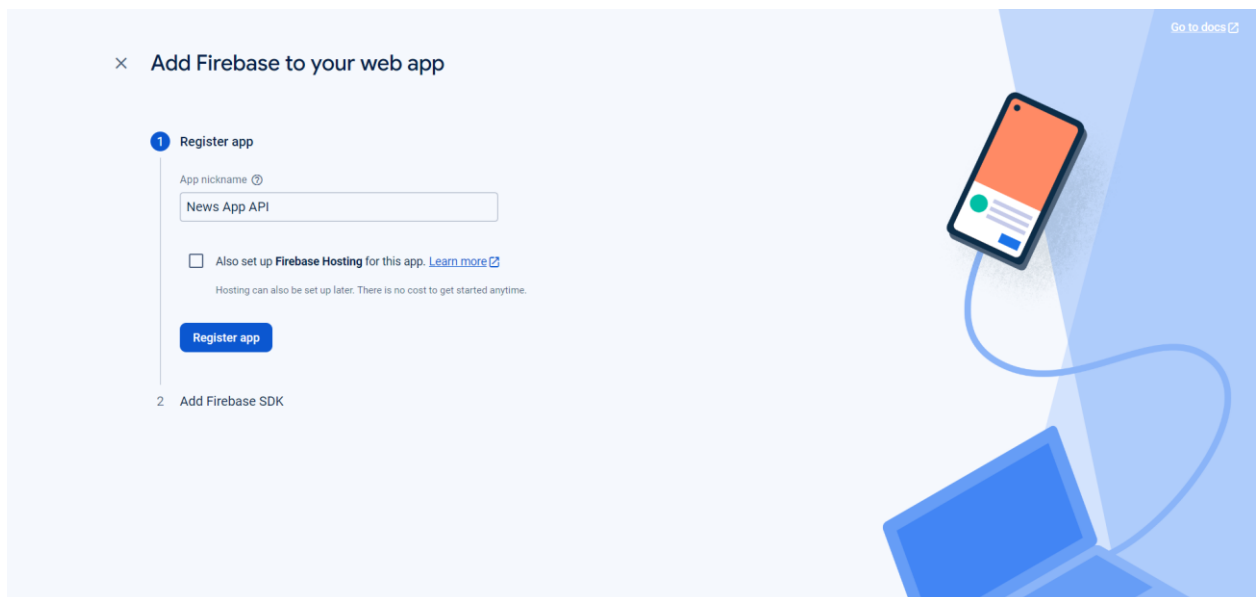


Bước 7. Chọn Continue.

c. **Thêm firebase và firebase SDK vào app**



Bước 8. Chọn vào Web ở Project Overview.

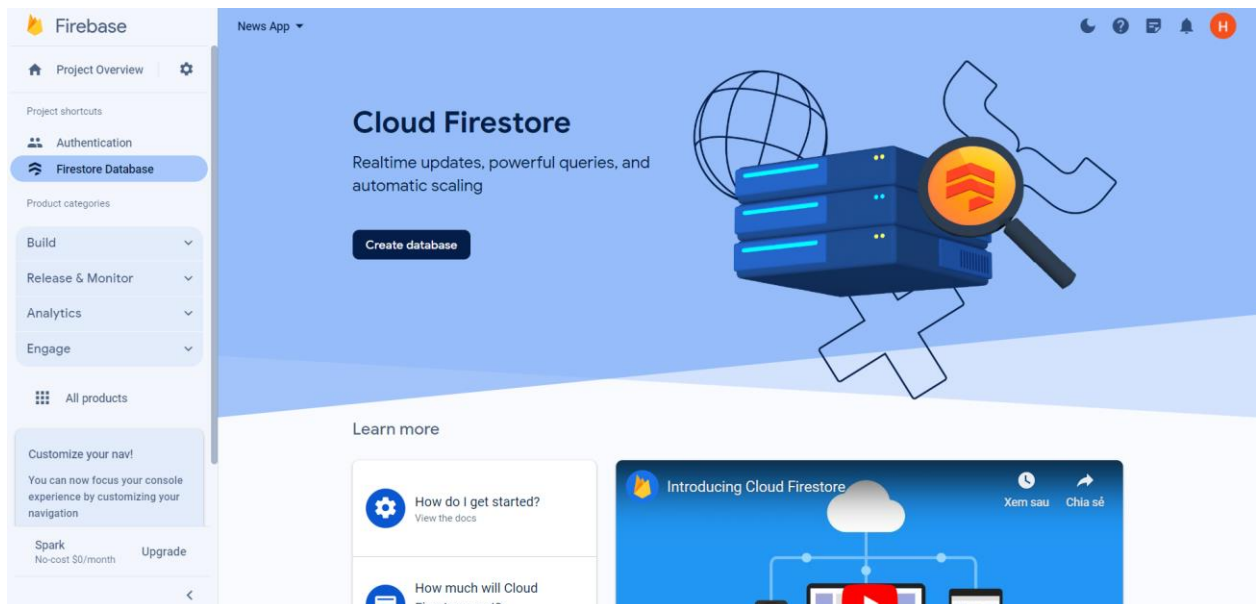


Bước 9. Đặt tên ở mục Register app và chọn vào nút Register app.

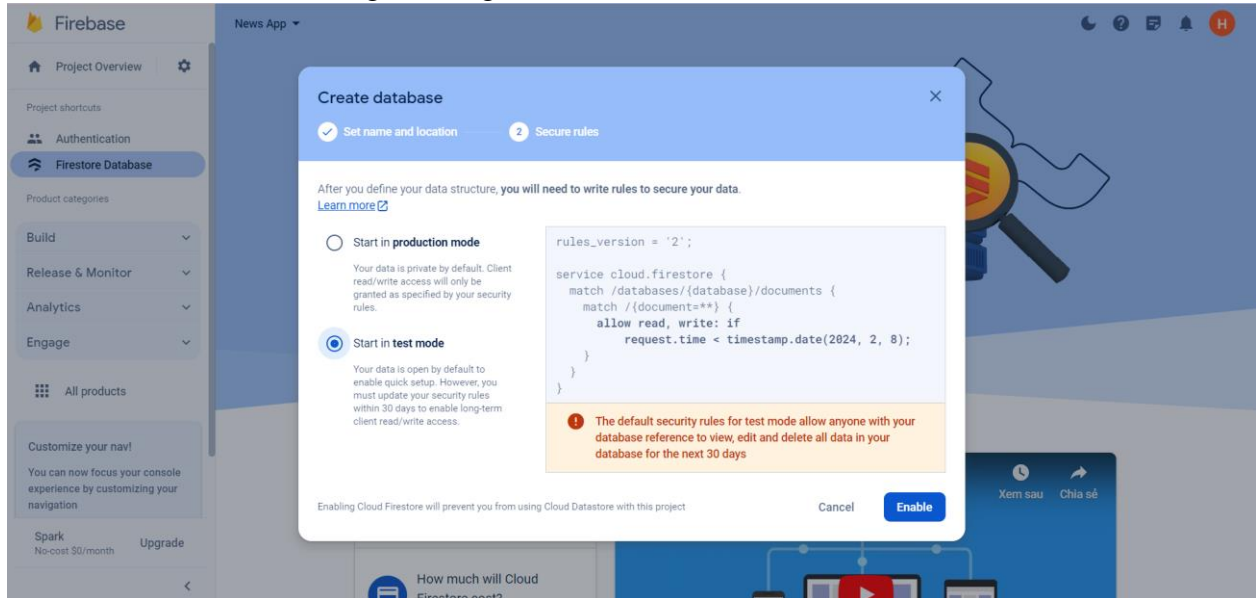


Bước 10. Cài đặt firebase trên terminal và add Firebase SDK. Rồi chọn Continue to console

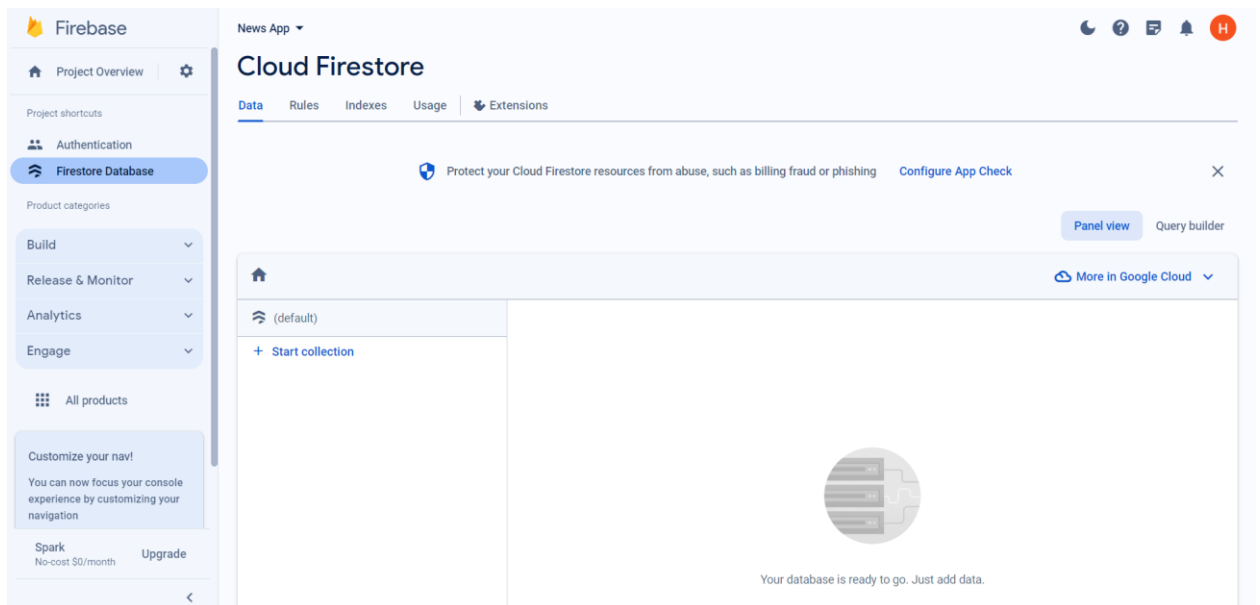
d. Tạo database



Bước 11. Chọn Create database ở phần Firestore Database

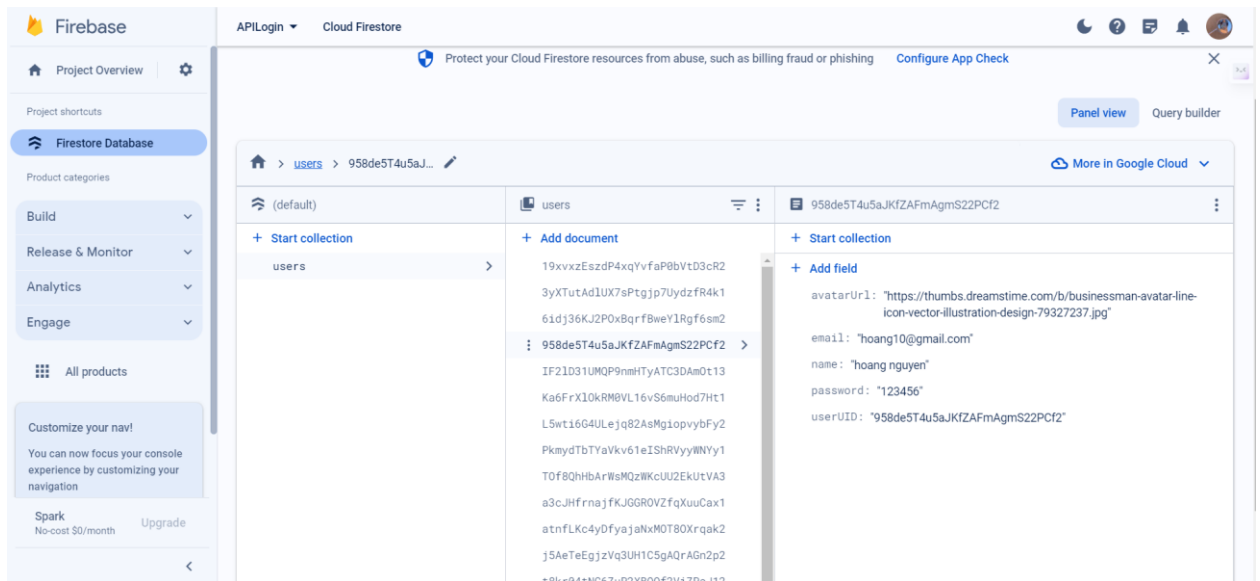


Bước 12. Chọn Start in test mode rồi chọn Enable



Bước 13. Tạo thành công Cloud Firestore để chứa data

e. Thực hiện lưu trữ data lên firebase bằng cách tạo user trên app



Kết quả: Sau khi đăng ký tài khoản mới ở màn hình register của app, thì data sẽ được lưu trữ ở mục user

3. Postman

3.1. Tổng quan

Postman là một công cụ phần mềm được sử dụng để thực hiện các hoạt động liên quan đến việc kiểm thử, gỡ lỗi và phân tích các API (Application Programming Interface). Nó cho phép người dùng tạo, gửi và xem kết quả từ các yêu cầu API một cách dễ dàng và hiệu quả. Postman cung cấp giao diện người dùng đồ họa, cho phép người dùng xây dựng, test và document các API một cách trực quan. Ngoài ra, Postman cũng cung cấp tính năng để tự động hóa các bước kiểm thử API và chia sẻ các collections (tập hợp các yêu cầu API liên quan) với các thành viên khác trong nhóm làm việc. Điều này giúp tăng cường hiệu suất và khả năng hợp tác trong quá trình phát triển và quản lý API. *Tạo yêu cầu HTTP:* Cho phép người dùng tạo các yêu cầu HTTP GET, POST, PUT, DELETE, PATCH và các loại yêu cầu khác để gửi đến server. Có các tính năng như :

- *Xây dựng và test API:* Postman cho phép người dùng tạo và thử nghiệm các yêu cầu API một cách linh hoạt và hiệu quả.

- *Gỡ lỗi API:* Postman cung cấp các công cụ giúp người dùng gỡ rối và sửa chữa các lỗi trong quá trình giao tiếp với API.

- *Tài liệu hóa API:* Người dùng có thể sử dụng Postman để tạo tài liệu cho các API một cách tự động, giúp người dùng dễ dàng hiểu cách sử dụng và tương tác với API.

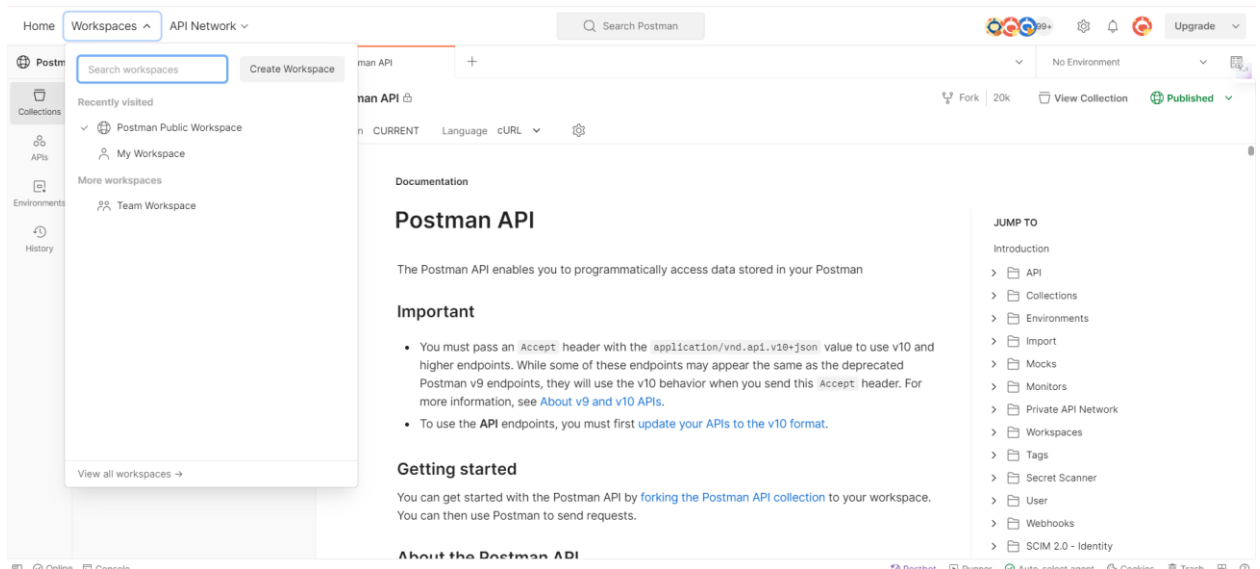
- *Tích hợp với các công cụ khác:* Postman có thể tích hợp với nhiều công cụ phát triển phổ biến khác nhau như GitHub, Jenkins, AWS, và nhiều công cụ quản lý dự án khác.

- *Kiểm thử tự động:* Postman cung cấp khả năng tự động hóa việc kiểm thử API, giúp tiết kiệm thời gian và tăng hiệu suất làm việc.

- *Chia sẻ và hợp tác:* Người dùng có thể chia sẻ collections (tập hợp các yêu cầu API) với các thành viên khác trong nhóm làm việc, tạo điều kiện cho việc hợp tác và đồng bộ hóa trong phát triển và quản lý API.

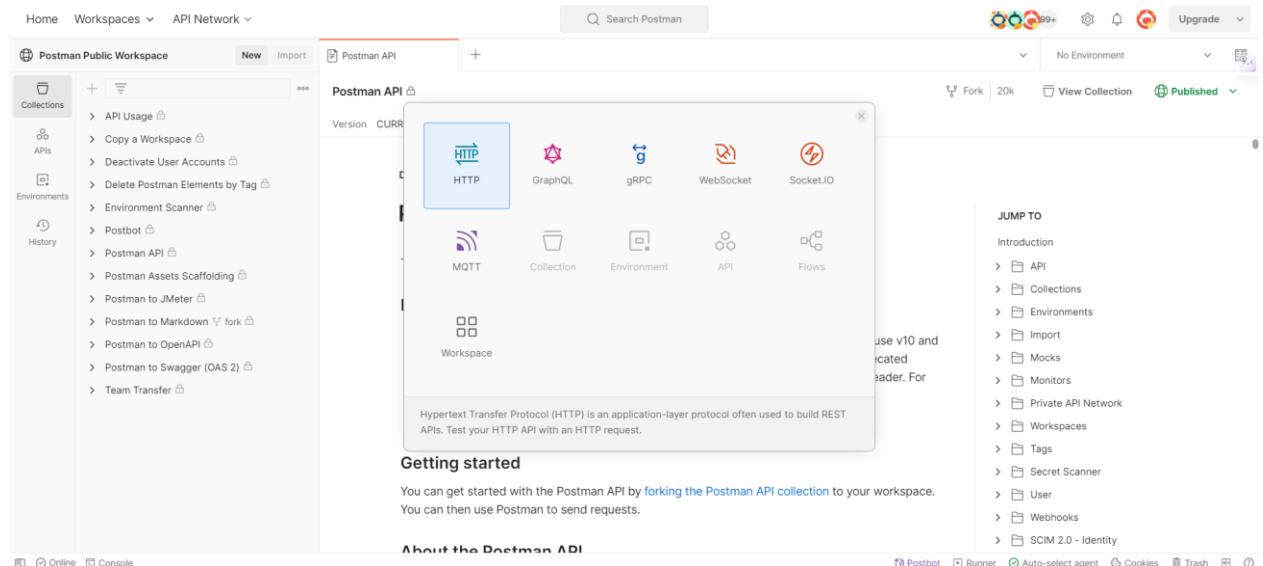
Postman là một công cụ hữu ích trong quá trình phát triển và kiểm thử API, giúp người dùng tạo, kiểm tra và debug các yêu cầu HTTP một cách hiệu quả và thuận tiện.

3.2. Cách sử dụng

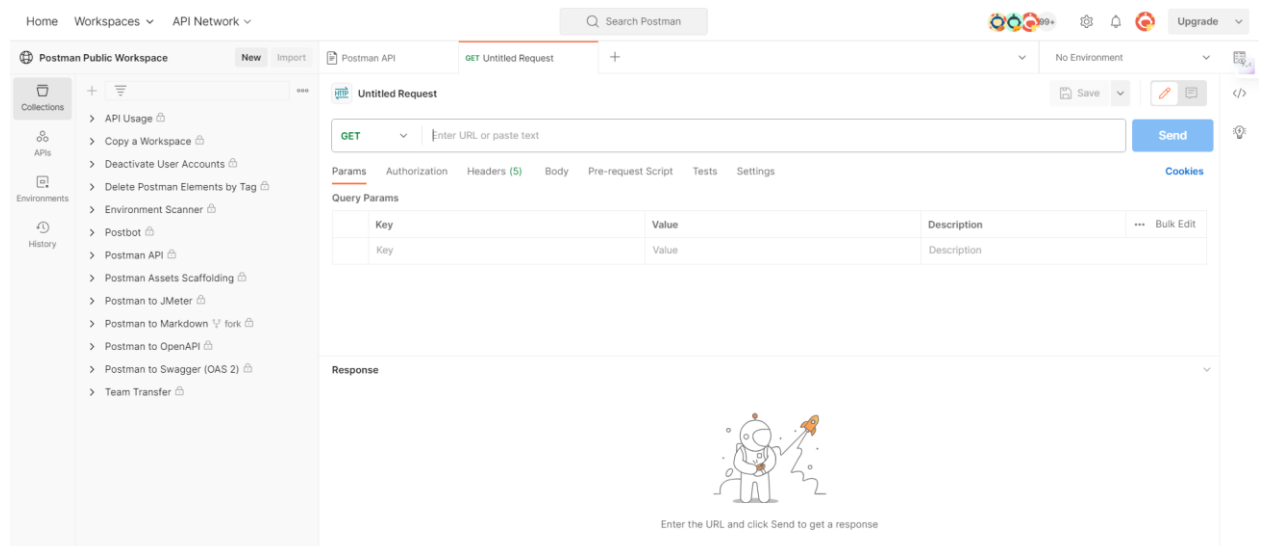


Bước 1 :Tạo 1 Workspaces

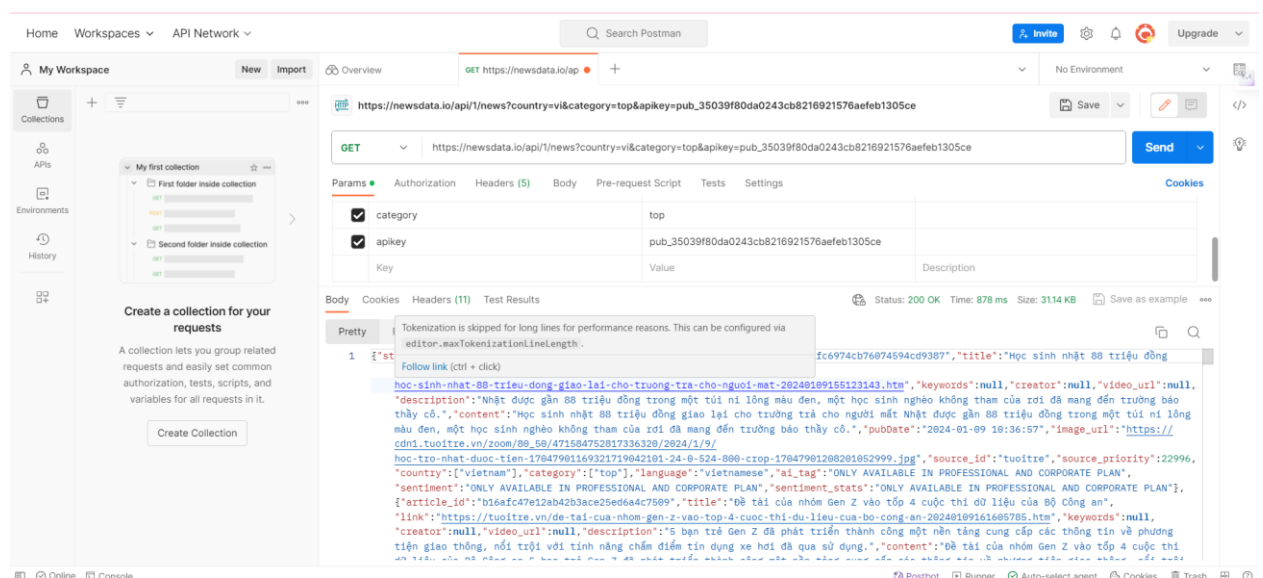
IE307 – Cross-Platform Programming



Bước 2 : Chọn new và chọn HTTP



Bước 3. Kiểm thử API ở khung GET.



Kết quả trả về từ việc kiểm thử đường link API.

Chương 3 ĐÁNH GIÁ VÀ KẾT LUẬN

1. Những điều làm được và chưa được

Phổ News News App là đồ án nhóm thực hiện về xây dựng và phát triển ứng dụng di động đọc báo trên online được thiết kế và thực hiện trong vòng 2-3 tháng bởi nhóm gồm 02 thành viên. Theo nhận định cá nhân, nhóm xin đưa ra các nhận định về những điều đã và chưa thực hiện được như sau:

• Điều làm được

- *Về mặt ý tưởng*: Đồ án đáp ứng được đầy đủ các nội dung chức năng trong việc thiết kế một ứng dụng di động đọc tin tức trong việc tương tác với các đối tượng dữ liệu: các bài báo, tương tác, đánh giá, yêu thích, phản hồi.
- *Về mặt kỹ thuật*:
 - + Đồ án xây dựng đầy đủ cấu trúc thành phần của hệ thống gồm lớp dữ liệu (database), lớp ứng dụng xử lý (backend) và lớp giao diện người dùng (frontend);
 - + Đồ án ứng dụng được các công nghệ cần thiết như Firebase, NodeJS, JavaScript và React Native từ cơ bản đến nâng cao trong mối quan hệ kết hợp, tương trợ nhau để cấu thành một hệ thống ứng dụng đa nền tảng đầy đủ, mạnh mẽ, linh hoạt và dễ quản lý.
- *Về mặt nhân lực*: Đồ án phân chia công việc cụ thể, rõ ràng giữa các thành viên trong nhóm, đồng thời tạo điều kiện để tiếp xúc nhiều công nghệ và công cụ mới, phát huy thế mạnh, học hỏi kiến thức và kinh nghiệm lẫn nhau, trau dồi khả năng làm việc nhóm và nâng cao tinh thần trách nhiệm chung trong cùng một dự án thực tế.

• Điều chưa làm được

- *Về mặt ý tưởng*: Nhóm tham vọng đồ án đạt được nhiều tiện ích hơn nữa như trang danh mục những bài báo đã yêu thích và đã từng lưu, tính năng post thêm bài báo theo mỗi cá nhân user,... tuy nhiên do thời gian và nhân lực có hạn, nên nhóm chủ yếu tập trung giải quyết bài toán sử dụng API để xây dựng ứng dụng. Hơn nữa, việc xây dựng back-end một cách chuyên sâu hơn sẽ là những tiện ích nâng cao trong việc nâng cao trải nghiệm người dùng cần được nghiên cứu và phát triển tích hợp thêm để có thể trở thành một ứng dụng thực sự.
- *Về mặt kỹ thuật*: Nhóm với tinh thần dám thử thách đã chọn những công nghệ, thành phần thiết kế giao diện mới, chưa từng được học và tiếp xúc nhiều. Đây vừa

là cơ hội học tập vừa là nguy cơ dẫn đến những hạn chế chưa tối ưu trong quá trình xử lý code là không thể tránh khỏi.

- *Về mặt nhân lực:* Đồ án đã đè nén nhiều áp lực về thời gian lên các thành viên và đặt ra nhiều thách thức về mặt kiến thức và khả năng lĩnh hội của mỗi người trong tình hình chung vừa học vừa làm. Điều này là yếu tố tiên quyết gây ra các mặt còn hạn chế trong ý tưởng và kỹ thuật vừa nêu.

2. Hướng phát triển và mở rộng của đồ án

Như đã đề cập, những tham vọng về mặt ý tưởng mà đồ án chưa thể đạt được sẽ là động lực để mở ra hướng phát triển và mở rộng hơn nữa trong tương lai, cụ thể:

- *Về tính năng:*

- + Xây dựng thêm một trang những bài báo đã yêu thích và có thể xem lại.
- + Xây dựng một cộng đồng sôi động, diễn đàn giao lưu trò chuyện trực tuyến, nơi mà những người đọc giả yêu thích những tin tức mỗi ngày có thể trao đổi, chia sẻ kiến thức, trải nghiệm và gắn kết với nhau.

- *Về tiện ích:*

- + Cung cấp thông và đề xuất các bài báo xu hướng theo từng loại khác nhau để các đọc giả có thể tham khảo và trải nghiệm một cách thực tế hơn.
- + Cung cấp báo cáo tài chính thông qua nhiều nền tảng mạng xã hội hơn.
- + Tích hợp nhiều hơn nữa các phương thức đăng nhập tiện lợi và dễ dàng hơn như : Facebook, Gmail...

- *Về giao diện:* tích hợp đồ họa 3D và giao diện động tăng xúc cảm trải nghiệm.

3. Kết luận

Dự án ứng dụng Phổ News News App về mặt ý tưởng có triển vọng đón đầu xu hướng thích đọc và tìm kiếm các tin nóng trên mạng xã hội; và về mặt công nghệ là sử dụng một nền tảng thiết kế React Native chưa bao giờ lỗi thời để cung cấp nhiều tính năng và tiện ích tuyệt vời như tham vọng thiết kế của nhóm và nhiều hơn thế nữa theo nhu cầu của xu hướng trong tương lai.

Trên tinh thần phát huy những điều đã làm được về công nghệ, cũng như phát triển thêm những điều chưa được hiện thực hoá trong ý tưởng, bên cạnh việc kết hợp lối thiết kế chuyên nghiệp và thẩm mỹ về mặt hình thức thể hiện giao diện cùng khả năng quản lý cơ sở dữ liệu toàn diện và hiệu quả, nhóm tin rằng đây sẽ không dừng lại ở mức độ chỉ là đồ án sinh viên, mà nó còn là mảnh đất màu mỡ, giàu nguồn tài nguyên với nhiều mảng sáng tối đa sắc cần được khai phá và giải phóng sức sáng tạo.

DANH MỤC NGUỒN THAM KHẢO

- ThS. Võ Ngọc Tân, “*Bài giảng môn Công nghệ lập trình đa nền tảng cho ứng dụng di động*” - Khoa Khoa học và Kỹ thuật thông tin, Đại học Công nghệ Thông tin ĐHQG TP. Hồ Chí Minh

Danh mục nguồn công nghệ và công cụ thực hiện

1. Firebase: <https://console.firebase.google.com/>
2. Nodejs: <https://nodejs.org/en>
3. React Native: <https://reactnative.dev/docs/view>
4. React Native Components: <https://aboutreact.com/components/>
5. React Router: <https://reactrouter.com/en/main>
6. Redux: <https://redux.js.org>
7. GitHub: <https://github.io>
8. Visual Studio Code: <https://code.visualstudio.com/download>