ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

KHOA KHOA HỌC VÀ KỸ THUẬT THÔNG TIN

VŨ CAO NGUYÊN

KHÓA LUẬN TỐT NGHIỆP

PHÁT TRIỂN ỨNG DỤNG DI ĐỘNG GỢI Ý PHIM CHO NGƯỜI DÙNG

A Mobile Application for Personalized Movie Recommendations

CỬ NHÂN NGÀNH CÔNG NGHỆ THÔNG TIN

GIẢNG VIÊN HƯỚNG DẪN

ThS. PHẠM THẾ SƠN

TP. HỒ CHÍ MINH, 2021

ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

KHOA KHOA HỌC VÀ KỸ THUẬT THÔNG TIN

VŨ CAO NGUYÊN – 16521756

KHÓA LUẬN TỐT NGHIỆP

PHÁT TRIỂN ỨNG DỤNG DI ĐỘNG GỢI Ý PHIM CHO NGƯỜI DÙNG

A Mobile Application for Personalized Movie Recommendations

CỬ NHÂN NGÀNH CÔNG NGHỆ THÔNG TIN

GIẢNG VIÊN HƯỚNG DẪN

ThS. PHẠM THẾ SƠN

TP. HỒ CHÍ MINH, 2021

THÔNG TIN HỘI ĐỒNG CHẤM KHÓA LUẬN TỐT NGHIỆP

Hội đồng chấm khóa luận tốt nghiệp, thành lập theo Quyết định số …… ngày …….. của Hiệu trưởng Trường Đại học Công nghệ Thông tin.

1. ThS. Mai Văn Cường – Chủ tịch.
2. ThS. Nguyễn Thị Anh Thư – Thư ký.
3. TS. Nguyễn Tấn Cầm – Ủy viên.

LỜI CẢM ƠN

Quá trình làm khóa luận gặp rất nhiều khó khăn vất vả nhưng chúng tôi vẫn nhận được nhiều sự quan tâm giúp đỡ từ nhiều phía khác nhau để hoàn thành đề tài khóa luận lần này.

Chúng tôi xin dành lời cảm ơn sâu sắc đến giảng viên hướng dẫn trong khóa luận lần này: ThS. Phạm Thế Sơn, thầy đã chỉ dạy, truyền đạt những kiến thức vô cùng quý báu. Thầy đã không ngại khó khăn, bận rộn để dành thời gian định hướng cho chúng tôi hoàn thành được đề tài khóa luận.

Chúng tôi xin chân thành cảm ơn Ban giám hiệu trường Đại học Công Nghệ Thông Tin, Đại học Quốc gia Thành phố Hồ Chí Minh, cùng với Khoa Khoa Học và Kỹ Thuật Thông Tin đã tạo điều kiện và hỗ trợ rất nhiều trong các quy trình thực hiện khóa luận.

Cám ơn tác giả Quách Phú Quốc và Nguyễn Tiến Đạt đã rất tận tình trong việc bàn giao đề tài khóa luận và truyền đạt kiến thức của hai tác giả để chúng tôi phát triển tại khóa luận lần này.

Chúng tôi xin chân thành cảm ơn.

TP. Hồ Chí Minh, tháng 8 năm 2021

Sinh viên thực hiện

Vũ Cao Nguyên

MỤC LỤC

DANH MỤC HÌNH

DANH MỤC TỪ VIẾT TẮT

DANH MỤC BẢNG

TÓM TẮT NỘI DUNG 1

Chương 1. TỔNG QUAN 2

1.1. Đặt vấn đề. 2

1.2. Đối tượng nghiên cứu. 2

1.3. Phạm vi nghiên cứu. 3

1.4. Phương pháp thực hiện. 3

1.5. Đóng góp khóa luận. 3

1.6. Cấu trúc khóa luận. 4

Chương 2. KHẢO SÁT 5

2.1. Khảo sát giao diện và chức năng 5

2.1.1. Ứng dụng ITunes Store. 5

2.1.2. Ứng dụng IMDb. 6

2.1.3. Ứng dụng Galaxy Play. 8

2.1.4. Kết luận về khảo sát giao diện và chức năng. 8

2.2. Tìm hiểu các công nghệ xây dựng ứng dụng Mobile 10

2.2.1. Native Code. 10

2.2.2. Cross Platform. 11

2.2.3. Hybird. 11

2.3. Tìm hiểu công nghệ xây dựng Web API 12

2.4. Tìm hiểu các quy trình phát triển ứng dụng 13
2.4.1. Mô hình thác nước. 13
2.4.2. Mô hình tạo bản mẫu. 14
2.4.3. Mô hình xoắn ốc. 15
Chương 3. PHÂN TÍCH YÊU CẦU ỨNG DỤNG 17
3.1. Phân tích use case. 17
3.1.1. Use case các chức năng trong ứng dụng. 17
3.1.2. Use case gợi ý phim cho người dùng. 25
3.2. Phân tích giao diện. 28
3.2.1. Đăng nhập. 28
3.2.2. Đăng ký. 29
3.2.3. Trang chủ. 29
3.2.4. Hiển thị danh sách phim. 29
3.2.5. Chi tiết phim. 30
3.2.6. Chi tiết thông tin người dùng. 30
3.2.7. Tìm kiếm. 30
3.2.8. Đánh giá. 30
3.2.9. Bình luận. 30
3.3. Phân tích các thực thể. 31
3.3.1. Xây dựng sơ đồ ERD. 31
Chương 4. THIẾT KẾ ỨNG DỤNG 34
4.1. Cơ sở dữ liệu. 34
4.2. Giao diện ứng dụng. 40

4.3. Thiết kế kiến trúc ứng dụng. 42
4.3.1. Flutter Framwork. 42
4.3.2. Kiến trúc hệ thống. 44
Chương 5. CÀI ĐẶT VÀ TRIỂN KHAI ỨNG DỤNG 46
5.1. Cài đặt và triển khai Web API 46
5.1.1. Cài đặt môi trường. 46
5.1.2. Triển khai các web api. 50
5.2. Cài đặt và triển khai ứng dụng. 59
5.2.1. Cài đặt SDK 59
5.2.2. Màn hình đăng nhập. 60
5.2.3. Màn hình đăng ký người dùng mới. 60
5.2.4. Màn hình trang chủ. 62
5.2.5. Màn hình xem thêm phim. 63
5.2.6. Màn hình xem chi tiết bộ phim. 64
5.2.7. Màn hình tìm kiếm phim. 68
5.2.8. Màn hình thông tin người dùng. 69
Chương 6. ĐÁNH GIÁ VỚI KHÓA LUẬN KẾ THỪA 70
6.1. Phát triển phần Web API. 70
6.2. Phát triển phần ứng dụng. 72
Chương 7. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 74
7.1. Kết luận 74
7.2. Hướng phát triển. 74
TÀI LIỆU THAM KHẢO 76

DANH MỤC HÌNH

Hình 2.1: Giao diện ứng dụng ITunes. 6

Hình 2.2: Giao diện ứng dụng IMDb. 7

Hình 2.3: Giao diện ứng dụng Galaxy Play. 8

Hình 2.4: Mô hình thác nước. 13

Hình 2.5: Mô hình tạo bản mẫu. 15

Hình 2.6: Mô hình xoắn ốc. 16

Hình 3.1: Use case diagram ứng dụng xem phim. 18

Hình 3.2: Sơ đồ ERD 32

Hình 4.1: Diagram cơ sở dữ liệu. 34

Hình 4.2: Đăng ký, đăng nhập, trang chủ. 40

Hình 4.3: Thông tin người dùng, chi tiết phim. 41

Hình 4.4: Phân chia lớp trong ứng dụng. 43

Hình 4.5: Kiến trúc hệ thống. 44

Hình 5.1: Active môi trường, cài đặt Django trong môi trường. 47

Hình 5. 2: Tạo mới một project movielens_api. 48

Hình 5.3: Tạo mới các app cho project movielens_api. 48

Hình 5.4: Cấu trúc thư mục movielens_api. 48

Hình 5.5: Khai báo các app trong file settings.py. 49

Hình 5.6: Khai báo database trong file settings.py. 50

Hình 5.7: Lấy thông tin bộ phim. 52

Hình 5.8: Thông báo đăng nhập thất bại. 53

Hình 5.9: Thông báo đăng nhập thành công. 53

Hình 5.10: Đăng ký người dùng mới thành công. 54

Hình 5.11: Đăng ký thất bại do trùng tên đăng nhập. 54

Hình 5.12: Đăng ký thất bại do trùng địa chỉ email. 55

Hình 5.13: Truy vấn thông tin người dùng 55

Hình 5.14: Cập nhật lại đánh giá phim. 56

Hình 5.15: Tạo mới một đánh giá phim. 56

Hình 5.16: Thống kê các đánh giá của một bộ phim. 56

Hình 5.17: Thêm bình luận thành công 57

Hình 5.18: Danh sách bình luận của bộ phim. 57

Hình 5.19: Danh sách movie_id các phim gợi ý. 58

Hình 5.20: Danh sách movie_id các bộ phim của một diễn viên. 58

Hình 5.21: Màn hình đăng nhập. 60

Hình 5.22: Màn hình đăng ký. 61

Hình 5.23: Màn hình trang chủ ứng dụng. 62

Hình 5.24: Màn hình xem thêm phim 63

Hình 5.25: Màn hình chi tiết phim. 64

Hình 5.26: Màn hình danh sách phim theo lựa chọn. 65

Hình 5.27: Màn hình bình luận phim. 66

Hình 5.28: Màn hình đánh giá phim. 67

Hình 5.29: Màn hình tìm kiếm phim. 68

Hình 5.30: Màn hình thông tin người dùng. 69

DANH MỤC BẢNG

Bảng 3.1: Use case đăng nhập. 19

Bảng 3.2: Use case đăng ký. 21

Bảng 3.3: Use case tìm kiếm. 22

Bảng 3.4: Use case xem danh sách phim. 22

Bảng 3.5: Use case xem chi tiết phim. 23

Bảng 3.6: Use case bình luận. 24

Bảng 3.7: Use case đánh giá. 24

Bảng 3.8: Use case xem trailer. 25

Bảng 3.9: Use case gợi ý phim cho người dùng cũ. 26

Bảng 3.10: Use case gợi ý phim cho người dùng mới. 27

Bảng 3.11: Use case gợi ý phim cho người dùng mới và có loại phim yêu thích. 28

Bảng 3.12: Bảng Mô tả đối tượng customer trong django server. 52

Bảng 4.1: Bảng Customer. 35

Bảng 4.2: Bảng movie. 37

Bảng 4.3: Bảng ratings. 38

Bảng 4.4: Bảng comment 39

Bảng 4.5: Bảng recommended movie. 39

Bảng 5.1: Mô tả đối tượng movie trong django server. 51

Bảng 6.1: So sánh web api. 71

Bảng 6.2: So sánh ứng dụng. 73

DANH MỤC TỪ VIẾT TẮT

STT	Viết tắt	Từ viết tắt
	ThS	Thạc sĩ
1	API	Application Programming Interface
2	UI	User Interface
3	HTML	Hypertext Markup Language
4	CSS	Cascading Style Sheets
5	ERD	Entity Relationship Diagram

TÓM TẮT NỘI DUNG

Trong khóa luận tốt nghiệp này, chúng tôi tập chung vào việc xây dựng ứng dụng gợi ý phim trên điện thoại di động dựa vào mô hình đã được xây dựng trong khóa luận tốt nghiệp[1] trước đó.

Về mặt tổng quan, ứng dụng được chia thành hai thành phần chính: (1) Xây dựng giao diện cho ứng dụng; (2) Xây dựng Web API server để thực hiện việc truyền dữ liệu. Về phần Web API server được xây dựng bằng Django rest framework và sử dụng cơ sở dữ liệu Postgresql. Phần giao diện điện thoại được xây dựng bằng flutter framework.

Khóa luận tốt nghiệp của chúng tôi sau khi hoàn thành đạt được một số kết quả sau: Xây dựng thành công ứng dụng xem phim trên điện thoại di động sử dụng hệ điều hành android, ứng dụng cho phép người dùng đăng ký mới, đăng nhập khi đã có tài khoản và có các chức năng như bình luận, đánh giá phim hoặc xem giới thiệu phim. Ngoài các chức năng thì ứng dụng còn đưa ra những gợi ý phim cho những người dùng cũ là những người dùng đã có mô hình gợi ý phim[2] và những người dùng đăng ký mới là những người dùng chưa có mô hình gợi ý phim. Kết quả cuối cùng là phần web api server cho phép nhận và trả thông tin từ cơ sở dữ liệu về cho ứng dụng.

Được mở rộng từ đề tài khóa luận tốt nghiệp "Phát triển hệ khuyến nghị phim cho ứng dụng di động" do Quách Phú Quốc và Nguyễn Tiến Đạt thực hiện trong năm học 2019-2020. Phần mở rộng trình bày trong đề cương.

[2] Mô hình gợi ý phim singular value decomposition được sử dụng trong khóa luận của chúng tôi là được kế thừa từ khóa luận "Phát triển hệ khuyến nghị phim cho ứng dụng di động".

Chương 1. TỔNG QUAN

Tóm tắt chương: *Trong chương 1, chúng tôi trình bày tổng quan về đề tài và lý do thực hiện đề tài. Qua đó thể hiện nội dung chính trong đề tài, làm rõ đối tượng nghiên cứu của để tài và phương pháp thực hiện của đề tài. Tiếp theo trình bày phạm vi và đóng góp khi thực hiện đề tài. Cuối cùng, trình bày cấu trúc đề tài khóa luận.*

1.1. Đặt vấn đề.

Đề tài phát triển ứng dụng gợi ý phim cho người dùng được kế thừa từ đề tài "Xây dựng hệ thống khuyến nghị cho người dùng xem phim trên điện thoại di động". Ở đề tài trước tác giả đã xây dựng thành công mô hình gợi ý phim cho người dùng thông qua việc xây dựng rất nhiều mô hình khác nhau. Mô hình được chọn để áp dụng vào ứng dụng xem phim trên di động là singular value decomposition [1]. Tại đề tài trước, nhóm tác giả đã giải quyết được vấn đề gợi ý phim cho người dùng đã xem phim và đã có đánh giá các bộ phim đã xem.

Đề tài hiện tại sẽ giải quyết vấn đề là cho phép người dùng mới đăng ký trong ứng dụng và người dùng mới sẽ được đưa ra các bộ phim gợi ý. Cùng với việc giải quyết vấn đề gợi ý phim cho người dùng mới thì đề tài cũng phát triển ứng dụng di động hoàn thiện cho phép người dùng mới đăng ký tài khoản vào hệ thống, thêm các chức năng bình luận phim, đánh giá phim và xem giới thiệu tóm tắt phim.

1.2. Đối tượng nghiên cứu.

Đề tài tập chung vào việc phát triển hoàn chỉnh ứng dụng gợi ý phim cho người dùng gồm các chức năng đăng ký, bình luận và đánh giá phim. Tìm hiểu các phương pháp và tiến hành xây dựng mô hình gợi ý phim cho người dùng mới.

1.3. Phạm vi nghiên cứu.

Trong phạm vi của để tài chúng tôi tập chung hoàn thành các nội dung sau:

- Tìm hiểu ngôn ngữ, Framework để xây dựng một giao diện ứng dụng di động.
- Tìm hiểu giao diện của một ứng dụng xem phim thông qua việc khảo sát các ứng dụng hiện có trên CH Play, Apple Store.
- Tìm hiểu việc xây dựng một web api cho phép truyền và nhận dữ liệu sau đó lưu và trả dữ liệu về cho ứng dụng di động.
- Xây dựng hoàn chỉnh ứng dụng gợi ý phim và chạy trên máy áo và thiết bị thật.

1.4. Phương pháp thực hiện.

Trong đề tài này, chúng tôi thực hiện bằng phương pháp:

- Chúng tôi sử dụng mô hình gợi ý phim đã được nhóm tác giả xây dựng thành công là singular value decomposition cho bộ dữ liệu Movielens.
- Trong ứng dụng chúng tôi đã sử dụng một số kết quả nhóm tác giả đã xây dựng như: Web API server cho chức năng đăng nhập, gợi ý phim cho mô hình singular value decomposition và một số giao diện trong ứng dụng.
- Sử dụng Flutter Framework để xây dựng giao diện cho ứng dụng, xây dựng các chức năng và kết nối với Web API server.

- Sử dụng Django Rest Framework xây dựng Web API server để thực hiện việc truyền, nhận dữ liệu cho ứng dụng.
- Sử dụng hệ quản trị cơ sở dữ liệu Postgresql.

1.5. Đóng góp khóa luận.

Một số đóng góp của khóa luận mà tôi thực hiện:

- Tổng quan việc xây dựng một ứng dụng gợi ý phim, kiến thức về các Framework.
    - Xây dựng Web API server dựa trên Django rest framework.
    - Xây dựng giao diện ứng dụng xem phim dựa trên Flutter Framework.
- Xây dựng ứng dụng điện thoại di động gợi ý phim. Ứng dụng cho phép người dùng đăng ký, đăng nhập tài khoản sau khi đăng nhập sẽ nhận được những bộ phim gợi ý phù hợp. Cùng với việc đưa ra gợi ý phim thì ứng dụng cũng có các chức năng đánh giá, bình luận phim.

1.6. Cấu trúc khóa luận.

Cấu trúc của khóa luận:

- Chương 1: Tổng quan.
- Chương 2: Khảo sát.
- Chương 3: Phân tích yêu cầu ứng dụng.
- Chương 4: Thiết kế ứng dụng.
- Chương 5: Cài đặt và triển khai ứng dụng.
- Chương 6: Đánh giá kết quả với khóa luận kế thừa.
- Chương 7: Kết luận và hướng phát triển.

Chương 2. **KHẢO SÁT**

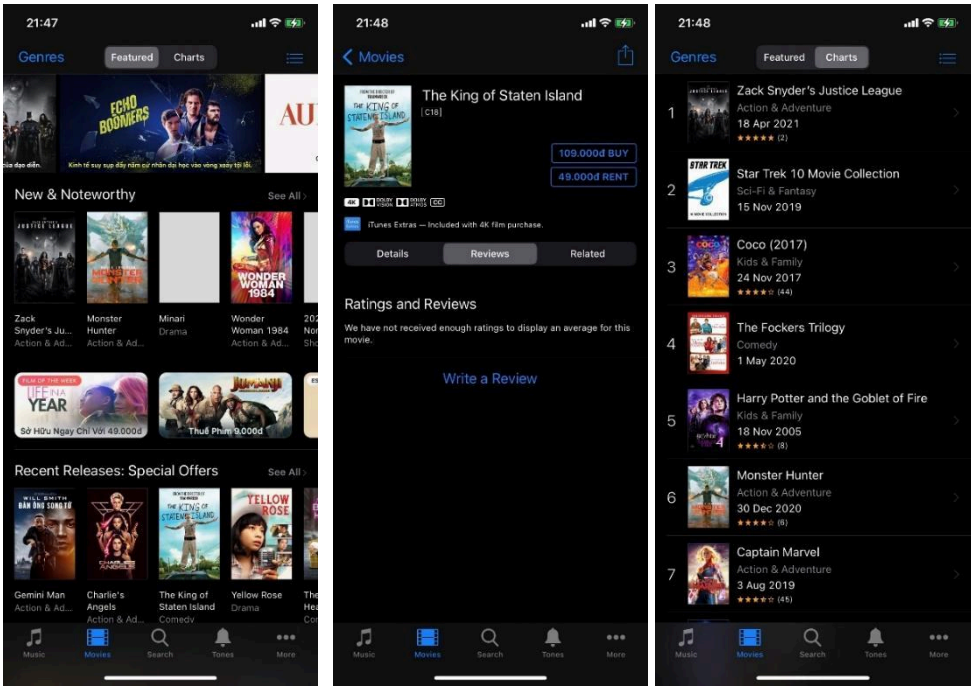
Tóm tắt chương: *Trong chương 2, chúng tôi tiến hành trải nghiệm và tìm hiểu về các ứng dụng xem phim phổ biến. Từ đó chúng tôi có cái nhìn tổng quan và kiến thức trong giao diện và các chức năng của một ứng dụng sau áp dụng vào trong thiết kế ứng dụng gợi ý phim. Ngoài ra chúng tôi tìm hiểu các framework và các quy trình để triển khai một ứng dụng di động.*

2.1. Khảo sát giao diện và chức năng

Giải trí bằng việc xem phim hiện nay rất phổ biển. Xem phim có thể thông qua xem trực tiếp tại các rạp chiếu phim, xem qua các kênh truyền hình, các trang web phim hoặc trên ứng dụng di động. Về ứng dụng xem phim trên điện thoại di động hiện nay đã có nhiều trên nền tảng IOS và Android. Chúng tôi sẽ tiến hành tìm hiểu các ứng dụng xem phim về phương diện giao diện và các chức năng thường có trong ứng dụng. Từ những khảo sát chúng tôi sẽ có cái nhìn tổng quan cùng với kiến thức để tiến hành xây dựng giao diện và chức năng cho ứng dụng gợi ý phim.

2.1.1. Ứng dụng ITunes Store.

ITunes Store là ứng dụng phổ biến với những người sử dụng các thiết bị của Apple như Iphone, Ipad, MacBook. Được phát hành bởi Apple Inc vào ngày 28, tháng tư, năm 2003. ITunes Store cung cấp 60 triệu bài hát, 65 nghìn bộ phim và 25 nghìn chương trình truyền hình tính đến tháng 4 năm 2020. Ứng dụng cho phép người dùng xem phim hoặc nghe nhạc với việc bỏ một khoản tiền ra để mua phim hoặc có thể thuê phim với giá rẻ hơn. Ngoài các thiết bị của Apple thì ITunes cũng có trên hệ điều hành Windows



Hình 2.1: Giao diện ứng dụng ITunes.

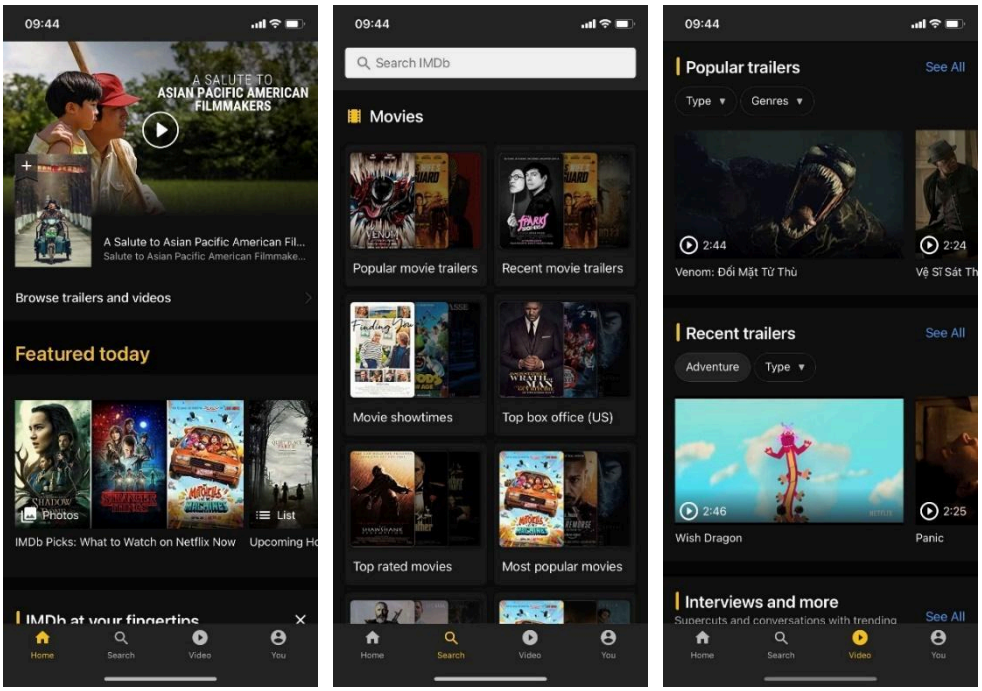
Qua việc tìm hiểu ứng dụng ITunes Store chúng tôi đưa ra một số nhận xét về giao diện cùng với các chức năng:

- Giao diện có nền màu đen kết hợp với chữ màu trắng, các nút nhấn khi được nhấn hoặc vị trí có thể nhấn được sẽ có màu xanh da trời. Thanh điều hướng được bố trí ở dưới cùng màn hình với năm nút điều hướng. Giao diện đơn giản, dễ dàng sử dụng.
- Chức năng đăng nhập, đăng ký mới, xem phim, xem trailer, nhận thông báo, viết đánh giá phim, mua phim, thuê phim, chia sẻ và tìm kiếm. Các chức năng của ứng dụng rất nhiều và có thể đáp ứng nhu cầu sử dụng của người dùng.

2.1.2. Ứng dụng IMDb.

Ứng dụng IMDb là viết tắt của Internet Movie Database dịch ra tiếng việt là cơ sở dữ liệu phim trên mạng. Ứng dụng cung cấp cho người dùng thông tin

của bộ phim như đạo diễn, diễn viên, mức độ phổ biến và điểm đánh giá trung bình. Ứng dụng có trên cả thiết bị chạy hệ điều hành Android, IOS và còn có website để truy cập.



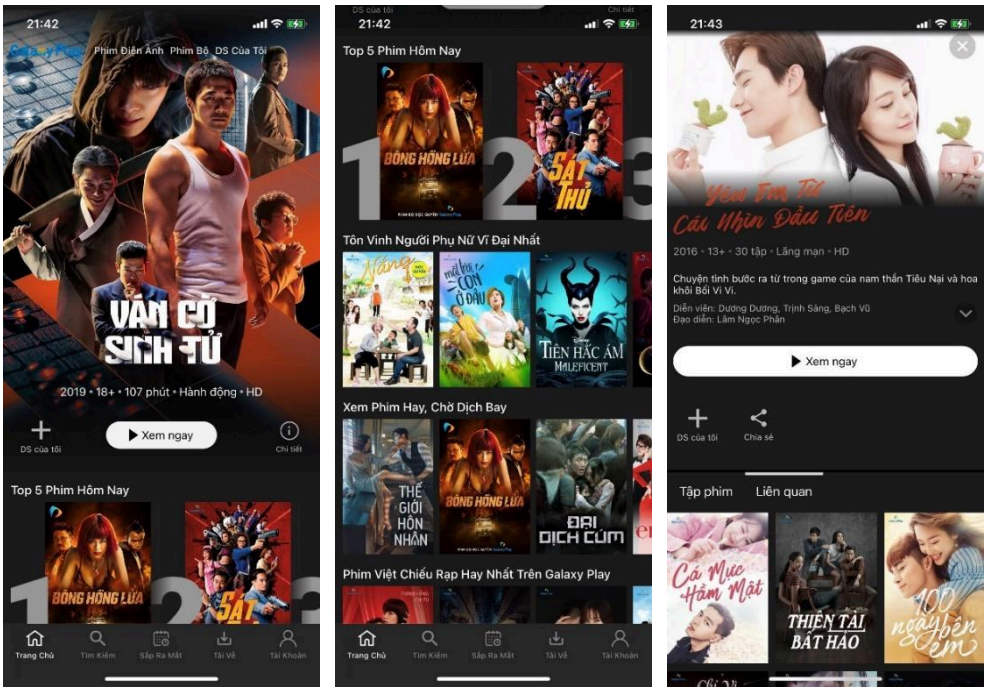
Hình 2.2: Giao diện ứng dụng IMDb.

Từ những trải nghiệm về ứng dụng chúng tôi đưa ra một số đánh giá về giao diện và về các chức năng.

- Giao diện của ứng dụng có hai lựa chọn giao diện sáng và tối.Giao diện sáng có nền trắng và chữ đen. Nền và màu chữ có độ tương phản làm cho nội dung dễ nhìn. Các nút điều hướng được nhấn sẽ có màu vàng và được đặt vị trí cuối của màn hình. Giao diện đơn giản dễ dàng sử dụng cho những người mới.
- Chức năng gồm có đăng nhập, đăng ký mới, xem giới thiệu phim, tìm kiếm và đặt vé xem phim nhưng chức năng đặt vé chỉ có ở một số nước. Tóm lại chức năng của ứng dụng không quá nhiều nhưng đã gồm có những chức năng cần thiết để người dùng sử dụng.

2.1.3. Ứng dụng Galaxy Play.

Ứng dụng Galaxy Play là tiền thân của film+ hiện nay có mặt cả trên nền tảng Android, IOS và Smart TV. Ứng dụng được ra đời năm 2015 gồm nhiều bộ phim Việt Nam độc quyền của Galaxy Play sản xuất.



Hình 2.3: Giao diện ứng dụng Galaxy Play.

Từ những trải nghiệm ứng dụng, chúng tôi đưa ra đánh giá về giao diện và các chức năng của ứng dung.

- Giao diện ứng dụng có màu nền màu đen chữ xám. Các nút nhấn khi được nhấn hoặc các vị trí có thể nhấn được sẽ có màu trắng để người dùng biết. Các nút điều hướng nằm ở dưới cùng màn hình. Giao diện đơn giản dễ dàng sử dụng cho những người dùng mới.
- Chức năng của ứng dụng gồm có đăng nhập, đăng ký, tìm kiếm, chia sẻ, tải về thiết bị, thêm vào danh sách yêu thích.

2.1.4. Kết luận về khảo sát giao diện và chức năng.

2.1.4.1. Giao diện.

Từ những khảo trải nghiệm, tìm hiểu các ứng dụng trên chúng tôi đưa ra các gợi ý cho việc thiết kế giao diện cho ứng dụng của mình.

- Màu nền và màu chữ cần phải có độ tương phản để nội dung được hiển thị rõ ràng và không gây cảm giác khó nhìn cho người dùng. Màu nền hay được sử dụng là nền tối chữ sáng, lý do là xem phim thường là những thời gian rảnh và ở không gian tối cho nên nền màu tối sẽ được ưa chuộng hơn.
- Các nút điều hướng khi được nhấn cần có màu khác biệt để người dùng biết mình đang ở đâu trong ứng dụng.
- Màn hình chính sẽ hiển thị các bộ phim gợi ý khi người dùng đăng nhập.
- Thông tin hiển thị gồm hình ảnh phim và tên phim được thể hiện trong một khung chữ nhật đứng hoặc ngang tùy theo màn hình. Giao diện chi tiết phim thì cần thể hiện các nội dung như tên phim, tên các diễn viên, thời lượng bộ phim, thể loại của bộ phim và mô tả ngắn về phim.

2.1.4.2. Chức năng.

Một ứng dụng có thể có rất nhiều chức năng, tùy theo đặc thù của từng ứng dụng mà có những chức năng cần phải có trong ứng dụng đó. Qua việc đã trải nghiệm và tìm hiểu thì đối với ứng dụng gợi ý phim của chúng tôi cần có các chức năng là: Đăng nhập, đăng ký, xem trailer, bình luận phim, đánh giá phim.

- Đăng nhập vào ứng dụng bằng tên đăng nhập và mật khẩu mà người dùng đã đăng ký.
- Đăng ký gồm các thông tin như: Tên đăng nhập, mật khẩu, email, ngày sinh, địa chỉ và thể loại phim yêu tích.
- Do không có nguồn phim nên chúng tôi chỉ dùng ở chức năng xem giới thiệu phim.
- Bình luận phim sẽ là các đánh giá, nhận xét, bày tỏ cảm xúc của người xem qua từ ngữ, ký tự biểu cảm.
- Đánh giá phim cho phép người dùng đánh giá điểm của bộ phim thông qua sao. Số sao cao nhất là 5 và thấp nhất 1 sao.

2.2. Tìm hiểu các công nghệ xây dựng ứng dụng Mobile

Để xây dựng một ứng dụng di động hiện nay có 3 cách để xây dựng đó là sử dụng Native Code, Cross Platform và Hybird. Chúng tôi đã tiến hành tìm hiểu về cách triển khai của từng công nghệ xây dựng ứng dụng và đưa ra bài tổng hợp.

2.2.1. Native Code.

Native code cách viết code bằng ngôn ngữ gốc và là ngôn ngữ được dành riêng cho từng nền tảng cụ thể. Với các ứng dụng chạy trên nền tảng di động IOS thì được viết bằng ngôn ngữ lập trình Swift, Objective-C. Trên các nền tảng Android các ứng dụng được viết bằng ngôn ngữ lập trình Java, Kotlin.

Ưu điểm:

- Ứng dụng hoạt động nhanh, hiệu quả nhất trong các cách xây dựng ứng dụng.
- Tình trạng xảy ra lỗi khi đang sử dụng rất thấp.

- Vì là ứng dụng được xây dựng từ ngôn ngữ gốc được dành riêng cho hệ điều hành nên các ngôn ngữ khác ngôn ngữ lập trình đều không thể sánh bằng.
- Cho phép sử dụng các tính năng của thiết bị di động như máy ảnh, la bàn, định vị vị trí.
- Dễ triển khai trên các cửa hàng ứng dụng.

Nhược điểm:

- Ứng dụng native chỉ hoạt động trên hệ điều hành nhất định do vậy trên hai hệ điều hành khác nhau cần xây dựng hai ứng dụng khác nhau.
- Quá trình bảo trì, nâng cấp ứng dụng mất khá nhiều thời gian.
- Chi phí xây dựng ứng dụng Native cao.

2.2.2. Cross Platform.

Cross Platform là cách xây dựng một ứng dụng đa nền tảng. Với Cross Platform thì chỉ với một bản code, chúng ta có thể phiên dịch và sử dụng chúng trên nhiều nền tảng khác nhau, điều đó giúp những lập trình viên tiết kiệm thời gian và công sức. Các framework hiện nay được sử dụng nhiều gồm: React Native, Flutter, Xamarin, Ionic.

Ưu điểm:

- Tiết kiệm thời gian và công sức xây dựng ứng dụng.
- Chỉ cần một lần code có thể chạy trên nhiều nền tảng khác nhau.
- Có thể xây dựng các ứng dụng lớn, linh hoạt và đa dạng.

Nhược điểm:

- Không thể triển khai thư viện của nền tảng Android, IOS.

- Quá trình cấp phép cho ứng dụng diễn ra phức tạp.

2.2.3. Hybird.

Ứng dụng xây dựng bằng Hybird được xây dựng bằng cách kết hợp 3 nền tảng là HTML, CSS, JavaScript. Ứng dụng xây dựng bằng hybrid có thể chạy trên được các nền tảng di động hiện nay là Android và IOS.

Ưu điểm:

- Tương thích với các thiết bị di động khác nhau, do vậy tiết kiệm chi phí và thời gian.
- Sử dụng được các tính năng như cảm biến, máy ảnh hay định vị cả khi không có kết nối mạng.

Nhược điểm:

- Ứng dụng chạy không được mượt, trải nghiệm người dùng không tốt.

2.3. Tìm hiểu công nghệ xây dựng Web API

API dùng để giao tiếp, trao đổi dữ liệu giữa các ứng dụng.

Hiện nay có rất nhiều framework hỗ để xây dựng Web Api một cách đơn giản và dễ dàng. Các framework để xây dựng web api như: Nodejs, Go, Flask, Django, Spring Boot, FastAPI, Laravel, Phoenix, ASP.NET Core, Ruby on Rails, ….

Trong quá trình tìm hiểu ưu điểm, nhược điểm của các framework xây dựng web api, chúng tôi lựa chọn Django để xây dựng web api.

Django Framework sử dụng ngôn ngữ Python. Nó là một framework hỗ trợ đầy đủ các thư viện, module cho web-developer. Django sử dụng mô hình MVC và được phát triển bởi Django Software Foundation.

Lý do chúng tôi sử dụng Django?

- Giúp xây dựng web nhanh chóng, đơn giản. [2]
- Có đầy đủ các thư viện và cá module cần thiết ví dụ: site maps, RSS feed, v.v… [2]
- Bảo mật: Django giúp nhà phát triển tránh được nhiều lỗi phổ biến hiện nay, chẳng hạn SQL injection, cross-site scripting và quản lý tài khoản, mật khẩu người dùng một cách an toàn. [2]
- Khả năng mở rộng cao: Django có thể đáp ứng được lượng truy cập lớn, chúng ta không cần lo lắng về khả năng mở rộng sản phẩm của mình nữa.
- Linh loạt trong xây dựng website: Tất cả website đều có thể xây dựng bằng Django chẳng hạn mạng xã hội, trang web thương mại điện tử, …

Để xây dựng Web API cho ứng dụng chúng tôi sử dụng Django Rest Framework. Nó là bộ công cụ mạnh mẽ để xây dựng các Web API bởi vì các lý do sau:

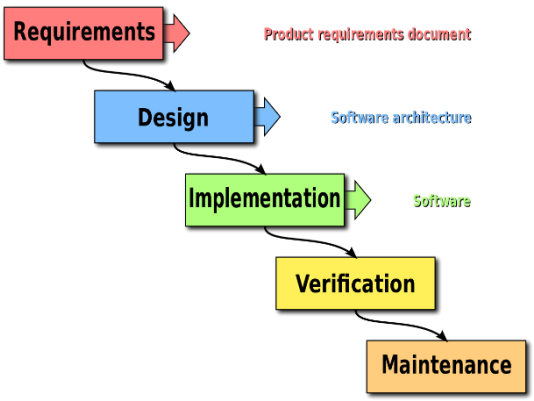
- Cộng đồng hỗ trợ, tài liệu mở rộng tuyệt vời.
- Được sử dụng bởi các công ty quốc tế như Red Hat, Mozilla, Heroku.
- Browsable API.
- Hỗ trợ xác thực gồm các gói OAuth1a, OAuth2
- Serialization: Hỗ trợ chuyển đổi model thành Json object.

2.4. Tìm hiểu các quy trình phát triển ứng dụng

Có nhiều mô hình phát triển ứng phần mềm, ứng dụng khác nhau hiện có trên thị trường. Với những mô hình thì có những ưu điểm và nhược điểm. Chúng tôi sẽ tìm hiểu các mô hình và lựa chọn mô hình phát triển ứng dụng phù hợp cho khóa luận tốt nghiệp “Xây dựng ứng dụng gợi ý phim di động cho người dùng”.

2.4.1. **Mô hình thác nước.**

Trong mô hình thác nước thì các bước triển khai xây dựng ứng dụng được thực hiện tuần tự từ đầu đến cuối và không có sự lặp lại trong mô hình thác nước. Khi một quá trình kết thúc thì mới bắt đầu một quá trình mới trong quy trình.



Hình 2.4: Mô hình thác nước.

(nguồn https://en.wikipedia.org/wiki/Waterfall_model)

Các thành phần trong mô hình thác nước:

- **Requirements:** Lấy yêu cầu của khách hàng, người dùng.
- **Design:** Thiết kế hệ thống ứng dụng.
- **Implementation:** Các Developer tiến hành triển khai, phát triển hệ thống.
- **Verification:** Tester kiểm thử hệ thống và đợi khách hàng xác nhận.
- **Maintenance:** Bảo trì, nâng cấp, hoặc thêm các chức năng mới.

Ưu điểm:

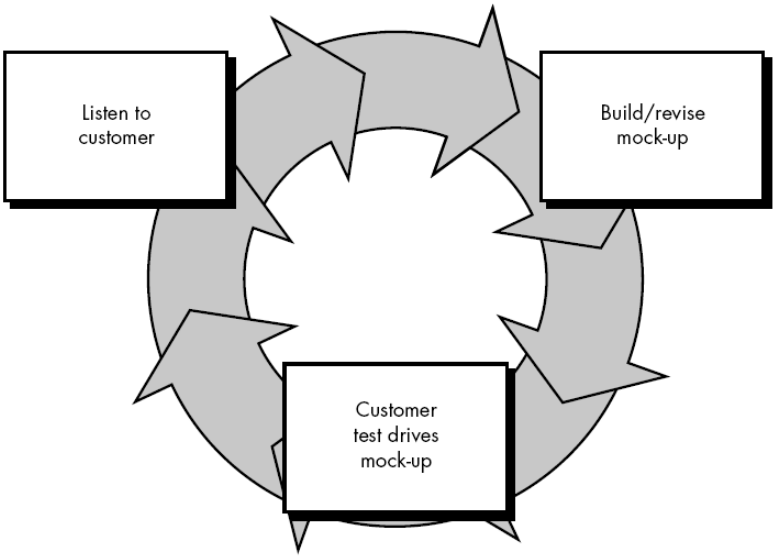
- Mô hình dễ hiểu và áp dụng.
- Dễ dàng nắm bắt được nội dung của các giai đoạn.

Nhược điểm:

- Khi giai đoạn trước có lỗi rất khó để sửa đổi hoặc mất nhiều thời gian và chí phí để sửa đổi.

2.4.2. **Mô hình tạo bản mẫu.**

Mô hình tạo bản mẫu sẽ xây dựng một bản mẫu cho người dùng sử dụng và đánh giá, sau đó bản mẫu đó sẽ được chỉnh sửa nhiều lần cho đến khi đạt đến được yêu cầu của khách hàng.



Hình 2.5: Mô hình tạo bản mẫu.

Ưu điểm:

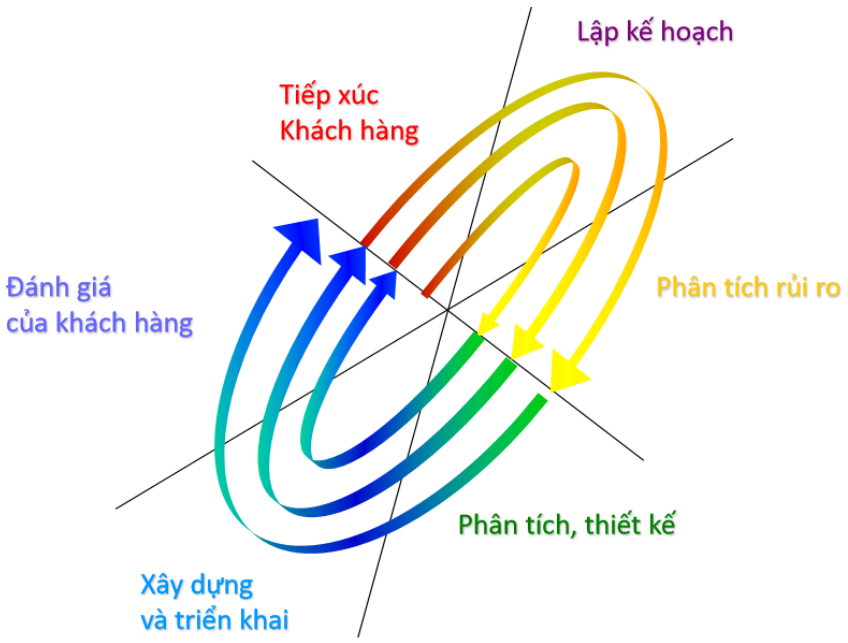
- Khách hàng sớm được sử dụng, tương tác với hệ thống. Người phát triển và khách hàng trao đổi và đánh giá để đưa ra hướng phát triển mới
- Nhờ bản mẫu những nhà phát triển có thể xác định nhanh các yêu cầu của khách hàng.
- Từ bản mẫu khách hàng và người phát triển có thể đưa ra nhiều yêu cầu hoặc tính năng mới và cần thiết cho ứng dụng.

Nhược điểm:

- Do phát triển sản phẩm nhanh chóng nên thiếu phần tài liệu, những phân tích về ứng dụng, dẫn đến vòng lặp code rồi sửa lỗi.
- Hệ thống để xây dựng có thể có cấu trúc không tốt do lựa chọn nhanh chóng để xây dựng ra bản mẫu và dẫn đến việc hiệu năng về sau này.

2.4.3. **Mô hình xoắn ốc.**

Mô hình dạng xoắn ốc được xây dựng dựa trên mô hình thác nước kế hợp với quá trình lặp theo vòng.



Hình 2.6: Mô hình xoắn ốc.

Một vòng lặp trong mô hình xoắn ốc bắt đầu tiếp xúc khách hàng => Lập kế hoạch => Phân tích rủi ro => Phân tích thiết kế => Xây dựng và triển khai => Đánh giá của khách hàng. Số lần lặp lại cho đến khi sản phẩm ứng dụng hoàn thiện và đạt yêu cầu của khách hàng.

Ưu điểm:

- Việc phân tích rủi ro nhiều lần cho nên tránh được hầu hết các rủi ro.
- Mô hình ứng dụng tốt vào trong các dự án lớn.
- Tài liệu được viết đầy đủ và kiểm duyệt liên tục.
- Khách hàng có thể phản hồi về sản phẩm và có thể thay đổi yêu cầu.

Nhược điểm:

- Việc phân tích rủi ro ban đầu khó khăn và cần chuyên gia có nhiều kinh nghiệm.
- Ứng dụng mô hình vào các dự án nhỏ không thực sự tốt.
- Thời gian để phát triển sản phẩm là vô hạn dẫn đến việc mất nhiều thời gian và tài chính.

Chương 3. PHÂN TÍCH YÊU CẦU ỨNG DỤNG

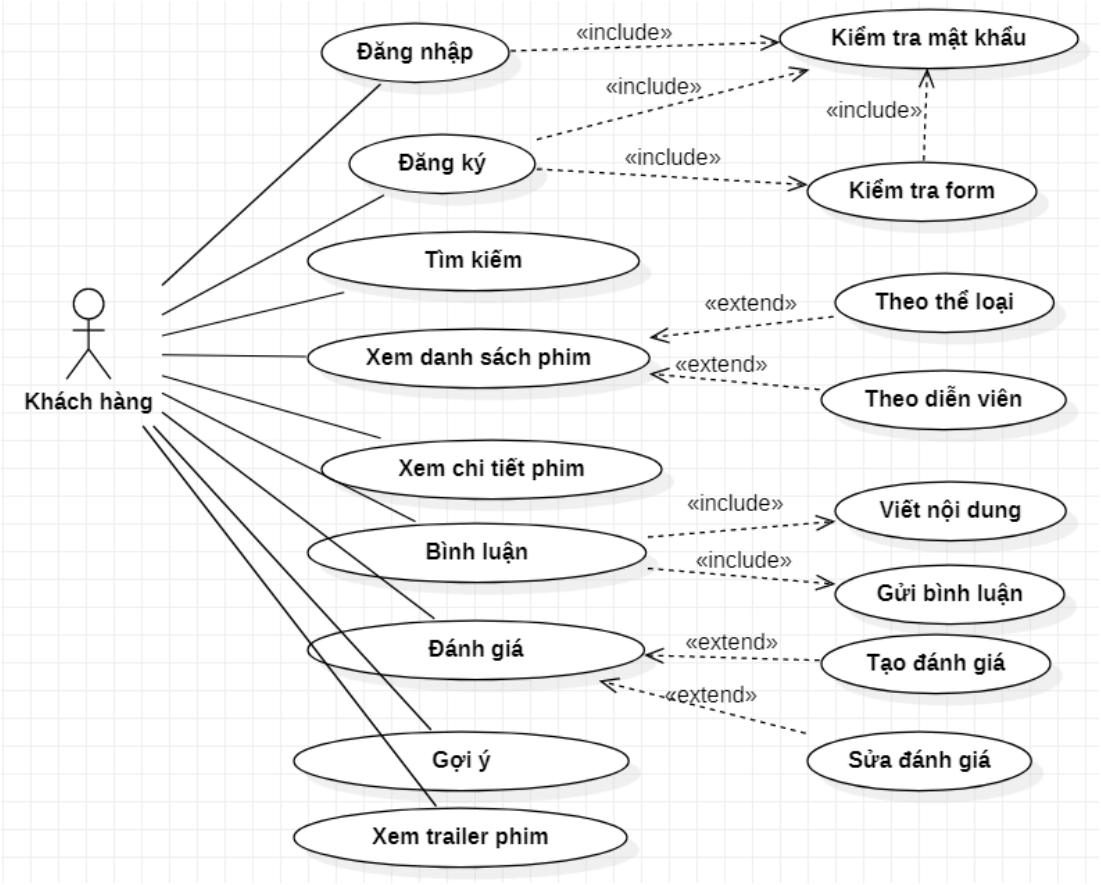
Tóm tắt chương: *Chương 3 chúng tôi sẽ trình bày các phân tích về ứng dụng: (1) phân tích use case; (2) phân tích giao diện; (3) phân tích các thực thể. Qua những phân tích, chúng tôi có căn cứ để tạo các mẫu thiết kế cho ứng*

3.1. Phân tích use case.

Trong phần này chúng tôi phân tích use case có trong ứng dụng để có cái nhìn tổng quan cho phần thiết kế về sau.

3.1.1. Use case các chức năng trong ứng dụng.

Trong ứng dụng chúng tôi phân tích thành các chức năng của người dùng gồm có: Đăng ký, đăng nhập, tìm kiếm, bình luận, đánh giá và xem trailer. Với từng chức năng chúng tôi tiến hành phân tích các use case và tổng hợp trong bảng đặc trả use case.



Hình 3.1: Use case diagram ứng dụng xem phim.

- Use case kiểm tra mật khẩu: Khi người dùng nhập mật khẩu vào trong trường mật khẩu => kiểm tra mật khẩu:
    - Nếu mật khẩu lớn hơn 3 ký tự thì hợp lệ và cho phép gửi thông tin
    - Nếu mật khẩu nhỏ hơn 3 ký tự thì không cho phép gửi thông tin
- Use case đăng nhập.

Use case	Nội dung
Tên	Đăng nhập
Mô tả	Cho phép người dùng đăng nhập vào ứng dụng
Tiền điều kiện	1. Người dùng chưa đăng nhập 2. Tài khoản đã được tạo

	3. Thiết bị người dùng được kết nối internet khi thực hiện đăng nhập
Hậu điều kiện	Người dùng được chuyển đến màn hình trang chủ phù hợp
Luồng sự kiện chính	1. Người dùng vào màn hình đăng nhập 2. Người dùng nhập tên đăng nhập và mật khẩu 3. Nhấn nút đăng nhập 4. Hệ thống kiểm tra thông tin đăng nhập 5. Nếu thông tin đăng nhập hợp lệ thì hệ thống chuyển người dùng đến màn hình trang chủ phù hợp
Luồng sự kiện phụ	A. Đăng nhập thất bại 1. Nếu đăng nhập thất bại thì hệ thống chuyển người dùng về màn hình đăng nhập. 2. Hệ thống sẽ hiển thị thông báo lỗi "Thông tin đăng nhập không hợp lệ" 3. Trở lại bước 2 của luồng sự kiện chính B. Người dùng thoát khỏi màn hình đăng nhập 1. Người dùng nhấn thoát khỏi giao đăng nhập 2. Hệ thống chuyển người dùng về lại màn hình xin chào của ứng dụng

Bảng 3.1: Use case đăng nhập.

- Use case kiểm tra form

  Trước khi gửi đăng ký thì thông tin người dùng cần được nhập đầy đủ. Trong form đăng ký kiểm tra các thông tin:

  - Mật khẩu lớn hơn 3 ký tự

- Địa chỉ email có ký tự "@" trong tên email.

Sau khi gửi form đăng ký hệ thống sẽ tiến hành kiểm tra thông tin:

- Username có trùng trong cơ sở dữ liệu khách hàng không, nếu có trùng thì trả về kết quả đăng ký thất bại
- Kiểm tra địa chỉ email đã có trong cơ sở dữ liệu khách hàng chưa, nếu có trả kết quả đăng ký thất bại

- Use case đăng ký.

Use case	Nội dung
Tên	Đăng ký
Mô tả	Cho phép người dùng đăng ký tài khoản mới
Tiền điều kiện	1. Người dùng chưa có tài khoản 2. Người dùng muốn sử dụng ứng dụng xem phim 3. Thiết bị phải được kết nối internet
Hậu điều kiện	Tài khoản người dùng tạo thành công
Luồng sự kiện chính	1. Điền thông tin đăng ký tại màn hình đăng ký 2. Kiểm tra form 3. Gửi thông tin đăng ký 4. Kiểm tra thông tin đăng ký hợp lệ 5. Gửi thông báo đăng ký thành công 6. Chuyển về màn hình đăng nhập
Luồng sự kiện phụ	2a. Mật khẩu phải có độ dài từ 3 ký tự 2b. Địa chỉ email phải có chứa "@" 4a. Kiểm tra tên người dùng đã tồn tại trong hệ thống chưa nếu tồn tại quay lại bước 1 trong luồng sự kiện chính và thông báo đăng ký thất bại 4b. Kiểm tra email có tồn tại trong hệ thống chưa nếu tồn tại quay lại bước 1 trong luồng sự kiện chính và gửi thông báo đăng ký thất bại

Bảng 3.2: Use case đăng ký.

- Use case tìm kiếm.

Use case	Nội dung
Tên	Tìm kiếm
Mô tả	Cho phép người dùng nhập từ khóa vào ô tìm kiếm và trả về danh sách các phim có chứa từ khóa người dùng đã nhập.
Tiền điều kiện	Không có
Hậu điều kiện	Trả về kết quả danh sách các phim có chứa từ người dùng đã khóa tìm
Luồng sự kiện chính	1. Người dùng nhập từ khóa vào ô tìm kiếm 2. Nhấn nút tìm kiếm 3. Hệ thống kiểm tra từ khóa tìm kiếm 4. Nếu từ khóa tìm kiếm có trong cơ sở dữ liệu phim thì trả về danh sách các bộ phim có chứa từ khóa tìm kiếm
Luồng sự kiện phụ	A. Từ khóa tìm kiếm quá ngắn 1. Nếu từ khóa tìm kiếm ít hơn 2 ký tự ứng dụng sẽ không thực hiện chức năng tìm kiếm và thông báo lỗi "Từ khóa tìm kiếm phải lớn hơn 2 ký tự" 2. Trở lại bước 1 luồng sự kiện chính B. Tìm kiếm thất bại 1. Nếu tìm kiếm thất bại hệ thống trả về màn hình không có bộ phim nào phù hợp

	2. Trở lại bước 1 luồng sự kiện chính C. Thoát khỏi màn hình tìm kiếm 1. Người dùng nhấn nút quay trở lại 2. Ứng dụng sẽ chuyển người dùng về màn hình trang chủ

Bảng 3.3: Use case tìm kiếm.

- Use case xem danh sách phim

Use case	Nội dung
Tên	Xem danh sách phim
Mô tả	Cho phép người dùng chọn danh sách phim muốn xem
Tiền điều kiện	Người dùng đang ở trong màn hình chi tiết phim
Hậu điều kiện	Người dùng chọn danh sách phim theo thể loại hoặc theo diễn viên
Luồng sự kiện chính	1. Người dùng nhấn chọn hình đại diện của diễn viên hoặc chọn thể loại phim. 2. Người dùng được chuyển đến trang danh sách phim đã chọn. 3. Hiện thị danh sách các bộ phim liên quan đến nội dung người dùng chọn
Luồng sự kiện phụ	A. Tải danh sách các bộ phim thất bại 1. Mạng internet bị ngắt kết nối

Bảng 3.4: Use case xem danh sách phim.

- Use case xem chi tiết phim

Use case	Nội dung
Tên	Xem chi tiết phim
Mô tả	Cho phép người dùng xem chi tiết thông tin của bộ phim
Tiền điều kiện	Người dùng chọn một bộ phim
Hậu điều kiện	Người dùng được chuyển đến trang chi tiết bộ phim
Luồng sự kiện chính	1. Người dùng nhấn chọn hình đại bộ phim muốn xem 2. Người dùng được chuyển đến màn hình chi tiết phim 3. Màn hình chi tiết phim hiển thị các thông tin về bộ phim: thời lượng, diễn viên, thể loại, tóm tắt nội dung
Luồng sự kiện phụ	A. Tải màn hình chi tiết phim thất bại 1. Mạng internet bị ngắt kết nối

Bảng 3.5: Use case xem chi tiết phim.

- Use case bình luận

Use case	Nội dung
Tên	Bình luận
Mô tả	Cho phép người dùng gửi bình luận bằng cách
Tiền điều kiện	Người dùng đang ở màn hình chi tiết phim và muốn đưa ra bình luận của mình về bộ phim
Hậu điều kiện	Bình luận của người dùng được lưu vào cơ sở dữ liệu và hiển thị trên màn hình bình luận tất cả các bình luận.
Luồng sự kiện chính	4. Người dùng nhập nội dung bình luận 5. Nhấn nút gửi bình luận 6. Bình luận được thêm vào cơ sở dữ liệu 7. Tải lại tất cả các bình luận của phim
Luồng sự kiện phụ	2. Bình luận thất bại

	2. Mạng internet bị ngắt kết nối 3. Trạng thái gửi bình luận và chờ phản hồi internet 4. Bình luận không được thêm vào cơ sở dữ liệu

Bảng 3.6: Use case bình luận.

- Use case đánh giá

Use case	Nội dung
Tên	Đánh giá phim
Mô tả	Cho phép người dùng đánh giá bộ phim hiện tại đang theo dõi và có thể chỉnh sửa đánh giá phim
Tiền điều kiện	Người dùng muốn đánh giá bộ phim đang xem hoặc sửa đánh giá của mình trước đó
Hậu điều kiện	Đánh giá được cập nhật mới với đánh giá của người dùng
Luồng sự kiện chính	1. Người dùng chọn số lượng sao muốn đánh giá chop him 2. Nhấn nút gửi đánh giá 3. Đánh giá gửi thành công 4. Cập nhật lại thống kê số lượng đánh giá
Luồng sự kiện phụ	A. Đánh giá thất bại 1. Nếu đánh giá của người dùng bị gián đoạn thì màn hình đánh giá không được tải lại 2. Trạng thái gửi bình luận sẽ chờ cho đến khi có kết nối.

Bảng 3.7: Use case đánh giá.

- Use case xem trailer

Use case	Nội dung
Tên	Xem trailer
Mô tả	Video trailer của bộ phim sẽ được phát dựa trên đường dẫn video trailer
Tiền điều kiện	Người dùng muốn xem trailer
Hậu điều kiện	Video được phát
Luồng sự kiện chính	1. Người dùng nhấn vào nút xem trailer 2. Video trailer được phát trên màn hình mới
Luồng sự kiện phụ	A. Thoát xem trailer 1. Người dùng nhấn vào nút trở về để thoát xem trailer B. Trailer không tải được 1. Nếu không tải được đường dẫn trailer thì màn hình sẽ hiển thị hình ảnh "Coming soon"

Bảng 3.8: Use case xem trailer.

3.1.2. Use case gợi ý phim cho người dùng.

Trong ứng dụng gợi ý phim sẽ chia người dùng ra ba danh mục rồi sau đó tiến hành đưa ra gợi ý cho từng người dùng theo những danh mục đã phân loại. Các danh mục người dùng gồm:

- Người dùng cũ đã có trong hệ thống và đã có mô hình gợi ý phim được xây dựng cho những người dùng này.
- Người dùng đăng ký mới, có tích chọn thể loại phim yêu thích trong khi đăng ký và chưa có mô hình gợi ý nào được xây dựng cho người dùng này.
- Người dùng đăng ký mới, không tích chọn thể loại phim yêu thích và chưa có mô hình gợi ý phim nào được xây dựng cho người dùng này.

- Người dùng cũ và đã có mô hình gợi ý phim.

Use case	Nội dung
Tên	Gợi ý phim cho người dùng cũ
Mô tả	Khi người dùng đăng nhập thành công ứng dụng sẽ kiểm tra xem là người dùng cũ đã có mô hình gợi ý hay là người dùng mới đăng ký
Tiền điều kiện	Người dùng đăng nhập thành công
Hậu điều kiện	Hiển thị danh sách các bộ phim được gợi ý cho người dùng
Luồng sự kiện chính	1. Người dùng đăng nhập thành công 2. Kiểm tra người dùng mới hay cũ 3. Chuyển người dùng qua màn hình trang chủ 4. Danh sách các bộ phim được gợi ý cho người dùng
Luồng sự kiện phụ	Không có

Bảng 3.9: Use case gợi ý phim cho người dùng cũ.

- Người dùng đăng ký mới và không có thể loại phim yêu thích.

Use case	Nội dung
Tên	Gợi ý phim cho người dùng mới và không có thể loại phim yêu thích
Mô tả	Khi người dùng đăng nhập thành công ứng dụng sẽ kiểm tra xem là người dùng cũ đã có mô hình gợi ý hay là người dùng mới đăng ký và kiểm tra người dùng có thể loại phim yêu thích hay không
Tiền điều kiện	Người dùng đăng nhập thành công
Hậu điều kiện	Danh sách các bộ phim được gợi ý cho người dùng mới

Luồng sự kiện chính	1. Người dùng đăng nhập thành công 2. Kiểm tra người dùng mới hay cũ 3. Chuyển người dùng qua màn hình trang chủ 4. Danh mục các bộ phim mới phát hành và các bộ phim gợi ý ngẫu nhiên 5. Người dùng nhấn vào một bộ phim bất kỳ sẽ được chuyển đến màn hình chi tiết phim. 6. Trong màn hình chi tiết phim thì ứng dụng sẽ gợi ý cho người dùng danh sách các bộ phim có cùng thể loại với bộ phim người dùng đã chọn
Luồng sự kiện phụ	Không có

Bảng 3.10: Use case gợi ý phim cho người dùng mới.

- Người dùng đăng ký mới và có thể loại phim yêu thích.

Use case	Nội dung
Tên	Gợi ý phim cho người dùng mới và có thể loại phim yêu thích
Mô tả	Khi người dùng đăng nhập thành công ứng dụng sẽ kiểm tra xem là người dùng cũ đã có mô hình gợi ý hay là người dùng mới đăng ký và kiểm tra người dùng có thể loại phim yêu thích hay không
Tiền điều kiện	Người dùng đăng nhập thành công
Hậu điều kiện	Danh sách các bộ phim được gợi ý cho người dùng mới
Luồng sự kiện chính	1. Người dùng đăng nhập thành công

	2. Kiểm tra người dùng mới hay cũ và cho thể loại phim yêu thích hay không 3. Nếu có chuyển người dùng sang màn hình trang chủ 4. Danh mục các bộ phim được gợi ý cho người dùng là các bộ phim có chứa thể loại yêu thích của người dùng và các bộ phim đó đồng thời được đánh giá cao trong bảng xếp hạng phim.
Luồng sự kiện phụ	Không có

Bảng 3.11: Use case gợi ý phim cho người dùng mới và có loại phim yêu thích.

3.2. **Phân tích giao diện.**

Trong phần giao diện ứng dụng chúng tôi sẽ tiến hành phân tích các nội dung hiện thị trong các màn hình dự kiến xây dựng. Các màn hình được xây dựng dựa trên các chức năng, các use case đã phân tích.

3.2.1. Đăng nhập.

Màn hình đăng nhập của ứng dụng chúng tôi sẽ sử dụng các thuộc tính đăng nhập là tên đăng nhập và mật khẩu.

Trong màn hình đăng nhập người dùng có thể chuyển đến trang đăng ký nếu trường hợp người dùng không có tài khoản.

Trường hợp người dùng không quên mật khẩu thì sẽ được chuyển đến trang lấy lại mật khẩu.

Có thể cho người dùng đăng nhập bằng các tài khoản online khác như:

facebook, google để sử dụng ứng dụng.

3.2.2. Đăng ký.

Giao diện màn hình đăng ký chúng tôi sẽ xây dựng gồm các nội dung:

- Các trường thông tin của người dùng: tên đăng nhập, mật khẩu, tên, ngày sinh, địa chỉ, địa chỉ email, thể loại phim yêu thích.
- Nếu trường hợp người dùng đã có tài khoản thì có thể chuyển đến trang đăng nhập và tiến hành đăng nhập sau đó sử dụng ứng dụng.
- Khi người dùng nhập thông tin thì sẽ tiến hành kiểm tra các định dạng trước khi gửi form đăng ký lên trên API. Các thông tin như mật khẩu, địa chỉ email.
- Nếu form đăng ký gửi lên API thì sau một khoảng thời gian người dùng sẽ nhận được thông báo về việc đăng ký thành công hay thất bại.

3.2.3. Trang chủ.

Qua quá trình khảo sát chúng tôi tiến hành phân tích các thành phần cần có trong màn hình trang chủ:

- Có thanh điều hướng để cho phép người dùng chọn các nội dung như: Xem thông tin người dùng hoặc chuyển đến trang chủ, xem thông báo.
- Chức năng tìm kiếm đặt ở màn hình trang chủ để người dùng có thể sử dụng nhanh nhất có thể.
- Hiển thị danh sách các bộ phim được phân chia theo danh mục: danh mục phim mới, phim phổ biến, phim gợi ý cho người dùng.

3.2.4. Hiển thị danh sách phim.

Màn hình hiển thị danh sách phim sẽ được gọi sau khi người dùng nhấn vào nút xem thêm của các danh mục phim, danh sách phim của diễn viên hoặc danh sách phim cùng thể loại.

Nội dung hiển thị dưới dạng danh sách có thể cuộn được theo chiều dọc. Thông tin các bộ phim hiển thị trong danh sách gồm thông tin về hình ảnh quảng cáo của bộ phim, tên bộ phim, số sao đánh giá, thời lượng phim, thể loại của phim.

3.2.5. Chi tiết phim.

Màn hình chi tiết phim chúng tôi phân tích gồm các nội dung:

- Hiển thị hình ảnh quảng cáo của bộ phim.
- Thông tin chi tiết của bộ phim: Tên phim, số sao đánh giá, danh sách các diễn viên của bộ phim, thể loại của bộ phim, tóm tắt phim, thời lượng, năm phát hành, doanh thu.
- Trong màn hình chi tiết người dùng có thể nhấn để chuyển đến màn hình xem danh sách phim, xem trailer phim.
- Phần cuối màn hình sẽ để các bộ phim gợi ý dành cho người dùng.

3.2.6. Chi tiết thông tin người dùng.

Giao diện màn hình chi tiết người dùng sẽ hiển thị các thông tin của người dùng. Tại giao diện sẽ cho người dùng thay đổi thông tin email, mật khẩu, thể loại phim yêu thích.

3.2.7. Tìm kiếm.

Màn hình tìm kiếm thiết kế có trường để cho người dùng nhập từ khóa tìm kiếm, từ khóa tìm kiếm cần có ràng buộc lớn hơn 2 ký tự để có kết quả tìm kiếm tốt. Sau khi tìm kiếm người dùng sẽ được trả kết quả là danh sách các bộ phim liên quan đến từ khóa.

3.2.8. Đánh giá.

Giao diện đánh giá hiển thị số lượng các đánh giá từ 1 đến 5 sao. Trong giao diện cho phép người dùng tạo đánh giá cho bộ phim. Có nút gửi đánh giá

và sau khi gửi màn hình sẽ tự động tải lại để cập nhật lại thống kê số lượng đánh giá.

3.2.9. Bình luận.

Giao diện bình luận gồm các nội dung:

- Người dùng có thể nhập văn bản đánh giá, các biểu cảm icon.
- Có nút gửi bình luận, sau khi gửi thì bình luận sẽ được thêm vào danh sách các bình luận của bộ phim.
- Danh sách bình luận sẽ có thể cuộn theo chiều dọc vì một bộ phim có thể có nhiều người bình luận.
- Thông tin một bình luận sẽ hiển thị tên người dùng, thời gian bình luận, nội dung của người viết.

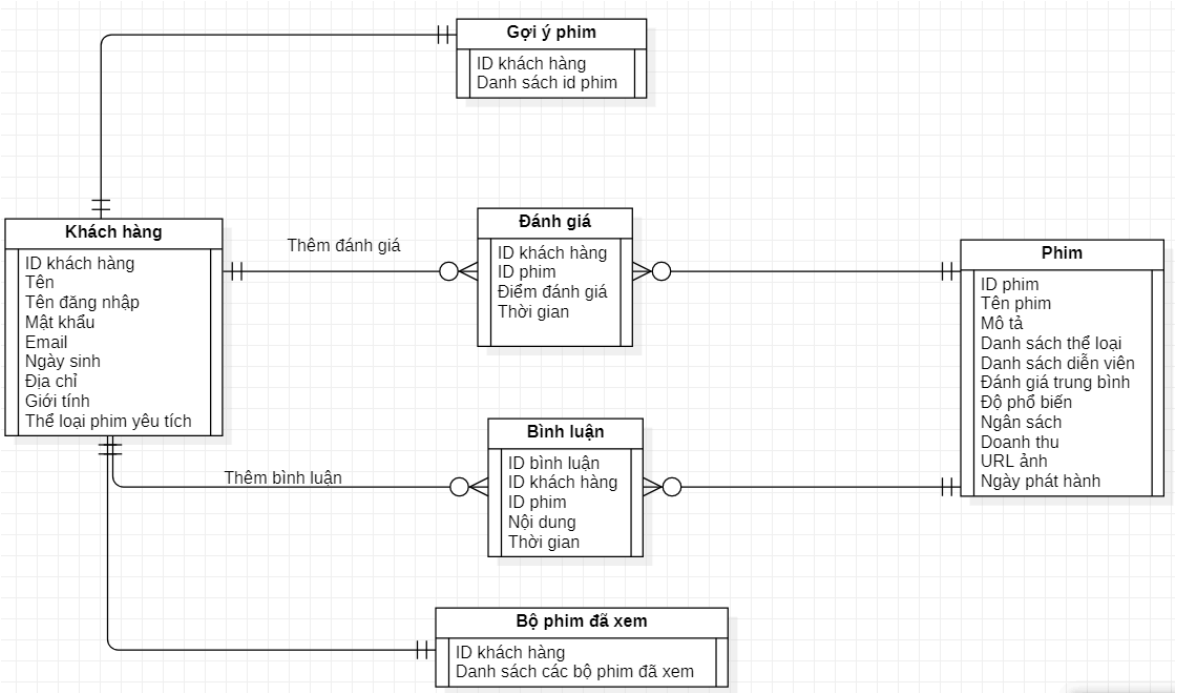
3.3. Phân tích các thực thể.

Dựa vào bộ dữ liệu movielens chúng tôi sẽ xác định các thực thể để xây dựng cơ sở dữ liệu. Các thực thể chính chính trong cơ sở dữ liệu phim gồm:

- Khách hàng người sử dụng ứng dụng, các chức năng được xây dựng cho người dùng có đăng ký, đăng nhập, đánh giá, bình luận, tìm kiếm, lọc danh sách phim, đưa ra phim gợi.
- Bộ phim chứa các thuộc tính của phim như: tên phim, nội dung tóm tắt, danh sách các diễn viên, danh sách các thể loại, mức độ phổ biến

3.3.1. Xây dựng sơ đồ ERD.

Sau khi phân tích các thực thể và mối liên hệ, chúng tôi xây dựng sơ đồ mối liên hệ thực thể như sau:



Hình 3.2: Sơ đồ ERD

Chi tiết mối quan hệ giữa các thực thể trong sơ đồ ERD:

- Thực thể khách hàng quan hệ 1-1 với thực thể gợi ý phim: Một khách hàng sẽ có một danh sách các bộ phim gợi ý, danh sách các bộ phim gợi ý được xây dựng từ mô hình Singular value decomposition của nhóm tác giả khóa luận trước, nhóm tác giả và tên khóa luận chúng tôi có trích dẫn trong phần tài liệu tham khảo.
- Thực thể khách hàng quan hệ 1-1 với thực thể danh sách phim đã xem: Khi khách hàng xem một bộ phim nào đó trong danh sách phim gợi ý thì bảng danh sách phim đã xem sẽ tự động thêm ID bộ phim mà người dùng đã xem.
- Thực thể khách hàng quan hệ 1 - N với thực thể đánh giá: Một khách hàng có thể có một hoặc nhiều đánh giá, với một bộ phim khách hàng đánh giá thì không thể thêm đánh giá mới nhưng có thể cập nhật, trong

bảng đánh giá khóa chính gồm ID khách hàng và ID phim vì vậy không thể có 2 đánh giá của người dùng cùng một bộ phim.

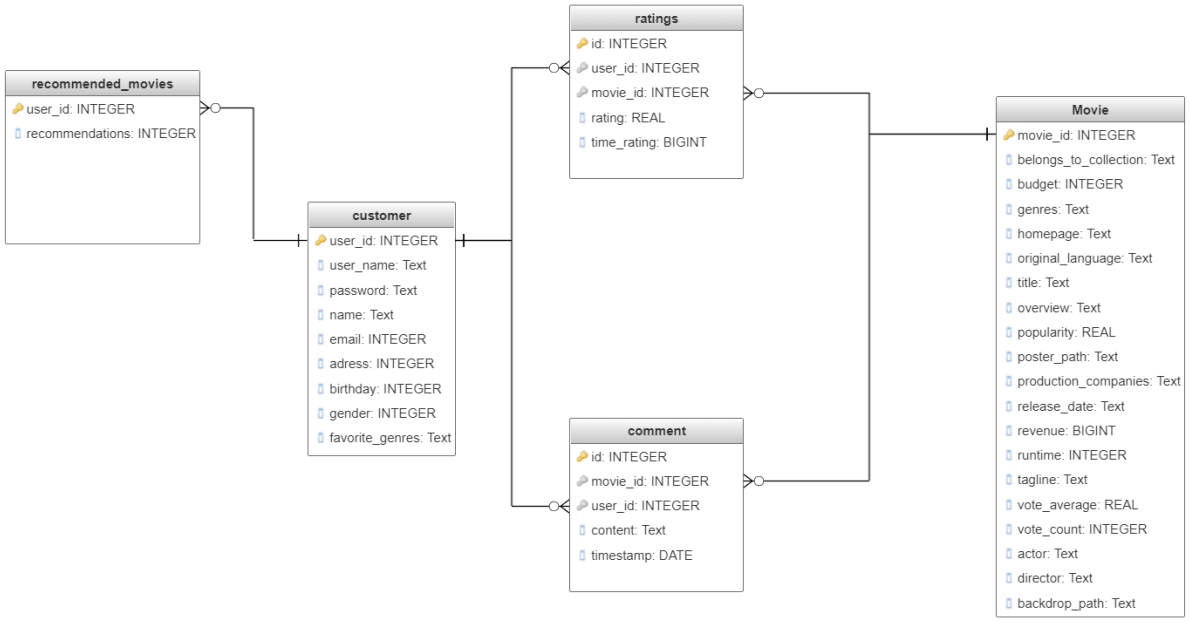
- Thực thể khách hàng quan hệ N-N với thực thể bình luận: Nhiều khách hàng có thể tạo nhiều bình luận hoặc một khác hàng có thể tạo nhiều bình luận, hoặc bình luận được tạo từ nhiều khách hàng.
- Thực thể phim quan hệ 1-N với thực thể đánh giá: Một bộ phim có thể có nhiều đánh giá khác nhau từ nhiều người dùng. Trong thực thể đánh giá có khóa chính là ID khách hàng và ID phim nên một người chỉ được tạo 1 đánh giá cho một bộ phim hoặc một người có thể đánh giá cho nhiều bộ phim.
- Thực thể phim quan hệ 1=N với thực thể bình luận: Một bộ phim có thể có nhiều bình luận.

Chương 4. THIẾT KẾ ỨNG DỤNG

Tóm tắt chương: *Chương 4 chúng tôi sẽ trình bày nội dung về các bản thiết kế: (1) thiết kế cơ sở dữ liệu; (2) thiết kế giao diện; (3) thiết kế kiến trúc ứng dụng.*

4.1. Cơ sở dữ liệu.

Trong khóa luận chúng tôi sử dụng cơ sở dữ liệu quan hệ Postgresql. Cơ sở dữ liệu xây dựng gồm 5 bảng.



Hình 4.1: Diagram cơ sở dữ liệu.

- Bảng customer.

Trong bảng customer sẽ lưu thông tin của người dùng. Khi người dùng đăng nhập vào hệ thống thì tiến hành truy cập vào bảng customer và xác định tên đăng nhập và mật khẩu người dùng. Ngoài ra nếu đăng ký mới người dùng phải cung

cấp thông tin cho các trường có trong bảng customer. Khóa chính là user_id cũng là khóa ngoại của bảng comment và ratings

STT	Tên trường	Mô tả	Kiểu dữ liệu	Quan hệ
1	user_id	Khóa chính của bảng customer	Integer	Khóa chính của bảng customer
2	user_name	Tên đăng nhập của người dùng	Text, not null	Không
3	password	Mật khẩu đăng nhập của người dùng	Text, not null	Không
4	name	Tên của người dùng khi đăng ký	Text, not null	Không
5	email	Địa chỉ email	Text, not null	Không
6	adress	Địa chỉ nơi ở	Text, not null	Không
7	birthday	Ngày sinh	Date, not null	Không
8	gender	Giới tính	Text, not null	Không
9	favorite_genres	Thể loại phim yêu thích	Text, allow null	Không

Bảng 4.1: Bảng Customer.

- Bảng movie.

STT	Tên trường	Mô tả	Kiểu dữ liệu	Quan hệ

1	movie_id	Khóa chính của bảng movie	Integer, not null	Khóa chính
2	belongs_to_collection	Thể hiện bộ sưu tập mà bộ phim đó thuộc về	Text, allow null	Không
3	budget	Ngân sách của bộ phim(dollar)	Integer, default=0	Không
4	genres	Danh sách các thể loại của bộ phim	Text, not null	Không
5	homepage	Trang chủ chính thức của bộ phim	Text, allow null	Không
6	original_language	Ngôn ngữ gốc của phim	Text, allow null	Không
7	title	Tiêu đề của phim	Text, not null	Không
8	overview	Sơ lược nội dung bộ phim	Text, not null	Không

9	popularity	Mức độ phổ biến do IMDb đánh giá	Real, allow null	Không
10	poster_path	Đường dẫn url đến hình ảnh giới thiệu phim	Text, allow null	Không
11	production_companies	Danh sách các công ty liên quan đến sản xuất phim	Text, allow null	Không
12	release_date	Ngày phát hành phim	Text, not null	Không
13	revenue	Doanh thu phim	Bigint, allow null	Không
14	runtime	Thời lượng của bộ phim tính bằng phút	Integer, not null	Không
15	tagline		Text, allow null	Không
16	vote_average	Đánh giá trung bình của phim trên IMDb	Real, not null	Không

17	vote_count	Số lượng người dùng bình chọn trung bình trên IMDb	Integer, allow null	Không
18	actor	Danh sách các diễn viên tham gia đóng phim	Text, not null	Không
19	director	Giám đốc sản xuất phim	Text, allow null	Không
20	backdrop_path	Đường dẫn url đến ảnh quảng cáo phim	Text, allow null	Không

Bảng 4.2: Bảng movie.

- Bảng ratings.

STT	Tên trường	Mô tả	Kiểu dữ liệu	Quan hệ
1	user_id	User id của bảng customer	Integer, not null	Khóa chính bảng rating, tham chiếu đến bảng customer

2	movie_id	Movie id của bảng movie	Integer, not null	Khóa chính bảng rating, tham chiếu đến bảng movie
3	rating	Số điểm rating	Real, not null	Không
4	time_rating	Thời gian rating	Bigint, not null	Không

Bảng 4.3: Bảng ratings.

- Bảng comment.

STT	Tên trường	Mô tả	Kiểu dữ liệu	Quan hệ
1	id	Khóa chính của bảng comment	Integer, not null	Khóa chính
2	user_id	User id của bảng customer	Integer, not null	Khóa ngoại bảng comment, tham chiếu đến bảng customer
3	movie_id	Movie id của bảng movie	Integer, not null	Khóa ngoại của bảng comment,

				tham chiếu đến bảng movie
4	content	Nội dung comment	Text, not null	Không
5	time_rating	Thời gian rating	Date, not null	Không

Bảng 4.4: Bảng comment

- Bảng recommended_movies.

STT	Tên trường	Mô tả	Kiểu dữ liệu	Quan hệ
1	user_id	User id của bảng customer	Integer, not null	Khóa chính bảng recommended movie, tham chiếu đến bảng customer
2	recommendations	Danh sách các movie id dùng để gợi ý cho user id	Text, not null	Không

Bảng 4.5: Bảng recommended movie.

Cài đặt hệ quản trị cơ sở dữ liệu Postgresql. Tải bản cài đặt Postgresql tại địa chỉ https://www.enterprisedb.com/downloads/postgres-postgresql-downloads.

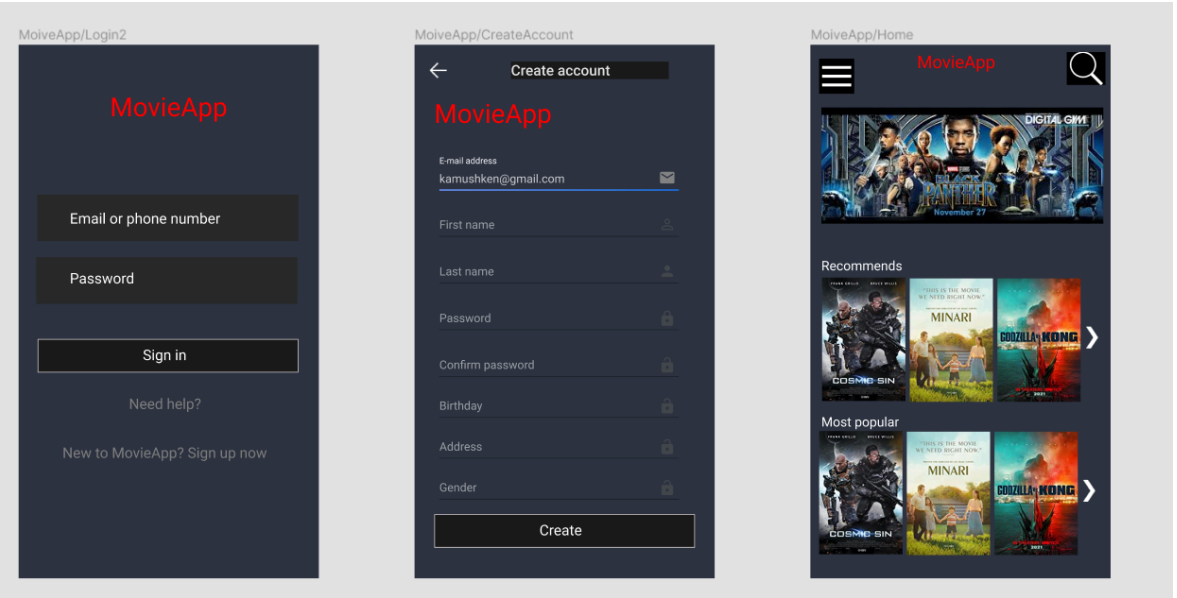
Chọn hệ điều hành đang sử dụng và tải về. Chúng tôi sử dụng hệ điều hành windows 10 64bit.

- Tạo mới một cơ sở dữ liệu trong postgresql đặt tên là movie_db.
- Tiến hành khôi phục cơ sở dữ liệu đã có từ khóa luận được kế thừa. Trong cơ sở dữ liệu kế thừa gồm các bảng customer, movie và recommended_movies.
- Sau khi đã khôi phục cơ sở dữ liệu, chúng tôi tiến hành tạo mới bảng comment và ratings.
- Cuối cùng tạo các ràng buộc khóa chính, khóa ngoại.

4.2. Giao diện ứng dụng.

Trong phần thiết kế giao diện ứng dụng chúng tôi dựa và những phân tích giao diện trong chương 3 để tiến hành phác thảo ra một giao diện gần đúng với những phân tích.

Chúng tôi sử dụng phần mềm Figma để tiến hành vẽ bản phác thảo giao diện ứng dụng và thực hiện các prototype của sản phẩm trên phần mềm.



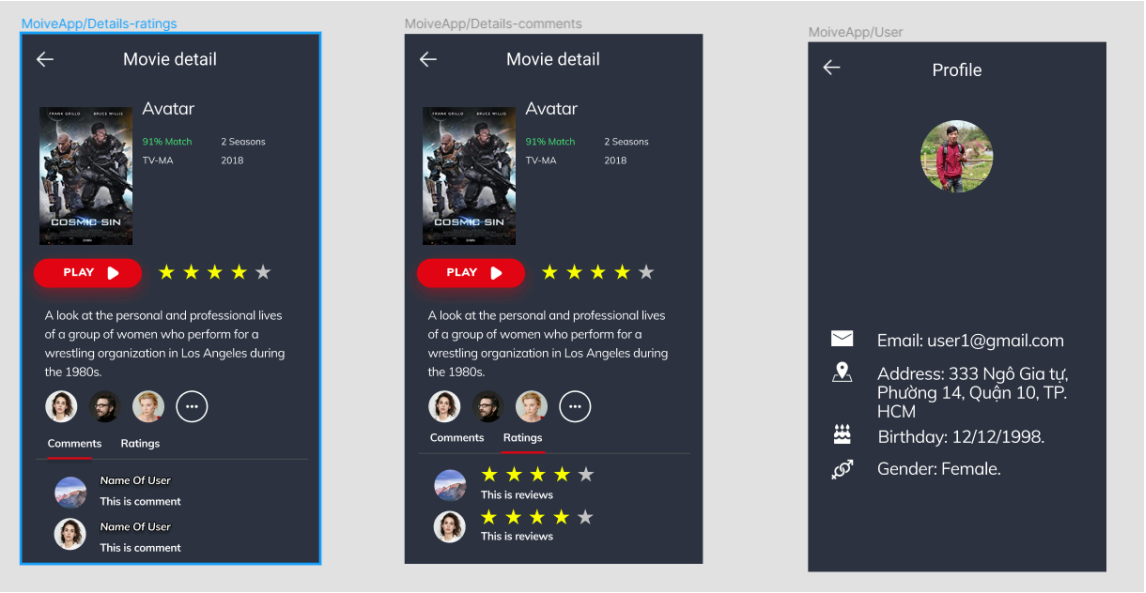
Hình 4.2: Đăng ký, đăng nhập, trang chủ.

Màu nền kết hợp giữa màu đen với chữ màu trắng thích hợp cho người dùng sử dụng trong không gian tối không chói mắt.

Màn hình đăng nhập gồm các trường cho người dùng nhập thông tin tên đăng nhập, mật khẩu và nút đăng nhập để người dùng gửi thông tin đăng nhập. Ngoài ra còn có thể chuyển đến màn hình đăng ký nếu người dùng không có tài khoản và muốn đăng ký mới.

Màn hình đăng ký cho phép người dùng đăng ký mới với việc điền các thông tin được yêu cầu trong màn hình. Người dùng có thể quay trở lại màn hình đăng ký nếu như người dùng đã có tài khoản.

Màn hình trang chủ hiển thị thông tin các bộ phim phổ biến, danh sách các bộ phim gợi ý. Bên trái là nút menu để chuyển đổi các trang, bên phải là icon tìm kiếm tương ứng với chức năng tìm kiếm.



Hình 4.3: Thông tin người dùng, chi tiết phim.

Màn hình chi tiết phim sẽ hiển thị thông tin về bộ phim, cho phép người dùng xem trailer phim, hiển thị các thông tin của diễn viên và có thể chọn để xem các bộ phim liên quan đến diễn viên.

Màn hình chi tiết phim còn có các tab cho các chức năng: Bình luân, đánh giá.

Màn hình thông tin người dùng sẽ hiện những thông tin như: tên, địa chỉ email, ngày sinh, giới tính.

4.3. Thiết kế kiến trúc ứng dụng.

Trong phần này chúng tôi trình bày kiến trúc trong xây dựng ứng dụng, kiến trúc để ứng dụng lấy thông tin từ cơ sở dữ liệu và lưu thông tin vào cơ sở dữ liệu.

4.3.1. Flutter Framwork.

Ngôn ngữ sử dụng để xây dựng giao diện và logic cho ứng dụng là ngôn ngữ dart. Dart được phát triển bởi google là ngôn ngữ lập trình hướng đối tượng, có cú pháp kiểu C, mã code Dart có thể biên dịch thành javascript để chạy trên trình duyệt.

Flutter được google phát triển để xây dựng giao diện một cách nhanh chóng và hiệu suất trải nghiệm tốt trên nền tảng mobile.

Các tính năng nổi bật của flutter:

- **Fast Development**: Google tạo ra tính năng Hot Reload hoạt động trong milliseconds để hiển thị giao diện trên thiết bị. Ngoài ra còn có các widget được dựng sẵn giúp cho việc xây dựng giao diện nhanh chóng. [5]
- **Express and Flexible UI:** Các widget được xây dựng đẹp mắt theo Material Design và Cupertino, các API chuyển động, điều khiển thanh cuộn mượt mà. [5]

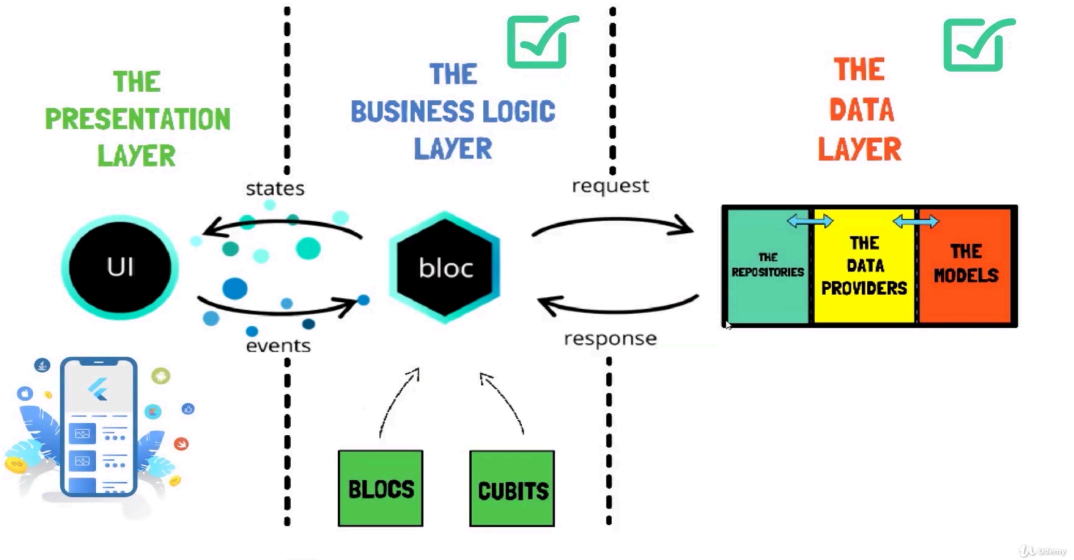
- **Native Performance:** Trên các nền tảng khác nhau thì các widget của flutter có sự khác biệt ví dụ: icons, scrolling, navigations, font vậy nên cung cấp hiệu năng tốt cho từng nền tảng. Code flutter được biên dịch thành mã máy ARM gốc bằng trình biên dịch gốc của Dart. [5]

Flutter framework được phát triển bởi google, là một bộ UI toolkit dùng để xây dựng giao diện ứng dụng. Thành phần trong xây dựng giao diện ứng dụng gồm các Widgets. Những widgets thường được sử dụng: Text, Column, Row, Container, … [5]

Widget có hai loại: StatefulWidget và StatelessWidget. Hiểu đơn giản có hai loại widget có state và widget không có state. Những widget có state thì khi thay đổi state thì sẽ làm thay đổi giao diện ứng dụng. [5]

Để quản lý các state trong ứng dụng flutter có nhiều hướng quản lý: setState là quản lý đơn giản nhất, Provider, Bloc, Getx, …

Trong ứng dụng gợi ý phim của chúng tôi xây dựng gồm 3 lớp khác nhau:



Hình 4.4: Phân chia lớp trong ứng dụng.

(*nguồn: Udemy, Flutter BLoC - From Zero to Hero Complete Course.*)

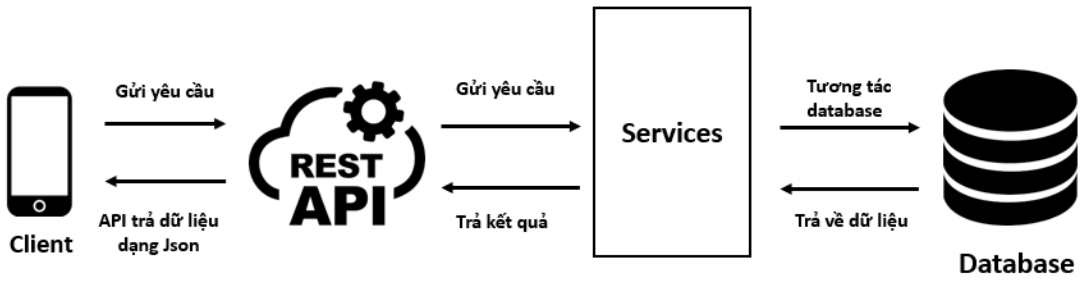
The Presentation Layer: Có nhiệm vụ hiển thị thông tin từ dữ liệu được trả về, ví dụ: thông tin chi tiết phim, thông tin người dùng, …

The Business Logic Layer: Có trách nhiệm xử lý các sự kiện như load dữ liệu, xử lý các sự kiện đăng nhập, đăng ký, …

The Data Layer: Nhiệm vụ khai báo các model, xây dựng các phương thức lấy, gửi dữ liệu cho web api, sau đó trả dữ liệu về dạng Json hoặc models về cho lớp business Logic Layer và sau đó trả về các state cho lớp the presentation layer.

4.3.2. Kiến trúc hệ thống.

Trong mục này chúng tôi trình bày kiến trúc hệ thống của ứng dụng gợi ý phim.



Hình 4.5: Kiến trúc hệ thống.

- Người dùng ứng dụng trên thiết bị di động: Ứng dụng được xây dựng bằng flutter framework, khi người dùng gửi một yêu cầu thì điểm nhận dữ liệu người dùng là Web API.

- Web API chịu tránh nhiệm nhận các yêu cầu từ người dùng sau đó sử dụng các dịch vụ tương ứng để xử lý yêu cầu của người dùng gửi. Dữ liệu trả về là Json cho người dùng để hiện thị trên ứng dụng.
- Các services tương tác với cơ sở dữ liệu để xử lý yêu cầu từ người dùng. Services được xây dựng bằng Django rest framework
    - Service người dùng cho chức năng đăng nhập, đăng ký, truy xuất thông tin.
    - Service gợi ý phim: nhận thông tin của người dùng và trả về danh sách các bộ phim gợi ý.
    - Service bình luận: nhận bình luận gửi từ người dùng sau đó lưu vào cơ sở dữ liệu, trả lại dữ liệu là danh sách các bình luận.
    - Service đánh giá: nhận đánh giá được gửi từ người dùng sau đó lưu và cơ sở dữ liệu, dữ liệu trả về là số lượng các đánh giá.
- Cơ sở dữ liệu thực hiện nhiệm vụ lưu trữ thông tin như người dùng, các bộ phim, bình luận, đánh giá.

Chương 5. CÀI ĐẶT VÀ TRIỂN KHAI ỨNG DỤNG

Tóm tắt chương: *Chương 5 chúng tôi sẽ trình bày các cài đặt framework, triển khai và phát triển thành ứng dụng di động. Chúng tôi trình bày các kết quả xây dựng và triển khai ứng dụng*

5.1. Cài đặt và triển khai Web API

Trong phần này chúng tôi tiến hành các bước cài đặt với Django rest framework. Sau đó là quá trình xây dựng và kết quả triển khai web api.

5.1.1. Cài đặt môi trường.

Django Rest Framework sử dụng ngôn ngữ python nên đầu tiên chúng tôi tiến hành cài đặt môi trường để viết ngôn ngữ python:

- Tiến hành tải bản release python tại https://www.python.org/downloads/. Trong khóa luận này tôi sử dụng Python phiên bản 3.8.2.

Sau khi cài đặt xong chúng ta có thể chạy lệnh python trong command prompt để kiểm tra quá trình cài đặt đã thành công hay chưa.

Cài đặt Django. [3]

- Sau khi cài đặt xong python, chúng tôi tiến hành Django bằng lệnh trên command prompt.

  python -m pip install Django

- Tiếp theo tiến hành cài môi trường ảo để phát triển web api bằng Django.

  virtualenv env -p python

- Nếu virtualenv chưa được cài đặt thì có thể cài đặt dễ dàng bằng lệnh

  pip install virtualenv

- Sau khi tạo mới môi trường, tiếp theo tiến hành active môi trường.
- Đi vào thư mục env bằng lệnh *cd env.* Sau đó để active chạy lệnh

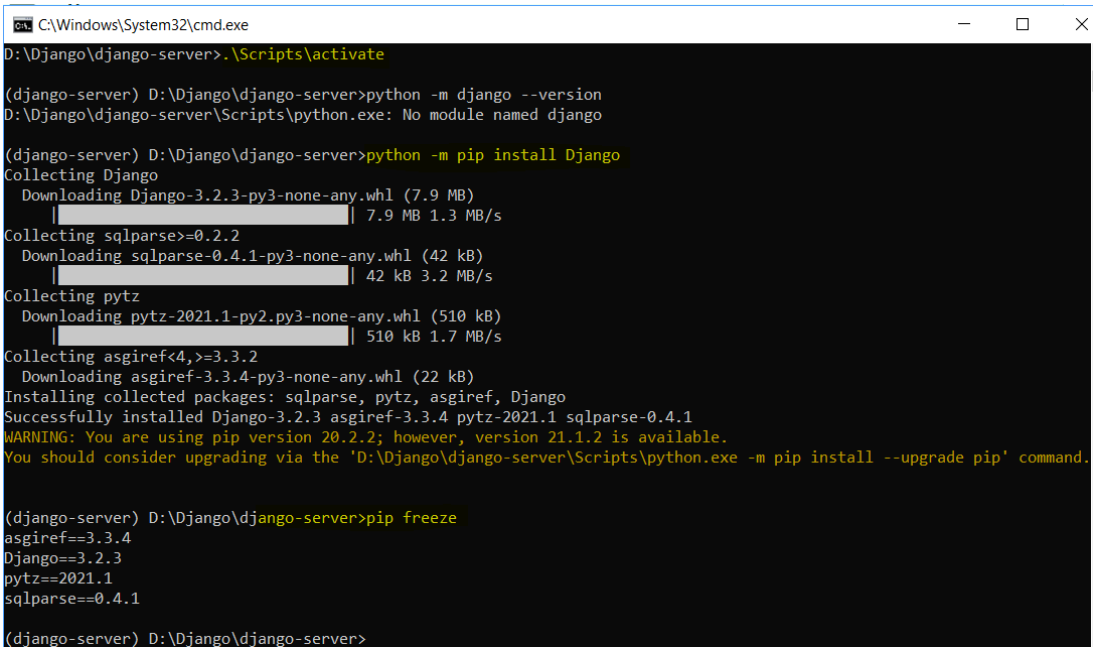
  .\Scripts\activate

- Tiếp theo cài đặt Django vào môi trường đã active.

  python -m pip install Django

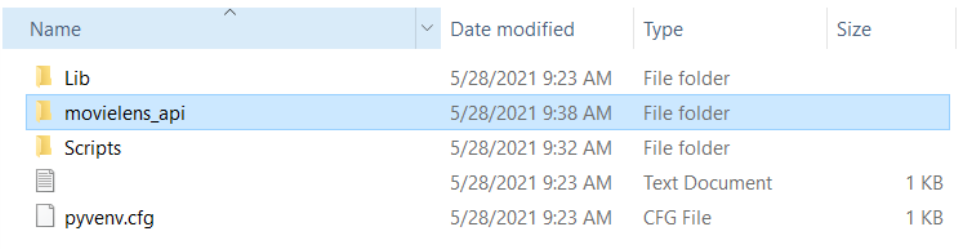
- Sau đó kiểm tra môi trường bằng lệnh

  pip freeze



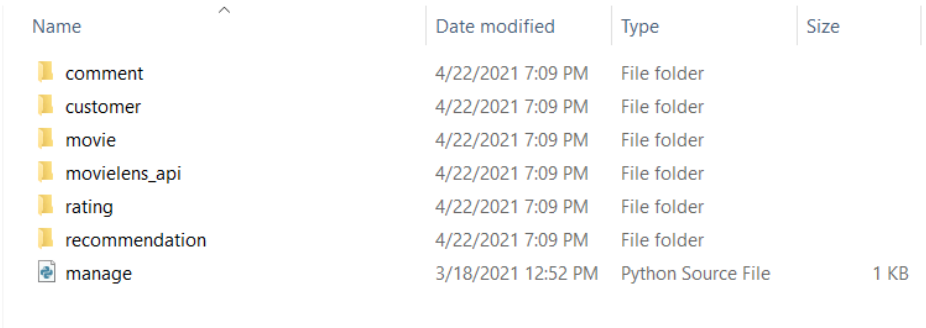
Hình 5.1: Active môi trường, cài đặt Django trong môi trường.

- Sau khi đã cài Django thành công trong môi trường thì có thể tạo mới một project bằng lệnh *django-admin startproject movielens_api.*



Hình 5. 2: Tạo mới một project movielens_api.

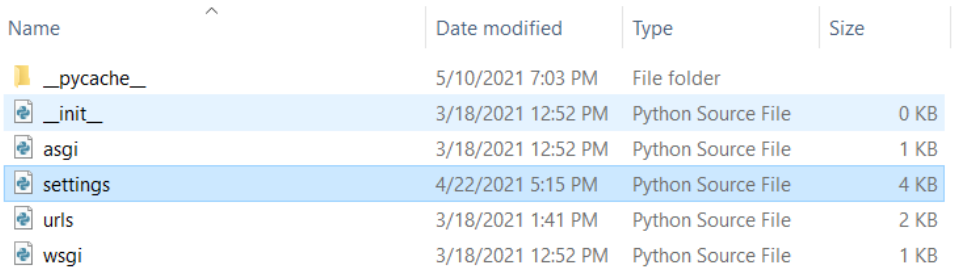
- Trong project movielens_api tôi sẽ chia ra các thư mục hay còn gọi là app tương đương với từng bảng trong database đã thiết kế.
- Tạo thư mục trong movielens_api bằng lệnh *django-admin startapp "app-name"*
- Tạo app movie '*django-admin startapp movie*'.
- Ngoài app movie chúng tôi cần tạo các app: comment, rating, customer, recommendation.



Hình 5.3: Tạo mới các app cho project movielens_api.

- File manage.py là dùng để quản lý các thay đổi của các models trong các app hoặc chạy server.

Cấu trúc thư mục, cấu hình trong thư mục movielens_api.



Name	Date modified	Type	Size
__pycache__	5/10/2021 7:03 PM	File folder	
__init__	3/18/2021 12:52 PM	Python Source File	0 KB
asgi	3/18/2021 12:52 PM	Python Source File	1 KB
settings	4/22/2021 5:15 PM	Python Source File	4 KB
urls	3/18/2021 1:41 PM	Python Source File	2 KB
wsgi	3/18/2021 12:52 PM	Python Source File	1 KB

Hình 5.4: Cấu trúc thư mục movielens_api.

- __init__.py: là một file rỗng chỉ định việc đường dẫn folder này sẽ được xem như là một Python package.
- settings.py: là file chứa các cài đặt, cấu hình cho project.
- urls.py: file khai báo các URL của project (kiểu như routing, với địa chỉ nào thì sẽ thực thi hàm nào).
- wsgi.py: là file dùng deploy project lên server online.

- Trong thư mục movielens_api chúng tôi cấu hình file settings.py và urls.py.

- Cấu hình trong file settings.py

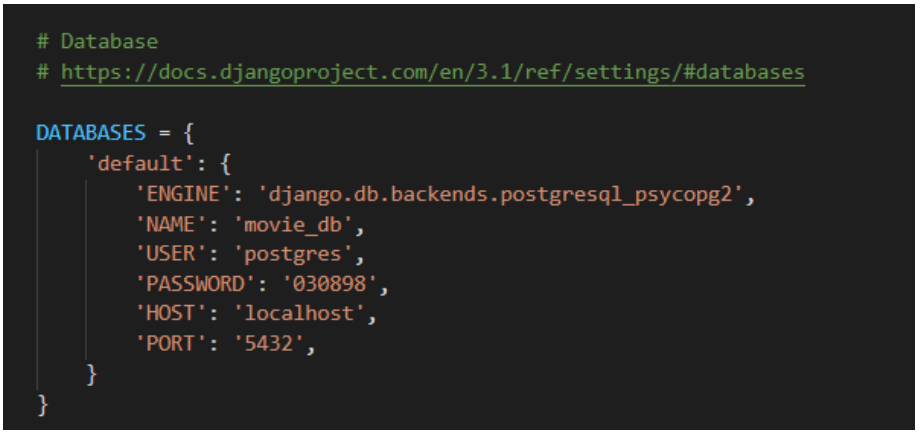
  Khai báo các app trong phần INSTALLED_APPS. Khai báo rest_framework, customer, comment, movie, rating, recommendation

```
INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'rest_framework',
    'customer',
    'comment',
    'movie',
    'rating',
    'recommendation',

]
```

Hình 5.5: Khai báo các app trong file settings.py.

Khai báo tên database và hệ quản trị cơ sở dữ liệu sử dụng để lưu trữ thông tin.



Hình 5.6: Khai báo database trong file settings.py.

- Tại ALLOWED_HOST tiến hành thêm các địa chỉ 10.0.2.2, localhost.

  ALLOWED_HOSTS = ['10.0.2.2','localhost', '127.0.0.1']

5.1.2. **Triển khai các web api.**

Tiếp theo chúng tôi sẽ xây dựng Web API cho từng app trong project movielens_api.

- App movie.

Trong app movie sẽ tiến hành xây dựng models Movies để thực hiện lưu dữ liệu của đối tượng phim.

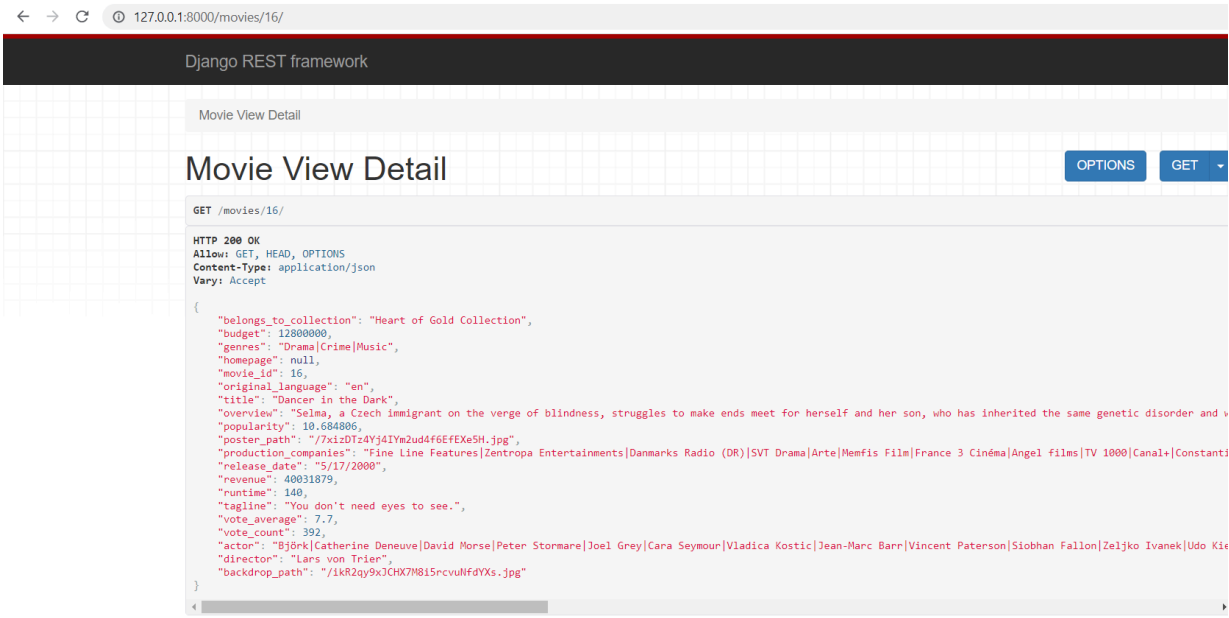
Đối tượng phim có các thuộc tính:

STT	Tên biến	Đặc tả của đối tượng
1	belongs_to_collection	TextField

2	budget	IntegerField
3	genres	TextField
4	homepage	TextField
5	movie_id	IntegerField
6	original_language	TextField
7	title	TextField
8	overview	TextField
9	popularity	FloatField
10	poster_path	TextField
11	production_companies	TextField
12	release_date	TextField
13	revenue	BigIntegerField
14	runtime	IntegerField
15	tagline	TextField
16	vote_average	FloatField
17	vote_count	IntegerField
18	actor	TextField
19	director	TextField
20	backdrop_path	TextField

Bảng 5.1: Mô tả đối tượng movie trong django server.

Web API truy xuất thông tin phim, chúng tôi đã viết một phương thức Get để lấy thông tin của bộ phim.



Hình 5.7: Lấy thông tin bộ phim.

Khởi động server, tại trình duyệt bất kỳ tiến hành truy cập http://127.0.0.1:8000/movies/< movie_id >/, với tham số tùy chọn là movie_id. Tại ví dụ trên tôi truyền một phương thức get với movie_id là 16.

- App customer.

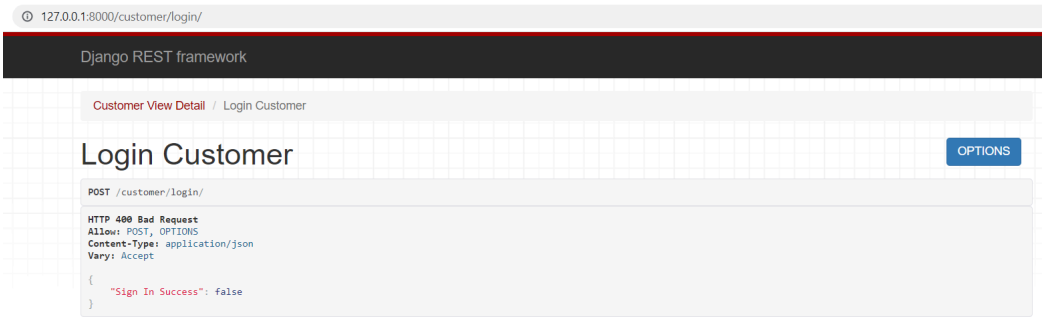
Định nghĩa đối tượng customer trong app customer và dùng để lưu trữ thông tin đối tượng khách trong database và truyền thông tin đó lên web api.

STT	Tên biến	Đặc tả của đối tượng
1	user_id	IntegerField
2	User_name	TextField
3	Password	TextField
4	Name	TextField
5	email	IntegerField
6	Adress	TextField
7	Birthday	TextField
8	Gender	TextField
9	Favorite_genres	TextField

Bảng 3.12: Bảng Mô tả đối tượng customer trong django server.

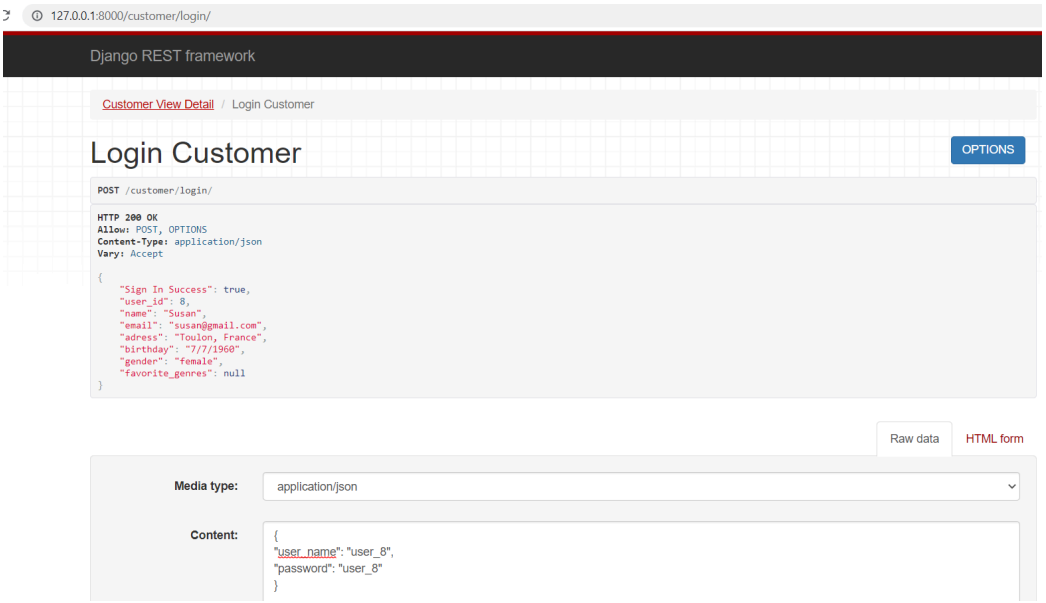
Web api của customer gồm có: đăng ký, đăng nhập, truy xuất thông tin khách hàng.

- Đăng nhập sử dụng phương thức POST với thông tin truyền vào là tên đăng nhập vào mật khẩu. Khi thông tin được truyền lên thì mật khẩu của người dùng sẽ được mã hóa dưới dạng md5 rồi sau đó sẽ kiểm tra thông tin tên người dùng mà mật khẩu sau khi mã hóa có trong database hay không và trả về kết quả.
    - Đăng nhập thất bại sẽ trả về thông tin là đăng nhập thất bại.



Hình 5.8: Thông báo đăng nhập thất bại.

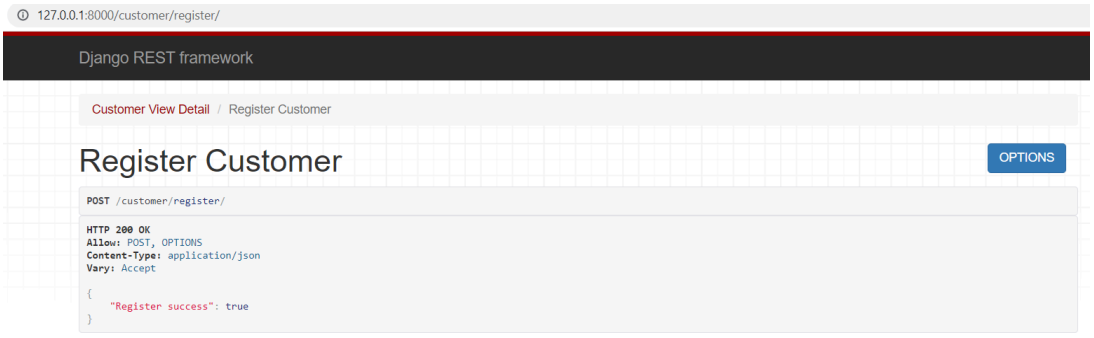
- Đăng nhập thành công sẽ trả về một dạng json là thông tin của người dùng và không bao gồm mật khẩu.



Hình 5.9: Thông báo đăng nhập thành công.

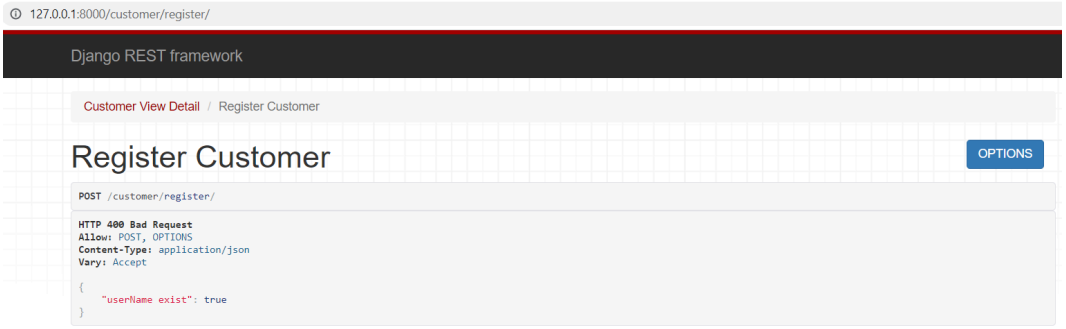
- Đăng ký người dùng mới vào trong hệ thống với các thông tin tên người dùng, tên đăng nhập, mật khẩu, địa chỉ email, địa chỉ nơi ở, ngày sinh, giới tính và thể loại phim yêu thích. Tương tự như đăng nhập thì mật khẩu người dùng khi đăng ký sẽ được mã hóa dạng md5 rồi mới được lưu trữ vào cơ sở dữ liệu.

- Đăng ký người dùng mới thành công sẽ trả về thông báo thành công.

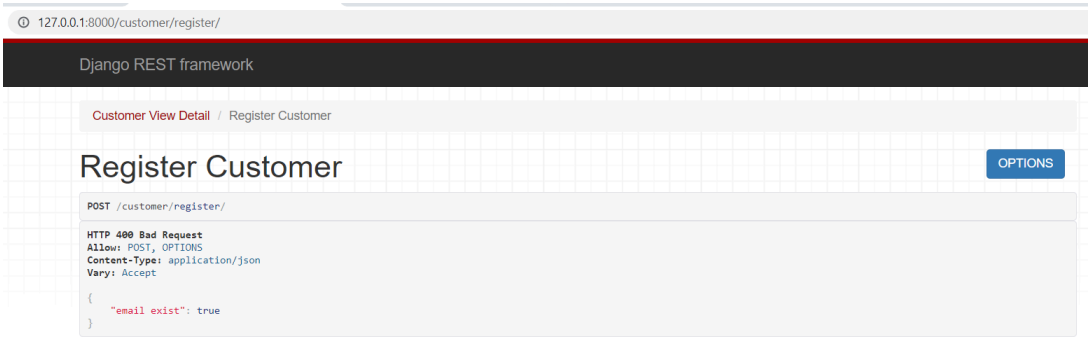


Hình 5.10: Đăng ký người dùng mới thành công.

- Đăng ký người dùng thất bại sẽ trả về kết quả thất bại.



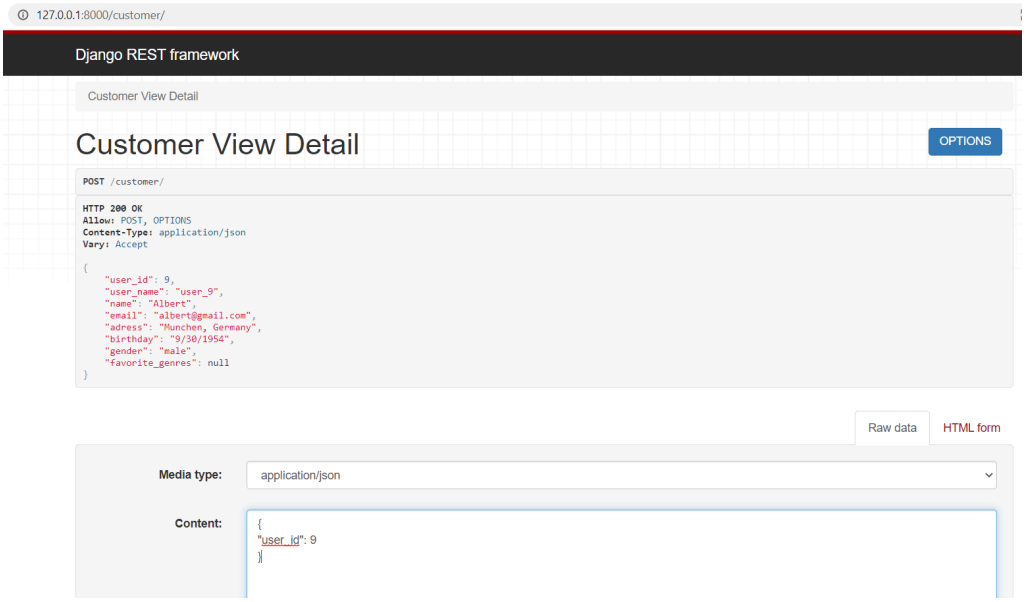
Hình 5.11: Đăng ký thất bại do trùng tên đăng nhập.



Hình 5.12: Đăng ký thất bại do trùng địa chỉ email.

- Truy xuất thông tin người dùng. Để truy vấn thông tin người dùng tôi sử dụng phương thức POST với địa chỉ http://127.0.0.1:8000/customer và sử dụng cú pháp json trong phương thức POST là:

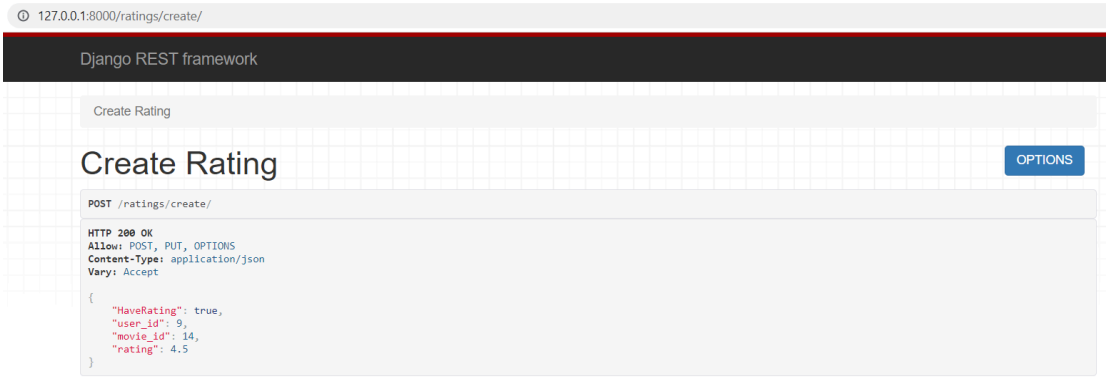
  { “user_id”: <user_id>}



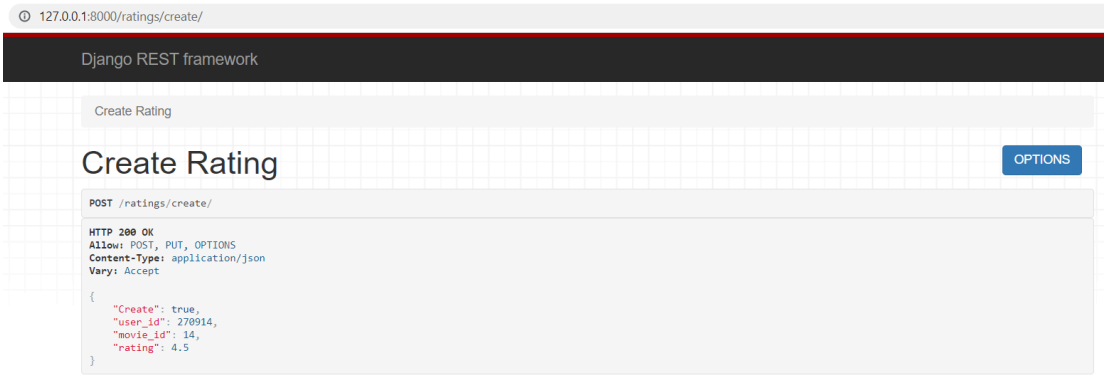
Hình 5.13: Truy vấn thông tin người dùng

- App rating.

Xây dựng phương thức POST cho hành động gửi đánh giá phim của một người dùng lên trên cơ sở dữ liệu. Nếu đánh giá phim đã có trong cơ sở dữ liệu thì sẽ tiến hành cập nhật lại đánh giá và lưu lại trong cơ sở dữ liệu.

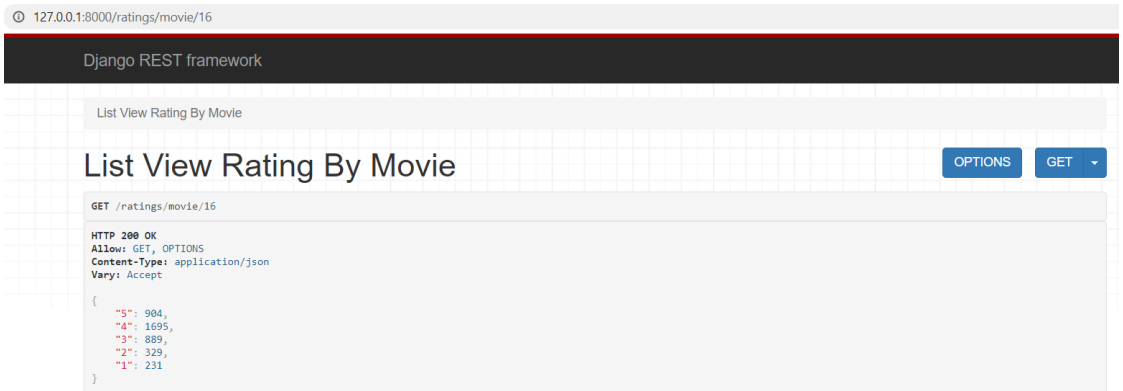


Hình 5.14: Cập nhật lại đánh giá phim.



Hình 5.15: Tạo mới một đánh giá phim.

Thống kê các đánh giá từ 1 đến 5 sao và được thống kê theo tổng số lượng.

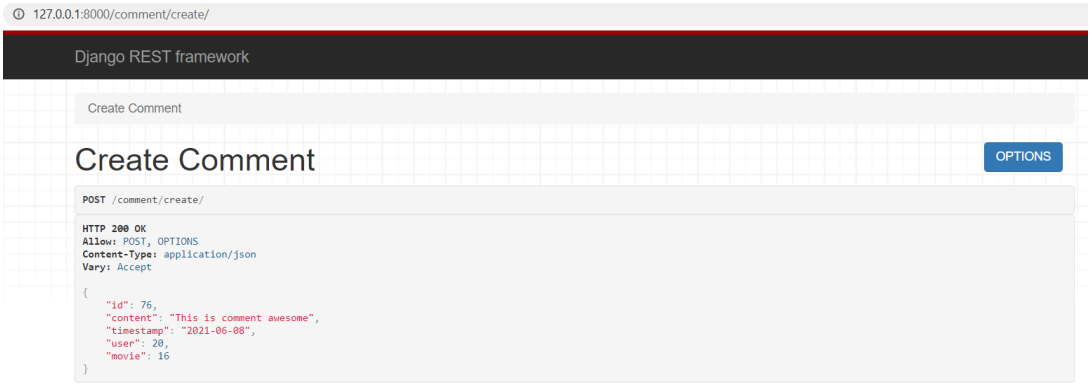


Hình 5.16: Thống kê các đánh giá của một bộ phim.

- App comment.

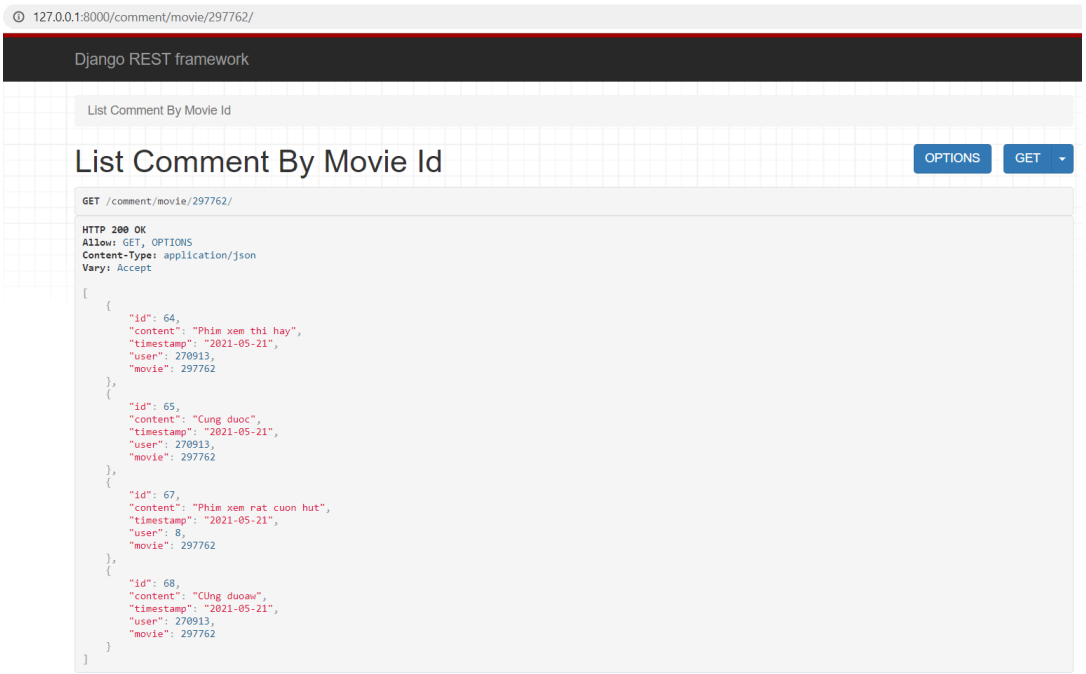
Tại app comment chúng tôi xây dựng Web API tạo bình luận, hiển thị bình luận thuộc về bộ phim.

- Tạo bình luận.



Hình 5.17: Thêm bình luận thành công

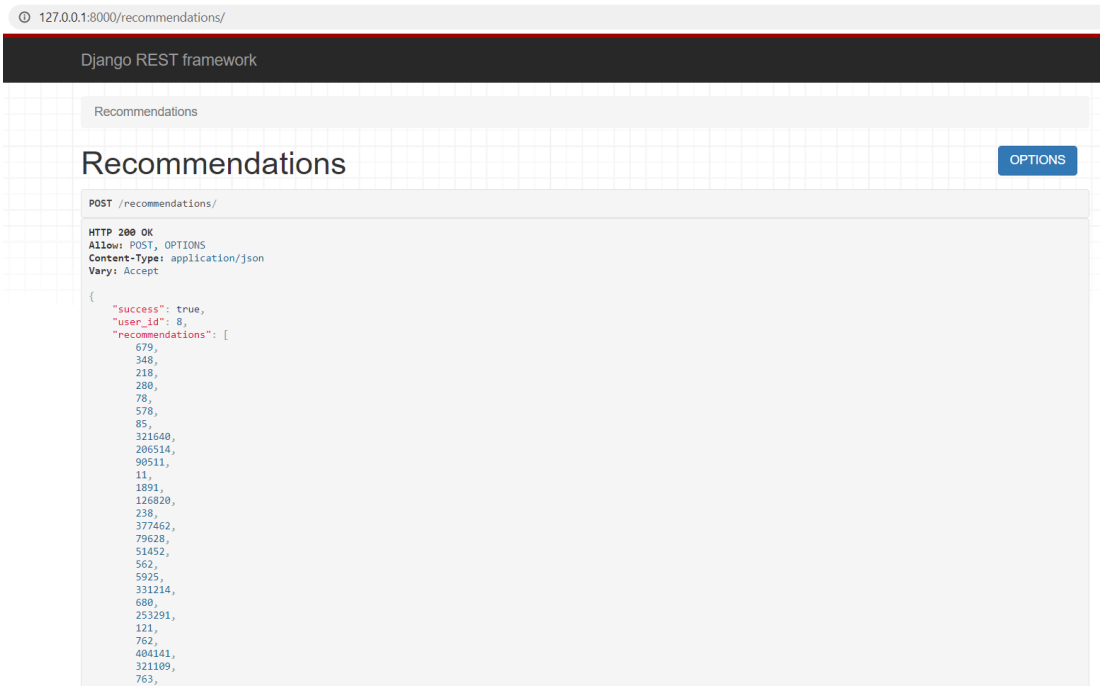
- Hiển thị danh sách bình luận của bộ phim.



Hình 5.18: Danh sách bình luận của bộ phim.

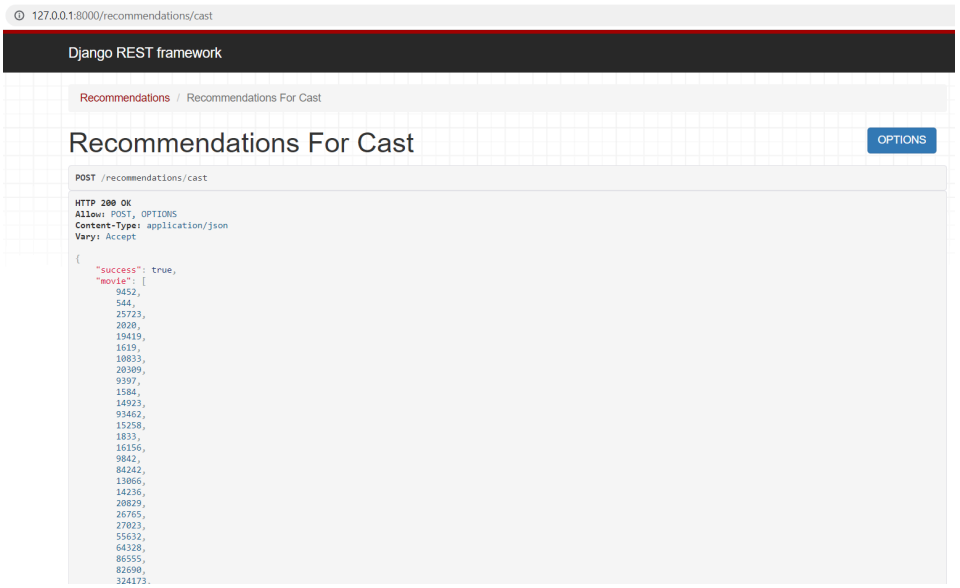
- App recommendation.

Xây dựng Web API trả về danh sách các gợi ý phim cho người dùng. Trong danh sách trả về ngoài các bộ phim gợi ý thì còn danh sách các bộ phim phổ biến, các bộ phim mới ra mắt.



Hình 5.19: Danh sách movie_id các phim gợi ý.

- Xây dựng api trả về danh sách phim của diễn viên. Truyền tham số là tên diễn viên kết quả trả về là danh sách movie_id của các bộ phim mà diễn viên được tìm kiếm có tham gia đóng phim.



Hình 5.20: Danh sách movie_id các bộ phim của một diễn viên.

5.2. Cài đặt và triển khai ứng dụng.

Trong phần này chúng tôi sẽ trình bày các bước cài đặt framework và kết quả xây dựng ứng dụng.

5.2.1. Cài đặt SDK

Trong phạm khóa luận này, chúng tôi sẽ cài đặt Flutter trên hệ điều hành Windows 10. Tải Flutter SDK.

- Tải xuống bản stable channel (Windows) phiên bản 2.0.3 tại trang Flutter SDK releases – Flutter.
- Giải nén tệp .zip vừa tải vào đường dẫn như "C:\src\"

Cài đặt Path

Để chạy các lệnh Flutter trong giao diện console Windows thông thường, hãy thực hiện các bước sau để thêm Flutter vào biến môi trường PATH:

- Nhấn nút Windows, gõ 'env' và chọn "Edit the system environment variables", sau đó chọn "Environment Variables".
- Trong phần System variables chọn "Path" và thêm đường dẫn tuyệt đối đến "C:\src\flutter\bin".
- Kiểm tra Path bằng cách mở Command Prompt và gõ "flutter –version".

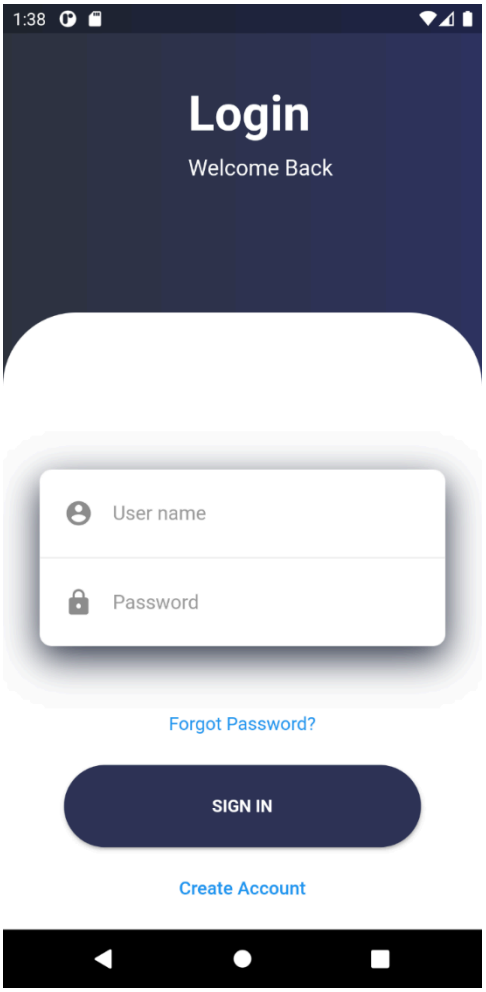
Cài đặt plugin trong IDE

Chúng tôi sử dụng IDE Visual Studio Code để thực hiện xây dựng ứng dụng. Sau khi cài đặt IDE, tiến hành thêm extension Flutter và Dart.

- Mở IDE chọn Extensions nhập Flutter và cài đặt.

5.2.2. Màn hình đăng nhập.

Trong màn hình đăng nhập thì người dùng phải điền thông tin đăng nhập là tên đăng nhập và mật khẩu. Nếu người dùng chưa có tài khoản có thể nhấn vào dòng chữ tạo tài khoản.



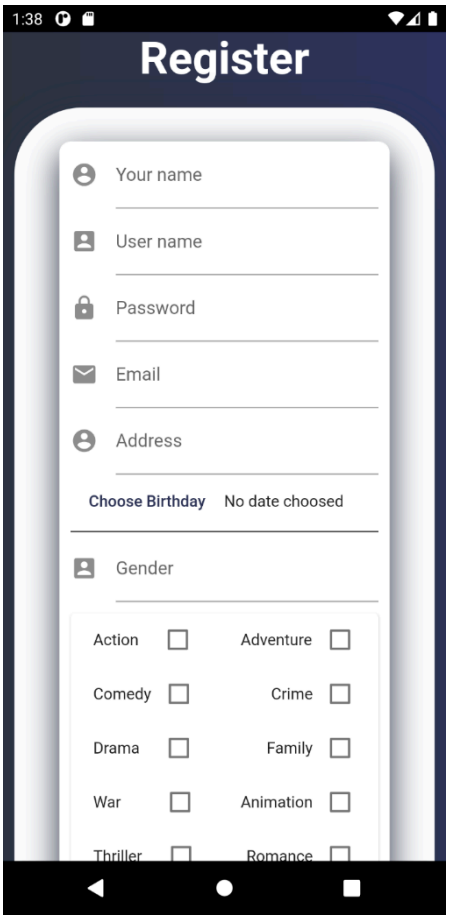
Hình 5.21: Màn hình đăng nhập.

5.2.3. Màn hình đăng ký người dùng mới.

Tại giao diện đăng ký người dùng mới cho phép người dùng chuyển về màn hình đăng nhập nếu như người dùng đã có tài khoản.

Ràng buộc khi đăng ký:

- Tên đăng nhập yêu cầu phải là duy nhất trong hệ thống, nếu có tồn tại tên đăng nhập thì sẽ trả về kết quả đăng ký thất bại.
- Mật khẩu của người dùng yêu cầu phải có độ dài lớn hơn ba ký tự.
- Thể loại phim yêu thích khi người dùng đăng ký có thể tích chọn một hoặc nhiều thể loại phim yêu thích hoặc là không tích chọn.



Hình 5.22: Màn hình đăng ký.

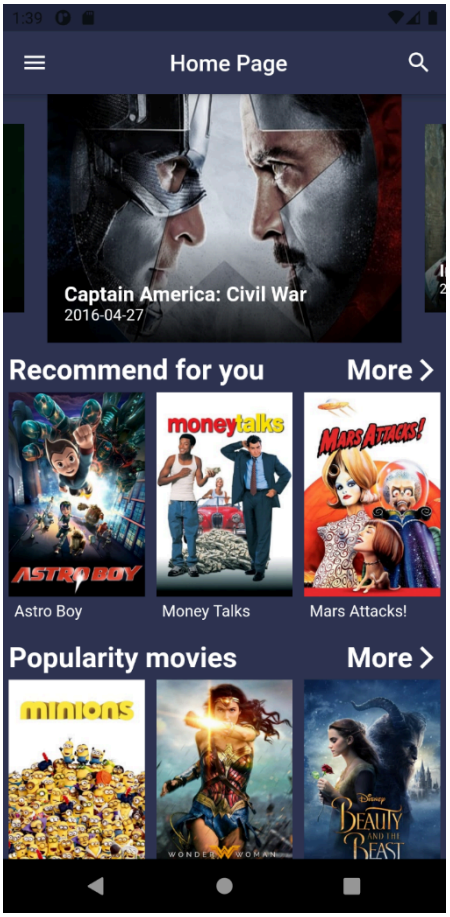
Giao diện đăng ký người dùng mới gồm các thông tin:

- Tên người dùng (Your name).
- Tên đăng nhập (User name).
- Mật khẩu đăng nhập (Password).
- Địa chỉ email (Email).

- Địa chỉ người dùng(Address).
- Ngày sinh (Birthday).
- Giới tính (Gender).
- Chọn các thể loại phim yêu thích.

5.2.4. Màn hình trang chủ.

Trong giao diện màn hình chính gồm các bộ phim được gợi ý, những bộ phim mới nhất và những bộ phim được đánh giá là điểm phổ biến cao.



Hình 5.23: Màn hình trang chủ ứng dụng.

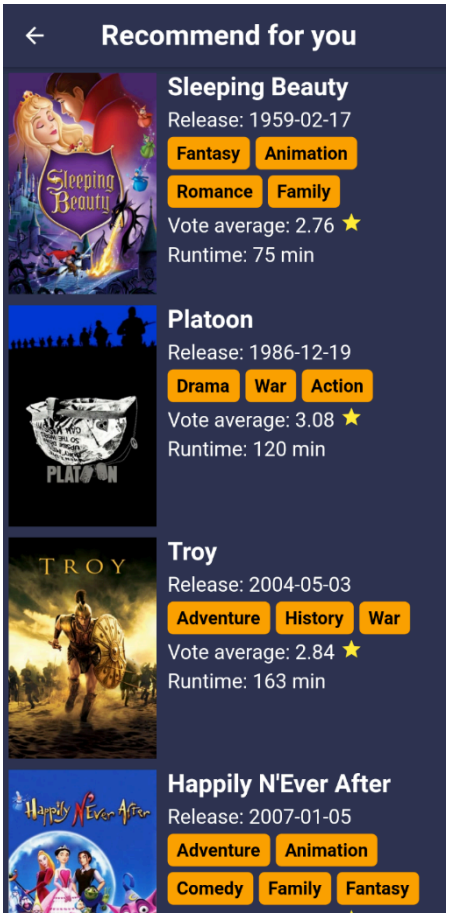
Thao tác trong màn hình:

- Phần trên góc trái khi nhấp vào sẽ là chuyển đổi giữa màn hình trang chủ và thông tin người dùng.

- Phần góc bên phải là có chức năng tìm kiếm phim.
- Khi người dùng nhấn vào “More” sẽ hiển thị thêm nhiều bộ phim thuộc danh sách như danh sách phim gợi ý, danh sách phim có đánh giá cao, danh sách những bộ phim mới ra mắt.

5.2.5. Màn hình xem thêm phim.

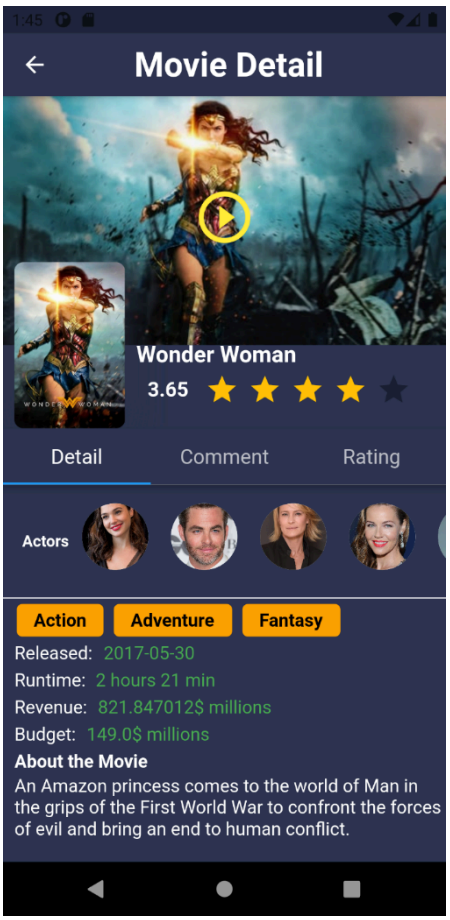
Giao diện xem thêm phim hiển thị danh sách các bộ phim theo danh sách ví dụ xem phêm phim gợi ý, xem thêm phim có độ đánh giá cao, xem thêm phim ra mắt mới nhất.



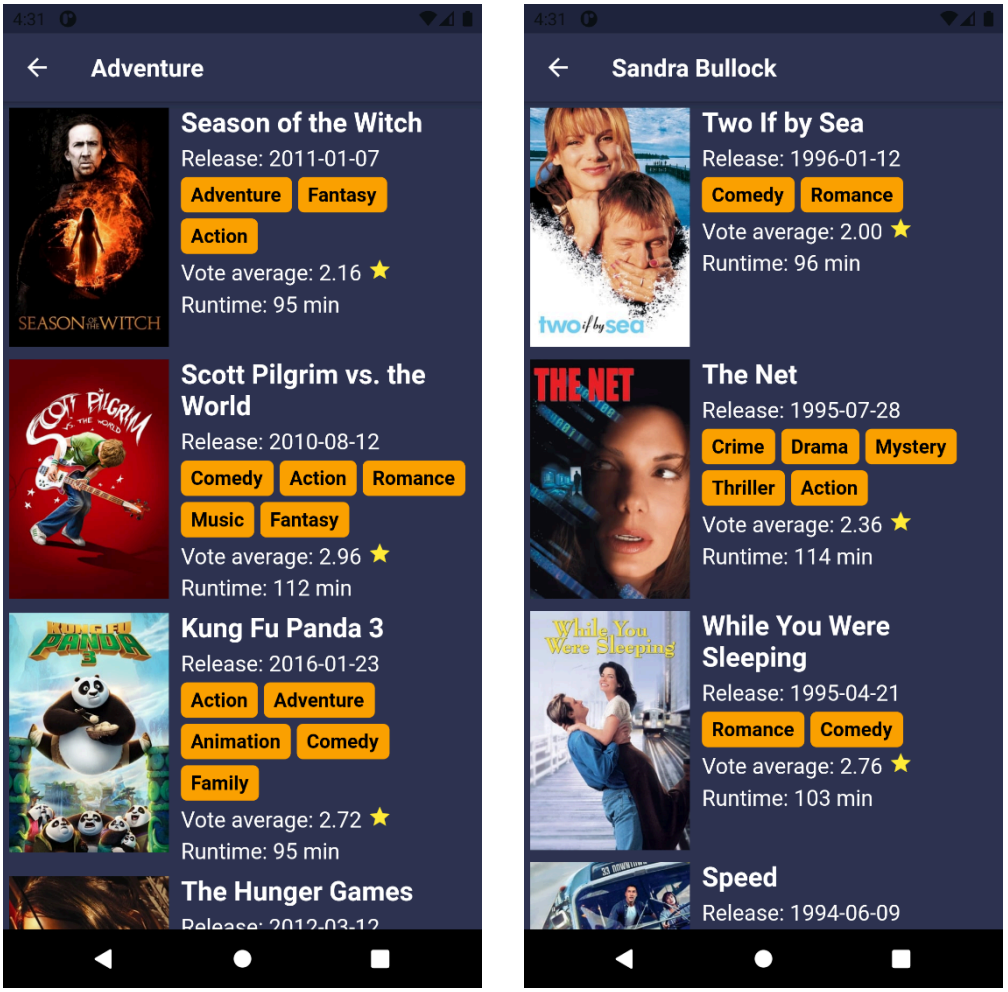
Hình 5.24: Màn hình xem thêm phim

5.2.6. Màn hình xem chi tiết bộ phim.

Tại màn hình chi tiết của bộ phim hiển thị các thông tin chi tiết như: Tên phim, số sao đánh giá, danh sách các diễn viên, ngày ra mắt, thời lượng phim, ngân sách làm phim, doanh thu phim và mô tả ngắn về phim.

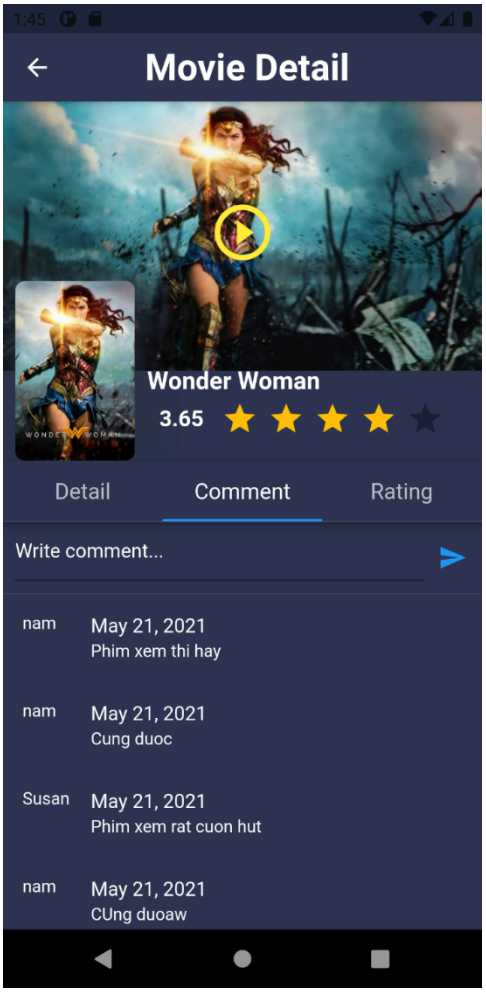


Hình 5.25: Màn hình chi tiết phim.



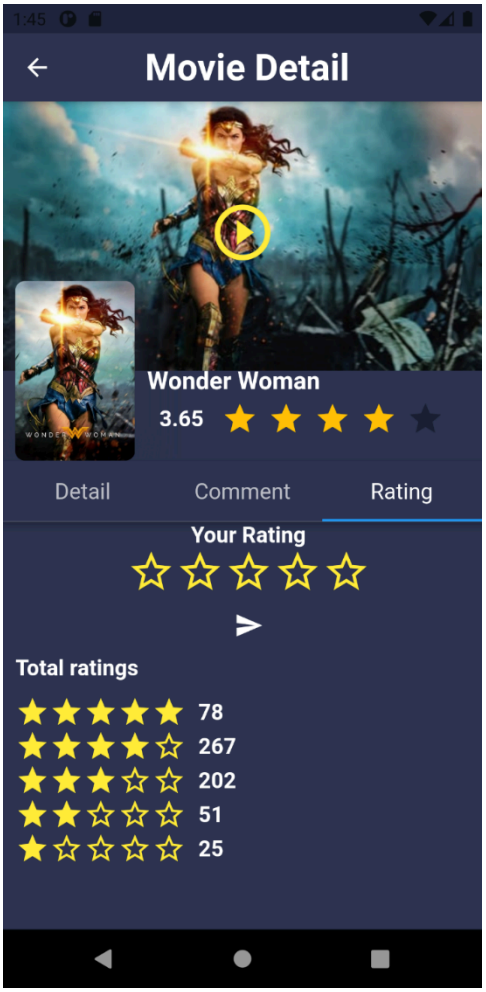
(a) thể loại phim (b) tên diễn viên

Hình 5.26: Màn hình danh sách phim theo lựa chọn.



Hình 5.27: Màn hình bình luận phim.

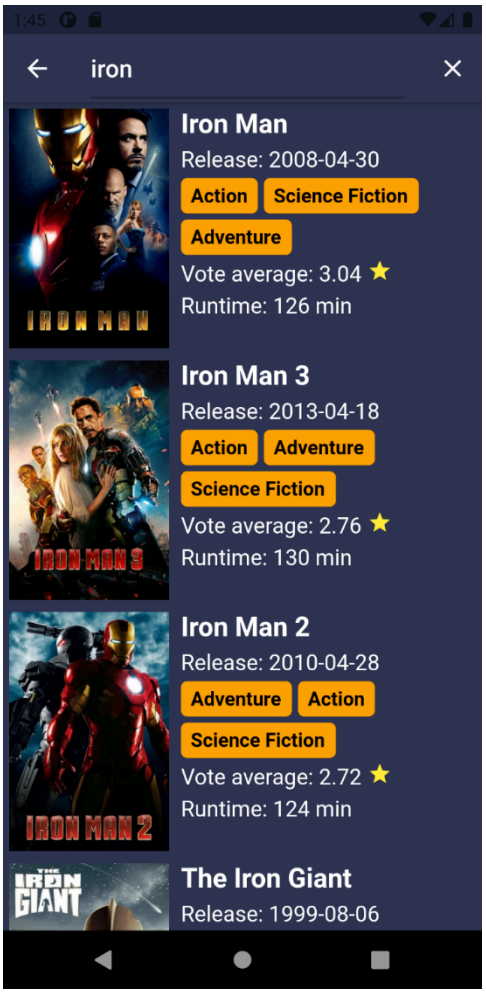
Giao diện đánh giá phim sẽ hiển thị tổng số lượng đánh giá theo số sao từ 1 đến 5 sao. Trong màn hình cho phép người dùng gửi số sao đánh giá hoặc sửa số sao đánh giá.



Hình 5.28: Màn hình đánh giá phim.

5.2.7. Màn hình tìm kiếm phim.

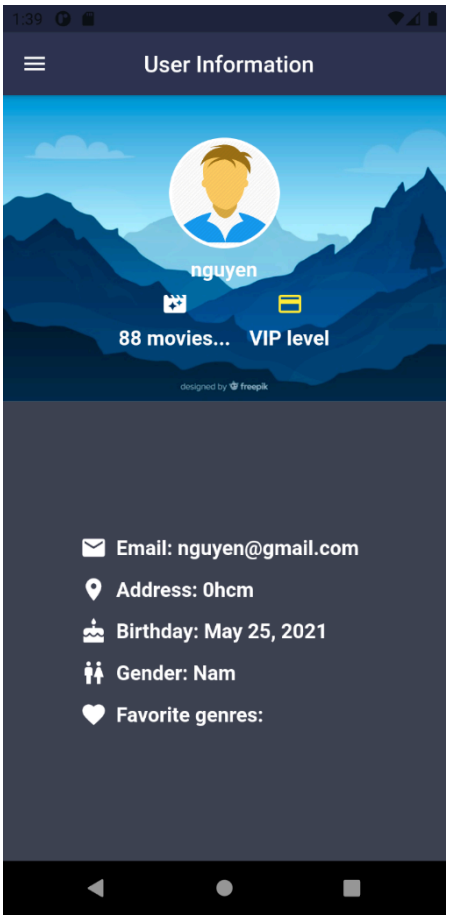
Màn hình tìm kiếm hiển thị các bộ phim bao gồm ký tự mà người dùng đã nhập vào để tìm kiếm. Từ khóa tìm kiếm bắt buộc phải lớn hơn 3 ký tự.



Hình 5.29: Màn hình tìm kiếm phim.

5.2.8. Màn hình thông tin người dùng.

Hiển thị các một số thông tin mà người dùng đã đăng ký như tên, địa chỉ email.



Hình 5.30: Màn hình thông tin người dùng.

Chương 6. ĐÁNH GIÁ VỚI KHÓA LUẬN KẾ THỪA

Tóm tắt chương: *Chương 6 sẽ trình bày những vấn đề được cải thiện và phát triển mới từ nền tảng khóa luận tốt nghiệp được kế thừa. Những phát triển về web api, giao diện ứng dụng.*

6.1. Phát triển phần Web API.

Web API chúng tôi đã xây dựng thêm các service để thực hiện các chức năng mới trong ứng dụng như: Đăng ký, bình luận, đánh giá.

STT	Nội dung	Khóa luận được kế thừa	Khóa luận hiện tại
1	Mã hóa mật khẩu người dùng	Trong bảng người dùng có thuộc tính password không được mã hóa	Mã hóa tất cả các mật khẩu của người dùng cũ trong bảng customer và khi người dùng mới đăng ký mật khẩu cũng được mã hóa một chiều dưới dạng MD5
2	Api cho chức năng bình luận	Chưa có	Đã xây dựng thành công api đăng ký thành viên mới thông qua phương thức Post

3	Api cho chức năng đánh giá phim	Chưa có	Xây dựng thành công api để thực hiện tạo bình luận cho phim
4	Api cho chức năng lọc phim theo thể loại	Chưa có	Với thể loại phim được chọn api sẽ trả về danh sách các bộ phim có chứa thể loại mà người dùng đã chọn và các bộ phim đã được đánh giá cao trong danh sách phim
5	Api cho chức năng lọc phim theo diễn viên	Chưa có	Với diễn viên được chọn api sẽ trả về danh sách các bộ phim mà có tên diễn viên trong danh sách các diễn viên của bộ phim và các bộ phim đã được đánh giá cao trong danh sách các bộ phim có trong cơ sở dữ liệu
6	Api gợi ý cho người dùng mới có thể loại phim yêu thích	Chưa có	Xây dựng thành công api khi người dùng mới đăng nhập thành công sẽ trả về danh sách các bộ phim gợi ý cho người dùng mới

7			

Bảng 6.1: So sánh web api.

6.2. Phát triển phần ứng dụng.

Ứng dụng có các chức năng đăng ký, bình luận, đánh giá số sao và một số những chi tiết khác trong phần giao diện so với khóa luận kế thừa. Chúng tôi so sánh trong bảng 6.2.

STT	Nội dung	Khóa luận được kế thừa	Khóa luận hiện tại
1	Màn hình đăng nhập	Đã có	Sử dụng lại màn hình đăng nhập
2	Màn hình đăng ký	Chưa có	Xây dựng thành công màn hình đăng ký. Trong màn hình đăng ký người dùng sẽ phải điền những thông tin bắt buộc và có những thông tin là tùy chọn có thể bỏ qua.
3	Màn hình trang chủ	Đã có	Sử dụng lại màn hình trang chủ Thêm Animation cho phần đầu trang
4	Màn hình tìm kiếm	Đã có	Thêm gợi ý phim mặc định khi vào màn hình tìm kiếm phim. Sửa lỗi hiển thị khi thông tin bị thiếu

5	Màn hình chi tiết người dùng	Đã có	Sử dụng lại Thay đổi một vài chi tiết nhỏ như hình nền, thêm thông tin thể loại phim yêu thích
6	Màn hình chi tiết phim	Đã có	Sử dụng lại một phần giao diện cũ. Phát triển thêm chức năng hiển thị chi tiết phim, tạo bình luận, tạo đánh giá phim trong giao diện chi tiết phim Trong màn hình chi tiết người dùng có thể nhấn chọn vào hình ảnh diễn viên hoặc thể loại phim để được gợi ý phim liên quan đến diễn viên hoặc thể loại phim mà người dùng đã chọn
7	Chức năng bình luận phim	Chưa có	Xây dựng thành công chức năng tạo bình luận phim cho người dùng
8	Chức năng lọc phim khi chọn diễn viên	Chưa có	Xây dựng thành công chức năng lọc phim theo

			diễn viên khi người dùng nhấn chọn diễn
9	Chức năng lọc phim khi chọn thể loại	Chưa có	Xây dựng thành công chức năng lọc phim theo thể loại khi người dùng nhấn vào thể loại phim

Bảng 6.2: So sánh ứng dụng.

Chương 7. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Tóm tắt chương: *Trong chương 7 chúng tôi sẽ trình bày những kết quả đã đạt được và những hướng mở rộng và phát triển của khóa luận.*

7.1. Kết luận

Trong tình hình dịch bệnh hiện nay nhu cầu giải trí lại càng rất lớn, xem phim giải trí là lựa chọn mà rất nhiều người thích. Xem phim thường là người dùng xem những phim được người quen giới thiệu, những bộ phim được đánh giá cao trong bảng xếp hạng hoặc những bộ phim có thể loại yêu thích của người dùng. Việc đưa ra gợi ý phim và được người dùng chọn xem đem đến lợi ích rất lớn. Trong đề tài này của chúng tôi áp dụng mô hình gợi ý phim vào trong ứng dụng.

Trải qua quá trình xây dựng ứng dụng trên thiết bị di động chúng tôi đạt được một số kết quả như sau:

- Xây dựng thành công Web API thực hiện công việc xử lý dữ liệu được gửi từ người dùng và trả lại kết quả Json.
- Chúng tôi xây dựng thành công ứng dụng gợi ý phim trên thiết bị Android, ứng dụng chạy trên thiết bị ảo và thiết bị thật hoạt động ổn định.

Kết quả đã đạt được mục tiêu đề ra nhưng vẫn chưa được hoàn thiện và còn nhiều thiếu sót. Từ những kết quả đã đạt được chúng tôi sẽ có tiền đề để phát triển thêm những chức năng, hiệu suất, bảo mật ứng dụng trong thời gian tới.

7.2. Hướng phát triển.

Trong tương lại, chúng tôi mong muốn có thể hoàn thiện tốt hơn đề tài này về một số điểm sau:

- Chạy ứng dụng trên nền tảng ios và kiểm tra hiệu năng.
- Phát triển trên nền tảng Desktop, Website.
- Triển khai trên CH Play và Apple Store để đánh giá độ hài lòng của người dùng cho ứng dụng.
- Phần đăng nhập cho phép người dùng sử dụng tài khoản bên thứ ba để đăng nhập như tài khoàn google, facebook, v.v…
- Xây dựng thêm phần loại bỏ phim đã xem trong phần gợi ý phim.
- Tìm hiểu và kiểm tra độ bảo mật ứng dụng và bảo mật web api.

TÀI LIỆU THAM KHẢO

Tài liệu khóa luận

[1] Q.P. Quốc và N.T. Đạt, "Xây dựng hệ thống khuyến nghị cho người dùng xem phim trên điện thoại di động", Cử nhân, Trường Đại học Công nghệ Thông tin, TP. Hồ Chí Minh, 2020.

Tài liệu online

[2] "Django Overview" https://www.djangoproject.com/start/overview (30/07/2021)

[3] "How to install Django" https://docs.djangoproject.com/en/3.0/topics/install/#installingdevelopment-version (30/07/2021)

[4] "Home – Django Rest Framework" https://www.django-rest-framework.org (30/07/2021)

[5] "Flutter Documentation" https://flutter.dev/. (30/07/2021)

[6] "Install Flutter." https://flutter.dev/docs/get-started/install. (30/07/2021)

[7] "API overview – The Movie Database" https://www.themoviedb.org/documentation/api (30/07/2020).