**BỘ GIÁO DỤC VÀ ĐÀO TẠO**



**TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI PHÂN HIỆU TẠI TP. HỒ CHÍ MINH**

**BỘ MÔN CÔNG NGHỆ THÔNG TIN**

**BÁO CÁO ĐỒ ÁN TỐT NGHIỆP**

**ĐỀ TÀI:**

**XÂY DỰNG ỨNG DỤNG XEM PHIM TRỰC TUYẾN VÀ NGHIÊN CỨU THUẬT TOÁN CONTENT-BASED FILTERING VÀO HỆ THỐNG ĐỀ XUẤT NỘI DUNG**

|  |  |
| --- | --- |
| Giảng viên hướng dẫn | : Ths.Trần Phong Nhã |
| Sinh viên thực hiện | : Nguyễn Gia Ngọc |
| Lớp | : Công nghệ thông tin |
| Khóa | : K60 |
| Mã số sinh viên | : 6051071077 |

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**



**TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI PHÂN HIỆU TẠI TP. HỒ CHÍ MINH**

**BỘ MÔN CÔNG NGHỆ THÔNG TIN**

**BÁO CÁO ĐỒ ÁN TỐT NGHIỆP**

**ĐỀ TÀI:**

**XÂY DỰNG ỨNG DỤNG XEM PHIM TRỰC TUYẾN VÀ NGHIÊN CỨU THUẬT TOÁN CONTENT-BASED FILTERING VÀO HỆ THỐNG ĐỀ XUẤT NỘI DUNG**

|  |  |
| --- | --- |
| Giảng viên hướng dẫn | : Ths.Trần Phong Nhã |
| Sinh viên thực hiện | : Nguyễn Gia Ngọc |
| Lớp | : Công nghệ thông tin |
| Khóa | : K60 |
| Mã số sinh viên | : 6051071077 |

|  |  |
| --- | --- |
| **TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI**  **PHÂN HIỆU TẠI TP. HỒ CHÍ MINH**  \*\*\* | **CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM**  Độc lập – Tự do – Hạnh phúc |

## NHIỆM VỤ THIẾT KẾ TỐT NGHIỆP

**BỘ MÔN:** **CÔNG NGHỆ THÔNG TIN**

\*\*\*

**Mã sinh viên:** 6051071077.......................... **Họ tên SV:** Nguyễn Gia Ngọc…………..

**Khóa:** 60......................................................... **Lớp:** Công nghệ thông tin………………

### Tên đề tài

* Xây dựng ứng dụng xem phim trực tuyến

### Mục đích, yêu cầu

* 1. **Mục đích**
* Xây dựng ứng dụng xem phim trực tuyến, để giải trí, xem những bộ phim hay và mới nhất.
* Hỗ trợ người dùng xem phim khi họ có thời gian với nhu cầu giải trí.
  1. **Yêu cầu**
* Tìm hiểu về JAVA, Android Studio.
* Tìm hiểu về Firebase, Firebase Function
* Tìm hiểu về Nodejs
* Tìm hiểu và áp dụng các API.
* Tìm hiểu thuật toán content based filtering.
* Xây dựng, thiết kế

### Nội dung và phạm vi đề tài

* 1. **Nội dung đề tài**
* Xây dựng ứng dụng xem phim gồm 2 giao diện người dùng và admin
  + Về giao diện người dùng cho phép người dùng có thể xem, tìm kiếm, đăng ký tài khoản và tiến hành xem những bộ phim yêu thích. Ngoài ra còn có gợi ý những bộ phim tương tự cho người dung tham khảo.
  + Về giao diện admin cho phép quản lý những bộ phim, quản lý người dung, thêm sửa xóa các bộ phim, người dùng qua Firebase.
  1. **Phạm vi đề tài**
* Đối tượng nghiên cứu của đề tài là những người yêu thích xem phim và sử dụng các ứng dụng xem phim trực tuyến.

### Công nghệ, công cụ và ngôn ngữ lập trình

* 1. **Ngôn ngữ lập trình và công nghệ.**
* Java
* Firebase, firebase Function
* Nodejs
  1. **Công cụ**
* Androids Studio, SDK 32, Máy ảo pixel 6 Pro API 28.
* Firebase, NodeJs

### Các kết quả chính dự kiến sẽ đạt được và ứng dụng

* Hoàn chỉnh bản báo cáo đề tài.
* Xây dựng đầu đủ các chức năng cần thiết cho ứng dụng xem phim.
* Xây dựng giao diện thuận tiện cho người sử dụng.
* Nắm được quy trình nghiệp vụ của ứng dụng.
* Xây dựng thành công ứng dụng.

### Giáo viên và cán bộ hướng dẫn

Họ tên: Ths.Trần Phong Nhã

Đơn vị công tác: Bộ môn Công nghệ thông tin – Trường đại học Giao Thông Vận tải phân hiệu tại TPHCM.

Điện thoại: 0981058040 Email: tpnha@st.utc2.edu.vn

|  |  |
| --- | --- |
| **Trưởng BM Công nghệ Thông tin** | **Giáo viên hướng dẫn** |
| **ThS. Trần Phong Nhã** | **ThS. Trần Phong Nhã** |

Sinh viên: Nguyễn Gia Ngọc Ký tên:

Điện thoại: 0346325760 Email: [6051071077@st.utc2.edu.vn](mailto:6051071077@st.utc2.edu.vn)

LỜI CẢM ƠN

Lời nói đầu tiên, em xin gửi tới Quý thầy cô trong Bộ môn Công Nghệ Thông Tin, cũng như Ban Giám Hiệu Trường Đại học Giao thông Vận tải phân hiệu tại Thành phố Hồ Chí Minh lời chúc sức khỏe và lời cảm ơn sâu sắc. Với sự quan tâm dạy dỗ, chỉ bảo tận tình chu đáo của thầy cô, nay em có thể hoàn thành đề tài tốt nghiệp “Xây dựng ứng dụng xem phim trực tuyến”.

Để hoàn thành nhiệm vụ được giao này, ngoài sự nỗ lực học hỏi không ngừng của bản thân còn có sự hướng dẫn tận tình của các giảng viên trong 4 năm vừa qua, đặc biệt hơn hết nhờ có giảng viên **Trần Phong Nhã**, người đã hướng dẫn cho tôi những hướng đi, truyền đạt cho em những kiến thức, kỹ năng để tôi có thể hoàn thành đề tài tốt nghiệp này.

Mặc dù đã cố gắng hết sức để hoàn thành đề tài, nhưng chắc chắn rằng sẽ khó tránh khỏi những thiếu sót. Em rất mong nhận được những sự đánh giá, góp ý của Quý thầy cô để em có thể rút ra cho mình những bài học, kinh nghiệm quý báu.

Sau cùng, em cũng không biết nói gì hơn ngoài kính chúc Quý thầy cô trong Bộ môn Công Nghệ Thông Tin và đặc biệt là Thầy **Trần Phong Nhã** thật dồi dào sức khỏe và ngày càng gặt hái được nhiều thành công hơn nữa trong cuộc sống cũng như trong sự nghiệp giảng dạy của mình.

Em xin chân thành cảm ơn!

TP. Hồ Chí Minh, ngày tháng năm 2023

Sinh viên thực hiện

**Nguyễn Gia Ngọc**

NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN

## ........................................................................................................................

## ........................................................................................................................

## ........................................................................................................................

## ........................................................................................................................

## ........................................................................................................................

## ........................................................................................................................

## ........................................................................................................................

## ........................................................................................................................

## ........................................................................................................................

## ........................................................................................................................

## ........................................................................................................................

## ........................................................................................................................

## ........................................................................................................................

## ........................................................................................................................

## ........................................................................................................................

## ........................................................................................................................

Tp. Hồ Chí Minh, ngày ….… tháng ….… năm ….…

**Giảng viên hướng dẫn**

**Ths. Trần Phong Nhã**

MỤC LỤC

[**LỜI CẢM ƠN i**](#_Toc138378617)

[**NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN ii**](#_Toc138378618)

[**MỤC LỤC iii**](#_Toc138378619)

[**DANH MỤC CHỮ VIẾT TẮT vi**](#_Toc138378620)

[**DANH MỤC HÌNH VẼ vii**](#_Toc138378621)

[**TỔNG QUAN 1**](#_Toc138378622)

[Tổng quan đề tài 1](#_Toc138378623)

[Mục tiêu nghiên cứu 1](#_Toc138378624)

[Phạm vi nghiên cứu 2](#_Toc138378625)

[Cách tiếp cận, phương pháp nghiên cứu và thực hiện 2](#_Toc138378626)

[Cấu trúc báo cáo đồ án tốt nghiệp 2](#_Toc138378627)

[CHƯƠNG 1 CƠ SỞ LÝ THUYẾT 4](#_Toc138378628)

[1.1 Giới thiệu về Java trong Android Studio 4](#_Toc138378629)

[1.1.1 Lịch sử phát triển của Java trong Android Studio: 4](#_Toc138378630)

[1.1.2 Đặc điểm của Android 4](#_Toc138378631)

[1.2 Nội dung cơ bản về Java trong Android 6](#_Toc138378632)

[1.2.1 Đặc điểm của Java trong Android Studio: 6](#_Toc138378633)

[1.2.2 Ứng dụng của Java trong Android Studio 7](#_Toc138378634)

[1.2.3 Khái niệm cơ bản về Java 7](#_Toc138378635)

[1.2.4 Lập trình hướng đối tượng trong Java 8](#_Toc138378636)

[1.2.5 Thành phần trong Android Studio sử dụng Java 9](#_Toc138378637)

[1.3 Tổng quan về Firebase và Firebase Function 10](#_Toc138378638)

[1.3.1 Tổng quan về Firebase 10](#_Toc138378639)

[1.3.2 Tổng quan về Firebase Function 12](#_Toc138378640)

[1.4 Tổng quan về Node.js 13](#_Toc138378641)

[1.5 Tổng quan về API 15](#_Toc138378642)

[CHƯƠNG 2 PHÂN TÍCH THUẬT TOÁN CONTENT BASED FILTERING 17](#_Toc138378643)

[2.1 Giới thiệu về Content-Based Filtering 17](#_Toc138378644)

[2.2 Khái niệm và đặc trưng của Content-Based Filtering 17](#_Toc138378645)

[2.3 Nguyên lý hoạt động của Content-Based Filtering: 19](#_Toc138378646)

[2.3.1 Các bước thực hiện của Content-Based Filtering: 19](#_Toc138378647)

[2.3.2 Các thuật toán phổ biến trong Content-based Filtering: 19](#_Toc138378648)

[2.3.3 Phương pháp gợi ý dựa theo nội dung 20](#_Toc138378649)

[2.4 Xây dựng mô hình và ví dụ Content Based Filtering 22](#_Toc138378650)

[2.4.1 Giải thuật về thuật toán Content-Based Filtering 22](#_Toc138378651)

[2.4.2 Tính toán độ tương đồng Euclid, Jaccard, Consine 23](#_Toc138378652)

[2.4.3 Ví dụ về thuật toán Content-Based Filtering 28](#_Toc138378653)

[2.5 Kết luận 28](#_Toc138378654)

[CHƯƠNG 3 PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 29](#_Toc138378655)

[3.1 Khảo sát và đánh giá hệ thống 29](#_Toc138378656)

[3.2 Quy trình thực hiện quản trị hệ thống của nhân viên quản trị 29](#_Toc138378657)

[3.3 Yêu cầu hệ thống 30](#_Toc138378658)

[3.3.1 Yêu cầu chức năng 30](#_Toc138378659)

[3.3.2 Yêu cầu phi chức năng 30](#_Toc138378660)

[3.3.3 Yêu cầu khác: 30](#_Toc138378661)

[3.4 Sơ đồ phân rã chức năng và luồng dữ liệu 30](#_Toc138378662)

[3.4.1 Yêu cầu người dùng 31](#_Toc138378663)

[3.4.2 Sơ đồ phân rã chức năng 31](#_Toc138378664)

[3.4.3 Sơ đồ luồng dữ liệu 32](#_Toc138378665)

[3.5 Biểu đồ UseCase 33](#_Toc138378666)

[3.6 Sơ đồ ER 37](#_Toc138378667)

[3.7 Sơ đồ hoạt động 38](#_Toc138378668)

[3.7.1 Với người dùng 38](#_Toc138378669)

[3.7.2 Với người quản trị 45](#_Toc138378670)

[3.8 Sơ đồ Database Diagram 52](#_Toc138378671)

[3.9 Sồ đồ ClassDiagram 53](#_Toc138378672)

[CHƯƠNG 4 TRIỂN KHAI CHƯƠNG TRÌNH ỨNG DỤNG XEM PHIM TRỰC TUYẾN 54](#_Toc138378673)

[4.1 Tồng quan ứng dụng 54](#_Toc138378674)

[4.2 Chương trình demo 54](#_Toc138378675)

[4.2.1 Giao diện chức năng đăng nhập và đăng ký: 54](#_Toc138378676)

[4.2.2 Giao diện chức năng trạng chủ 58](#_Toc138378677)

[4.2.3 Giao diện tìm kiếm 60](#_Toc138378678)

[4.2.4 Giao diện thông tin và chỉnh sửa thông tin người dùng 62](#_Toc138378679)

[4.2.5 Giao diện thêm phim 64](#_Toc138378680)

[4.2.6 Giao diện yêu thích, đánh giá, bình luận và danh sách phim yêu thích, đánh giá, bình luận 65](#_Toc138378681)

[4.2.7 Giao diện đề xuất phim 68](#_Toc138378682)

[4.2.8 Giao diện xem phim 69](#_Toc138378683)

[**KẾT LUẬN VÀ KIẾN NGHỊ 70**](#_Toc138378684)

[**TÀI LIỆU THAM KHẢO 71**](#_Toc138378685)

## 

DANH MỤC CHỮ VIẾT TẮT

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Từ viết tắc** | **Thuật ngữ** | **Ý nghĩa** |
| 1 | JVM | Java Virtual Machine | Là một phần mềm đọc mã Java và chạy các chương trình Java. |
| 3 | NFC | Near Field Communication | Là một công nghệ không dây cho phép truyền tải dữ liệu giữa các thiết bị gần nhau một cách an toàn và nhanh chóng |
| 4 | SSL | Secure Sockets Layer | Là một giao thức bảo mật được sử dụng để thiết lập một kết nối mật giữa hai máy tính trên Internet |
| 5 | CDN | Content Delivery Network | Là một mạng lưới phân phối nội dung được sử dụng để cung cấp dữ liệu và tài nguyên trên Internet |
| 6 | APK | Android Package Kit | Là một định dạng tệp tin được sử dụng trên hệ điều hành Android để cài đặt và phân phối các ứng dụng cho các thiết bị Android. |
| 7 | SDK | Software Development Kit | Là một bộ công cụ phần mềm được cung cấp cho nhà phát triển |
| 8 | IDE | Integrated Development Environment | Cung cấp cho người dùng một giao diện đồ họa đơn giản và thuận tiện triển phần mềm |

DANH MỤC HÌNH VẼ

[Hình 1 Sơ đồ phân rã chức năng ứng dụng xem phim 31](#_Toc138378582)

[Hình 2 Sơ đồ phân rã chức năng Quản lý ứng dụng 32](#_Toc138378583)

[Hình 3 Sơ đồ luồng dữ liệu 32](#_Toc138378584)

[Hình 4 Biểu đồ UseCase tổng quát 33](#_Toc138378585)

[Hình 5 UseCase Admin 35](#_Toc138378586)

[Hình 6 UseCase Người dùng 35](file:///C:\Users\GIANGOC\Desktop\DATN%201.docx#_Toc138378587)

[Hình 7 Sơ đồ Usecase đăng nhập 36](#_Toc138378588)

[Hình 8 Sơ đồ ER 37](#_Toc138378589)

[Hình 9 Sơ đồ hoạt động chức năng đăng nhập 38](file:///C:\Users\GIANGOC\Desktop\DATN%201.docx#_Toc138378590)

[Hình 10 Sơ đồ hoạt động chức năng đăng ký 39](#_Toc138378591)

[Hình 11 Sơ đồ hoạt động chức năng tìm kiếm phim 40](#_Toc138378592)

[Hình 12 Sơ đồ hoạt động chức năng xem chi tiết phim 41](#_Toc138378593)

[Hình 13 Sơ đồ hoạt động chức năng đánh giá phim 42](#_Toc138378594)

[Hình 14 Sơ đồ hoạt động chức năng bình luận phim 43](#_Toc138378595)

[Hình 15 Sơ đồ hoạt động chức năng yêu thích phim 44](#_Toc138378596)

[Hình 16 Sơ đồ hoạt động chức năng tìm kiếm phim 45](#_Toc138378597)

[Hình 17 Sơ đồ hoạt động chức năng thêm phim 46](#_Toc138378598)

[Hình 18 Sơ đồ hoạt động chức năng xoá phim 47](#_Toc138378599)

[Hình 19 Sơ đồ hoạt động chức năng sửa phim 48](#_Toc138378600)

[Hình 20 : Sơ đồ hoạt động chức năng thêm tài khoản 49](#_Toc138378601)

[Hình 21 Sơ đồ hoạt động chức năng sửa tài khoản 50](#_Toc138378602)

[Hình 22 Sơ đồ hoạt động chức năng xoá tài khoản 51](#_Toc138378603)

[Hình 23 Sơ đồ Diagram 52](file:///C:\Users\GIANGOC\Desktop\DATN%201.docx#_Toc138378604)

[Hình 24 Sơ đồ ClassDiagram 53](#_Toc138378605)

[Hình 25 Giao diện chức năng đăng nhập 55](#_Toc138378606)

[Hình 26 Giao diện chức năng đăng ký 56](#_Toc138378607)

[Hình 27 Giao diện trạng chủ 58](#_Toc138378608)

[Hình 28 Giao diện tìm kiếm 60](#_Toc138378609)

[Hình 29 Giao diện thông tin và chỉnh sửa thông tin người dùng 62](#_Toc138378610)

[Hình 30 Giao diện thêm phim 64](#_Toc138378611)

[Hình 31 Giao diện yêu thích và danh sách phim yêu thích 65](#_Toc138378612)

[Hình 32 Giao diện yêu đánh giá và danh sách đánh giá 66](#_Toc138378613)

[Hình 33 Giao diện bình luận và danh sách phim đã bình luận 67](#_Toc138378614)

[Hình 34 Giao diện đề xuất phim 68](#_Toc138378615)

[Hình 35 Giao diện xem phim 69](#_Toc138378616)

# 

TỔNG QUAN

Tổng quan đề tài

Ngày nay với các công nghệ hiện đại đã thay đổi cuộc sống của chúng ta rất nhiều. Giờ đây với sự phổ biến của thiết bị điện tử và bùng nổ công nghệ, bất kì ai cũng có thể tìm kiếm những thông tin và phim cần thiết chỉ bằng vài thao tác tìm kiếm. Trong đó mô hình được nhiều ưa thích vì nó có thể cho phép người dùng ở bất kì đâu, chỉ cần có kết nối Internet là dễ dàng truy cập vào và sử dụng. Có rất nhiều lĩnh vực ứng dụng như là web xem phim, ứng dụng, thương mại điện tử, y tế, giáo dục và đào tạo, vui chơi giải trí,..

Ứng dụng xem phim là một ứng dụng giải trí cho phép người dùng xem các bộ phim trên điện thoại di động của mình. Các ứng dụng này cung cấp cho người dùng một thư viện phim đa dạng và có thể được sử dụng để xem các bộ phim mới nhất và cũ nhất. Ngoài ra, nhiều ứng dụng xem phim còn cung cấp cho người dùng các tính năng khác như tải xuống phim để xem ngoại tuyến, chia sẻ phim với bạn bè và gia đình và tùy chỉnh chất lượng video. Các ứng dụng xem phim phổ biến hiện nay bao gồm Netflix, Amazon Prime Video, Disney+, HBO Max và Hulu.

Trong đó đồ án này, với mục đích xây dựng một hệ thống xem phim trực tuyến, chúng tôi thiết kế để xây dựng một cho phép người dùng có thể truy cập vào để xem phim trực tuyến thông qua mạng Internet. Hệ thống được lập trình bằng ngôn ngữ Java và hệ thống QTCSDL Firebase với những tính năng cơ bản như đăng ký người dùng, xem phim, tìm phim và các phản hồi ý kiến liên quan.

Mục tiêu nghiên cứu

* Cung cấp cho người dùng một trải nghiệm xem phim trực tuyến tốt hơn và thuận tiện hơn so với các phương thức truyền thống khác như xem phim trên đĩa DVD hoặc điện thoại di động.
* Tạo ra một nền tảng xem phim trực tuyến chất lượng cao với các tính năng như chất lượng hình ảnh và âm thanh tốt nhất có thể, tính năng tìm kiếm và lọc phim thông minh, và tính năng gợi ý phim dựa trên sở thích của người dùng.
* Phát triển các công nghệ mới để cải thiện trải nghiệm xem phim trực tuyến, bao gồm việc sử dụng trí tuệ nhân tạo để cải thiện khả năng gợi ý phim và việc sử dụng công nghệ đám mây để cải thiện khả năng lưu trữ và phân phối phim.
* Hoàn thành nghiên cứu và xây dựng thành công ứng dụng trên trước thời gian báo cáo.

Phạm vi nghiên cứu

* Sử dụng Android Studio Xây dựng giao diện người dùng bằng Layout Editor (Trình chỉnh sửa bố cục).
* Sử dụng ngôn ngữ lập trình Java và Firebase, Nodejs để hoàn thành ứng dụng.

Cách tiếp cận, phương pháp nghiên cứu và thực hiện

Để có thể hiểu rõ, nắm bắt được đề tài và có cái nhìn tổng quan về vấn đề đang nghiên cứu, cần thực hiện:

* Tìm hiểu phân tích và thiết kế hướng đối tượng có cấu trúc.
* Khảo sát tình hình thực tế về nhu cầu của các người dùng khi xem phim trực tuyến và chất lượng các ứng dụng xem phim hiện nay.
* Tìm hiểu thiết kế, quy trình hoạt động của các ứng dụng đó.
* Tổng hợp các tài liệu cần thiết về đề tài.
* Xây dựng và phát triển chương trình theo một quy trình nhất định đã được vạch ra từ trước.
* Tổ chức code dễ đọc, dễ bảo trì.

Cấu trúc báo cáo đồ án tốt nghiệp

**Tổng quan**

* Tổng quan về đề tài đồ án tốt nghiệp.
* Nêu ra mục tiêu và pham vi nghiên cứu.
* Cách tiếp cận đề tài, phương pháp nghiên cứu và thực hiện.

**Chương 1: Cơ sở lý thuyết**

* Giới thiệu về các ngôn ngữ đã sử dụng.
* Giới thiệu về framework.
* Các kỹ thuật và thư viện đã sử dụng trong quá trình thực hiện đề tài

**Chương 2: Phân tích thuật toán content based filtering comender**

* Giới thiệu
* Các bước thực hiện thuật toán Content-Based Filtering
* Các phương pháp tiếp cận trong Content-Based Filtering
* Cách giải và ví dụ về thuật toán Content-Based Filtering
* Kết luận

**Chương 3: Phân tích và thiết kế**

* Yêu cầu hệ thống.
* Sơ đồ rã chức năng.
* Sơ đồ luồng dữ liệu.
* Sơ đồ usecase.
* Sơ đồ hoạt động.
* Mô hình ERD.
* Cơ sở dữ liệu.

**Chương 4: Triển khai chương trình**

* Giao diện ứng dụng.

**Kết quả - Kiến nghị**

* Kết luận.
* Tài liệu tham khảo.

1. CƠ SỞ LÝ THUYẾT
   1. Giới thiệu về Java trong Android Studio
      1. Lịch sử phát triển của Java trong Android Studio:

Lịch sử phát triển của Java trong Android Studio được bắt đầu từ khi Google mua lại Android Inc. vào năm 2005. Khi đó, Android Inc. đã sử dụng ngôn ngữ lập trình Java để phát triển hệ điều hành Android. Sau đó, Google phát triển Android Studio là một công cụ phát triển tích hợp (IDE) cho phát triển ứng dụng Android, và Java tiếp tục được sử dụng làm ngôn ngữ lập trình chính cho phát triển ứng dụng trong Android Studio.

Trong những năm đầu, để phát triển ứng dụng Android, người lập trình cần phải sử dụng Eclipse IDE và plugin ADT (Android Development Tools) để lập trình bằng Java. Tuy nhiên, vào năm 2013, Google đã chính thức ra mắt Android Studio, công cụ phát triển tích hợp mới cho phát triển ứng dụng Android, với nhiều tính năng mới và cải tiến so với Eclipse.

Android Studio được xây dựng trên nền tảng IntelliJ IDEA của JetBrains, và hỗ trợ nhiều tính năng tiên tiến cho phát triển ứng dụng Android, bao gồm thiết kế giao diện người dùng, lập trình Java và Kotlin, gỡ lỗi, thử nghiệm và triển khai ứng dụng.

Trong quá trình phát triển, Java trong Android Studio đã trải qua nhiều cải tiến và cập nhật để tối ưu hóa hiệu suất và tính năng cho phát triển ứng dụng Android. Ngoài ra, Kotlin, một ngôn ngữ lập trình khác, cũng được hỗ trợ trong Android Studio và được xem là một sự lựa chọn thay thế cho Java trong phát triển ứng dụng Android. Tuy nhiên, Java vẫn là một ngôn ngữ lập trình chính trong Android Studio và được sử dụng rộng rãi bởi cộng đồng lập trình viên Android.

* + 1. Đặc điểm của Android

Android là hệ điều hành mã nguồn mở được phát triển bởi Google và được sử dụng rộng rãi trên các thiết bị di động như điện thoại thông minh, máy tính bảng, đồng hồ thông minh, TV và các thiết bị IoT. Dưới đây là một số đặc điểm của Android:

* Mã nguồn mở: Android là một hệ điều hành mã nguồn mở, cho phép các nhà phát triển có thể tùy chỉnh và phát triển theo nhu cầu của họ.
* Đa nền tảng: Android là một hệ điều hành đa nền tảng, hỗ trợ cho nhiều loại thiết bị di động khác nhau.
* Tích hợp dịch vụ Google: Android tích hợp các dịch vụ của Google như Google Maps, Google Drive, Google Play Store, Google Assistant, v.v. giúp người dùng dễ dàng truy cập vào các dịch vụ này.
* Hỗ trợ ứng dụng thứ ba: Android cho phép người dùng cài đặt và sử dụng các ứng dụng thứ ba bên ngoài Google Play Store.
* Tùy chỉnh giao diện người dùng: Android cho phép người dùng tùy chỉnh giao diện theo ý muốn, từ hình nền cho đến bố cục các ứng dụng.
* Hỗ trợ kết nối mạng: Android hỗ trợ nhiều loại kết nối mạng như Wi-Fi, 3G, 4G, Bluetooth, NFC, v.v.
* Hỗ trợ đa nhiệm: Android cho phép người dùng sử dụng nhiều ứng dụng cùng một lúc và chuyển đổi giữa chúng dễ dàng.
* Hỗ trợ đa phương tiện: Android hỗ trợ nhiều định dạng âm thanh, video và hình ảnh khác nhau.
* Bảo mật: Android được thiết kế để đảm bảo tính bảo mật cho người dùng và các dữ liệu của họ.
* Cộng đồng phát triển mạnh mẽ: Android có cộng đồng phát triển lớn và nhiều tài liệu hướng dẫn sẵn có, giúp các nhà phát triển dễ dàng phát triển ứng dụng trên nền tảng này.

**Ưu điểm của Android Studio**

* Tích hợp đầy đủ: Android Studio tích hợp tất cả các công cụ cần thiết để phát triển ứng dụng Android, bao gồm cả bộ SDK, trình biên dịch và máy ảo.
* Tính linh hoạt: Android Studio hỗ trợ nhiều ngôn ngữ lập trình khác nhau, cho phép bạn phát triển ứng dụng bằng Java, Kotlin và nhiều ngôn ngữ khác.
* Công cụ tốt cho thiết kế giao diện: Android Studio cung cấp cho người dùng một số công cụ thiết kế giao diện tốt như Layout Editor, giúp thiết kế giao diện trở nên dễ dàng hơn.
* Được hỗ trợ bởi cộng đồng lớn: Android Studio được hỗ trợ bởi cộng đồng lớn của các nhà phát triển Android, điều này có nghĩa là bạn có thể tìm thấy hỗ trợ và tài liệu rất nhiều trên mạng.
* Hỗ trợ kiểm tra và gỡ lỗi chuyên nghiệp: Android Studio cung cấp cho người dùng nhiều công cụ để kiểm tra và gỡ lỗi ứng dụng, giúp bạn tìm ra và sửa lỗi nhanh chóng.

**Nhược điểm của Android Studio**

* Yêu cầu tài nguyên hệ thống cao: Android Studio là một phần mềm khá nặng và yêu cầu một hệ thống mạnh để chạy mượt mà.
* Học tập khá phức tạp: Android Studio đòi hỏi người sử dụng có kiến thức về lập trình, điều này có thể gây khó khăn cho những người mới bắt đầu phát triển ứng dụng.
* Đôi khi chậm và không ổn định: Mặc dù Android Studio đã được cải tiến rất nhiều, nhưng vẫn có thể gặp phải các vấn đề về tốc độ và ổn định.
* Không hỗ trợ tốt cho các phiên bản Android cũ: Android Studio hỗ trợ tốt nhất cho các phiên bản Android mới nhất, điều này có nghĩa là nếu bạn muốn phát triển ứng dụng cho các phiên bản cũ hơn của Android, bạn sẽ phải tìm kiếm giải pháp khác.
  1. Nội dung cơ bản về Java trong Android
     1. Đặc điểm của Java trong Android Studio:
* Đa nền tảng: Java là một ngôn ngữ lập trình đa nền tảng, có thể sử dụng trên nhiều hệ điều hành khác nhau bao gồm cả Android, iOS và máy tính để bàn.
* Tính đồng nhất: Java được sử dụng rộng rãi trong cộng đồng lập trình viên Android, do đó có sự đồng nhất trong cách lập trình và chia sẻ mã giữa các dự án khác nhau.
* Tính bảo mật: Java được phát triển với mục đích tăng cường tính bảo mật, đặc biệt là trong việc xử lý dữ liệu đầu vào và tiếp cận cơ sở dữ liệu.
* Tính linh hoạt: Java cung cấp nhiều tính năng linh hoạt cho phát triển ứng dụng Android, bao gồm hỗ trợ các thư viện bên thứ ba, khả năng xử lý dữ liệu theo thời gian thực và tính tương thích với các công nghệ khác.
* Tính hiệu suất: Java được tối ưu hóa để đạt hiệu suất cao, đặc biệt là trong việc xử lý dữ liệu lớn và tương tác với các cơ sở dữ liệu.
* Tính khả dụng cao: Java là một ngôn ngữ lập trình rất phổ biến và có nhiều tài liệu và cộng đồng hỗ trợ, do đó rất dễ dàng để tìm kiếm giải pháp khi gặp vấn đề trong quá trình phát triển ứng ụng Android.
  + 1. Ứng dụng của Java trong Android Studio

Java là ngôn ngữ lập trình chính được sử dụng trong Android Studio để phát triển ứng dụng Android. Dưới đây là một số ứng dụng của Java trong Android Studio:

* Xây dựng giao diện người dùng: Java được sử dụng để xây dựng các thành phần giao diện người dùng trong Android, bao gồm TextView, EditText, Button, ImageView và các thành phần khác.
* Quản lý dữ liệu: Java được sử dụng để lưu trữ, truy xuất và xử lý dữ liệu trong ứng dụng Android. Nó được sử dụng để kết nối với cơ sở dữ liệu SQLite, truy vấn dữ liệu và xử lý các tác vụ liên quan đến dữ liệu.
* Xử lý sự kiện: Java được sử dụng để xử lý các sự kiện người dùng, bao gồm cả các sự kiện cảm ứng màn hình, như chạm và vuốt trên màn hình, và các sự kiện khác như nhấn nút.
* Xử lý thông báo: Java được sử dụng để xử lý các thông báo trong ứng dụng Android, bao gồm cả các thông báo từ hệ thống và thông báo tùy chỉnh.
* Kết nối mạng: Java được sử dụng để kết nối với các dịch vụ web và truy cập các tài nguyên mạng trong ứng dụng Android.
* Tương tác với các thành phần khác: Java được sử dụng để tương tác với các thành phần khác trong Android, bao gồm Service, BroadcastReceiver .
* Xử lý ngoại lệ: Java được sử dụng để xử lý các ngoại lệ trong ứng dụng Android, bao gồm các lỗi hệ thống và các lỗi phát sinh trong quá trình thực thi ứng dụng.
  + 1. Khái niệm cơ bản về Java
* Ngôn ngữ lập trình: Java là một ngôn ngữ lập trình, được sử dụng để tạo ra các ứng dụng và phần mềm trên nhiều nền tảng khác nhau.
* Cấu trúc: Java có cấu trúc chương trình được xây dựng trên các lớp (class) và đối tượng (object), với sự hỗ trợ của các khái niệm như kế thừa (inheritance), đa hình (polymorphism) và trừu tượng hóa (abstraction).
* Độc lập nền tảng: Java là một ngôn ngữ độc lập nền tảng, có thể chạy trên nhiều hệ điều hành khác nhau mà không cần thay đổi mã nguồn.
* Cú pháp: Java có cú pháp đơn giản và dễ hiểu, với nhiều từ khóa và quy tắc viết code.
* Thư viện: Java cung cấp nhiều thư viện chuẩn để hỗ trợ phát triển ứng dụng, bao gồm các thư viện đồ họa, xử lý dữ liệu và kết nối mạng.
* JVM: Java Virtual Machine (JVM) là một máy ảo được sử dụng để chạy các ứng dụng Java trên các hệ điều hành khác nhau.
* Xử lý ngoại lệ: Java có cơ chế xử lý ngoại lệ (exception handling) để quản lý các lỗi và ngoại lệ trong quá trình thực thi chương trình.
* Bảo mật: Java được thiết kế để đảm bảo tính bảo mật và an toàn trong quá trình thực thi chương trình.
* Hỗ trợ đa luồng: Java hỗ trợ đa luồng (multithreading), cho phép chương trình thực thi nhiều tác vụ cùng một lúc.
* Phát triển ứng dụng web: Java cũng được sử dụng để phát triển các ứng dụng web, với các khái niệm như Servlet, JSP và Enterprise JavaBeans (EJB).
  + 1. Lập trình hướng đối tượng trong Java

Lập trình hướng đối tượng (OOP) là một phương pháp lập trình phổ biến trong Java, nơi các đối tượng được tạo ra để đại diện cho các thực thể trong thế giới thực.

* Lớp (class): Lớp là một mô tả chung về một nhóm đối tượng có các thuộc tính và phương thức chung. Ví dụ, một lớp có thể đại diện cho một đối tượng như con mèo, với các thuộc tính như tên, màu sắc và tuổi.
* Đối tượng (object): Đối tượng là một thể hiện của một lớp. Ví dụ, một đối tượng con mèo có thể có các giá trị cụ thể cho các thuộc tính như tên là "Miu Miu", màu sắc là "đen" và tuổi là "3".
* Kế thừa (inheritance): Kế thừa là một khái niệm trong OOP cho phép một lớp con (subclass) kế thừa các thuộc tính và phương thức từ một lớp cha (superclass).
* Đa hình (polymorphism): Đa hình là một khái niệm trong OOP cho phép một đối tượng của một lớp có thể được sử dụng như một đối tượng của một lớp khác.
* Trừu tượng (abstraction): Trừu tượng là một khái niệm trong OOP cho phép che giấu các chi tiết cài đặt của một đối tượng và tập trung vào các tính năng của đối tượng đó.
* Giao diện (interface): Giao diện là một khái niệm trong OOP cho phép định nghĩa các phương thức mà các lớp khác có thể triển khai.
  + 1. Thành phần trong Android Studio sử dụng Java

Activity là thành phần chính của một ứng dụng Android. Nó đại diện cho một màn hình giao diện người dùng, mà người dùng có thể tương tác để thực hiện các hành động nhất định. Activity được viết bằng Java và là một lớp Java.

Service là một thành phần được sử dụng để thực hiện các tác vụ dài hạn mà không cần tương tác với người dùng. Nó được sử dụng để thực hiện các tác vụ nền trong khi người dùng đang sử dụng các ứng dụng khác. Service được viết bằng Java và là một lớp Java.

Broadcast Receiver là một thành phần được sử dụng để nhận các thông báo từ hệ thống hoặc ứng dụng khác. Nó được sử dụng để xử lý các sự kiện hệ thống như kết nối mạng thay đổi, pin yếu và các sự kiện khác. Broadcast Receiver được viết bằng Java và là một lớp Java.

Content Provider là một thành phần được sử dụng để quản lý và chia sẻ dữ liệu giữa các ứng dụng khác nhau. Nó cung cấp cơ chế để truy cập và thao tác với dữ liệu được lưu trữ trong cơ sở dữ liệu của ứng dụng. Content Provider được viết bằng Java và là một lớp Java.

Fragment là một thành phần được sử dụng để tạo ra một phần của giao diện người dùng trong một Activity. Nó cho phép bạn tạo ra giao diện người dùng động và linh hoạt hơn. Fragment được viết bằng Java và là một lớp Java.

Layout là một thành phần được sử dụng để định dạng giao diện người dùng. Nó đại diện cho cách các thành phần khác nhau như TextView, EditText, Button... được sắp xếp trên màn hình. Layout được viết bằng XML, tuy nhiên, Java cũng được sử dụng để thao tác với các thành phần của Layout.

View là một thành phần cơ bản của giao diện người dùng. Nó đại diện cho các thành phần như TextView, EditText, Button, CheckBox... được hiển thị trên màn hình. ViewGroup là một lớp cha của các thành phần View và được sử dụng để sắp xếp các thành phần View trên màn hình. Cả View và ViewGroup được viết bằng Java và là các lớp Java.

Style và Theme là hai thành phần được sử dụng để định dạng giao diện người dùng. Style được sử dụng để định dạng một thành phần cụ thể, trong khi Theme được sử dụng để định dạng toàn bộ ứng dụng. Cả Style và Theme được viết bằng XML.

Intent là một thành phần được sử dụng để kết nối các thành phần khác nhau của ứng dụng Android. Nó được sử dụng để chuyển dữ liệu và thực hiện các hành động nhất định. Intent được viết bằng Java và là một lớp Java.

AsyncTask và Thread là hai thành phần được sử dụng để thực hiện các tác vụ nền trong ứng dụng Android. AsyncTask được sử dụng để thực hiện các tác vụ nền nhỏ nhẹ, trong khi Thread được sử dụng để thực hiện các tác vụ nền nặng hơn và cần nhiều thời gian hơn. Cả AsyncTask và Thread đều được viết bằng Java và là các lớp Java.

**Ưu điểm**

* Tính đa nền tảng: Java là một ngôn ngữ lập trình đa nền tảng, cho phép các nhà phát triển viết một lần và chạy trên nhiều nền tảng khác nhau.
* Tính bảo mật: Java có các tính năng bảo mật để giúp bảo vệ ứng dụng Android của bạn.
* Tính khả năng mở rộng: Java có tính khả năng mở rộng cao, giúp các nhà phát triển có thể thêm các tính năng mới vào ứng dụng một cách dễ dàng.
* Tính dễ học: Java là một ngôn ngữ lập trình dễ học và dễ sử dụng, đặc biệt là đối với những người mới bắt đầu học lập trình.
* Tính hiệu suất: Java có tính hiệu suất tốt, giúp ứng dụng Android chạy mượt mà và nhanh chóng.

**Nhược điểm**

* Tính khả năng mở rộng: Mặc dù Java có tính khả năng mở rộng cao, nhưng cũng có thể gặp phải các vấn đề về quản lý mã nguồn khi ứng dụng phát triển.
* Tính tương thích: Một số vấn đề tương thích có thể xảy ra khi sử dụng Java trong Android, đặc biệt là đối với các phiên bản Android cũ hơn.
* Tính cồng kềnh: Một số người cho rằng Java có tính cồng kềnh và không linh hoạt trong việc phát triển ứng dụng Android.
* Tính tốc độ: Một số người cho rằng Java không có tốc độ cao nhất trong việc phát triển ứng dụng Android.
  1. Tổng quan về Firebase và Firebase Function
     1. Tổng quan về Firebase

Firebase được thành lập năm 2011 bởi James Tamplin và Andrew Lee. Ban đầu, Firebase là một nền tảng cho phép các nhà phát triển xây dựng ứng dụng di động nhanh chóng và dễ dàng bằng cách cung cấp các công cụ để lưu trữ và đồng bộ dữ liệu.

Năm 2014, Firebase đã được mua lại bởi Google với mục đích cung cấp cho các nhà phát triển các công cụ và dịch vụ để xây dựng các ứng dụng di động và web hiệu quả. Sau đó, Firebase đã trở thành một phần của Google Cloud Platform.

Firebase là một nền tảng phát triển ứng dụng di động và web của Google, cung cấp các công cụ và dịch vụ để giúp các nhà phát triển xây dựng các ứng dụng nhanh chóng và dễ dàng hơn. Firebase bao gồm nhiều tính năng như lưu trữ dữ liệu (Realtime Database và Cloud Firestore), xác thực người dùng, phân tích, thông báo đẩy, hosting và nhiều tính năng khác

**Ưu điểm**

* Tính dễ sử dụng: Firebase cung cấp các công cụ và dịch vụ giúp các nhà phát triển xây dựng các ứng dụng di động và web một cách nhanh chóng và dễ dàng hơn.
* Tính linh hoạt: Firebase cung cấp nhiều tính năng và dịch vụ để giúp các nhà phát triển xây dựng các ứng dụng phức tạp và linh hoạt.
* Tính khả dụng: Firebase cung cấp các tính năng và dịch vụ mạnh mẽ để giúp các nhà phát triển xây dựng các ứng dụng di động và web phức tạp.
* Tính tiết kiệm chi phí: Firebase cho phép bạn trả tiền theo mức sử dụng, giúp bạn tiết kiệm chi phí so với việc tự quản lý cơ sở hạ tầng máy chủ.
* Tính bảo mật: Firebase cung cấp các tính năng bảo mật để giúp bảo vệ dữ liệu của bạn.
* Tính khả năng mở rộng: Firebase cung cấp các tính năng và dịch vụ để giúp các ứng dụng của bạn có tính khả năng mở rộng cao.

**Nhược điểm:**

* Giới hạn: Firebase có giới hạn về số lượng request và lưu trữ trong một thời gian nhất định.
* Tính linh hoạt về cấu hình: Firebase không cung cấp cho người dùng quyền truy cập root vào các cấu hình máy chủ.
* Tính khả dụng của định dạng dữ liệu: Firebase chỉ hỗ trợ một số định dạng dữ liệu như JSON và XML, giới hạn khả năng tùy chỉnh của ứng dụng.
* Tính ổn định: Firebase có thể gặp phải các vấn đề về ổn định trong quá trình hoạt động.
  + 1. Tổng quan về Firebase Function

Firebase Functions là một dịch vụ tích hợp trong Firebase, cho phép bạn viết và triển khai các hàm backend chạy trên đám mây. Nó giúp cho các nhà phát triển ứng dụng có thể xử lý các tác vụ phức tạp, giảm thiểu tải cho máy chủ của bạn và tăng tính linh hoạt trong việc quản lý ứng dụng của bạn.

Firebase Functions cho phép bạn viết mã ứng dụng của mình bằng các ngôn ngữ lập trình như JavaScript, TypeScript và Python. Sau đó, bạn có thể triển khai mã này trực tiếp trên Firebase Cloud Functions để thực thi các tác vụ như tính toán, gửi email, xử lý định dạng dữ liệu, v.v.

**Ưu điểm**

* Dễ dàng triển khai: Firebase Functions cho phép người dùng triển khai các hàm backend một cách dễ dàng và nhanh chóng, đặc biệt là khi kết hợp với các dịch vụ khác của Firebase.
* Tiết kiệm chi phí: Firebase Functions chỉ tính phí dựa trên lượng tài nguyên sử dụng, số lần kích hoạt và lưu lượng mạng, giúp người dùng tiết kiệm chi phí cho việc triển khai và vận hành các hàm backend.
* Tính linh hoạt: Firebase Functions cho phép người dùng viết mã ứng dụng bằng các ngôn ngữ lập trình khác nhau và triển khai chúng trên nền tảng đám mây của Firebase.
* Tích hợp tốt với các dịch vụ khác của Firebase: Firebase Functions tích hợp tốt với các dịch vụ khác của Firebase như Realtime Database, Cloud Firestore, Firebase Authentication, Firebase Hosting, Firebase Storage, Firebase Analytics và Firebase Remote Config.
* Độ tin cậy cao: Firebase Functions được xây dựng trên cơ sở hạ tầng đám mây của Google, giúp đảm bảo tính tin cậy và khả năng sẵn sàng cao của dịch vụ.

**Nhược điểm:**

* Thời gian khởi động chậm: Firebase Functions có thời gian khởi động ban đầu khá chậm, đặc biệt là khi bạn triển khai nhiều hàm cùng lúc.
* Giới hạn lưu trữ: Firebase Functions chỉ cho phép lưu trữ dữ liệu tạm thời, giới hạn lượng dữ liệu có thể lưu trữ và thời gian lưu trữ.
* Khó khăn trong việc debug: Do Firebase Functions chạy trên đám mây, việc debug có thể khó khăn hơn so với việc debug trên máy tính cá nhân.
  1. Tổng quan về Node.js

Node.js là một nền tảng phát triển ứng dụng web và backend được xây dựng trên JavaScript. Nó cho phép các nhà phát triển sử dụng JavaScript để viết các ứng dụng web, thay vì phải sử dụng các ngôn ngữ khác như PHP hoặc Ruby. Node.js được phát triển bởi Ryan Dahl vào năm 2009 và nhanh chóng trở thành một trong những công nghệ phát triển ứng dụng backend phổ biến nhất hiện nay.

Node.js sử dụng mô hình xử lý không đồng bộ (asynchronous) để xử lý các yêu cầu và phản hồi (request/response) trong khi giữ cho các tiến trình khác vẫn hoạt động. Điều này làm cho Node.js rất hiệu quả trong việc xử lý các ứng dụng web với lưu lượng truy cập lớn.

Node.js cũng cung cấp một số module tích hợp sẵn, cho phép các nhà phát triển xây dựng các ứng dụng web hoàn chỉnh với các chức năng như routing, xử lý yêu cầu HTTP, xử lý tệp, xử lý dữ liệu JSON và nhiều tính năng khác.

Node.js cũng có thể được sử dụng để xây dựng các ứng dụng desktop và mobile, nhờ các framework như Electron và React Native.

Node.js là một công nghệ phát triển ứng dụng web và backend phổ biến, được sử dụng bởi hàng ngàn các ứng dụng web và các công ty lớn trên toàn thế giới.

**Đặc điểm của Node.js:**

* Single-threaded: Node.js là single-threaded, điều này cho phép nó xử lý nhiều yêu cầu đồng thời mà không tốn quá nhiều tài nguyên hệ thống.
* Asynchronous: Node.js sử dụng mô hình lập trình bất đồng bộ để tối đa hóa hiệu suất. Việc sử dụng callback hoặc promise cho phép Node.js xử lý nhiều yêu cầu mà không bị chặn bởi các hoạt động đồng bộ.
* Event-driven: Node.js sử dụng mô hình lập trình dựa trên sự kiện, cho phép các ứng dụng được xây dựng để phản ứng nhanh chóng với các sự kiện như yêu cầu HTTP, các kết nối cơ sở dữ liệu, hoặc các sự kiện khác.
* Cross-platform: Node.js được hỗ trợ trên nhiều nền tảng khác nhau bao gồm Windows, Linux và macOS.
* Lightweight and scalable: Node.js được thiết kế để là một nền tảng nhẹ và có thể mở rộng linh hoạt. Nó có thể xử lý hàng ngàn kết nối đồng thời và có thể được triển khai trên nhiều máy chủ để tạo thành một hệ thống phân tán.
* NPM: Node.js đi kèm với npm - một hệ thống quản lý gói phổ biến, cho phép các nhà phát triển dễ dàng chia sẻ và tái sử dụng mã nguồn mở.

**Ưu điểm của Node.js:**

* Hiệusuất cao: Node.js được thiết kế để xử lý các yêu cầu và phản hồi trong khi giữ cho các tiến trình khác vẫn hoạt động, điều này giúp tối ưu hóa hiệu suất của ứng dụng web.
* Xử lý không đồng bộ: Node.js sử dụng mô hình xử lý không đồng bộ, cho phép xử lý đồng thời nhiều yêu cầu mà không cần đợi hoàn tất từng yêu cầu trước.
* Dễ dàng mở rộng: Node.js cho phép mở rộng dễ dàng bằng cách sử dụng các module và thư viện và cũng hỗ trợ các kiến trúc mở rộng như microservices.
* Cộng đồng phát triển lớn: Node.js có một cộng đồng phát triển lớn và năng động, với nhiều module và thư viện hữu ích được phát triển liên tục.
* Sử dụng JavaScript: Với Node.js, các nhà phát triển có thể sử dụng JavaScript để phát triển cả các ứng dụng frontend và backend, giúp giảm thời gian học và tăng tính hiệu quả.

**Nhược điểm của Node.js:**

* Không thích hợp cho các ứng dụng nặng: Node.js không phù hợp cho các ứng dụng có lưu lượng truy cập lớn và yêu cầu xử lý dữ liệu nặng.
* Không hỗ trợ đa luồng: Node.js không hỗ trợ đa luồng, điều này có thể dẫn đến khó khăn trong việc xử lý các tác vụ phức tạp và yêu cầu sử dụng các module và thư viện để giải quyết vấn đề này.
* Khó khăn trong việc quản lý callback: Vì Node.js sử dụng mô hình xử lý không đồng bộ và callback, việc quản lý callback có thể gây khó khăn cho các nhà phát triển.
* Module không đồng nhất: Node.js có nhiều module và thư viện được phát triển bởi cộng đồng, điều này dẫn đến khó khăn trong việc đảm bảo tính đồng nhất và chất lượng của các module và thư viện này.
  1. Tổng quan về API

API (Application Programming Interface) là một tập hợp các quy tắc, giao thức và công nghệ được sử dụng để tương tác với một ứng dụng hoặc hệ thống khác. API cho phép các ứng dụng khác nhau trao đổi dữ liệu và chức năng với nhau một cách đơn giản và hiệu quả.

**Đặc điểm về API:**

* Standardization: API thường được thiết kế để tuân thủ các chuẩn định dạng và giao thức nhất định, giúp các ứng dụng tương tác với nhau một cách dễ dàng và hiệu quả.
* Interoperability: API cho phép các ứng dụng khác nhau tương tác với nhau bất kể hệ thống hoặc ngôn ngữ lập trình được sử dụng. Vì vậy, các ứng dụng có thể được phát triển độc lập và sau đó kết hợp lại với nhau thông qua API.
* Scalability: API có thể được sử dụng để kết nối các ứng dụng với số lượng lớn người dùng hoặc đối tác. Vì vậy, nó cũng cung cấp khả năng mở rộng cao để hỗ trợ các kịch bản tăng trưởng hoặc tải.
* Security: API cung cấp các tính năng bảo mật để đảm bảo rằng dữ liệu được truyền đi và được truy cập chỉ bởi các ứng dụng được ủy quyền và các người dùng được phép.
* Documentation and testing: API cung cấp tài liệu và công cụ kiểm thử để giúp các nhà phát triển dễ dàng sử dụng và kiểm tra tính năng của API.
* Versioning: API có thể được phát triển và cập nhật mà không ảnh hưởng đến các ứng dụng sử dụng nó bằng cách sử dụng các phiên bản khác nhau của API.

**Ưu điểm của API:**

* Tính linh hoạt: API cho phép các ứng dụng khác nhau tương tác với nhau một cách độc lập và linh hoạt. Các ứng dụng có thể được phát triển trên các nền tảng khác nhau và vẫn có thể sử dụng API để tương tác với nhau.
* Tính khả chuyển: API cho phép các ứng dụng khác nhau tương tác với nhau một cách hiệu quả và chính xác, giúp giảm thiểu sự nhầm lẫn và tăng tính khả chuyển của hệ thống.
* Tăng tốc độ phát triển: API cho phép các nhà phát triển đơn giản hóa quá trình tương tác giữa các ứng dụng. Điều này giúp giảm thời gian và chi phí phát triển, đồng thời tăng tốc độ triển khai ứng dụng.

**Nhược điểm của API:**

* Độ phức tạp: Việc phát triển và triển khai API có thể rất phức tạp và đòi hỏi kiến thức chuyên môn cao.
* Bảo mật: API có thể trở thành một điểm yếu trong hệ thống, nếu không được thiết kế và triển khai đúng cách, có thể dẫn đến các vấn đề bảo mật.
* Không ổn định: API có thể thay đổi theo thời gian, do đó các ứng dụng phải được thiết kế để có thể xử lý các thay đổi này một cách linh hoạt.

.

1. PHÂN TÍCH THUẬT TOÁN CONTENT BASED FILTERING
   1. Giới thiệu về Content-Based Filtering

Content-Based Filtering là một thuật toán lọc cộng tác được sử dụng trong hệ thống khuyến nghị để đưa ra các khuyến nghị dựa trên nội dung của các mục đã được người dùng đánh giá hoặc xem trước đó. Nó dựa trên việc phân tích nội dung của các mục để đưa ra các khuyến nghị dựa trên sự tương đồng của các mục với nhau.

Content-Based Filtering áp dụng các kỹ thuật xử lý ngôn ngữ tự nhiên để phân tích nội dung của các mục và trích xuất các đặc trưng của chúng để xây dựng mô hình dự đoán. Các đặc trưng này có thể bao gồm các thuộc tính như tác giả, thể loại, chủ đề, từ khóa, mô tả, v.v.

Content-Based Filtering thường được sử dụng trong các hệ thống khuyến nghị cho các sản phẩm, phim, sách, bài báo, v.v. Nó cho phép các hệ thống khuyến nghị đưa ra các khuyến nghị dựa trên sở thích cá nhân của người dùng và cải thiện trải nghiệm người dùng.

* 1. Khái niệm và đặc trưng của Content-Based Filtering

**Khái niệm của Content-Based Filtering:**

Content-Based Filtering là một thuật toán lọc cộng tác trong hệ thống khuyến nghị, được sử dụng để đưa ra các khuyến nghị dựa trên nội dung của các mục đã được người dùng đánh giá hoặc xem trước đó. Nó dựa trên việc phân tích nội dung của các mục để đưa ra các khuyến nghị dựa trên sự tương đồng của các mục với nhau.

**Đặc trưng của Content-Based Filtering:**

* Phân tích nội dung: Content-Based Filtering sử dụng các kỹ thuật phân tích nội dung để trích xuất các đặc trưng của các mục, bao gồm các thuộc tính như tác giả, thể loại, chủ đề, từ khóa, mô tả, v.v.
* Đánh giá sự tương đồng: Content-Based Filtering sử dụng các phương pháp đo độ tương đồng để so sánh các mục với nhau, bao gồm phương pháp đo độ tương đồng dựa trên cosine, Euclidean distance, Pearson correlation, v.v.
* Xây dựng mô hình dự đoán: Content-Based Filtering sử dụng các đặc trưng của các mục để xây dựng mô hình dự đoán, dựa trên các thuật toán như k-Nearest Neighbors, Naive Bayes, Decision Trees, v.v.
* Tính cá nhân hóa: Content-Based Filtering cho phép tính cá nhân hóa trong khuyến nghị bằng cách đưa ra các khuyến nghị dựa trên sở thích cá nhân của người dùng.
* Độc lập với dữ liệu người dùng: Content-Based Filtering không yêu cầu thông tin về các người dùng khác nhau, chỉ cần dữ liệu về các mục trong hệ thống.
* Hạn chế trong việc đưa ra các khuyến nghị mới: Content-Based Filtering có hạn chế trong việc đưa ra các khuyến nghị mới cho người dùng, do nó chỉ đưa ra các khuyến nghị dựa trên các mục tương tự với các mục đã được người dùng đánh giá hoặc xem trước đó.

**Ưu điểm của Content-Based Filtering:**

* Độc lập với dữ liệu người dùng: Content-Based Filtering không yêu cầu thông tin về các người dùng khác nhau, chỉ cần dữ liệu về các mục trong hệ thống.
* Tính cá nhân hóa: Content-Based Filtering cho phép tính cá nhân hóa trong khuyến nghị bằng cách đưa ra các khuyến nghị dựa trên sở thích cá nhân của người dùng.
* Hiệu quả với các mục mới: Content-Based Filtering có thể đưa ra các khuyến nghị cho các mục mới trong hệ thống khi có đủ thông tin về các đặc trưng của chúng.
* Dễ triển khai và thực hiện: Content-Based Filtering có tính đơn giản và dễ dàng triển khai trong các hệ thống khuyến nghị.
* Không bị ảnh hưởng bởi các đánh giá sai lệch: Content-Based Filtering không bị ảnh hưởng bởi các đánh giá sai lệch của người dùng.

**Nhược điểm của Content-Based Filtering:**

* Hạn chế trong việc đưa ra các khuyến nghị mới: Content-Based Filtering có hạn chế trong việc đưa ra các khuyến nghị mới cho người dùng, do nó chỉ đưa ra các khuyến nghị dựa trên các mục tương tự với các mục đã được người dùng đánh giá hoặc xem trước đó.
* Cần nhiều dữ liệu và đặc trưng: Content-Based Filtering cần có nhiều dữ liệu và đặc trưng của các mục để đưa ra các khuyến nghị chính xác.
* Không hiệu quả với các mục phức tạp: Content-Based Filtering không hiệu quả với các mục có nội dung phức tạp và không thể phân tích một cách đầy đủ.
* Không đưa ra khuyến nghị xã hội: Content-Based Filtering không đưa ra khuyến nghị dựa trên các mối quan hệ xã hội giữa các người dùng.
* Khó đo lường độ tương đồng giữa các mục: Content-Based Filtering có thể gặp khó khăn trong việc đo lường độ tương đồng giữa các mục nếu chúng có nội dung phức tạp và cần phân tích chi tiết hơn.
  1. Nguyên lý hoạt động của Content-Based Filtering:
     1. Các bước thực hiện của Content-Based Filtering:
* Tiền xử lý dữ liệu: Trước khi bắt đầu phân tích nội dung, dữ liệu cần được tiền xử lý để loại bỏ các dữ liệu không cần thiết và chuẩn hóa các dữ liệu còn lại.
* Trích xuất đặc trưng: Sau khi tiền xử lý dữ liệu, các đặc trưng của các mục cần được trích xuất. Các đặc trưng này bao gồm các thuộc tính như tác giả, thể loại, chủ đề, từ khóa, mô tả, v.v.
* Đánh giá độ tương đồng giữa các mục: Sau khi đã trích xuất các đặc trưng, các đặc trưng này cần được sử dụng để đánh giá độ tương đồng giữa các mục. Các phương pháp đánh giá độ tương đồng bao gồm phương pháp đo độ tương đồng dựa trên cosine, Euclidean distance, Pearson correlation, v.v.
* Xây dựng mô hình dự đoán: Sau khi đã đánh giá độ tương đồng giữa các mục, các đặc trưng này có thể được sử dụng để xây dựng mô hình dự đoán. Các thuật toán xây dựng mô hình dự đoán bao gồm k-Nearest Neighbors, Naive Bayes, Decision.
* Đánh giá và kiểm tra mô hình: Cuối cùng, mô hình cần được đánh giá và kiểm tra để đảm bảo rằng nó đưa ra các khuyến nghị chính xác cho người dùng. Các phương pháp đánh giá và kiểm tra bao gồm phân chia dữ liệu thành hai phần (train và test) và sử dụng các phương pháp đánh giá khác nhau như accuracy, precision, recall, F1-score, ROC curve, v.v.
  + 1. Các thuật toán phổ biến trong Content-based Filtering:
* TF-IDF (Term Frequency-Inverse Document Frequency): tính toán trọng số của các từ khóa trong văn bản để biểu diễn nội dung.
* Word2Vec: biểu diễn các từ thành các vector đặc trưng dựa trên ngữ cảnh xuất hiện của chúng.
* Decision Trees: sử dụng cây quyết định để học mô hình content-based và dự đoán sở thích của người dùng.
* Bayesian Networks: sử dụng mạng Bayesian để học mô hình content-based và dự đoán sở thích của người dùng.
* Neural Networks: sử dụng mạng neuron để học mô hình content-based và dự đoán sở thích củangười dùng.
  + 1. Phương pháp gợi ý dựa theo nội dung

Ý tưởng của thuật toán này là, từ thông tin mô tả của item, biểu diễn item dưới dạng vec-tơ thuộc tính. Sau đó dùng các vec-tơ này để học mô hình của mỗi user, là ma trận trọng số của user với mỗi item.

Như vậy, thuật toán content-based gồm 2 bước:

* Bước 1: Biểu diễn items dưới dạng vec-tơ thuộc tính - item profile
* Bước 2: Học mô hình của mỗi user

Một số phương pháp thường được sử dụng để xây dựng feature vector là:

* Sử dụng TF-IDF
* Sử dụng biểu diễn nhị phân

**Sử dụng TF-IDF:**

* TF(t, d) = ( số lần từ t xuất hiện trong văn bản d) / (tổng số từ trong văn bản d)
* IDF(t, D) = log\_e( Tổng số văn bản trong tập mẫu D/ Số văn bản có chứa từ t )

Công thức để tính trọng số TF-IDF của một từ trong một văn bản được xác định bởi công thức sau:

Trong đó:

* TF(t, d) là tần số xuất hiện của từ t trong văn bản d.
* N là tổng số văn bản trong tập dữ liệu.
* DF(t) là số văn bản trong tập dữ liệu chứa từ t.
* = log(N/DF) (1.1)

**Ví dụ**: Giả sử chúng ta tìm kiếm về "IoT and analytics" trên Internet và tìm được những bài báo dưới đây:

A picture containing text, screenshot, line, number

Description automatically generated

Trong các bài báo, 5000 chứa analytics, 50.000 chứa data và số lượng tương tự dànhcho các từ khác. Chúng ta hãy giả sử rằng tổng số tài liệu là 1 triệu (10 ^ 6).

* Tính TF: Ta sẽ tính TF cho mỗi từ trong mỗi bài báo

A picture containing text, screenshot, font, number

Description automatically generatedVí dụ, TF(analytics) = 1 + lg21 = 2.32

* Tính IDF:

A close-up of a number

Description automatically generated with low confidenceIDF được tính bằng cách lấy nghịch đảo logarit của tần số tài liệu trong toàn bộ kho tài liệu. Vì vậy, nếu có tổng số 1 triệu tài liệu được trả về bởi truy vấn tìm kiếm của chúng tôi và trong số các tài liệu đó, ví dụ: nếu từ smart xuất hiện trong 0,5 triệu lần trong tài liệu, giá trị IDF của nó sẽ là: Log10 (10 ^ 6/5000000) = 0,30.

* Tính trọng số TF-IDF: Đầu tiên chúng ta sử dụng công thức sau để tính tf-df:

A picture containing text, screenshot, font, number

Description automatically generatedSau đó, chuẩn hóa feature vector bằng cách chia vector cho độ dài của chính nó.

Như vậy, ta có thể có được các vector đặc trưng cho từng bài báo. Sau đó chúng ta có thể sử dụng độ tương đồng cosin để tính khoảng cách giữa chúng.

* 1. Xây dựng mô hình và ví dụ Content Based Filtering
     1. Giải thuật về thuật toán Content-Based Filtering

Xây dựng mô hình đặc trưng: Đầu tiên, chúng ta cần xây dựng một mô hình để trích xuất các đặc trưng của các mục trong hệ thống khuyến nghị. Điều này có thể được thực hiện bằng cách sử dụng các phương pháp phân tích nội dung như xử lý ngôn ngữ tự nhiên (Natural Language Processing - NLP), trích xuất thông tin (Information Retrieval - IR) và phân tích dữ liệu.

Đánh giá độ tương đồng: Tiếp theo, chúng ta cần đánh giá độ tương đồng giữa các mục trong hệ thống khuyến nghị, dựa trên các đặc trưng được trích xuất từ mô hình đặc trưng. Các phương pháp thường được sử dụng để đánh giá độ tương đồng bao gồm đo độ tương đồng cosine, độ tương đồng Jaccard và khoảng cách Euclid

Tạo ra danh sách khuyến nghị: Sau khi đã đánh giá độ tương đồng giữa các mục, chúng ta có thể tạo ra danh sách các mục tương tự nhất, dựa trên sở thích của người dùng. Điều này có thể được thực hiện bằng cách sử dụng các thuật toán phù hợp để lọc và sắp xếp các mục.

Đưa ra khuyến nghị: Cuối cùng, chúng ta có thể đưa ra các khuyến nghị cho người dùng, dựa trên danh sách các mục tương tự nhất được tạo ra từ bước trước đó. Các khuyến nghị này có thể được đưa ra dưới dạng danh sách các mục được sắp xếp theo độ tương đồng giữa chúng hoặc được đưa ra dưới dạng các bản ghi đề xuất.

* + 1. Tính toán độ tương đồng Euclid, Jaccard, Consine

**A picture containing screenshot

Description automatically generated****Khoảng cách Euclid**: là một khái niệm trong toán học, đo lường khoảng cách giữa hai điểm trong không gian Euclid bằng cách tính độ dài của đoạn thẳng nối hai điểm đó.

* Trong không gian hai chiều, khoảng cách Euclid giữa hai điểm A(x1, y1) và B(x2, y2) được tính bằng công thức:
* Trong không gian ba chiều, khoảng cách Euclid giữa hai điểm A(x1, y1, z1) và B(x2, y2, z2) được tính bằng công thức:

Ví dụ, giả sử chúng ta có một cơ sở dữ liệu các phim và mỗi phim có các đặc trưng như sau:

Phim 1: [1, 0, 1, 4.5]

Phim 2: [0, 1, 1, 3.0]

Phim 3: [1, 1, 0, 4.8]

Phim 4: [0, 1, 0, 2.5]

Trong đó, đặc trưng thứ nhất là thể loại hành động (1 nếu phim thuộc thể loại hành động, 0 nếu không), đặc trưng thứ hai là thể loại tình cảm, đặc trưng thứ ba là đạo diễn nam (1 nếu phim được đạo diễn bởi nam giới, 0 nếu ngược lại), và đặc trưng thứ tư là đánh giá trung bình của người dùng.

Để tính toán sự tương tự giữa các phim, ta có thể sử dụng công thức khoảng cách Euclid như đã mô tả ở trên. Ví dụ, nếu ta muốn tính toán khoảng cách Euclid giữa phim 1 và phim 2, ta sẽ có:

d(Phim 1, Phim 2) = √[(0 - 1)² + (1 - 0)² + (1 - 1)² + (3.0 - 4.5)²]

= √[1 + 1 + 2.25 + 2.25]

= √6.5

≈ 2.55

**Ưu điểm của Euclid:**

* Dễ hiểu và tính toán: Công thức khoảng cách Euclid rất đơn giản và dễ hiểu, có thể áp dụng cho nhiều loại dữ liệu và không yêu cầu nhiều kỹ năng toán học phức tạp.
* Tính toán nhanh: Việc tính toán khoảng cách Euclid rất nhanh chóng và hiệu quả đối với các bộ dữ liệu có kích thước nhỏ đến trung bình.
* Đo đạc khoảng cách chính xác: Khoảng cách Euclid đo đạc khoảng cách chính xác giữa hai điểm trong không gian và có thể được sử dụng để tính toán các phương trình đường thẳng, đường cong và các hình học khác trong không gian hai hoặc ba chiều.

**Nhược điểm của Euclid:**

* Nhạy cảm với nhiễu: Khoảng cách Euclid không phù hợp cho các bộ dữ liệu có nhiễu hoặc giá trị bất thường (outliers), vì một giá trị bất thường có thể ảnh hưởng đến khoảng cách giữa các điểm khác.
* Không phù hợp cho các bộ dữ liệu thưa: Khoảng cách Euclid không phù hợp cho các bộ dữ liệu thưa (sparse data), nghĩa là các điểm trong không gian của dữ liệu được phân bố rải rác.
* Không phù hợp cho các bộ dữ liệu với định dạng khác nhau: Khoảng cách Euclid không phù hợp cho các bộ dữ liệu có định dạng khác nhau. Điều này có nghĩa là nếu các đặc trưng của các điểm không có thang đo chung, các khoảng cách Euclid sẽ không có ý nghĩa.

**Độ tương đồng Jaccard:** là một công thức đo lường sự tương đồng giữa hai tập hợp. Độ tương đồng Jaccard được tính bằng tỉ lệ giữa số phần tử chung của hai tập hợp và tổng số phần tử trong hai tập hợp đó. Công thức tính độ tương đồng Jaccard như sau:

Trong đó:

* A và B là hai tập hợp cần tính độ tương đồng.
* |A ∩ B| là số phần tử chung của hai tập hợp A và B.
* |A ∪ B| là tổng số phần tử trong hai tập hợp A và B.

Ví dụ: giả sử có hai tập hợp A và B như sau:

A = {1, 2, 3, 4}

B = {2, 4, 6, 8}

Để tính độ tương đồng Jaccard giữa hai tập hợp này, ta tính số phần tử chung và tổng số phần tử như sau:

|A ∩ B| = {2, 4} => có 2 phần tử chung giữa hai tập hợp A và B.

|A ∪ B| = {1, 2, 3, 4, 6, 8} => có tổng cộng 6 phần tử trong hai tập hợp A và B.

Áp dụng công thức tính độ tương đồng Jaccard, ta có:

J(A, B) = |A ∩ B| / |A ∪ B| = 2 / 6 = 0.33

Vậy, độ tương đồng Jaccard giữa hai tập hợp A và B là 0.33.

**Ưu điểm của độ tương đồng Jaccard:**

* Độ tương đồng Jaccard đơn giản và dễ dàng tính toán, không yêu cầu nhiều tài nguyên tính toán.
* Thường được sử dụng trong các bài toán phân cụm và phát hiện bất thường trong dữ liệu.
* Giá trị độ tương đồng Jaccard giữa hai tập hợp nằm trong khoảng từ 0 đến 1, dễ dàng hiểu và giải thích.
* **Nhược điểm của độ tương đồng Jaccard:**
* Độ tương đồng Jaccard không lưu lại thông tin về mức độ tương đồng giữa các phần tử của hai tập hợp. Nó chỉ cho biết mức độ tương đồng giữa hai tập hợp như một toàn thể.
* Độ tương đồng Jaccard không phù hợp trong các trường hợp tập hợp cần so sánh có kích thước lớn và không có nhiều phần tử chung. Khi đó, nó không cho kết quả chính xác về độ tương đồng giữa hai tập hợp và cần phải sử dụng các phương pháp khác để tính toán.
* Độ tương đồng Jaccard không phù hợp cho các tập hợp có sự khác biệt về kích thước một cách lớn, vì nó có thể dẫn đến sai sót trong việc so sánh độ tương đồng giữa hai tập hợp.

**Độ tương đồng của Consine:** là một phương pháp đo độ tương đồng giữa hai vector trong không gian đa chiều. Nó được sử dụng rộng rãi trong các ứng dụng liên quan đến xử lý ngôn ngữ tự nhiên, khai phá dữ liệu và học máy để so sánh các tài liệu văn bản và tìm kiếm các mẫu trong dữ liệu.

Công thức tính độ tương đồng Cosine giữa hai vector x và y được tính như sau:

Trong đó:

* x và y là hai vector cần tính độ tương đồng.
* (x . y) là tích vô hướng của hai vector x và y.
* ||x|| và ||y|| là độ dài của hai vector x và y.

Giá trị độ tương đồng Cosine dao động từ -1 đến 1, với 1 nghĩa là hai vector hoàn toàn giống nhau và -1 nghĩa là hai vector hoàn toàn trái ngược nhau. Giá trị 0 nghĩa là hai vector không có tương đồng với nhau.

**Ưu điểm của độ tương đồng Cosine:**

* Độ tương đồng Cosine tính toán nhanh và hiệu quả với những vector có số chiều lớn.
* Nó không bị ảnh hưởng bởi kích thước của vector.
* Nó không yêu cầu các vector phải có cùng số chiều.

**Nhược điểm của độ tương đồng Cosine:**

* Nó không giải quyết được vấn đề về tần suất xuất hiện của các thuộc tính.
* Nó không xem xét mối quan hệ giữa các chiều trong không gian vector.
  + 1. Ví dụ về thuật toán Content-Based Filtering

Giả sử chúng ta có một hệ thống khuyến nghị phim sử dụng Content-Based Filtering để đưa ra các khuyến nghị dựa trên đặc trưng của các phim. Các đặc trưng của các phim bao gồm thể loại, diễn viên, đạo diễn, năm sản xuất, mô tả v.v.

Để xây dựng mô hình Content-Based Filtering, chúng ta sử dụng các phương pháp phân tích nội dung để trích xuất các đặc trưng của các phim và đánh giá độ tương đồng giữa các phim. Chẳng hạn, chúng ta có thể sử dụng phương pháp đo độ tương đồng cosine để so sánh đặc trưng của các phim.

Sau khi đã xây dựng mô hình, chúng ta có thể đưa ra các khuyến nghị dựa trên sở thích của người dùng. Ví dụ, nếu người dùng đã đánh giá cao các phim thuộc thể loại hành động, chúng ta có thể đưa ra các khuyến nghị về các phim thuộc thể loại tương tự. Nếu người dùng đã xem các phim của một diễn viên cụ thể, chúng ta có thể đưa ra các khuyến nghị về các phim có sự tham gia của diễn viên đó.

* 1. Kết luận

Content-based filtering là một phương pháp lọc dựa trên nội dung của các sản phẩm để đưa ra các gợi ý sản phẩm tương đồng. Phương pháp này đánh giá độ tương đồng giữa các sản phẩm bằng cách so sánh các thuộc tính của chúng, như mô tả sản phẩm, thể loại, tác giả, diễn viên, đạo diễn, v.v. từ đó đưa ra các đề xuất cho người dùng.

* Giới hạn của phương pháp này là chỉ đề xuất các sản phẩm tương đồng với những sản phẩm mà người dùng đã xem hoặc mua trước đó. Nó không đưa ra các đề xuất mới hoàn toàn.
* Nó không xem xét các thông tin liên quan đến người dùng, như lịch sử tìm kiếm, lịch sử mua hàng, sở thích cá nhân, v.v. Do đó, nó có thể không đưa ra các đề xuất phù hợp với sở thích của người dùng.
* Nếu các sản phẩm không có các thuộc tính phù hợp hoặc các thuộc tính quá giống nhau, phương pháp này có thể không đưa ra các đề xuất chính xác.

1. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG
   1. Khảo sát và đánh giá hệ thống

Hiện nay, có rất nhiều ứng dụng xem phim trên internet. Nhưng từ quá trình khảo sát, em nhận thấy ngoài các ứng dụng có số lượng đông đảo người sử dụng như:

* Clip TV
* VTV Go
* Zing TV
* FPT Play
* Netflix

Qua quá trình khảo sát, hệ thống xem phim được chia ra làm hai phần chính:

* Phần thứ nhất là giao dịch với khách hàng trên ứng dụng.
* Phần thứ hai là dành cho nhân viên quản trị hệ thống.

Mỗi phần có chức năng con khác nhau phục vụ cho mục đích quản lý.

Sau khi khảo sát các hệ thống trên, em muốn xây dựng một ứng dụng có các chức năng tương tự để nhằm mục đích nâng cao trình độ lập trình, khả năng phân tích hệ thống và khả năng tự tìm hiểu.

* 1. Quy trình thực hiện quản trị hệ thống của nhân viên quản trị

Để truy cập vào chức năng quản trị hệ thống, quản trị viên cần có một tài khoản để đăng nhập, sau khi đăng nhập thì có thể thực hiện các chức năng sau:

* Quản trị danh mục loại phim: Bao gồm các công việc thêm, xóa, cập nhật thông tin về phim.
* Quản trị phim: Thêm, xóa và cập nhật thông tin về mỗi bộ phim.
* Quản trị người dùng: Thêm, xóa và cập nhật thông tin người dùng và theo dõi người dùng.
* Quản trị bình luận, yêu thích, đánh giá: Xem được những đánh giá, bình luận, lượt yêu thích phim của người dùng.
  1. Yêu cầu hệ thống
     1. Yêu cầu chức năng
* Ở giao diện người quản trị ứng dụng
* Quản lý danh phim
* Lọc, tìm kiếm phim theo nhiều tiêu chí.
* Thêm sửa xóa phim, người dùng.
* Ở giao diện người dùng:
* Chức năng đăng nhập, đăng ký tài khoản.
* Chức năng đánh giá, bình luận, yêu thích.
* Chức năng tìm kiếm, lọc phim
  + 1. Yêu cầu phi chức năng
* Giao diện ứng dụng xem phim trực tuyến phải đơn giản, thân thiện với người dùng, tạo cho khách hàng cảm giác thích thú mà muốn khám phá ngay lần đầu tiên khi truy cập vào ứng dụng.
* Bố cục ứng dụng phải gần gũi, phù hợp với người dùng.
  + 1. Yêu cầu khác:
* Ứng dụng phải đơn giản, dễ sử dụng.
* Tốc độ tải và tốc độ mở ứng không được quá chậm, tránh việc gây khó chịu cho người dùng.
* Khi có thay đổi trong quá trình xử lý hay tạm ngừng ứng dụng để bảo trì, sửa chữa phải có thông báo cho người dùng.
* Chạy ổn định trên android studio.
  1. Sơ đồ phân rã chức năng và luồng dữ liệu

Sau khi khảo sát và đánh giá một số ứng dụng xem phim, em đã phân tích quy trình hoạt động của những ứng dụng đó và đưa ra được sơ đồ phân rã chức năng như sau:

* + 1. Yêu cầu người dùng
* Đối với tất cả người dùng:
* Cần có thiết bị để kết nối internet để truy cập ứng dụng
* Người dùng chưa đăng ký có thể tra cứu và xem thông tin phim, cần đăng ký để có thể xem phim.
* Đối với quản trị viên:
* Cần có tài khoản được cấp quyền admin để truy cập hệ thống.
  + 1. Sơ đồ phân rã chức năng

A screenshot of a computer

Description automatically generated with medium confidence

Hình 1 Sơ đồ phân rã chức năng ứng dụng xem phim

A picture containing text, screenshot, font, rectangle

Description automatically generated

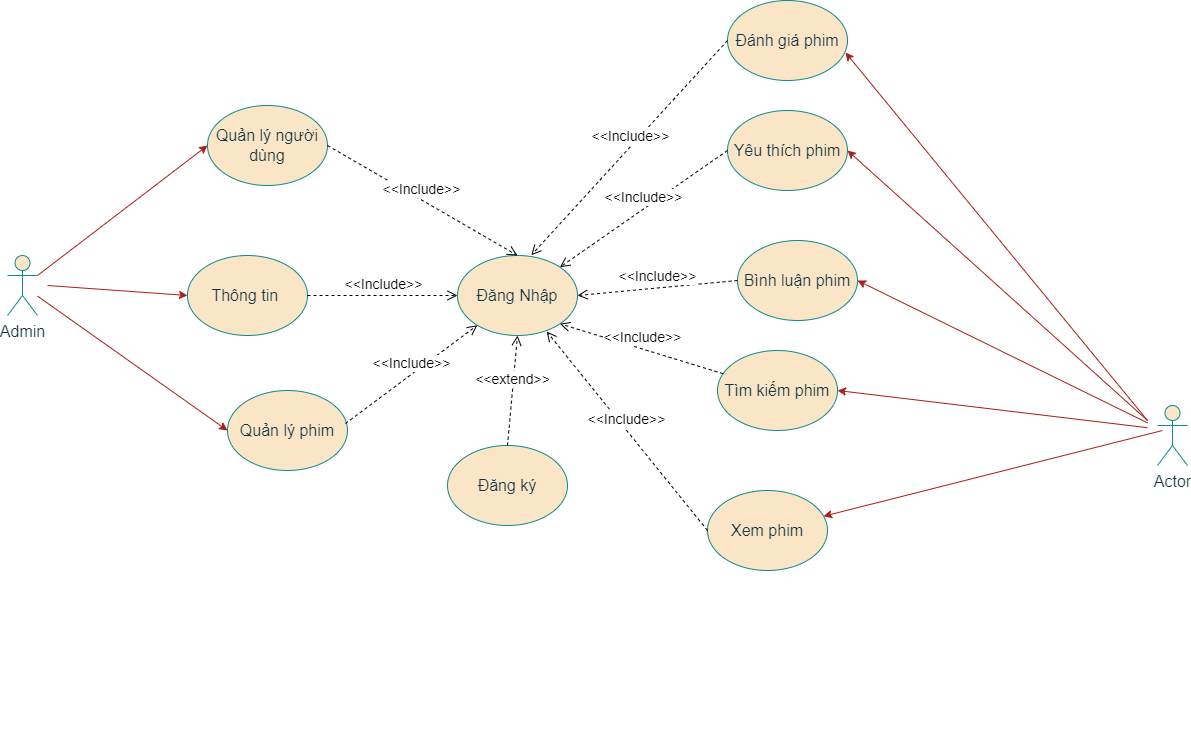
Hình 2 Sơ đồ phân rã chức năng Quản lý ứng dụng

* + 1. Sơ đồ luồng dữ liệu

**A picture containing text, screenshot, font, design

Description automatically generated**

Hình 3 Sơ đồ luồng dữ liệu

* 1. Biểu đồ UseCase
     + 1. Biểu đồ UseCase tổng quát

Hình 4 Biểu đồ UseCase tổng quát

* + - 1. Biểu đồ UseCase Admin

A picture containing text, screenshot, circle, font

Description automatically generated

Hình 5 UseCase Admin

* + - 1. A picture containing screenshot, circle, text, astronomy

         Description automatically generatedBiểu đồ UseCase người dùng

Hình 6 UseCase Người dùng

* + - 1. UseCase đăng nhập



Hình 7 Sơ đồ Usecase đăng nhập

**Mô Tả**

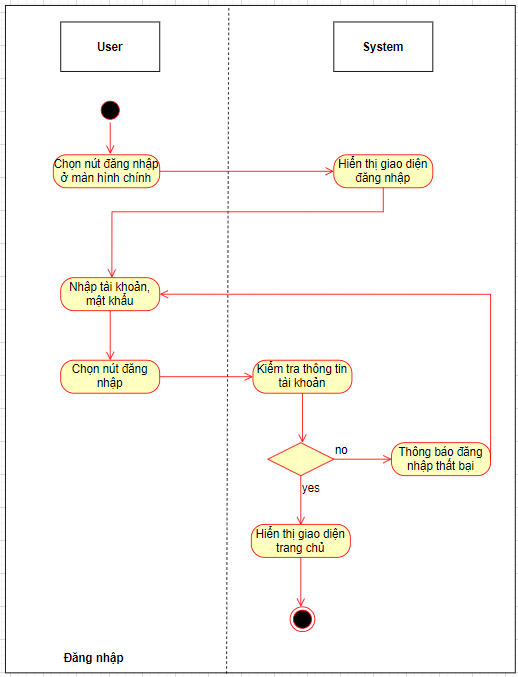
Biểu đồ usecase biểu diễn sơ đồ chức năng của hệ thống. Từ các yêu cầu của hệ thống, biểu đồ usecase chỉ ra hệ thống cần thực hiện những điều gì để đáp ứng nhu cầu của hệ thống sơ đồ luồng dữ liệu. Các chức năng của từng tác nân được thể hiện như sau:

* Người quản trị ứng cần đăng nhập để thực hiện các chức năng quản lý phim, quản lý người dùng.
* Khách hàng có thể đăng ký tài khoản, xem, tìm kiếm phim.
* Khách hàng sau khi đăng nhập thực hiện đánh giá, bình luận hoặc thêm phim vào mục ưa thích.
  1. Sơ đồ ER

A picture containing circle, screenshot

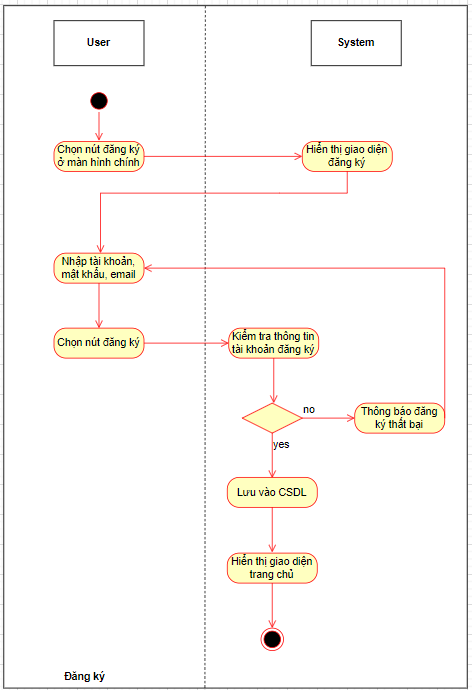
Description automatically generated

Hình 8 Sơ đồ ER

* 1. Sơ đồ hoạt động
     1. Với người dùng
        1. Đăng nhập

Hình 9 Sơ đồ hoạt động chức năng đăng nhập

* + - 1. Đăng ký



Hình 10 Sơ đồ hoạt động chức năng đăng ký

* + - 1. Tìm kiếm phim

A screenshot of a computer

Description automatically generated with medium confidence

Hình 11 Sơ đồ hoạt động chức năng tìm kiếm phim

* + - 1. Xem thông tin phim

A screenshot of a computer screen

Description automatically generated with low confidence

Hình 12 Sơ đồ hoạt động chức năng xem chi tiết phim

* + - 1. Đánh giá phim

A screen shot of a diagram

Description automatically generated with low confidence

Hình 13 Sơ đồ hoạt động chức năng đánh giá phim

* + - 1. Bình luận phim

A screen shot of a computer

Description automatically generated with low confidence

Hình 14 Sơ đồ hoạt động chức năng bình luận phim

*j. yêu thích phim*

A screenshot of a computer

Description automatically generated with medium confidence

Hình 15 Sơ đồ hoạt động chức năng yêu thích phim

* + 1. Với người quản trị
       1. Tìm kiếm phim

A screenshot of a diagram

Description automatically generated with medium confidence

Hình 16 Sơ đồ hoạt động chức năng tìm kiếm phim

* + - 1. Thêm phim

A screenshot of a computer screen

Description automatically generated with low confidence

Hình 17 Sơ đồ hoạt động chức năng thêm phim

* + - 1. Xoá phim

A screenshot of a computer

Description automatically generated with low confidence

Hình 18 Sơ đồ hoạt động chức năng xoá phim

* + - 1. Sửa phim

A screenshot of a computer

Description automatically generated with medium confidence

Hình 19 Sơ đồ hoạt động chức năng sửa phim

* + - 1. Thêm tài khoản

A screenshot of a diagram

Description automatically generated with medium confidence

Hình 20 : Sơ đồ hoạt động chức năng thêm tài khoản

* + - 1. Sửa tài khoản

A screenshot of a computer screen

Description automatically generated with low confidence

Hình 21 Sơ đồ hoạt động chức năng sửa tài khoản

* + - 1. Xoá tài khoản

A screenshot of a computer screen

Description automatically generated with low confidence

Hình 22 Sơ đồ hoạt động chức năng xoá tài khoản

* 1. A screenshot of a computer

     Description automatically generated with medium confidenceSơ đồ Database Diagram

Hình 23 Sơ đồ Diagram

* 1. Sồ đồ ClassDiagram

A picture containing screenshot, black, space

Description automatically generated

Hình 24 Sơ đồ ClassDiagram

1. TRIỂN KHAI CHƯƠNG TRÌNH ỨNG DỤNG XEM PHIM TRỰC TUYẾN
   1. Tồng quan ứng dụng

Ứng dụng xem phim là một loại ứng dụng giải trí phổ biến trên các thiết bị di động, máy tính bảng và smart TV. Ứng dụng này cung cấp cho người dùng các tính năng để tìm kiếm, xem và quản lý các bộ phim, chương trình truyền hình, và nội dung giải trí khác.

Các tính năng chính của ứng dụng xem phim bao gồm:

Tìm kiếm nội dung: Người dùng có thể tìm kiếm các bộ phim, chương trình truyền hình, và nội dung giải trí khác bằng cách sử dụng các từ khóa hoặc thể loại.

Xem phim: Người dùng có thể xem các bộ phim và chương trình truyền hình với chất lượng cao trên thiết bị của mình. Nhiều ứng dụng cung cấp tính năng xem trực tuyến hoặc tải xuống nội dung để xem ngoại tuyến.

Đề xuất nội dung: Các ứng dụng xem phim có thể đề xuất cho người dùng các bộ phim và chương trình truyền hình dựa trên sở thích và lịch sử xem của người dùng.

Cập nhật thông tin mới nhất: Người dùng có thể cập nhật các thông tin mới nhất về các bộ phim và chương trình truyền hình, bao gồm thông tin về diễn viên, đạo diễn và nhà sản xuất.

* 1. Chương trình demo

Khi truy cập vào ứng dụng nếu chưa đăng nhập hệ thống sẽ tự điều hướng đến trang đăng nhập.

* + 1. Giao diện chức năng đăng nhập và đăng ký:

**Giao diện chức năng đăng nhập**

Khi người dùng chạy ứng dụng thì bước đầu tiên để đăng nhập vào ứng dụng thì chúng ta cần phải đăng nhâp tài khoản và mật khẩu (nếu đã có tài khoản trước), Nếu chưa có tài khoản thì chúng ta cần phải đăng ký tài khoản để đăng nhập nhập vào ứng dụng.

A screenshot of a phone

Description automatically generated with medium confidence

Hình 25 Giao diện chức năng đăng nhập

**Giao diện chức năng đăng ký**

A screenshot of a phone

Description automatically generated with medium confidenceVới chức năng đăng ký thì người dùng sẽ điền vào những thông tin cần thiết mà ứng dụng đã yêu cầu. Sau khi nhập xong thông tin thì chúng ta tích vào điều khoản và bắt đầu đăng ký tài khoản riêng cho mình. Sau đó ta sẽ tài khoản của mình và bắt đầu đăng nhập vào ứng dụng

Hình 26 Giao diện chức năng đăng ký

* + 1. Giao diện chức năng trạng chủ

Giao diện trang chủ là trang đầu tiên hiển thị cho khách hàng truy cập vào ứng dụng. Tại đây người dùng có thể xem thông tin về các bộ phim, hiển thị danh sách phim mới nhất, phim đề cử và thể loại phim.

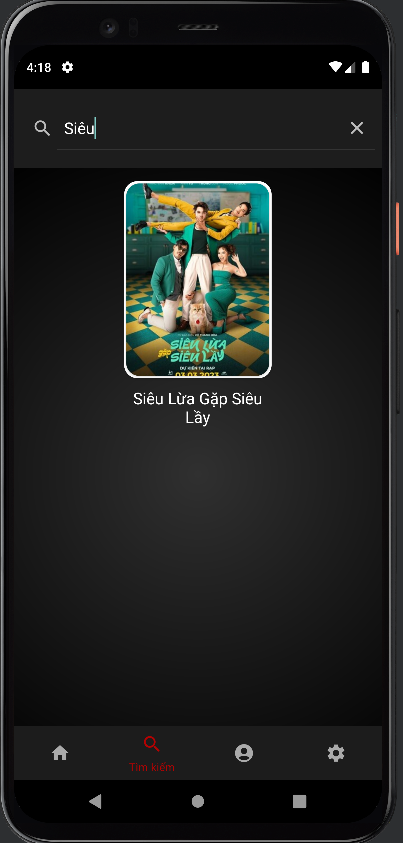
A screenshot of a phone

Description automatically generated with medium confidence

Hình 27 Giao diện trạng chủ

* + 1. Giao diện tìm kiếm

Khi người đùng truy cập vào mục tìm kiếm và bắt đầu tìm kiếm thông tin phim thì hệ thống sẽ hiển thị ra những phim mà người dùng muốn xem



Hình 28 Giao diện tìm kiếm

* + 1. Giao diện thông tin và chỉnh sửa thông tin người dùng

Với giao diện thông tin người dùng thì người dùng có thể xem lại hồ sơ của mình và chỉnh sửa lại thông tin của mình.

Screens screenshot of a phone

Description automatically generated with low confidence

Hình 29 Giao diện thông tin và chỉnh sửa thông tin người dùng

* + 1. Giao diện thêm phim

Giao diện thêm phim chỉ dùng cho admin thêm một bộ phim mới nhất

A screenshot of a phone

Description automatically generated with medium confidence

Hình 30 Giao diện thêm phim

* + 1. Giao diện yêu thích, đánh giá, bình luận và danh sách phim yêu thích, đánh giá, bình luận

**Giao diện yêu thích**

Khi người dùng nhấn vào yêu thích phim thì những bộ phim mà người dùng đã yêu thích sẽ lưu vào danh mục danh sách phim mà mình đã yêu thích

A screenshot of a phone

Description automatically generated with low confidence

Hình 31 Giao diện yêu thích và danh sách phim yêu thích

**Giao diện đánh giá và danh sách phim đã đánh giá**

Khi người dùng đánh giá phim thì những bộ phim mà người dùng đã đánh giá sẽ lưu vào danh mục danh sách phim mà mình đã đánh giá

A screenshot of a phone

Description automatically generated with low confidence

Hình 32 Giao diện yêu đánh giá và danh sách đánh giá

**Giao diện bình luận và danh sách phim đã bình luận**

Khi người dùng bình luận một bô phim mà mình đã xem thì những bộ phim mà người dùng đã bình luận sẽ lưu vào danh sách các phim mà người dùng đã bình luận

A screenshot of a phone

Description automatically generated with low confidence

Hình 33 Giao diện bình luận và danh sách phim đã bình luận

* + 1. Giao diện đề xuất phim

Khi người dùng click vào những bộ phim có thể loại như hành động, hoạt hình, tình cảm, … Thì hệ thống sẽ đề xuất cho người dụng những bộ phim tương tự

A screenshot of a phone

Description automatically generated with low confidence

Hình 34 Giao diện đề xuất phim

* + 1. Giao diện xem phim

Khi người dùng click vào phát phim hoặc click vào các tập phim thì hệ thống sẽ hiển thị ra giao diện như dưới đây và người dùng có thể trải nghiệm những bộ phim mà họ muốn xem

A screen shot of a cartoon

Description automatically generated with medium confidence

Hình 35 Giao diện xem phim

KẾT LUẬN VÀ KIẾN NGHỊ

### Kết quả đạt được

Ứng dụng xem phim trực tuyến tuy không phải là đề tài mới nhưng nó sẽ dần trở thành một phần quan trọng trong ứng dụng công nghệ thông tin vào cuộc sống. Tuy nhiên hệ thống vẫn còn đang ở mức độ cơ bản, sơ khai chưa có một chuẩn rõ ràng, hoàn chỉnh. Trong khuôn khổ cho phép và kiến thức kinh nghiệm học trong suốt thời gian qua em đã cố gắng xây dựng hệ thống một cách tổng quan nhất. Qua quá trình khảo sát, đồ án “Xây dựng ứng dụng xem phim trực tuyến” đã đạt được những kết quả nhất định.

Tìm hiểu nội dung yêu cầu bài toán và các ứng dụng - Nắm bắt kỹ năng phân tích thiết kế hướng đối tượng với UML, tìm hiểu công nghệ phát triển app.

Về mặt chức năng, hệ thống hỗ trợ các tính năng cơ bản trong việc giúp người dùng thuận tiện trong việc mua hàng, tích hợp thanh toàn trực tuyến. Đối với app quản lý giúp quản trị viên dễ dàng quản lý thông tin.

### Hạn chế

Tuy nhiên, còn nhiều chức năng chương trình còn chưa hoàn thiện, giao diện chưa đẹp, các giao diện còn chưa tường minh. Chưa tối ưu quá trình tìm kiếm, nếu dữ liệu quá lớn thì hệ thống sẽ hoạt động kém hiệu quả và chưa đa dạng các thuộc tính tìm kiếm. Đưa ra những bộ phim tương tự, phim gợi ý cho người dùng chưa đa đạng. Tốc độ chạy của ứng dụng chưa được nhanh, chạy phim vẫn còn thiếu sót ở phần giao diện. Tính xác thực về dữ liệu phim chưa được chính xác. Sử dụng quá nhiều thư viện có sẵn.

### Hướng phát triển

* Khắc phục những nhược điểm trên.
* Phát triển thêm ứng dụng có thể đăng kí tài khoản VIP, và thanh toán trực tiếp trên ứng dụng.
* Tạo thêm ứng dụng giúp quản trị viên có thể cập nhật thông tin phim trên ứng dụng
* Phát triển một ứng dụng trên nền tảng di động (android, ios) để nâng cao trải nghiệm của khách hàng, phục vụ nhu cầu sử dụng ứng dụng trên thiết bị di động.
* Phát triển thêm các công cụ gợi ý phim tốt hơn cho người dùng.
* Ứng dụng các thuật toán học máy để dự đoán phim hay xem nhiều nhất.

TÀI LIỆU THAM KHẢO

1. Bill Phillips, Chris Stewart, Kristin Marsicano, và Joe Hummel, Android Programming: - The Big Nerd Ranch Guide (4th Edition), Nhà xuất bản: Big Nerd Ranch Guides (2019).
2. Cay S. Horstmann, Java Concepts: Late Objects, Nhà xuất bản: John Wiley & Sons (2018)
3. Joshua Bloch, Effective Java, Nhà xuất bản: Addison-Wesley Professional (2018)
4. Marko Gargenta và Masumi Nakamura, Learning Android: Develop Mobile Apps Using Java and Eclipse, Nhà xuất bản: O'Reilly Media (2018)
5. Ted Hagos, Android Programming: The Big Nerd Ranch Guide, Nhà xuất bản: Big Nerd Ranch Guides (2019)
6. Jannach, Dietmar và Markus Zanker, Recommender Systems: An Introduction, Nhà xuất bản: Cambridge University Press (2019)
7. Robin Burke, Hybrid Recommender Systems Handbook, Nhà xuất bản: Springer (2019)
8. Paul Deitel và Harvey Deitel, Android 6 for Programmers: An App-Driven Approach, Nhà xuất bản: Prentice Hall (2018)
9. Neil Smyth, Firebase Essentials for Android, Nhà xuất bản: Payload Media (2018).
10. Suresh Kumar Gorakala, Building Recommender Systems: An Introduction, Nhà xuất bản: Apress (2018)
11. Charu Aggarwal, Recommender Systems: The Textbook, Nhà xuất bản: Springer (2018).
12. <https://developer.android.com/guide?hl=vi>
13. https://machinelearningcoban.com/2017/05/17/contentbasedrecommendersys/
14. <https://developer.android.com/studio/write/firebase?hl=vi>
15. <https://viblo.asia/p/tim-hieu-so-luoc-ve-firebase-Eb85oeOmZ2G>
16. https://freetuts.net/tai-lieu-hoc-java-android-kotlin-lo-trinh-hoc-android-2243.html
17. https://www.javatpoint.com/android-tutorial
18. <https://www.w3schools.com/sql/default.asp>
19. <https://www.tutorialspoint.com/firebase/firebase_overview.htm>
20. https://www.javacodegeeks.com/