BỘ GIÁO DỤC VÀ ĐÀO TẠO

**TRƯỜNG ĐẠI HỌC NHA TRANG**

**KHOA CÔNG NGHỆ THÔNG TIN**

A blue and red logo

Description automatically generated

**BÁO CÁO ĐỒ ÁN TỐT NGHIỆP**

**XÂY DỰNG ỨNG DỤNG DI ĐỘNG QUẢN LÝ TÀI CHÍNH CÁ NHÂN**





Nha Trang, tháng 3 năm 2024

| **TRƯỜNG ĐẠI HỌC NHA TRANG**  KHOA CÔNG NGHỆ THÔNG TIN |  |
| --- | --- |

**PHIẾU THEO DÕI TIẾN ĐỘ VÀ ĐÁNH GIÁ**

**ĐỒ ÁN/KHÓA LUẬN/CHUYÊN ĐỀ TỐT NGHIỆP**

**(Dùng cho CBHD và nộp cùng báo cáo ĐA/KL/CĐTN của sinh viên)**

Tên đề tài: Xây Dựng Ứng Dụng Di Động Quản Lý Tài Chính Cá Nhân

Giảng viên hướng dẫn: Ths. Nguyễn Hải Triều

Sinh viên được hướng dẫn: Nguyễn Duy Thiên MSSV: 62131996

Khóa: 62 Ngành: Công nghệ thông tin

| **Lần KT** | **Ngày** | **Nội dung** | **Nhận xét của GVHD** |
| --- | --- | --- | --- |
| 1 | 19/02/2024 | Tìm hiểu về đề tài, các chức năng thống, phân tích thiết kế hệ thống. | Đạt |
| 12/04/2024 | Viết chi tiết báo cáo, làm thêm chức năng danh mục, filter, theo đổi số dư, chọn tài khoản, màn hình đăng xuất, | Đạt |
| 04/03/2024 | Phân tích thiết kế ứng dụng dựa trên hệ qtcsdl nonson sql thống kê, thông báo nhắc nhở người thêm dữ liệu cho app, api gemini để tạo app chat.  Sau khi làm hết các tính năng, tải thử 1 app chat trên CHPLAY, xem thử nó có tính năng khác không thì tham khảo, so sánh với app đã tải về nhà |  |
| 11/03/2024 | Thiết kế giao diện Trang chủ |  |
| **Kiểm tra giữa tiến độ của trưởng Bộ môn** | | | |
| Ngày kiểm tra:  ……………………. | | Đánh giá công việc hoàn thành: …..%  Được tiếp tục: € Không tiếp tục: € | Ký tên ………………………. |
|  | |  |  |
|  | |  |  |
|  | |  |  |
|  | |  |  |

Nhận xét chung (sau khi sinh viên hoàn thành ĐA/KL/CĐTN):

Điểm hình thức: 9/10 Điểm nội dung: 9/10 Điểm tổng kết: 9/10

+ Đối với ĐA/KLTN:

Kết luận sinh viên: Được bảo vệ: 🗹 Không được bảo vệ: €

Khánh Hòa, ngày…….tháng 07 năm 2021

Cán bộ hướng dẫn

(Ký và ghi rõ họ tên)

Nguyễn Hải Triều

**TRƯỜNG ĐẠI HỌC NHA TRANG**

**Khoa Công Nghệ Thông Tin**

**PHIẾU CHẤM ĐIỂM ĐỒ ÁN/ KHÓA LUẬN TỐT NGHIỆP**

***(Dành cho cán bộ chấm phản biện)***

Tên đề tài: Xây Dựng Ứng Dụng Di Động Quản Lý Tài Chính Cá Nhân

Chuyên ngành: Công nghệ thông tin.

Họ và tên sinh viên: Nguyễn Duy Thiên Mã sinh viên: 62131996

Người phản biện (học hàm, học vị, họ và tên):

Cơ quan công tác:

**Phần đánh giá và cho điểm của người phản biện (tính theo thang điểm 10)**

| **Tiêu chí đánh giá** | **Trọng số**  **(%)** | **Mô tả mức chất lượng** | | | | **Điểm** |
| --- | --- | --- | --- | --- | --- | --- |
| **Giỏi** | **Khá** | **Đạt yêu cầu** | **Không đạt** |
| **9 - 10** | **7 - 8** | **5 - 6** | **< 5** |
| Hình thức bản thuyết minh | 30 |  |  |  |  |  |
| Nội dung bản  thuyết minh | 30 |  |  |  |  |  |
| Kết quả nghiên cứu | 20 |  |  |  |  |  |
| Mức độ trích dẫn và sao chép | 20 |  |  |  |  |  |
| **ĐIỂM TỔNG** | | | | | |  |

*Ghi chú: Điểm tổng làm tròn đến 1 số lẻ.*

**Kết luận:**

Đồng ý cho sinh viên: Được bảo vệ: € Không được bảo vệ: €

*Khánh Hòa, ngày…….tháng………năm………..* **Cán bộ chấm phản biện** *(Ký và ghi rõ họ tên)*

**II. Phần nhận xét cụ thể (dựa theo phiếu chấm điểm và khung tiêu chí đánh giá theo Rubric)**

**II.1. Hình thức thuyết minh (tỉ trọng 30%)**

***\* Trình bày*** *(Rõ ràng, mạch lạc? Biểu bảng, hình vẽ trình bày rõ ràng, đúng quy cách?…)*

……………………………………………………………………………………………………

……………………………………………………………………………………………………

***\* Bố cục và lập luận*** *(Bố cục hợp lý? Tỉ trọng giữa các phần? Cơ sở lập luận?...)*

……………………………………………………………………………………………………

……………………………………………………………………………………………………

***\* Văn phong*** *(Gọn gàng, súc tích hay rườm rà, khó hiểu? Lỗi văn phạm và chính tả?…)*

……………………………………………………………………………………………………

**II.2. Nội dung thuyết minh** (tỉ trọng 30%)

***\* Mục tiêu nghiên cứu*** *(Trình bày rõ ràng? Ý nghĩa khoa học và thực tiễn? Tính khả thi?...)*

……………………………………………………………………………………………………

……………………………………………………………………………………………………

***\* Tổng quan tài liệu*** *(Phân tích và đánh giá? Độ tin cậy và chất lượng nguồn tài liệu?…)*

……………………………………………………………………………………………………

……………………………………………………………………………………………………

***\* Phương pháp nghiên cứu* (***Hiện đại?**Phù hợp với mục tiêu và nội dung nghiên cứu? Mô tả? Đánh giá và so sánh với các phương pháp khác?…)*

……………………………………………………………………………………………………

……………………………………………………………………………………………………

**II.3. Kết quả nghiên cứu (**tỉ trọng 20%)

***\* Kết quả đạt được*** *(Độ**tin cậy? Tính sáng tạo? Giá trị khoa học và thực tiễn?...)*

……………………………………………………………………………………………………

……………………………………………………………………………………………………

***\* Kết luận*** *(Đáp ứng mục tiêu nghiên cứu? Quan điểm của cá nhân? ...)*

……………………………………………………………………………………………………

……………………………………………………………………………………………………

**4. MỨC ĐỘ TRÍCH DẪN VÀ SAO CHÉP** (tỉ trọng 20%)

**\* Mức độ trích dẫn *(****Đúng quy định? Trung thực, đầy đủ, rõ ràng?Sắp xếp tài liệu tham khảo?...)*

……………………………………………………………………………………………………

**\* Mức độ sao chép** *(Tỉ lệ sao chép? Hình thức sao chép?...)*

……………………………………………………………………………………………………

**MỤC LỤC**

[CHƯƠNG 1. CƠ SỞ LÝ THUYẾT 1](#_heading=h.2et92p0)

[1.1.](#_heading=h.tyjcwt) Giới thiệu ứng dụng di động: 1

[1.2.](#_heading=h.3dy6vkm) Giới thiệu lập trình ứng dụng di động: 3

[CHƯƠNG 2. CÁC PHẦN MỀM VÀ CÔNG CỤ SỬ DỤNG 4](#_heading=h.1t3h5sf)

[2.1. Các phần mềm công cụ sử dụng: 4](#_heading=h.4d34og8)

[2.2. Giới thiệu về Dark – Flutter 4](#_heading=h.2s8eyo1)

[2.3.Giới thiệu về Firebase 6](#_heading=h.17dp8vu)

[2.3.1. Lịch sử phát triển Firebase 6](#_heading=h.3rdcrjn)

[2.3.1. Các tính năng chính của Firebase 7](#_heading=h.26in1rg)

[CHƯƠNG 3. KHẢO SÁT HIỆN TRẠNG 9](#_heading=h.lnxbz9)

[3.1. Khảo sát hiện trạng: 9](#_heading=h.35nkun2)

[CHƯƠNG 4. PHÂN TÍCH HỆ THỐNG 10](#_heading=h.1ksv4uv)

[4.1. Phân tích hệ thống: 10](#_heading=h.44sinio)

[CHƯƠNG 5. THIẾT KẾ DỮ LIỆU 11](#_heading=h.2jxsxqh)

[3.1. Thiết kế dữ liệu: 11](#_heading=h.z337ya)

**LỜI MỞ ĐẦU**

Trong thời đại số hóa ngày nay, việc quản lý tài chính cá nhân trở nên quan trọng hơn bao giờ hết. Với sự phát triển không ngừng của công nghệ di động, việc sử dụng ứng dụng di động để quản lý tài chính cá nhân đã trở thành một phần không thể thiếu trong cuộc sống hàng ngày của chúng ta. Đồng thời, việc xây dựng một ứng dụng di động hiệu quả và tiện ích trong việc quản lý tài chính cũng đặt ra nhiều thách thức đối với các nhà phát triển.

Đồ án tốt nghiệp này là kết quả của sự nỗ lực và nghiên cứu sâu sắc của em trong việc phát triển một ứng dụng di động quản lý tài chính cá nhân. Chúng tôi đã dành nhiều tháng để nghiên cứu, phân tích và thiết kế một hệ thống hoàn chỉnh, mang lại sự thuận tiện và hiệu quả cho người dùng trong việc quản lý tài chính của mình.

Trong báo cáo này, chúng tôi sẽ trình bày chi tiết về quá trình phát triển ứng dụng, từ việc nghiên cứu nhu cầu của người dùng, đến việc thiết kế giao diện và triển khai hệ thống. Chúng tôi cũng sẽ giới thiệu về các tính năng chính của ứng dụng và cách mà chúng mang lại giá trị thực cho người dùng.

Em hi vọng rằng thông qua báo cáo này, mọi người sẽ hiểu rõ hơn về quá trình phát triển ứng dụng di động quản lý tài chính cá nhân, cũng như đánh giá được công lao và sự cống hiến của em trong dự án này. Hy vọng rằng sản phẩm của em sẽ góp phần vào việc cải thiện quản lý tài chính và cuộc sống của người dùng trong thời đại số hóa ngày nay.

Repo dự án: [duythien02/expenses\_tracker (github.com)](https://github.com/duythien02/expenses_tracker)

**TỔNG QUAN VỀ ĐỀ TÀI**

***Lý do chọn đề tài***

Trong thời đại hiện đại với sự bùng nổ của công nghệ di động, quản lý tài chính cá nhân trở thành một vấn đề cực kỳ quan trọng. Việc đảm bảo sự quản lý tài chính hiệu quả và tiện lợi đã trở thành một ưu tiên hàng đầu của nhiều người. Chính vì vậy, việc phát triển một ứng dụng di động nhằm hỗ trợ quản lý tài chính cá nhân đã thu hút sự quan tâm lớn từ cộng đồng. Ứng dụng di động không chỉ mang lại sự thuận tiện cho người dùng trong việc theo dõi và quản lý tài chính của họ mọi lúc mọi nơi, mà còn giúp họ hiểu rõ hơn về tình hình tài chính của mình thông qua các công cụ và báo cáo phân tích, đồng thời giảm thiểu sai sót trong quá trình ghi chép và tính toán. Em đã quyết định chọn đề tài "Xây dựng ứng dụng di động quản lý tài chính cá nhân" vì nhận thức sâu sắc về tầm quan trọng của việc quản lý tài chính trong cuộc sống hàng ngày của mọi người và khả năng của công nghệ di động trong việc giải quyết vấn đề này một cách hiệu quả và tiện lợi.

***Mục tiêu đề tài***

Phát triển một ứng dụng di động hiệu quả, đơn giản và dễ sử dụng: Mục tiêu chính là tạo ra một ứng dụng di động mà người dùng có thể dễ dàng truy cập và sử dụng mọi lúc mọi nơi, mà không gặp phải các rắc rối về giao diện hoặc trải nghiệm người dùng. Cung cấp tính năng quản lý tài chính toàn diện: Ứng dụng sẽ cung cấp các tính năng như theo dõi chi tiêu, thiết lập ngưỡng chi tiêu hàng tháng, và gợi ý đề chi tiêu hợp lý dựa trên các hoạt động chi tiêu trước, tích hợp chatbot AI để hỏi các vấn đề liên quan.

**CHƯƠNG 1. CƠ SỞ LÝ THUYẾT**

* 1. **Giới thiệu ứng dụng di động:**

Ứng dụng di động, còn được gọi là ứng dụng di động hoặc ứng dụng di động, là phần mềm được thiết kế để chạy trên các thiết bị di động như điện thoại thông minh và máy tính bảng. Được xây dựng trên các nền tảng như iOS, Android hoặc Windows Phone, những ứng dụng này cung cấp một loạt các tính năng và dịch vụ từ giải trí đến công việc và học tập. Với sự phát triển của công nghệ di động, ứng dụng di động đã trở thành một phần không thể thiếu của cuộc sống hiện đại. Ứng dụng di động có thể phân loại theo mục đích sử dụng, bao gồm giải trí, mua sắm trực tuyến, mạng xã hội, sản phẩm và dịch vụ ngân hàng, y tế, giáo dục và nhiều hơn nữa. Ví dụ, các ứng dụng giải trí như YouTube, Netflix và TikTok cung cấp nội dung đa dạng từ video ngắn đến phim truyện và chương trình truyền hình. Các ứng dụng mua sắm như Amazon và eBay cho phép người dùng mua hàng trực tuyến và theo dõi đơn hàng của họ. Các ứng dụng mạng xã hội như Facebook, Instagram và Twitter giúp người dùng kết nối và chia sẻ thông tin với bạn bè và gia đình.

Ứng dụng di động cũng có thể được phân loại dựa trên công nghệ sử dụng. Có hai loại ứng dụng chính: ứng dụng di động dựa trên web và ứng dụng di động dựa trên nền tảng. Ứng dụng dựa trên web được thiết kế để chạy trên trình duyệt web của thiết bị di động và thường được xây dựng bằng các ngôn ngữ như HTML, CSS và JavaScript. Trong khi đó, ứng dụng dựa trên nền tảng được phát triển cho một nền tảng cụ thể như iOS hoặc Android bằng cách sử dụng các công nghệ và công cụ phát triển như Swift hoặc Objective-C cho iOS và Java hoặc Kotlin cho Android. Việc phát triển ứng dụng di động đòi hỏi kiến thức chuyên sâu về lập trình và thiết kế người dùng, cũng như kỹ năng thực hiện kiểm thử và triển khai. Các nhà phát triển ứng dụng cần phải hiểu rõ về các hệ thống điều hành di động, các giao diện lập trình ứng dụng (API), và các công cụ phát triển tích hợp (IDE) như Xcode cho iOS và Android Studio cho Android. Họ cũng cần phải có khả năng tương tác với các dịch vụ đám mây như AWS (Amazon Web Services) hoặc Firebase để lưu trữ dữ liệu và quản lý người dùng.

Ứng dụng di động không chỉ là công cụ giải trí và tiện ích mà còn là một phần của nền kinh tế kỹ thuật số ngày nay. Nhiều doanh nghiệp sử dụng ứng dụng di động như một cách để tiếp cận và tương tác với khách hàng của họ, cung cấp dịch vụ và sản phẩm thông qua nền tảng di động. Điều này đã tạo ra một thị trường phát triển mạnh mẽ cho các nhà phát triển ứng dụng và đã tạo ra hàng ngàn cơ hội việc làm trong ngành công nghiệp công nghệ thông tin.

* 1. **Giới thiệu lập trình ứng dụng di động:**

Lập trình ứng dụng di động là quá trình tạo ra các ứng dụng hoạt động trên các thiết bị di động như điện thoại thông minh và máy tính bảng. Việc phát triển ứng dụng di động đang trở thành một lĩnh vực ngày càng phát triển và phổ biến do sự gia tăng về số lượng người sử dụng thiết bị di động trên toàn cầu. Các ứng dụng di động cung cấp nhiều tính năng và dịch vụ khác nhau, từ giải trí, mạng xã hội đến công việc và giáo dục. Một phần quan trọng trong việc lập trình ứng dụng di động là việc sử dụng các ngôn ngữ lập trình và nền tảng phát triển phù hợp. Các ngôn ngữ lập trình phổ biến nhất cho việc phát triển ứng dụng di động bao gồm Java và Kotlin cho Android và Swift cho iOS. Ngoài ra, có các nền tảng phát triển như React Native và Flutter cho phép việc phát triển ứng dụng di động đa nền tảng, giúp tiết kiệm thời gian và công sức cho các nhà phát triển.

Quá trình lập trình ứng dụng di động bao gồm nhiều bước từ việc lên kế hoạch và thiết kế, viết mã, thử nghiệm và triển khai. Khi lập trình ứng dụng, các nhà phát triển cần cân nhắc về trải nghiệm người dùng, hiệu suất ứng dụng, bảo mật và tích hợp các tính năng mới như trí tuệ nhân tạo và thực tế ảo. Không chỉ là việc viết mã, lập trình ứng dụng di động cũng liên quan đến việc tối ưu hóa giao diện người dùng để đảm bảo sự thuận tiện và trải nghiệm người dùng tốt nhất. Điều này có thể bao gồm việc tối ưu hóa cho các kích thước màn hình khác nhau, cải thiện hiệu suất ứng dụng và tương tác người dùng một cách trơn tru.

Với sự phát triển không ngừng của công nghệ di động và nhu cầu ngày càng tăng về các ứng dụng di động, lập trình ứng dụng di động đang trở thành một lĩnh vực hấp dẫn và đầy tiềm năng cho các nhà phát triển. Điều này mở ra cơ hội cho việc sáng tạo và đổi mới trong việc tạo ra các ứng dụng có ảnh hưởng lớn đến cuộc sống hàng ngày và nền kinh tế.

**CHƯƠNG 2. CÁC PHẦN MỀM VÀ CÔNG CỤ SỬ DỤNG**

**2.1. Các phần mềm công cụ sử dụng:**

* Visual Studio Code.
* Firebase.
* Dark và Framework Flutter.
* Android Emulator.

**2.2. Giới thiệu về Dark – Flutter**

Đối với những nhà phát triển ứng dụng di động, việc chọn lựa một ngôn ngữ lập trình phù hợp là một phần quan trọng của quy trình phát triển. Trong thập kỷ gần đây, Flutter đã trở thành một lựa chọn phổ biến cho việc phát triển ứng dụng di động, nhất là so với các công nghệ khác như React Native hay native development. Để hiểu rõ hơn về cơ sở lý thuyết của Flutter, chúng ta sẽ so sánh nó với một số ngôn ngữ lập trình di động khác.



Flutter là một framework mã nguồn mở được phát triển bởi Google, cho phép nhà phát triển xây dựng ứng dụng di động và web từ một mã nguồn duy nhất. Ngôn ngữ chính trong Flutter là Dart, một ngôn ngữ lập trình cũng được Google phát triển. So với các ngôn ngữ lập trình di động khác như Java (cho Android) hoặc Swift (cho iOS), Dart được xem là một lựa chọn đầy tiềm năng nhờ vào các đặc tính sau:

* Tính nhất quán của ngôn ngữ: Dart là một ngôn ngữ có cú pháp dễ đọc và dễ hiểu. Điều này giúp cho các nhà phát triển dễ dàng học và triển khai ứng dụng một cách nhanh chóng. So với Java hay Objective-C, Dart có cú pháp gọn nhẹ hơn, giảm thiểu các đoạn mã không cần thiết.
* Sự ổn định và hiệu suất: Dart được thiết kế để tối ưu hiệu suất của ứng dụng, đặc biệt là trong việc xử lý các tác vụ đa luồng. So với JavaScript trong React Native, Dart thường cho hiệu suất cao hơn và ít xảy ra lỗi hơn. Điều này đặc biệt quan trọng đối với các ứng dụng yêu cầu độ tin cậy và độ trễ thấp.
* Widget-based UI development: Một trong những điểm mạnh của Flutter là sự linh hoạt trong phát triển giao diện người dùng. Flutter sử dụng một cấu trúc giao diện dựa trên các widget, cho phép nhà phát triển xây dựng giao diện một cách dễ dàng và tái sử dụng các thành phần. So với React Native, Flutter cung cấp một tập hợp đa dạng các widget được xây dựng sẵn, giúp cho việc phát triển UI nhanh chóng hơn và đồng nhất trên nhiều nền tảng.
* Hỗ trợ cộng đồng và tài liệu phong phú: Mặc dù Flutter và Dart là những công nghệ mới, nhưng họ đã nhận được sự quan tâm lớn từ cộng đồng phát triển. Điều này dẫn đến việc có một cộng đồng sôi nổi, nhiều tài liệu học tập và hỗ trợ từ các nhà phát triển khác. So với các công nghệ khác như Kotlin (cho Android) hay SwiftUI (cho iOS), Flutter có một cộng đồng phát triển đang phát triển mạnh mẽ, giúp cho việc giải quyết vấn đề và tìm kiếm giải pháp trở nên dễ dàng hơn.

A blue logo with a white background

Description automatically generated

Flutter và Dart đem lại nhiều lợi ích cho việc phát triển ứng dụng di động, từ tính nhất quán và hiệu suất đến sự linh hoạt trong phát triển giao diện người dùng và hỗ trợ từ cộng đồng. Mặc dù vẫn còn một số thách thức và hạn chế, nhưng sự phát triển nhanh chóng và sự quan tâm từ cộng đồng đã làm cho Flutter trở thành một lựa chọn hấp dẫn cho việc phát triển ứng dụng di động trong thời gian gần đây. Dưới đây là một số ưu điểm nối bật của Flutter:

* Đa nền tảng (Cross-platform): Flutter cho phép phát triển ứng dụng cho cả iOS và Android từ một mã nguồn duy nhất. Điều này giúp giảm thiểu thời gian và công sức cần thiết để phát triển và duy trì ứng dụng trên nhiều nền tảng.
* Hiệu suất cao: Flutter sử dụng một lớp giao diện người dùng riêng gọi là "widgets" và rendering engine của nó gọi là "Skia", giúp tối ưu hóa hiệu suất của ứng dụng. Điều này đảm bảo rằng ứng dụng Flutter chạy mượt mà và đáp ứng nhanh chóng ngay cả trên các thiết bị có cấu hình thấp.
* Giao diện người dùng tùy chỉnh: Flutter cung cấp một bộ công cụ mạnh mẽ để tạo và tùy chỉnh giao diện người dùng. Bạn có thể thiết kế giao diện đẹp mắt và linh hoạt bằng cách sử dụng các widget có sẵn hoặc tạo widget tùy chỉnh của riêng mình.
* Hot Reload: Hot Reload là một tính năng quan trọng trong Flutter, cho phép bạn thay đổi mã nguồn và xem kết quả ngay lập tức trên ứng dụng đang chạy mà không cần khởi động lại. Điều này giúp tăng tốc quá trình phát triển và giảm thời gian kiểm tra.
* Cộng đồng mạnh mẽ: Flutter có một cộng đồng rộng lớn, đam mê và nhiệt tình. Bạn có thể tìm thấy rất nhiều tài liệu, ví dụ và hỗ trợ từ cộng đồng này, giúp bạn giải quyết các vấn đề và tăng cường kỹ năng lập trình của mình.
* Dễ học và sử dụng: Với người mới bắt đầu, Flutter rất dễ học và sử dụng. Bạn chỉ cần kiến thức cơ bản về lập trình và một chút kiên nhẫn để bắt đầu phát triển ứng dụng của mình trong thời gian ngắn.

**2.3.Giới thiệu về Firebase**

**2.3.1. Lịch sử phát triển Firebase**

Gần một thập niên trước, Firebase ra đời với tiền thân là Envolve. Đây là một nền tảng đơn giản chuyên cung cấp những API cần thiết để tích hợp tính năng chat vào trang Web. Bên cạnh ứng dụng nhắn tin trực tuyến, Envolve còn được người dùng sử dụng để truyền và đồng bộ hóa dữ liệu cho những ứng dụng khác như các trò chơi trực tuyến. Do đó, các nhà sáng lập đã tách biệt hệ thống nhắn tin trực tuyến và đồng bộ dữ liệu thời gian thực thành hai phần riêng biệt.

Trên cơ sở đó, năm 2012, Firebase ra đời với sản phẩm cung cấp là dịch vụ Backend-as-a-Service. Tiếp đến, vào năm 2014, Google mua lại Firebase và phát triển nó thành một dịch vụ đa chức năng được hàng triệu người sử dụng cho đến hiện nay.



**2.3.1. Các tính năng chính của Firebase**

***Firebase Realtime Database:***

Khi đăng ký một tài khoản trên Firebase để tạo ứng dụng, bạn đã có một cơ sở dữ liệu thời gian thực. Dữ liệu bạn nhận được dưới dạng JSON. Đồng thời nó cũng luôn được đồng bộ thời gian thực đến mọi kết nối client. Đối với các ứng dụng đa nền tảng, tất cả các client đều sử dụng cùng một cơ sở dữ liệu. Nó được tự động cập nhật dữ liệu mới nhất bất cứ khi nào các lập trình viên phát triển ứng dụng. Cuối cùng, tất cả các dữ liệu này được truyền qua kết nối an toàn SSL có bảo mật với chứng nhận 2048 bit.

Trong trường hợp bị mất mạng, dữ liệu được lưu lại ở local, vì thế khi có mọi sự thay đổi nào đều được tự động cập nhật lên Server của Firebase. Bên cạnh đó, đối với các dữ liệu ở local cũ hơn với Server thì cũng tự động cập nhật để được dữ liệu mới nhất.

***Firebase Authentication:***

Hoạt động nổi bật của Firebase là xây dựng các bước xác thực người dùng bằng Nhóm email, Facebook, Twitter, GitHub, Google. Đồng thời cũng xác thực nặc danh cho các ứng dụng. Hoạt động xác thực có thể giúp thông tin cá nhân của người sử dụng được an toàn và đảm bảo không bị đánh cắp tài khoản

***Firebase Authentication:***

Firebase cũng cung cấp Cloud Firestore, một dịch vụ cơ sở dữ liệu dựa trên tài liệu (document-based) mới hơn. Firestore cung cấp hiệu suất tốt hơn và tính linh hoạt cao hơn so với Realtime Database, bao gồm các tính năng như truy vấn phức tạp và tự động đồng bộ hóa dữ liệu giữa các thiết bị của người dùng.

***Firebase Storage:***

Firebase còn cung cấp các dịch vụ như Firebase Storage để lưu trữ tệp tin như hình ảnh và video, Firebase Cloud Messaging để gửi thông báo đẩy tới người dùng, Firebase Hosting để đăng ký và quản lý tên miền, cũng như một loạt các công cụ phân tích và tích hợp với Google Analytics.

**CHƯƠNG 3. KHẢO SÁT HIỆN TRẠNG**

**3.1. Khảo sát hiện trạng:**

**tạo mẫu khảo sát : google form, tạo các mục: chi tiêu như thế, bạn quản lý chi tiêu…**

**chụp hình google form: nghiên cứu điều tra dữ liệu.**

**-> hầu hết sv trong diện khảo sát chưa có kết hoạch c hi tiêu hợp lý, dựa vào khảo sát tôi sẽ tiến hành xây dựng app quản lý tài chính cá nhân có các chức năng sau/;**

**CHƯƠNG 4. PHÂN TÍCH HỆ THỐNG**

**4.1. Phân tích hệ thống:**

**phân tích non sql: tìm hiểu**

**CHƯƠNG 5. THIẾT KẾ DỮ LIỆU**

**3.1. Thiết kế dữ liệu:**

**CHương 3: phân tích thiết kế ứng dụng**

**UI, UX,....**

**CHương 4 Xây dựng ứng dụng**

**CHương 5: Kết luận**

**Tài liệu tham khảo**