

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**KHOA CÔNG NGHỆ THÔNG TIN**

-----🙞🙜🕮🙞🙜-----



**BÁO CÁO THỰC NGHIỆM**

**PHÁT TRIỂN ỨNG DỤNG TRÊN THIẾT BỊ DI ĐỘNG**

**XÂY DỰNG ỨNG DỤNG ANDROID KLNEWS ĐỂ QUẢN LÝ ĐỌC TIN TỨC TRÊN INTERNET**

|  |  |
| --- | --- |
| Giáo viên hướng dẫn: | TS. Hà Mạnh Đào |
| Lớp: | 20231IT6029001 |
| Thành viên: | Nguyễn Duy Linh Nguyễn Thị Huyền Trang  Nguyễn Lam Trường |

**Hà Nội, 2024**

# MỤC LỤC

[MỤC LỤC 2](#_Toc155317079)

[DANH MỤC HÌNH ẢNH 4](#_Toc155317080)

[DANH MỤC BẢNG BIỂU 6](#_Toc155317081)

[LỜI CẢM ƠN 7](#_Toc155317082)

[MỞ ĐẦU 8](#_Toc155317083)

[CHƯƠNG 1. TỔNG QUAN VỀ NỘI DUNG NGHIÊN CỨU 9](#_Toc155317084)

[1.1. Giới thiệu chung 9](#_Toc155317085)

[1.2. Cơ sở lý thuyết 9](#_Toc155317086)

[1.2.1. Tổng quan về lập trình di động 9](#_Toc155317087)

[1.2.1.1. Hybrid App 9](#_Toc155317088)

[1.2.1.2. Native App 10](#_Toc155317089)

[1.2.1.3. Web App 10](#_Toc155317090)

[1.2.2 Khái niệm về ResAPI và cách thức hoạt động 11](#_Toc155317091)

[1.2.2.1 Khái niệm 11](#_Toc155317092)

[1.2.2.2 Cách thức hoạt động 12](#_Toc155317093)

[1.2.2.3 Ưu điểm 12](#_Toc155317094)

[1.2.3 Khái niệm Restrofit và cách thực hoạt động 13](#_Toc155317095)

[1.2.3.1 Khái niệm 13](#_Toc155317096)

[1.2.3.2 Cách thức hoạt động 13](#_Toc155317097)

[1.2.3.3 Ưu điểm 14](#_Toc155317098)

[CHƯƠNG 2. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 16](#_Toc155317099)

[2.1. Yêu cầu về chức năng 16](#_Toc155317100)

[2.2. Yêu cầu phi chức năng 16](#_Toc155317101)

[2.3. Thiết kế hệ thống 17](#_Toc155317102)

[2.3.1. Mô hình hóa Usecase 17](#_Toc155317103)

[2.3.1.1. Biểu đồ Usecase tổng quan 17](#_Toc155317104)

[2.3.1.3.Biểu đồ Use case người quản trị 19](#_Toc155317105)

[2.3.2. Dữ liệu 20](#_Toc155317106)

[2.4. Thiết kế hệ thống 21](#_Toc155317107)

[2.4.1.Mô tả chi tiết các use case: 21](#_Toc155317108)

[2.4.1.1.Mô tả use case Xem tin tức: 21](#_Toc155317109)

[2.4.1.2 Mô tả use case Xem thông tin cá nhân 23](#_Toc155317110)

[2.4.1.3.Mô tả use case Chỉnh sửa thông tin cá nhân 24](#_Toc155317111)

[2.4.1.4 Mô tả use case Xem tin tức theo danh mục 25](#_Toc155317112)

[2.4.1.5 Mô tả use case Đăng nhập 26](#_Toc155317113)

[2.1.4.6 Mô tả use case Đăng ký 27](#_Toc155317114)

[2.4.1.7 Mô tả use case Bảo trì danh mục 28](#_Toc155317115)

[2.4.1.8 Mô tả use case Bảo trì tài khoản 30](#_Toc155317116)

[2.4.1.9 Mô tả use case Bảo trì tin tức 32](#_Toc155317117)

[2.4.2 Biểu đồ phân tích Use case 34](#_Toc155317118)

[2.4.2.1 Use case tìm kiếm tin tức 34](#_Toc155317119)

[2.4.2.2.Use case Xem tin tức nổi bật 35](#_Toc155317120)

[2.4.2.3 Use case xem tin tức theo danh mục 36](#_Toc155317121)

[2.4.2.4 Use case xem chi tiết tin tức 37](#_Toc155317122)

[2.4.2.5 Use case xem thông tin cá nhân 38](#_Toc155317123)

[2.4.2.6.Use case chỉnh sửa thông tin cá nhân 39](#_Toc155317124)

[2.4.2.7. Use case đăng nhập 40](#_Toc155317125)

[2.4.2.8. Use case 41](#_Toc155317126)

[2.4.2.9. Use case bảo trì tài khoản 43](#_Toc155317127)

[2.4.2.10. Use case bảo trì bình luận 45](#_Toc155317128)

[2.4.2.11. Use case bảo trì danh mục 47](#_Toc155317129)

[2.4.2.12. Use case bảo trì tin tức 48](#_Toc155317130)

[2.4.3. Biểu đồ thực thể liên kết 50](#_Toc155317131)

[CHƯƠNG 3. KẾT QUẢ 51](#_Toc155317132)

[3.1. Kết quả đạt được 51](#_Toc155317133)

[3.2. Kết quả kiểm thử chức năng người quản trị 58](#_Toc155317134)

[3.2. Kiểm thử chức năng người quản trị 59](#_Toc155317135)

[KẾT LUẬN 60](#_Toc155317136)

[TÀI LIỆU THAM KHẢO 61](#_Toc155317137)

# DANH MỤC HÌNH ẢNH

[Hình 2. 1 : Biểu đồ usecase tổng quan 17](#_Toc155316619)

[Hình 2. 2: Biểu đồ usecase người dùng 18](#_Toc155316620)

[Hình 2. 3:Biểu đồ usecase người quản trị 19](#_Toc155316621)

[Hình 2. 4: Biểu đồ trình tự tìm kiếm tin tức 34](#_Toc155316622)

[Hình 2. 5: Biểu đồ phân tích usecase tìm kiếm tin tức 35](#_Toc155316623)

[Hình 2. 6: Biểu đồ tình tự xem tin tức nổi bật 35](#_Toc155316624)

[Hình 2. 7: Biểu đồ phân tích xem tin tức nổi bật 36](#_Toc155316625)

[Hình 2. 8: Biểu đồ trình tự xem tin tức theo danh mục 36](#_Toc155316626)

[Hình 2. 9: Biểu đồ phân tích xem tin tức theo danh mục 37](#_Toc155316627)

[Hình 2. 10: Biểu đồ trình tự xem chi tiết tin tức 37](#_Toc155316628)

[Hình 2. 11: Biểu đồ phân tích xem chi tiết tin tức 38](#_Toc155316629)

[Hình 2. 12: Biểu đồ trình tự xem thông tin cá nhân 38](#_Toc155316630)

[Hình 2. 13: Biểu đồ phân tích xem thông tin cá nhân 39](#_Toc155316631)

[Hình 2. 14: Biểu đồ trình tự chỉnh sửa thông tin cá nhân 39](#_Toc155316632)

[Hình 2. 15: Biểu đồ phân tích chỉnh sửa thông tin cá nhân 40](#_Toc155316633)

[Hình 2. 16: Biểu đồ trình tự đăng nhập 40](#_Toc155316634)

[Hình 2. 17: Biểu đồ phân tích đăng nhập 41](#_Toc155316635)

[Hình 2. 18: Biểu đồ trình tự đăng ký 41](#_Toc155316636)

[Hình 2. 19: Biểu đồ phân tích đăng ký 42](#_Toc155316637)

[Hình 2. 20: Biểu đồ trình tự bảo trì tài khoản 44](#_Toc155316638)

[Hình 2. 21: Biểu đồ phân tích bảo trì tài khoản 44](#_Toc155316639)

[Hình 2. 22: Biểu đồ trình tự bảo trì bình luận 45](#_Toc155316640)

[Hình 2. 23: Biểu đồ phân tích bảo trì bình luận 46](#_Toc155316641)

[Hình 2. 24: Biểu đồ trình tự bảo trì danh mục 47](#_Toc155316642)

[Hình 2. 25: Biểu đồ phân tích bảo trì danh mục 47](#_Toc155316643)

[Hình 2. 26: Biểu đồ trình tự bảo trì tin tức 49](#_Toc155316644)

[Hình 2. 27: Biểu đồ phân tích bảo trì tin tức 50](#_Toc155316645)

[Hình 2. 28:Biểu đồ thực thể liên kết mức vật lý 50](#_Toc155316646)

[Hình 3. 1: Xem chi tiết bài viết 51](#_Toc155316672)

[Hình 3. 2: Đăng nhập 52](#_Toc155316673)

[Hình 3. 3: Đăng ký 53](#_Toc155316674)

[Hình 3. 4: Tìm kiếm 54](#_Toc155316675)

[Hình 3. 5: Xem danh mục 55](#_Toc155316676)

[Hình 3. 6: Xem danh mục con 56](#_Toc155316677)

[Hình 3. 7: Xem bài viết theo danh mục 57](#_Toc155316678)

# DANH MỤC BẢNG BIỂU

[Bảng 2. 1 Bảng yêu cầu chức năng 16](#_Toc155316775)

[Bảng 2. 2 Bảng dữ liệu người dùng 20](#_Toc155316776)

[Bảng 2. 3: Bảng dữ liệu danh mục 20](#_Toc155316777)

[Bảng 2. 4: Bảng dữ liệu tin tức 21](#_Toc155316778)

[Bảng 2. 5: Mô tả usecase xem chi tiết tin tức 21](#_Toc155316779)

[Bảng 2. 6: Mô tả usecase xem tin tức nổi bật 22](#_Toc155316780)

[Bảng 2. 7: Mô tả usecase xem thông tin cá nhân 23](#_Toc155316781)

[Bảng 2. 8: Mô tả usecase chỉnh sửa thông tin cá nhân 24](#_Toc155316782)

[Bảng 2. 9: Bảng mô tả usecase xem tin tức theo danh mục 25](#_Toc155316783)

[Bảng 2. 10: Mô tả usecase đăng nhập 26](#_Toc155316784)

[Bảng 2. 11: Mô tả usecase Đăng ký 27](#_Toc155316785)

[Bảng 2. 12: Mô tả usecase bảo trì danh mục 28](#_Toc155316786)

[Bảng 2. 13: Mô tả usecase bảo trì tài khoản 30](#_Toc155316787)

[Bảng 2. 14: Mô tả usecase bảo trì tin tức 32](#_Toc155316788)

# LỜI CẢM ƠN

Trong suốt thời gian học tập và thực hiện bài tập lớn em luôn nhận được sự hướng dẫn nhiệt tình từ phía thầy giáo. Vì thế, lời đầu tiên em xin chân thành cảm ơn thầy **TS. Hà Mạnh Đào** đã tạo điều kiện thuận lợi và truyền dạy kiến thức cho em trong thời gian qua để em hoàn thành đề tài một cách tốt nhất. Và cảm ơn thầy đã tận tình giúp đỡ, định hướng nhóm chúng em rất nhiều trong suốt quá trình tìm hiểu nghiên cứu và hoàn thành bài tập lớn. Khi thực hiện nghiên cứu đề tài, nhóm chúng em đã cố gắng hoàn thành tốt nhất báo cáo bài tập lớn. Tuy nhiên kiến thức chuyên ngành của nhóm chúng em còn nhiều hạn chế. Vì vậy trong báo cáo không tránh khỏi những thiếu sót, nhóm chúng em rất mong nhận được sự đóng góp của thầy giáo để bài báo cáo của nhóm chúng em được đầy đủ và hoàn chỉnh hơn.

Nhóm 12 xin chân thành cảm ơn!

# MỞ ĐẦU

**1. Lý do chọn đề tài**

Trong thời đại thông tin ngày nay thì việc update tin tức nhanh và liên tục là vô cùng cần thiết. App tin tức là một giải pháp hiệu quả giúp cập nhật thông tin đến người tiêu dùng một cách dễ dàng và nhanh chóng.

So với việc mua báo giấy hay theo dõi tin tức trên tivi thì app tin tức có thể cung cấp thông tin mới nhất một cách chính xác và nhanh chóng. Ngoài ra, người dùng cũng có thể lựa chọn những chủ đề tin tức mà họ thích và cập nhật thông tin một cách nhanh chóng.

Với sự bùng nổ của công nghệ, app tin tức càng trở nên thịnh hành và được mọi người ưa thích. Tuy nhiên, muốn có một app tin tức chất lượng cao và hấp dẫn người sử dụng thì cần phải có một quá trình nghiên cứu và phát triển lâu dài.

**2. Đối tượng, mục tiêu và phạm vi nghiên cứu**

a. Mục tiêu nghiên cứu

- Phân tích, hiểu bài toán nghiệp vụ thực tế.

- Tìm hiểu công nghệ xây dựng trên nền tảng android.

- Xây dựng được một ứng dụng đọc tin tức KLNEWS

- Về phía người dùng, thao tác đơn giản dễ sử dụng.

b. Đối tượng nghiên cứu

- Ngôn ngữ lập trình Android (java)

**3. Phương pháp nghiên cứu**

- Phương pháp nghiên cứu về mặt lý thuyết:

▪ Nghiên cứu và tiến hành thu thập các tài liệu có liên quan đến đề tài.

▪ Tổng hợp các tài liệu đã thu thập và tiến hành phân tích.

▪ Chọn lọc các tài liệu nghiên cứu để báo cáo bài tập lớn.

- Phương pháp nghiên cứu trong thực nghiệm:

▪ Tìm hiểu cách thức, tiêu chí kiểm thử chức năng.

▪ Phân tích các ứng dụng có chức năng tương tự.

▪ Phân tích và tiến hành xây dựng ứng dụng theo phân tích và yêu cầu

# CHƯƠNG 1. TỔNG QUAN VỀ NỘI DUNG NGHIÊN CỨU

## 1.1. Giới thiệu chung

Trong thời đại thông tin ngày nay thì việc update tin tức nhanh và liên tục là vô cùng cần thiết. App tin tức là một giải pháp hiệu quả giúp cập nhật thông tin đến người tiêu dùng một cách dễ dàng và nhanh chóng.

So với việc mua báo giấy hay theo dõi tin tức trên tivi thì app tin tức có thể cung cấp thông tin mới nhất một cách chính xác và nhanh chóng. Ngoài ra, người dùng cũng có thể lựa chọn những chủ đề tin tức mà họ thích và cập nhật thông tin một cách nhanh chóng.

Với sự bùng nổ của công nghệ, app tin tức càng trở nên thịnh hành và được mọi người ưa thích. Tuy nhiên, muốn có một app tin tức chất lượng cao và hấp dẫn người sử dụng thì cần phải có một quá trình nghiên cứu và phát triển lâu dài.

## **1.2. Cơ sở lý thuyết**

### 1.2.1. Tổng quan về lập trình di động

Hiện nay có 3 hướng chính xây dựng và phát triển 1 ứng dụng di động là: Native app, Web App và Hybrid app. Mỗi hướng đều có ưu và nhược điểm và kỹ năng riêng.

1.2.1.1. Hybrid App

Hybrid App là ứng dụng kết hợp của cả Mobile Web App và Native App.

Theo đó, các phần cơ bản của ứng dụng vẫn viết bằng ngôn ngữ web, nhưng được đặt trong native container, nên vẫn có thể đưa lên các App Store.

• Ưu điểm:

+ Chỉ cần có kiến thức về HTML, CSS, JavaScript …

+ Viết một lần dùng được nhiều nơi.

+ Tận dụng được các chức năng của hệ thống.

+ Có thể chạy được ở chế độ offline

• Nhược điểm:

+ Hiệu năng chậm

+ Không ổn định do khó debug. Framework sẽ dịch code thành code native sửa lỗi khá khó khăn không biết được dịch như thế nào

#### 1.2.1.2. Native App

Native App là những ứng dụng được viết riêng cho một loại nền tảng như : iOS, Android, Windows Phone bằng các ngôn ngữ tương ứng của mọi nền tảng đó. Ví dụ Java trên Android, Object C trên iOS. Mỗi Native App chỉ chạy được trên một nền tảng và không thể mang sang các nền tảng khác

• Ưu điểm:

+ Tận dụng được tính năng có sẵn trên thiết bị như: GPS, Camera, thiết bị thu âm ...

+ Hiệu năng cao vì code native chạy trực tiếp trên máy

+ Có thế chạy được ở chế độ offline

• Nhược điểm:

+ Không thể kết hợp nhiều nền tảng. Mỗi một ứng dụng chỉ chạy trên 1 nền tảng nhất định

+ Mỗi hệ điều hành cân phải viết ứng dụng riêng khó đồng bộ giữa các ứng dụng

+ Việc bảo trì hay nâng cấp sẽ làm mất nhiều thời gian. Do phải sửa chữa từng app trên từng hệ điều hành

+ Xây dựng ứng dụng cần dùng các phần mềm riêng biệt theo hệ điều hành (dùng Xcode trên Mac để phát triển ứng dụng IOS, Android Studio để phát triển ứng dụng Android)

#### 1.2.1.3. Web App

Web app được phát triển khi đã có sẵn một website đang hoạt động. Ta tạo thêm một trang web riêng cho các thiết bị di động sử dụng HTML, CSS và một số thư viện khác hỗ trợ.

Web app được thiết kế chạy trên nền tảng web hoặc các trình duyệt của thiết bị di động cho phép người dùng thao tác như thao tác trang web giống nhu thao tác ứng dụng.

• Ưu điểm:

+ Có thể chạy trên tất cả trình duyệt của mobile hỗ trợ phiên bản

HTML và javascript

+ Không cần cài đặt trên máy miễn là máy có trình duyệt web

+ Một phiên bản duy nhất cho tất cả, nên giảm chi phí và thời gian cho phát triển, bảo trì, cũng như nâng cấp sau này.

+ Phiên bản được cập nhật liên tục không cần phải cập nhật trên chợ

• Nhược điểm:

+ Hiệu năng không được tốt như native app và luôn phải chạy online.

+ Không thể dùng được các tính năng tích hợp của di động:

Notification, chụp hình, nghiêng máy, định vị GPS, các sensor…

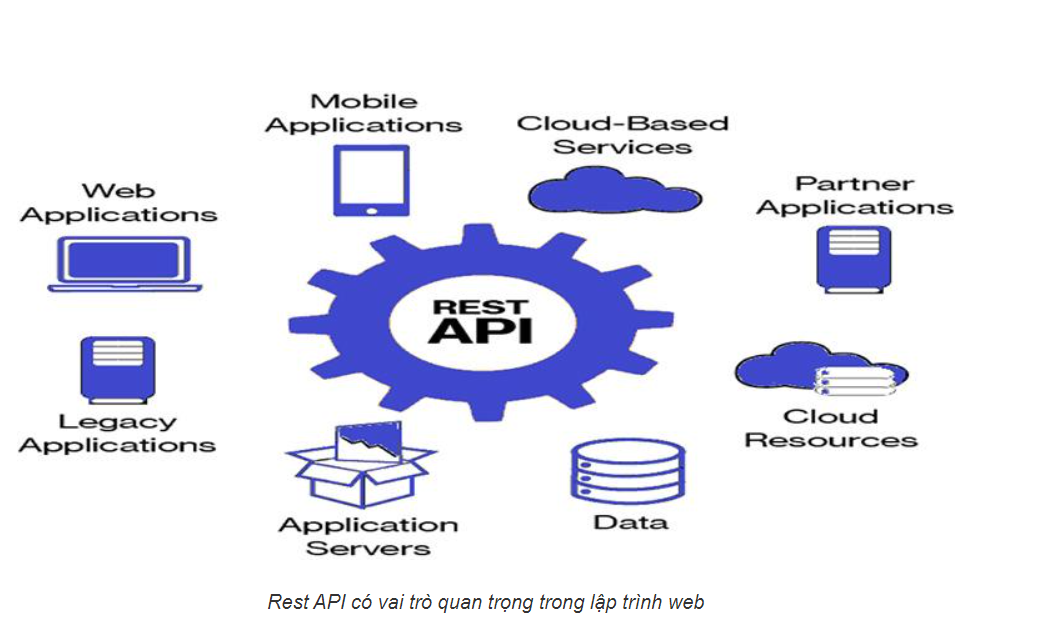
+Với một số máy đời cũ, Web app sẽ bị vỡ giao diện, hiển thị sai,

hoặc javascript không chạy

### 1.2.2 Khái niệm về ResAPI và cách thức hoạt động

#### 1.2.2.1 Khái niệm

Rest API là một ứng dụng được dùng để chuyển đổi cấu trúc của dữ liệu có những phương thức giúp kết nối với các thư viện và ứng dụng khác nhau. Rest Api thường không được xem là công nghệ, mà nó là giải pháp giúp tạo ra những ứng dụng web services chuyên dụng để thay thế cho nhiều kiểu khác như: SOAP, WSDL,…



#### 1.2.2.2 Cách thức hoạt động

Tương tự với các giao thức truyền thông hoặc các cấu trúc dữ liệu khác thì: để bạn có thể tìm hiểu được bản chất của vấn đề thì đầu tiên việc hiểu rõ nguyên lý hoạt động của nó vô cần thiết.

REST thường hoạt động chủ yếu dựa vào những giao thức HTTP, các cơ sở hoạt động cơ bản nêu trên sẽ được sử dụng những phương thức HTTP riêng biệt:

* GET: Trả về với một Resource hoặc có một danh sách Resource.
* POST: Nó hỗ trợ tạo mới một Resource.
* PUT: Thường hỗ trợ cho việc cập nhật các thông tin cho Resource.
* DELETE: Xóa một Resource.

Đây là phương thức hoạt động được gọi là Crud tương ứng với một Create, Read, Update, Delete ý nghĩa theo thứ tự là: Tạo, đọc, sửa và xóa.Cho đến hiện tại, thì đa số các lập trình viên viết Restful API đều lựa chọn JSON làm format chính thức, tuy nhiên vẫn còn nhiều người lựa chọn XML.

#### 1.2.2.3 Ưu điểm

Việc sử dụng Restful api mang lại vô và hiệu quả nhất định cho các lập trình viên. Vậy thì, Restful API mang lại những lợi ích gì?

* Giúp ứng dụng trông rõ ràng và dễ nhìn hơn.
* Rest URL còn là đại diện cho một resource chứ không phải là một hành động.
* Các dữ liệu thường sẽ được trả về dưới nhiều định dạng khác nhau bao gồm: xml, html hoặc json,...
* Code của Rest APi vô cùng đơn giản và ngắn gọn.
* REST thường cần chú trọng rất nhiều vào tài nguyên của các hệ thống. vận chuyển hàng không quốc tế

Hiện nay, có rất nhiều trang web sử dụng REST API để cho phép việc kết nối dữ liệu từ họ dễ dàng hơn. Và trong số đó, thì facebook cũng đang cung cấp cho các Rest API để ứng dụng bên ngoài vẫn có thể kết nối đến với dữ liệu của họ.

### 1.2.3 Khái niệm Restrofit và cách thực hoạt động

#### 1.2.3.1 Khái niệm

Retrofit là một khái niệm thường được sử dụng trong ngữ cảnh của kỹ thuật và xây dựng, đặc biệt là khi nâng cấp, cải tiến hoặc làm mới một hệ thống, thiết bị, hoặc công trình để đáp ứng các yêu cầu mới hoặc tiêu chuẩn hiện đại hóa.

Trong ngữ cảnh của phần mềm và công nghệ thông tin, Retrofit là một thư viện trong lập trình Android, được sử dụng để gửi các yêu cầu mạng và xử lý các phản hồi từ máy chủ. Retrofit giúp đơn giản hóa việc tương tác với các API (Application Programming Interface) và giúp các nhà phát triển xây dựng ứng dụng Android một cách hiệu quả và dễ dàng hơn.

#### 1.2.3.2 Cách thức hoạt động

Cách thức hoạt động của Retrofit trong ngữ cảnh Android:

- Định nghĩa API:

* Xác định các API endpoint mà ứng dụng của bạn sẽ tương tác.
* Xác định cấu trúc dữ liệu của các yêu cầu và phản hồi.

- Tạo Interface API:

* Sử dụng Retrofit để tạo một interface đại diện cho API, định nghĩa các phương thức tương ứng với các yêu cầu mạng (GET, POST, PUT, DELETE, v.v.).
* Sử dụng các annotation để đánh dấu các phần của yêu cầu (ví dụ: URL endpoint, tham số, phần thân của yêu cầu).

- Khởi tạo Retrofit:

* Tạo một đối tượng Retrofit thông qua Builder Pattern, cấu hình cơ bản như URL cơ sở, bộ chuyển đổi (converter) dữ liệu (JSON, XML, v.v.).

- Tạo đối tượng API Service:

* Sử dụng Retrofit để tạo một đối tượng thực thi của interface API.

- Thực hiện các yêu cầu mạng:

* Gọi các phương thức trên đối tượng API Service để thực hiện các yêu cầu mạng.
* Retrofit sẽ tự động xử lý các yêu cầu và phản hồi, sử dụng các bộ chuyển đổi để chuyển đổi dữ liệu giữa định dạng máy chủ và đối tượng trong ứng dụng.

- Xử lý kết quả:

* Xử lý kết quả trả về từ máy chủ trong các phương thức gọi API, thường thông qua callback hoặc sử dụng reactive programming (RxJava, LiveData).

#### 1.2.3.3 Ưu điểm

Retrofit giúp giảm độ phức tạp của việc xử lý các yêu cầu mạng và giúp ứng dụng Android tương tác với các dịch vụ web một cách dễ dàng và hiệu quả.

Retrofit là một thư viện phổ biến trong lập trình Android với nhiều ưu điểm nổi bật giúp đơn giản hóa việc giao tiếp mạng trong ứng dụng. Dưới đây là một số ưu điểm chính của Retrofit:

- Dễ sử dụng:

* Retrofit cung cấp một cách tiếp cận đơn giản và dễ hiểu để tương tác với API. Việc sử dụng các annotation giúp giảm độ phức tạp của mã nguồn.

- Đọc mã dễ hiểu:

* Cú pháp của Retrofit rất rõ ràng và dễ đọc. Các annotation như `@GET`, `@POST`, `@Query`, `@Body` giúp định rõ mục đích của từng phần mã nguồn.

- Tích hợp tốt với Gson:

* Retrofit tích hợp mạnh mẽ với thư viện Gson để tự động chuyển đổi dữ liệu JSON thành các đối tượng Java và ngược lại. Điều này giúp giảm bớt công việc lặp lại và giữ cho mã nguồn sạch sẽ.

- Hỗ trợ các bộ chuyển đổi (Converters):

* Retrofit hỗ trợ nhiều bộ chuyển đổi (converters) mặc định và có thể được mở rộng. Điều này cho phép ứng dụng tương tác với các định dạng dữ liệu khác nhau như JSON, XML, hoặc các định dạng tùy chỉnh.

- Quản lý trạng thái và lỗi:

* Retrofit giúp tự động xử lý mã trạng thái HTTP và quản lý lỗi trong quá trình thực hiện yêu cầu mạng. Điều này giúp giảm độ phức tạp khi xử lý lỗi.

- Hỗ trợ RxJava:

* Retrofit tích hợp tốt với RxJava, một thư viện lập trình không đồng bộ trong Java, giúp xử lý các yêu cầu mạng và phản hồi một cách linh hoạt và hiệu quả.

- Tiết kiệm thời gian và công sức:

* Sử dụng Retrofit giúp giảm độ phức tạp của việc giao tiếp mạng, giúp nhà phát triển tiết kiệm thời gian và công sức, đồng thời giữ cho mã nguồn dễ bảo trì.

- Tương thích với OkHttp:

* Retrofit được xây dựng trên nền tảng thư viện OkHttp, cho phép tận dụng những lợi ích của OkHttp như kết nối không đồng bộ, quản lý kết nối, và cơ sở hạ tầng mạng mạnh mẽ.

Những ưu điểm này khiến cho Retrofit trở thành một lựa chọn phổ biến cho việc thực hiện giao tiếp mạng trong ứng dụng Android.

# CHƯƠNG 2. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

## 2.1. Yêu cầu về chức năng

➢ Các chức năng chính của hệ thống:

Bảng 2. 1 Bảng yêu cầu chức năng

|  |  |
| --- | --- |
| Người dùng (Chưa đăng nhập) | Đăng nhập  Xem chi tiết tin tức  Đăng ký  Xem tin tức theo danh mục  Xem tin tức nổi bật |
| Người dùng | Xem chi tiết tin tức  Xem thông tin cá nhân  Chính sửa thông tin cá nhân  Xem tin tức đã lưu  Bình luận |
| Admin | Đăng nhập  Đăng ký  Bảo trì tin tức  Bảo trì danh mục  Bảo trì tài khoản  Thống kê số lượng bài viết |

## 2.2. Yêu cầu phi chức năng

Việc tra cứu đòi hỏi phải chuẩn xác.

Thông tin về sản phẩm, về nhà cung cấp,... cần phải lưu giữ định kì đề phòng những trục trặc bất ngờ xảy ra.

Chương trình phải chặn sự truy cập bất hợp pháp của người dùng nhằm phòng tránh những nguy cơ hỏng máy diễn ra.

Chương trình phải có chức năng dễ dàng cập nhật, nâng cấp hoặc chỉnh sửa khi cần nâng cấp hoặc cập nhật các chức năng mới.

Website phải được thiết kế làm sao cho dễ đọc, giao diện dễ dùng, bắt mắt và giúp cho người dùng tìm thấy các thông tin cần tìm kiếm, hoặc các thông tin quảng cáo thật hấp dẫn, để lôi cuốn sự chú ý của người dùng.

Ngôn ngữ phù hợp, thân thiện với người dùng.

## 2.3. Thiết kế hệ thống

### 2.3.1. Mô hình hóa Usecase

#### 2.3.1.1. Biểu đồ Usecase tổng quan



Hình 2. 1 : Biểu đồ usecase tổng quan

2.3.1.2. Biểu đồ Usecase người dùng



Hình 2. 2: Biểu đồ usecase người dùng

- Đăng nhập: Use case này cho phép người dùng đăng nhập vào hệ thống.

- Đăng ký: Use case này cho phép người dùng đăng ký tài khoản vào hệ thống.

- Xem chi tiết tin tức: Use case này cho phép người dùng xem chi tiết một tin tức.

- Xem tin tức theo danh mục: Use case này cho phép người dùng xem tin tức theo danh mục được chọn.

- Xem tin tức nổi bật: Use case này cho phép người dùng xem tin tức nổi bật của trang web

- Xem thông tin cá nhân: Use case này cho phép người dùng xem thông tin cá nhân

- Chính sửa thông tin cá nhân: Use case này cho phép người dùng chỉnh sửa thông tin các nhân.

- Xem tin tức đã lưu: Use case này cho phép người dùng xem các tin mình đã lưu.

- Bình luận: Use case này cho phép người dùng bình luận vào tin tức.

#### 2.3.1.3.Biểu đồ Use case người quản trị



Hình 2. 3:Biểu đồ usecase người quản trị

- Đăng nhập: Use case này cho phép người quản trị đăng nhập vào hệ thống quản trị.

- Đăng ký: Use case này cho phép người quản trị đăng ký vào hệ thống quản trị

- Bảo trì tin tức: Use case này cho phép người quản trị bảo trì tin tức với các chức năng thêm, sửa, xoá tin tức.

- Bảo trì danh mục: Use case này cho phép người quản trị bảo trì các danh mục tin tức với các chức năng thêm, sửa, xoá danh mục.

- Bảo trì tài khoản: Use case này cho phép người quản trị bảo trì tài khoản người dùng với các chức năng thêm, sửa, xoá tài khoán.

- Thống kê số lượng bài viết: Use case này cho phép người quản trị tài khoản xem số liệu thống kê có bao nhiêu bài viết.

### 2.3.2. Dữ liệu

Bảng 2. 2 Bảng dữ liệu người dùng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TÀI KHOẢN | | | | |
| Tên trường | Kiểu dữ liệu | Kích thước | Ràng buộc | Mô tả |
| maTaiKhoan | Bigint | 20 | Khóa chính | Mã tài khoản người dùng |
| tenTaiKhoan | Varchar | 20 | Not null | Tên tài khoản người dùng |
| hoTen | Char | 20 | Not null | Họ và tên người dùng |
| matKhau | Char | 20 | Not null | Mật khẩu đăng nhập |
| email | Char | 20 | Not null | Địa chỉ email đăng kí/đặng nhập |
| ngaySinh | Date |  | Not null | Ngày sinh người dùng |
| gioiTinh | Bool |  | Not null | Giới tính người dùng |
| soDienThoai | Varchar | 20 | Not null | Số điện thoại người dùng |
| diaChi | Char | 20 | Not null | Địa chỉ người dùng |
| anhDaidien | Varchar | 20 | Not null | Ảnh đại diện cho người dùng |

Bảng 2. 3: Bảng dữ liệu danh mục

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DANH MỤC | | | | |
| Tên trường | Kiểu dữ liệu | Kích thước | Ràng buộc | Mô tả |
| maDanhMuc | Bigint | 20 | Khóa chính | Mã dạnh mục tin tức |
| tenDanhMuc | Char | 20 | Not null | Tên dạnh mục tin tức |

Bảng 2. 4: Bảng dữ liệu tin tức

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TIN TỨC | | | | |
| Tên trường | Kiểu dữ liệu | Kích thước | Ràng buộc | Mô tả |
| maTinTuc | Bigint | 20 | Khóa chính | Mã tin tức |
| tenTinTuc | Varchar | 20 | Not null | Tên tin tức |
| tieuDe | Char | 20 | Not null | Tiêu đề của tin tức |
| noiDung | Varchar | 20 | Not null | Nội dung tin tức |
| image | Varchar | 20 | Not null | Ảnh đại diện cho người dùng |
| view | Interger |  | Not null | Số lượt xem |
| author | Varchar | 20 | Not null | Tác giả bài đăng |

## 2.4. Thiết kế hệ thống

### 2.4.1.Mô tả chi tiết các use case:

#### 2.4.1.1.Mô tả use case Xem tin tức:

Bảng 2. 5: Mô tả usecase xem chi tiết tin tức

|  |  |
| --- | --- |
| **Mô tả vắn tắt** | Use case này cho phép người dùng xem tin tức. |
| **Đối tượng** | Người dùng |
| **Tiền điều kiện** | Không có |
| **Luồng cơ bản** | 1. Use case này bắt đầu khi người dùng kích vào mục “Tin tức” trên thanh menu. Hệ thống sẽ lấy tin tức, tiêu đề, ảnh minh hoạ, nội dung của các tin tức từ bảng TINTUC trong cơ sở dữ liệu và hiển thị lên màn hình. 2. Người dùng kích vào ảnh tin tức hoặc tiêu đề tin tức hoặc nút xem thêm của một tin tức. Hệ thống sẽ lấy tên tin tức, tiêu đề, ảnh minh hoạ, nội dung từ bảng TINTUC trong cơ sở dữ liệu và hiển thị lên màn hình.   Use case kết thúc. |
| **Luồng rẽ nhánh** | 1. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc. |
| **Các yêu cầu đặc biệt** | Không có |
| **Hậu điều kiện** | Không có |

Bảng 2. 6: Mô tả usecase xem tin tức nổi bật

|  |  |
| --- | --- |
| **Mô tả vắn tắt** | Use case này cho phép người dùng xem tin tức nổi bật. |
| **Đối tượng** | Người dùng |
| **Tiền điều kiện** | Không có |
| **Luồng cơ bản** | 1. Use case này bắt đầu khi người dùng kích vào mục “Nổi bật” trên thanh menu. Hệ thống sẽ lấy tên tin tức, tiêu đề, ảnh minh hoạ, nội dung của các tin tức từ bảng TINTUC trong cơ sở dữ liệu và hiển thị lên màn hình. 2. Người dùng kích vào ảnh tin tức hoặc tiêu đề tin tức hoặc nút xem thêm của một tin tức. Hệ thống sẽ lấy tiêu đề tin tức, ảnh tin tức, nội dung tin tức từ bảng TINTUC trong cơ sở dữ liệu và hiển thị lên màn hình.   Use case kết thúc. |
| **Luồng rẽ nhánh** | 1. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc. |
| **Các yêu cầu đặc biệt** | Không có |
| **Hậu điều kiện** | Không có |

#### 2.4.1.2 Mô tả use case Xem thông tin cá nhân

Bảng 2. 7: Mô tả usecase xem thông tin cá nhân

|  |  |
| --- | --- |
| **Mô tả vắn tắt** | Use case này cho phép người dùng xem thông tin cá nhân. |
| **Đối tượng** | Người dùng |
| **Tiền điều kiện** | Use case này chỉ cho phép người dùng thực hiện khi đã đăng nhập. |
| **Luồng cơ bản** | 1. Use case này bắt đầu khi người dùng di chuyển con trỏ chuột vào biểu tượng người dùng hoặc tên tài khoản trên thanh menu. Hệ thống sẽ đưa ra những danh mục “Thông tin cá nhân”, “Tin tức yêu thích”, “Tin tức đã xem” hiển thị lên màn hình. 2. Người dùng kích vào “Thông tin cá nhân” trên màn hình. Hệ thống sẽ lấy họ tên người dùng, Email, mật khẩu, ngày sinh, giới tính, số điện thoại, địa chỉ và ảnh đại diện từ bảng TAIKHOAN trong cơ sở dữ liệu và hiển thị lên màn hình.   Use case kết thúc. |
| **Luồng rẽ nhánh** | 1. Tại bước 1 trong luồng cơ bản nếu người dùng chưa đăng nhập vào trang web thì hệ thống sẽ hiển thị yêu cầu đăng nhập/đăng ký lên màn hình. 2. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc. |
| **Các yêu cầu đặc biệt** | Không có |
| **Hậu điều kiện** | Không có |

#### 2.4.1.3.Mô tả use case Chỉnh sửa thông tin cá nhân

Bảng 2. 8: Mô tả usecase chỉnh sửa thông tin cá nhân

|  |  |
| --- | --- |
| **Mô tả vắn tắt** | Use case này cho phép người dùng chỉnh sửa thông tin cá nhân. |
| **Đối tượng** | Người dùng |
| **Tiền điều kiện** | Use case này chỉ cho phép người dùng thực hiện khi đã đăng nhập. |
| **Luồng cơ bản** | 1. Use case này bắt đầu khi người dùng kích vào “Thông tin cá nhân” trên màn hình. Hệ thống sẽ lấy họ tên, Email, mật khẩu, ngày sinh, giới tính, số điện thoại, địa chỉ và ảnh đại diện từ bảng TAIKHOAN trong cơ sở dữ liệu và hiển thị lên màn hình 2. Người dùng kích vào nút “thay đổi” của từng thông tin trên màn hình, nhập thông tin muốn thay đổi và kích vào nút đổi. Hệ thống sẽ chỉnh sửa thông tin vào bảng TAIKHOAN trong cơ sở dữ liệu và hiển thị lên màn hình thông báo “chỉnh sửa thành công”.   Use case kết thúc. |
| **Luồng rẽ nhánh** | 1. Tại bước 2, nếu người dùng kích vào nút “huỷ” thì hệ thống sẽ thông báo “huỷ chỉnh sửa”. 2. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc. |
| **Các yêu cầu đặc biệt** | Không có |
| **Hậu điều kiện** | Không có |

#### 2.4.1.4 Mô tả use case Xem tin tức theo danh mục

Bảng 2. 9: Bảng mô tả usecase xem tin tức theo danh mục

|  |  |
| --- | --- |
| **Mô tả vắn tắt** | Use case này cho phép người dùng xem tin tức theo danh mục. |
| **Đối tượng** | Người dùng |
| **Tiền điều kiện** | Không có. |
| **Luồng cơ bản** | 1. Use case này bắt đầu khi người dùng di chuyển chuột vào thanh menu và kích vào tên danh mục. Hệ thống sẽ lấy danh sách thông tin các tin tức (tên tin tức, tiêu đề, ảnh minh hoạ, nội dung) liên quan đến danh mục đã được chọn từ bảng TINTUC trong cơ sở dữ liệu và hiển thị lên màn hình. 2. Người dùng kích vào ảnh tin tức hoặc tiêu đề tin tức hoặc nút xem thêm của một tin tức. Hệ thống sẽ lấy tên tin tức, tiêu đề, ảnh minh hoạ, nội dung từ bảng TINTUC trong cơ sở dữ liệu và hiển thị lên màn hình.   Use case kết thúc. |
| **Luồng rẽ nhánh** | 1. Tại bước 1 trong luồng cơ bản nếu hệ thống không lấy được một tin tức nào thì hệ thống sẽ hiển thị thông báo “Không có tin tức nào được hiển thị!”. 2. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc. |
| **Các yêu cầu đặc biệt** | Không có |
| **Hậu điều kiện** | Không có |

#### 2.4.1.5 Mô tả use case Đăng nhập

Bảng 2. 10: Mô tả usecase đăng nhập

|  |  |
| --- | --- |
| **Mô tả vắn tắt** | Use case này cho phép người dùng đăng nhập để xác nhận quyền truy cập hệ thống. |
| **Đối tượng** | Người dùng |
| **Tiền điều kiện** | Không có. |
| **Luồng cơ bản** | 1. Use case này bắt đầu khi người dùng kích nào nút đăng nhập trên trang chủ hệ thống. Hệ thống hiện thị lên giao diện yêu cầu người dùng nhập Email và mật khẩu. 2. Người dùng nhập Email và mật khẩu sau đó click vào nút đăng nhập. Hệ thống kiểm tra Email và mật khẩu ở bảng TAIKHOAN trong cơ sở dữ liệu sau đó hiển thị giao diện chính.   Use case kết thúc. |
| **Luồng rẽ nhánh** | 1. Sai tên đăng nhập/mật khẩu: Tại bước 3 nếu người dùng nhập sai tên đăng nhập hoặc mật khẩu hệ thống sẽ hiển thị một thông báo lỗi. Người dùng có thể nhập lại, chọn chức năng lấy lại mật khẩu hoặc bỏ qua thì use case sẽ kết thúc. 2. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sử dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc. |
| **Các yêu cầu đặc biệt** | Không có |
| **Hậu điều kiện** | Nếu use case thành công người dùng sẽ vào được hệ thống. Nếu không trạng thái của hệ thống không thay đổi. |

#### 2.1.4.6 Mô tả use case Đăng ký

Bảng 2. 11: Mô tả usecase Đăng ký

|  |  |
| --- | --- |
| **Mô tả vắn tắt** | Use case này cho phép người dùng và người quản trị đăng kí tài khoản hệ thống. |
| **Đối tượng** | Người dùng |
| **Tiền điều kiện** | Không có. |
| **Luồng cơ bản** | 1. Use case này bắt đầu khi người dùng kích vào biểu tượng “Đăng kí đăng nhập” trên thanh menu và chọn vào “Đăng kí”. Hệ thống sẽ hiển thị màn hình đăng ký. 2. Người dùng điền tên, số điện thoại, Email, mật khẩu, nhập lại mật khẩu sau đó bấm đăng kí. Hệ thống lưu thông tin vừa nhập vào bảng TAIKHOAN usecase kết thúc.   Use case kết thúc. |
| **Luồng rẽ nhánh** | 1. Tại luồng cơ bản nếu người dùng nhập thiếu thông tin thì hệ thống sẽ hiện ra một dòng alert “Bạn chưa điền đủ thông tin. Xin hãy kiểm tra lại.”. Use case kết thúc. 2. Tại luồng cơ bản nếu người dùng nhập số điện thoại không đúng định dạng thì hệ thống sẽ hiện ra một dòng alert “Số điện thoại không đúng định dạng”. Use case kết thúc. 3. Tại luồng cơ bản nếu người dùng nhập Email không đúng định dạng thì hệ thống sẽ hiện ra một dòng alert “Sai định dạng Email”. Use case kết thúc. 4. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sử dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc. |
| **Các yêu cầu đặc biệt** | Không có |
| **Hậu điều kiện** | Nếu use case thành công người dùng sẽ vào được hệ thống. Nếu không trạng thái của hệ thống không thay đổi. |

#### 2.4.1.7 Mô tả use case Bảo trì danh mục

Bảng 2. 12: Mô tả usecase bảo trì danh mục

|  |  |
| --- | --- |
| **Mô tả vắn tắt** | Use case này cho phép Admin xem, thêm, sửa và xóa các sản phẩm trong bảng DANHMUC. |
| **Đối tượng** | Admin, người quản trị |
| **Tiền điều kiện** | Admin cần đăng nhập với vai trò quản trị hệ thống trước khi có thể thực hiện use case. |
| **Luồng cơ bản** | 1. Use case này bắt đầu khi người dùng Quản trị viên kích vào nút “Bảo trì danh mục” trong menu quản trị. 2. Hệ thống lấy thông tin chi tiết của các danh mục (Mã danh mục, Tên danh mục) từ bảng DANHMUC trong cơ sở dữ liệu và hiển thị lên màn hình. 3. Nếu người dùng Quản trị viên muốn thêm mới một danh mục, họ có thể kích vào nút “Thêm mới” trên cửa sổ danh mục để hiển thị màn hình yêu cầu nhập thông tin chi tiết cho danh mục (Tên danh mục) và kích vào nút “Tạo mới”. Nếu thông tin hợp lệ, hệ thống sẽ tạo một bản ghi mới trong bảng DANHMUC và hiển thị danh sách các danh mục được cập nhật. 4. Nếu người dùng Quản trị viên muốn sửa một danh mục đã có, họ có thể kích vào nút “Sửa” trên dòng sản phẩm đó để hiển thị một màn hình với các thông tin cũ của một danh mục trong các Textbox. Người dùng Quản trị viên sửa các thông tin của danh mục (Tên danh mục) và kích vào nút “Cập nhật”. Nếu thông tin hợp lệ, hệ thống sẽ cập nhật thông tin sản phẩm trong bảng DANHMUC và hiển thị danh sách danh mục đã cập nhật. 5. Nếu người dùng Quản trị viên muốn xóa một danh mục, họ có thể kích vào nút “Xóa” trên dòng sản phẩm đó để hiển thị một màn hình yêu cầu xác nhận xóa. Người dùng Quản trị viên kích vào nút “Đồng ý”. Hệ thống sẽ xóa sản phẩm khỏi bảng DANHMUC và hiển thị danh sách danh mục đã cập nhật.   Use case kết thúc. |
| **Luồng rẽ nhánh** | 1. Tại bước 3 hoặc 7 trong luồng cơ bản, nếu người dùng Quản trị viên nhập thông tin chi tiết sản phẩm không hợp lệ thì hệ thống sẽ hiển thị thông báo lỗi yêu cầu nhập lại. Người dùng Quản trị viên có thể nhập lại để tiếp tục hoặc kích chọn vào nút “Hủy bỏ” để kết thúc. 2. Tại bước 3 hoặc 7 trong luồng cơ bản, nếu người dùng Quản trị viên kích vào nút “Hủy bỏ” hệ thống sẽ bỏ qua thao tác thêm sản phẩm mới hoặc sửa tương ứng và hiển thị danh sách các sản phẩm trong bảng DANHMUC. 3. Tại bước 9 trong luồng cơ bản, nếu người dùng Quản trị viên không đồng ý xóa sản phẩm, họ có thể kích chọn vào nút “Không đồng ý” và hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc. |
| **Các yêu cầu đặc biệt** | Use case này chỉ cho phép một số vai trò như Quản trị viên, người chủ hệ thống thực hiện. |
| **Hậu điều kiện** | Nếu use case kết thúc thành công thì thông tin chi tiết của danh mục sẽ được cập nhật trong cơ sở dữ liệu. |

#### 2.4.1.8 Mô tả use case Bảo trì tài khoản

Bảng 2. 13: Mô tả usecase bảo trì tài khoản

|  |  |
| --- | --- |
| **Mô tả vắn tắt** | Use case này cho phép Admin quản lý các tài khoản người dùng trên trang web hoặc ứng dụng, bao gồm xem, thêm, sửa, xoá tài khoản. |
| **Đối tượng** | Admin, người quản trị |
| **Tiền điều kiện** | Admin cần đăng nhập với vai trò quản trị hệ thống trước khi có thể thực hiện use case. |
| **Luồng cơ bản** | 1. Use case bắt đầu khi Admin kích vào nút "Bảo trì tài khoản" trên menu quản trị. Hệ thống lấy thông tin chi tiết về các tài khoản người dùng từ bảng TAIKHOAN trong cơ sở dữ liệu và hiển thị lên màn hình danh sách các tài khoản. 2. Xem tài khoản: Admin có thể xem thông tin chi tiết của một tài khoản bằng cách kích vào tên người dùng hoặc mã tài khoản trong danh sách. 3. Sửa tài khoản: Admin có thể sửa thông tin chi tiết của một tài khoản bằng cách kích vào nút "Sửa" trên một dòng tài khoản trong danh sách. Hệ thống hiển thị một màn hình với các thông tin cũ của tài khoản trong các Textbox. Admin sửa thông tin và kích nút "Cập nhật". Hệ thống sẽ cập nhật thông tin tài khoản trong bảng TAIKHOAN và hiển thị danh sách tài khoản đã cập nhật. 4. Xóa tài khoản: Admin có thể xóa một tài khoản bằng cách kích vào nút "Xóa" trên một dòng tài khoản trong danh sách. Hệ thống hiển thị một màn hình yêu cầu xác nhận xóa. Nếu Admin chọn "Đồng ý", hệ thống sẽ xóa tài khoản khỏi bảng TAIKHOAN và hiển thị danh sách tài khoản đã cập nhật. |
| **Luồng rẽ nhánh** | 1. Tại bước 3 khi Admin nhập thông tin chi tiết tài khoản không hợp lệ, hệ thống sẽ hiển thị thông báo lỗi yêu cầu nhập lại. Admin có thể nhập lại để tiếp tục hoặc kích chọn vào nút "Hủy bỏ" để kết thúc. 2. Tại bước 3 hoặc 4 khi Admin kích vào nút "Hủy bỏ", hệ thống sẽ bỏ qua thao tác thêm mới hoặc sửa tương ứng và hiển thị danh sách các tài khoản trong bảng TAIKHOAN. 3. Tại bước 5 khi Admin kích vào nút "Không đồng ý", hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc. |
| **Các yêu cầu đặc biệt** | Use case này chỉ cho phép một số vai trò như Quản trị viên, người chủ hệ thống thực hiện.  Các tài khoản có thể bị xóa hoặc chỉnh sửa nếu chúng vi phạm các quy định và chính sách của trang web hoặc ứng dụng.  Cần cung cấp các thông báo cho người dùng khi có các thay đổi hoặc sự cố với tính năng quản lý tài khoản để họ có thể hiểu và sử dụng tính năng một cách hiệu quả. |
| **Hậu điều kiện** | Nếu use case kết thúc thành công thì thông tin chi tiết của tài khoản sẽ được cập nhật trong cơ sở dữ liệu. |

#### 2.4.1.9 Mô tả use case Bảo trì tin tức

Bảng 2. 14: Mô tả usecase bảo trì tin tức

|  |  |
| --- | --- |
| **Mô tả vắn tắt** | Use case này cho phép người quản trị xem, thêm, sửa và xóa thông tin các tin tức trong bảng TINTUC |
| **Đối tượng** | Admin, người quản trị |
| **Tiền điều kiện** | Admin cần đăng nhập với vai trò quản trị hệ thống trước khi có thể thực hiện use case. |
| **Luồng cơ bản** | 1. Use case này bắt đầu khi người quản trị kích vào nút “Tin tức” trên thanh menu quản trị. Hệ thống sẽ lấy mã tin tức tên tin tức, tiêu đề, nội dung, ảnh từ bảng TINTUC trong cơ sở dữ liệu và hiển thị lên màn hình. 2. Thêm tin tức:    1. Người quản trị kích vào nút “Thêm Tin Tức” trong màn hình bảo trì tin tức. Hệ thống sẽ tự sinh ra mã tin tức và hiển thị lên màn hình yêu cầu nhập các thông tin của tin tức (tên tin tức, tiêu đề, nội dung, ảnh).    2. Người quản trị nhập thông tin của tin tức và kích vào nút “Thêm”. Hệ thống sẽ tạo một bản ghi mới trong bảng TINTUC và hiển thị danh sách các tin tức sau khi cập nhật lên màn hình. 3. Cập nhật tin tức:    1. Người quản trị kích vào nút “Cập nhật” trên một dòng trong bảng danh sách các tin tức. Hệ thống sẽ lấy mã tin tức và các thông tin cũ của tin tức cần sửa (tên tin tức, tiêu đề, nội dung, ảnh) từ bảng TINTUC trong cơ sở dữ liệu và hiển thị một màn hình.    2. Người quản trị sửa các thông tin của tin tức (tên tin tức, tiêu đề, nội dung, ảnh) và kích vào nút “Cập nhật”. Hệ thống sẽ cập nhật thông tin của tin tức trong bảng TINTUC và hiển thị danh sách các tin tức sau khi cập nhật lên màn hình. 4. Xóa tin tức:    1. Người quản trị kích vào nút “Xóa” trên một dòng trong bảng danh sách các tin tức. Hệ thống sẽ hiển thị một yêu cầu xác nhận xóa.    2. Người quản trị kích vào nút “Có”. Hệ thống sẽ xóa thông tin tin tức ra khỏi bảng TINTUC và hiển thị danh sách các tin tức sau khi cập nhật lên màn hình.   Kết thúc use case. |
| **Luồng rẽ nhánh** | 1. Tại bước 1 trong luồng cơ bản nếu hệ thống không lấy được một tin tức nào thì sẽ hiển thị thông báo “Không có tin tức trong CSDL!” và use case kết thúc. 2. Tại bước 2.2 và 3.2 trong luồng cơ bản nếu người quản trị nhập thông tin không hợp lệ. Hệ thống sẽ hiển thị thông báo lỗi và yêu cầu nhập lại. 3. Tại bước 2.2 và 3.2 trong luồng cơ bản nếu người quản trị nhấn nút “Quay lại”. Hệ thống sẽ bỏ qua thao tác thêm mới hoặc cập nhật và hiển thị danh sách các tin tức trong bảng TINTUC. 4. Tại bước 4.2 trong luồng cơ bản nếu người quản trị nhấn nút “Không”. Hệ thống sẽ bỏ qua thao tác xóa và hiển thị danh sách các tin tức trong bảng TINTUC. 5. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc. |
| **Các yêu cầu đặc biệt** | Use case này chỉ cho phép một số vai trò như Quản trị viên, người chủ hệ thống thực hiện.  Các tài khoản có thể bị xóa hoặc chỉnh sửa nếu chúng vi phạm các quy định và chính sách của trang web hoặc ứng dụng.  Cần cung cấp các thông báo cho người dùng khi có các thay đổi hoặc sự cố với tính năng quản lý tài khoản để họ có thể hiểu và sử dụng tính năng một cách hiệu quả. |
| **Hậu điều kiện** | Nếu use case thành công thì thông tin về tin tức sẽ được cập nhật trong cơ sở dữ liệu. |

### 2.4.2 Biểu đồ phân tích Use case

#### 2.4.2.1 Use case tìm kiếm tin tức

a.Biểu đồ trình tự



Hình 2. 4: Biểu đồ trình tự tìm kiếm tin tức

b. Biểu đồ phân tích



Hình 2. 5: Biểu đồ phân tích usecase tìm kiếm tin tức

#### 2.4.2.2.Use case Xem tin tức nổi bật

a. Biểu đồ trình tự



Hình 2. 6: Biểu đồ tình tự xem tin tức nổi bật

b. Biểu đồ phân tích



Hình 2. 7: Biểu đồ phân tích xem tin tức nổi bật

#### 2.4.2.3 Use case xem tin tức theo danh mục

a.Biểu đồ trình tự



Hình 2. 8: Biểu đồ trình tự xem tin tức theo danh mục

b. Biểu đồ phân tích



Hình 2. 9: Biểu đồ phân tích xem tin tức theo danh mục

#### 2.4.2.4 Use case xem chi tiết tin tức

a. Biểu đồ trình tự



Hình 2. 10: Biểu đồ trình tự xem chi tiết tin tức

b. Biểu đồ phân tích



Hình 2. 11: Biểu đồ phân tích xem chi tiết tin tức

#### 2.4.2.5 Use case xem thông tin cá nhân

a. Biểu đồ trình tự



Hình 2. 12: Biểu đồ trình tự xem thông tin cá nhân

b.Biểu đồ phân tích



Hình 2. 13: Biểu đồ phân tích xem thông tin cá nhân

#### 2.4.2.6.Use case chỉnh sửa thông tin cá nhân

a.Biểu đồ trình tự



Hình 2. 14: Biểu đồ trình tự chỉnh sửa thông tin cá nhân

b.Biểu đồ phân tích



Hình 2. 15: Biểu đồ phân tích chỉnh sửa thông tin cá nhân

#### 2.4.2.7. Use case đăng nhập

a.Biểu đồ trình tự



Hình 2. 16: Biểu đồ trình tự đăng nhập

b.Biểu đồ phân tích



Hình 2. 17: Biểu đồ phân tích đăng nhập

#### 2.4.2.8. Use case

a.Biểu đồ trình tự



Hình 2. 18: Biểu đồ trình tự đăng ký

b.Biểu đồ phân tích

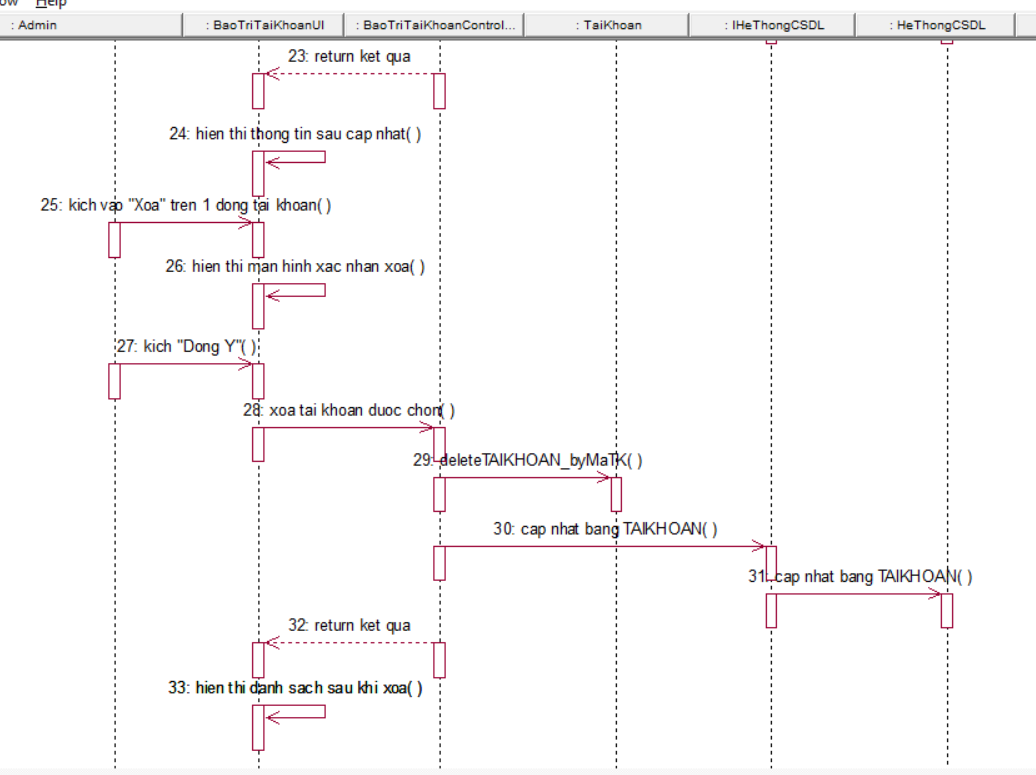


Hình 2. 19: Biểu đồ phân tích đăng ký

#### 2.4.2.9. Use case bảo trì tài khoản

a.Biểu đồ trình tự





Hình 2. 20: Biểu đồ trình tự bảo trì tài khoản

b. Biểu đồ phân tích



Hình 2. 21: Biểu đồ phân tích bảo trì tài khoản

#### 2.4.2.10. Use case bảo trì bình luận

a. Biểu đồ trình tự



Hình 2. 22: Biểu đồ trình tự bảo trì bình luận

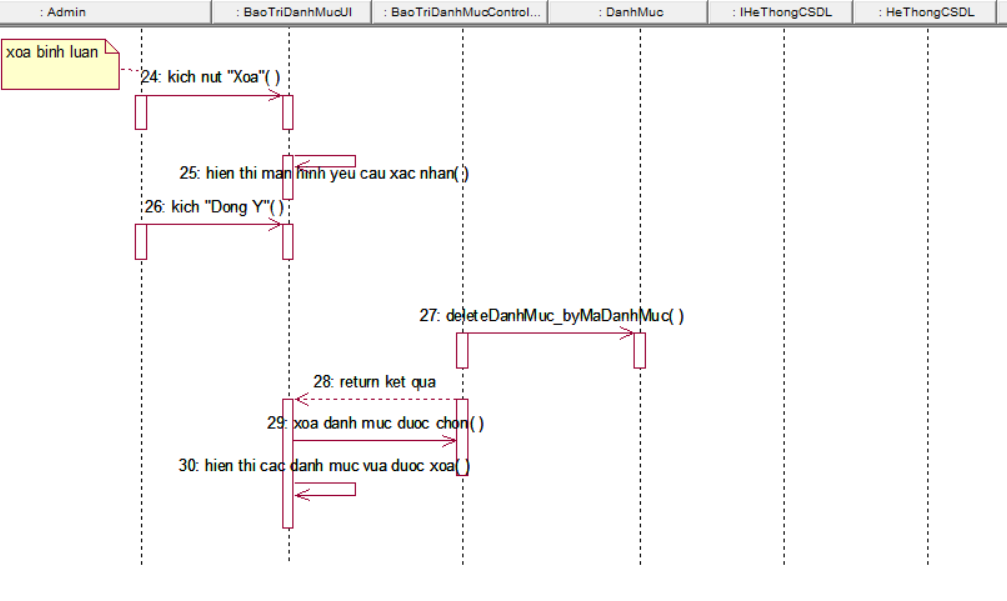
b. Biểu đồ phân tích



Hình 2. 23: Biểu đồ phân tích bảo trì bình luận

#### 2.4.2.11. Use case bảo trì danh mục

a.Biểu đồ trình tự



Hình 2. 24: Biểu đồ trình tự bảo trì danh mục

b.Biểu đồ phân tích

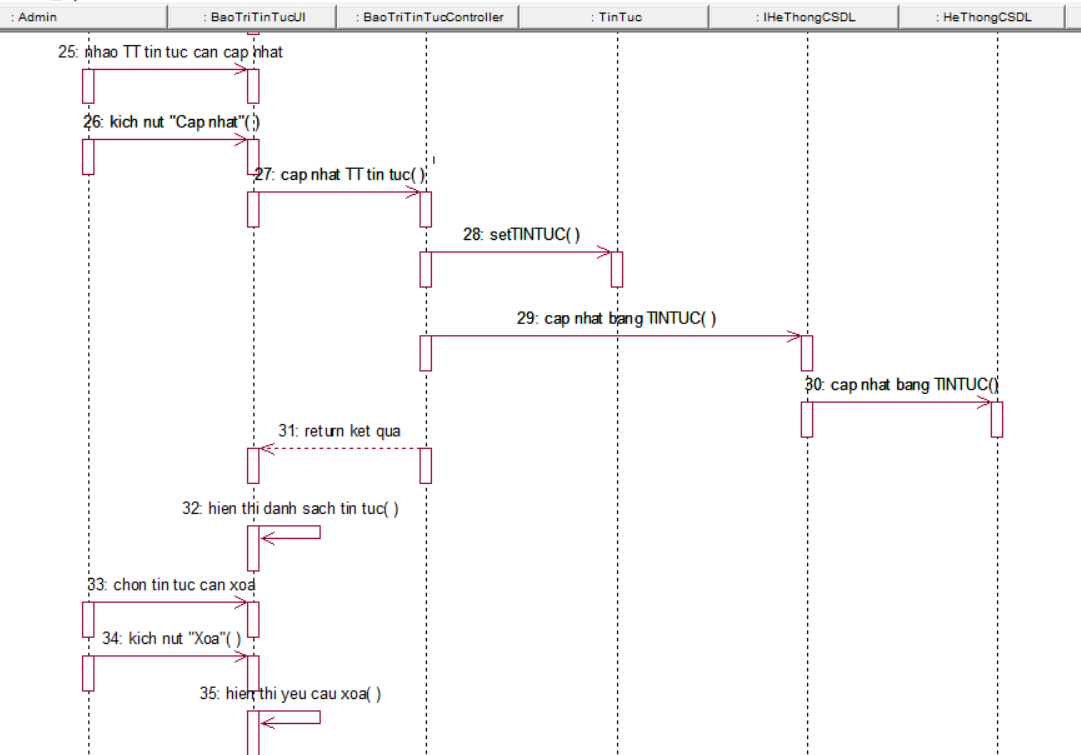


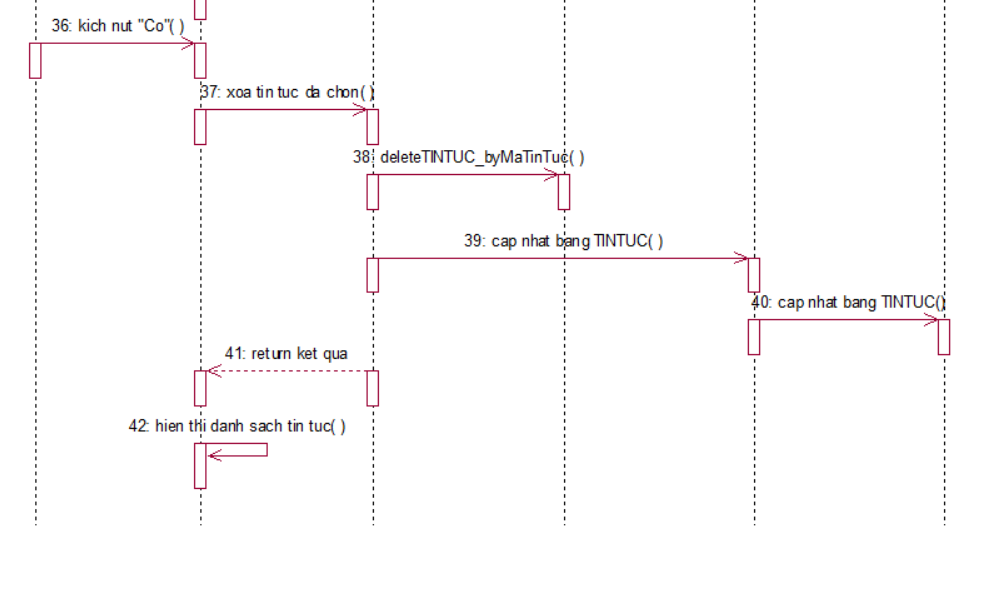
Hình 2. 25: Biểu đồ phân tích bảo trì danh mục

#### 2.4.2.12. Use case bảo trì tin tức

a.Biểu đồ trình tự







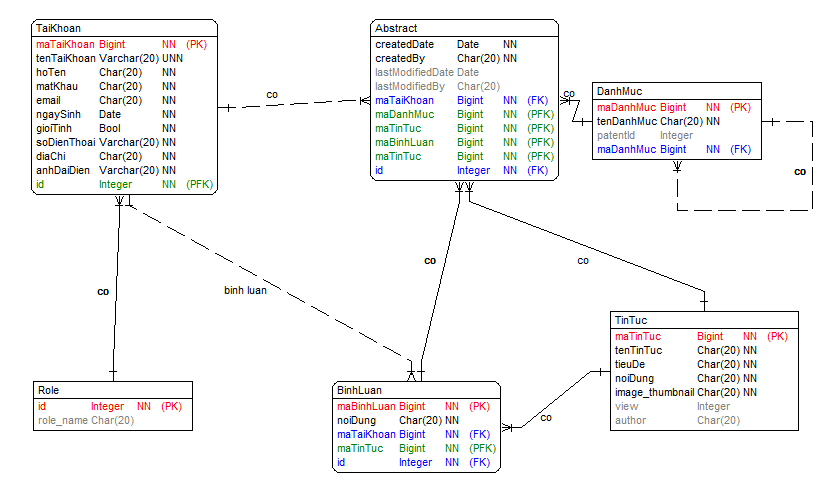
Hình 2. 26: Biểu đồ trình tự bảo trì tin tức

b. Biểu đồ phân tích



Hình 2. 27: Biểu đồ phân tích bảo trì tin tức

### 2.4.3. Biểu đồ thực thể liên kết



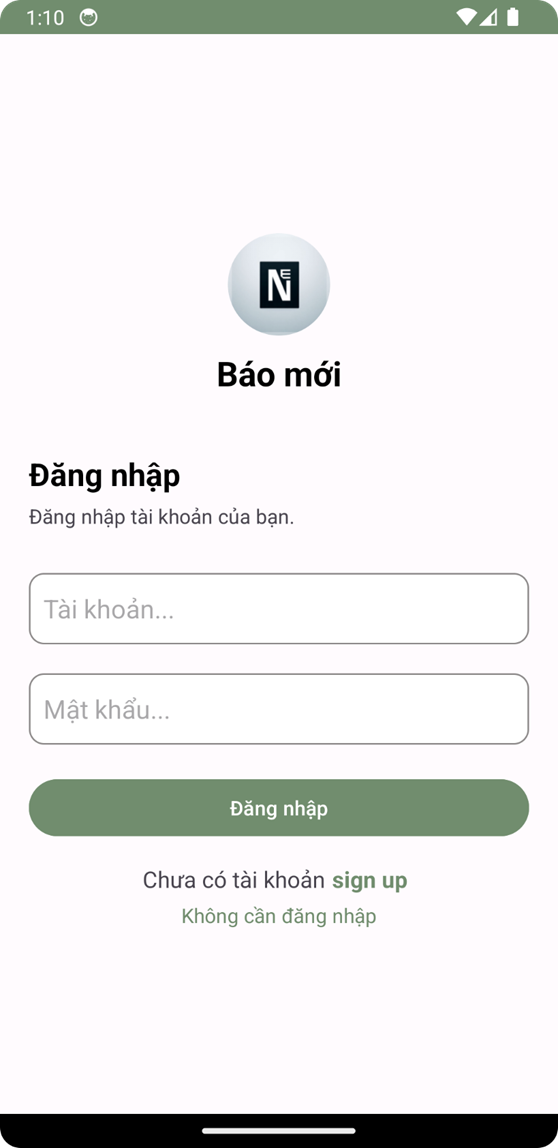
Hình 2. 28:Biểu đồ thực thể liên kết mức vật lý

# CHƯƠNG 3. KẾT QUẢ

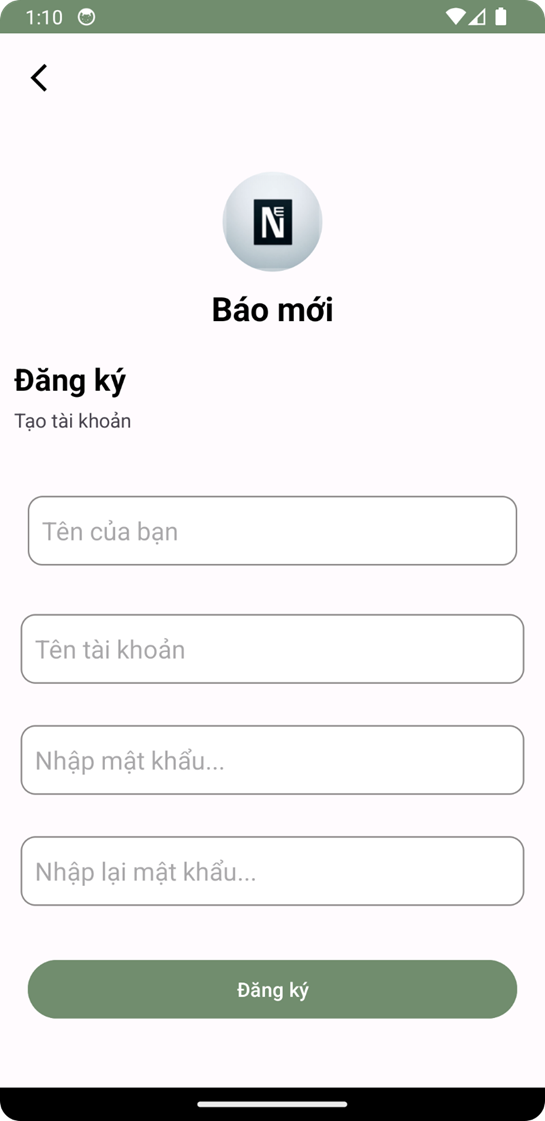
# 3.1. Kết quả đạt được



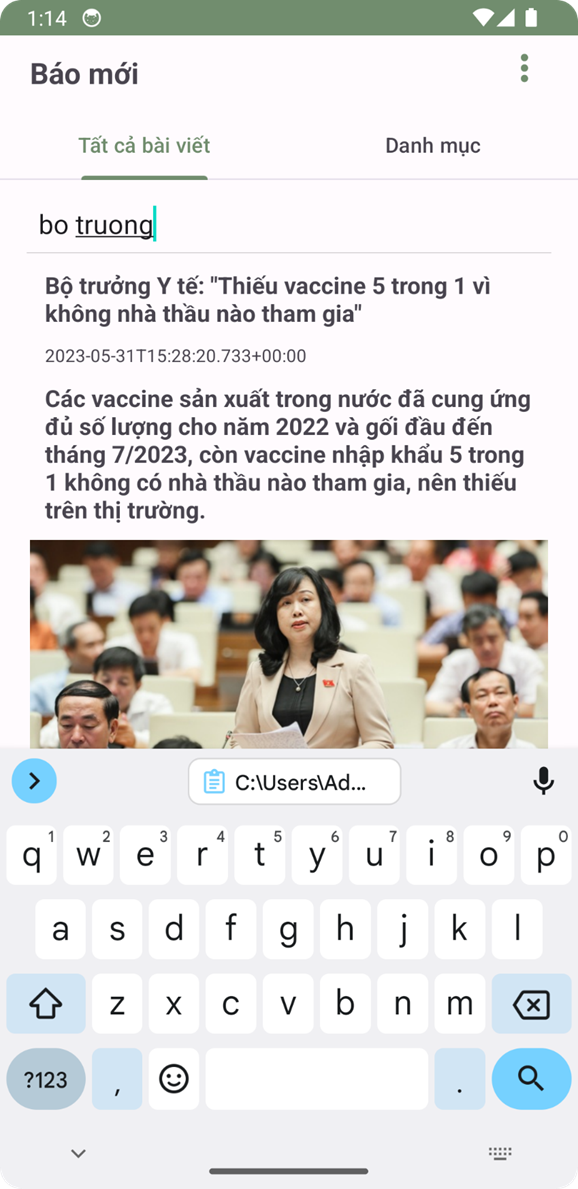
Hình 3. 1: Xem chi tiết bài viết



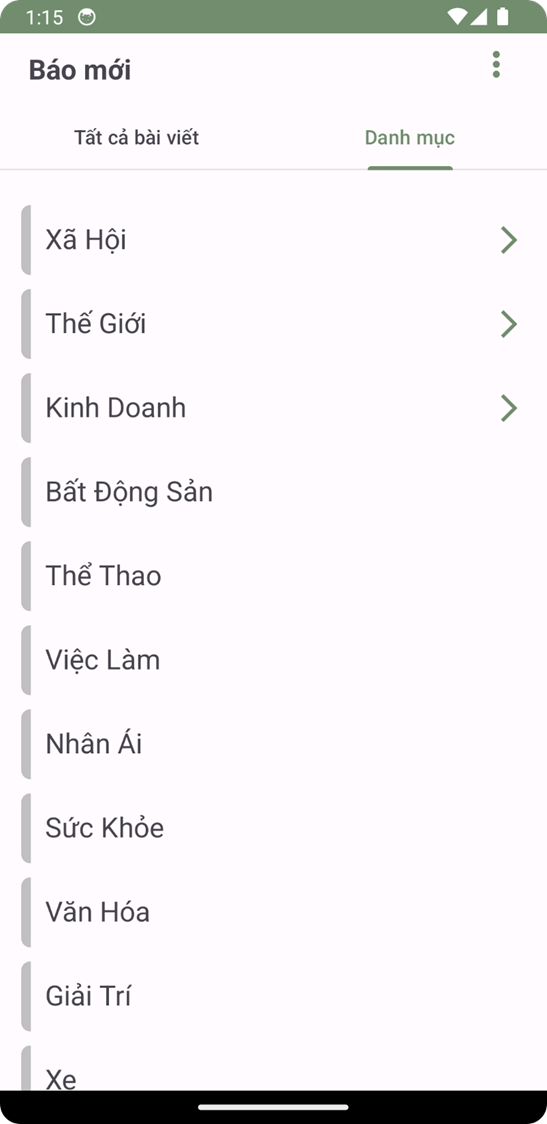
Hình 3. 2: Đăng nhập



Hình 3. 3: Đăng ký



Hình 3. 4: Tìm kiếm



Hình 3. 5: Xem danh mục



Hình 3. 6: Xem danh mục con



Hình 3. 7: Xem bài viết theo danh mục

## 3.2. Kết quả kiểm thử chức năng người quản trị

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Trường hợp | Đầu vào | Đầu ra mong muốn | Kết quả |
| 1 | Kiểm tra chức năng đăng ký tài khoản | Nhập thông tin đăng ký không hợp lệ | Hệ thống hiển thị thông báo nhập sai định dạng | Pass |
| Nhập thông tin đăng ký hợp lệ | Đăng ký thành công, tự động đăng nhập | Pass |
| 2 | Kiểm tra chức năng đăng nhập | Nhập thông tin tài khoản và mật khẩu hợp lệ | Đăng nhập thành công, hiển thị giao diện sau đăng nhập | Pass |
| Nhập thông tin tài khoản và mật khẩu không hợp lệ | Hệ thống thông báo lỗi | Pass |
| 3 | Kiểm tra chức năng đăng xuất | Đăng xuất tài khoản | Đăng xuất thành công, chuyển sang giao diện trang chủ | Pass |
| 4 | Kiểm tra chức năng tìm xem tin tức đã lưu | Tin tức đã lưu | Khi đăng nhập thành công với quyền user, hệ thống sẽ lưu lại những tin mà người dùng đã xem | Pass |

## 3.2. Kiểm thử chức năng người quản trị

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Trường hợp | Đầu vào | Đầu ra mong muốn | Kết quả |
| 1 | Kiểm tra chức năng đăng nhập | Nhập thông tin tài khoản và mật khẩu hợp lệ | Đăng nhập thành công, hiển thị giao diện sau đăng nhập với quyền super\_admin | Pass |
| Nhập thông tin tài khoản và mật khẩu không hợp lệ | Hệ thống thông báo lỗi | Pass |
| 2 | Kiểm tra chức năng đăng xuất | Đăng xuất tài khoản | Đăng xuất thành công, chuyển sang giao diện đăng nhập | Pass |
| 3 | Kiểm tra chức năng bảo trì danh sách người dùng | Đã đăng nhập hệ thống bằng tài khoản quản trị viên. | Xem danh sách người dùng, xem chi tiết thông tin người dùng, thêm người dùng, cập nhật người dùng, xóa người dùng. | Pass |
| 4 | Kiểm tra chức năng bảo trì danh sách hạng mục | Đã đăng nhập hệ thống bằng tài khoản quản trị viên. | Xem danh sách hạng mục, thêm hạng mục, sửa thông tin hạng mục, xoá hạng mục | Pass |
| 5 | Kiểm tra chức năng bảo trì danh sách tin tức | Đã đăng nhập hệ thống bằng tài khoản quản trị viên. | Xem danh sách tin tức, xem chi tiết tin tức, sửa thông tin tin tức, xoá tin tức | Pass |
| 7 | Kiểm tra chức năng thống kê | Đã đăng nhập hệ thống bằng tài khoản quản trị viên. | Hiển thị các bài viết có lượt xem nhiều nhất, số lượng bài viết | Pass |

# KẾT LUẬN

❖ Đánh giá kết quả

• Về mặt công nghệ, chúng em đã tìm hiểu và nắm bắt quy trình để xây dựng một ứng dụng trên android

• Hoàn thiện bài khảo sát và phân tích thiết kế hệ thống “đọc tin tức KLNews”.

• Lập trình thành thạo với ngôn ngữ Java.

❖ Hướng phát triển

• Thiết kế và lập trình mở rộng các chức năng: đăng nhập, đăng ký, đọc tin tức.

• Hỗ trợ giao diện dạng widget cho ứng dụng trên màn hình chính

của điện thoại.

• Hỗ trợ giao diện ứng dụng trên tablet và pc.

• Cải thiện sự mượt mà của ứng dụng. Nâng cao trải nghiệm người

dùng.

• Em rất mong tiếp tục nhận được sự giúp đỡ và tạo điều kiện của

thầy để nhóm em có cơ hội phát triển, hoàn thiện ứng dụng tốt hơn trong thời gian tới.

# TÀI LIỆU THAM KHẢO

1. Tổ HTTT Đại học Công Nghiệp Hà Nội – Giáo trình hệ quản trị cơ sở dữ liệu SQL.
2. Tổ HTTT Đại học Công Nghiệp Hà Nội – Giáo trình phân tích thiết kế hệ thống.

[3]. Stack overflow https://stackoverflow.com/