

**BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA KHOA HỌC VÀ KỸ THUẬT THÔNG TIN**



**ĐỒ ÁN MÔN HỌC
KỸ THUẬT LẬP TRÌNH PYTHON**

**ỨNG DỤNG DJANGO ĐỂ PHÁT TRIỂN WEB BÌNH
LUẬN PHIM**

Giảng viên hướng dẫn : ThS. Nguyễn Thanh Sơn

Sinh viên thực hiện : Nguyễn Trí Vũ

Mã số sinh viên : 20521042

Lớp : IE221.N11.VB2

Tp HCM, tháng 01 năm 2023

MỤC LỤC

MỤC LỤC.....	1
Chương 1: GIỚI THIỆU ĐỒ ÁN.....	5
1.1. Tên Website:.....	5
1.2. Mô tả Website:	5
1.3. Lý do chọn thiết kế Website:.....	5
1.1. Sơ đồ sitemap của Website:.....	5
1.4. Công nghệ sử dụng:.....	6
1.5. Công cụ hỗ trợ:	6
Chương 2: TÓM TẮT QUÁ TRÌNH THỰC HIỆN.....	7
2.1. Xây dựng model	7
2.2. Thiết kế, lập trình, kết nối view và template.	8
2.3. Tiến hành thiết lập settings và deploy trang web	13
Chương 3: KẾT QUẢ ĐẠT ĐƯỢC.....	14
3.1. Tạo cơ sở dữ liệu PostgresDB	14
3.2. Ứng dụng Web Bình Luận Phim.....	14
3.2.1. Trang chủ.....	14
3.2.2. Duyệt tìm phim.....	15
3.2.3. Trang Chi tiết phim	17
3.2.4. Trang Sửa đánh giá phim	19
3.2.5. Trang Đăng nhập	20
3.2.6. Trang Đăng ký.....	20
3.2.7. Trang Khởi động lại mật khẩu	21
3.2.8. Trang Xác nhận gửi mail khởi động mật khẩu.....	22
3.2.9. Trang Nhập mật khẩu mới.....	24
3.2.10. Trang Hoàn thành khởi động lại mật khẩu.....	24

3.2.11.	Trang Tài khoản cá nhân	25
3.2.12.	Trang Tạo danh sách phim mới.....	27
3.2.13.	Trang Chi tiết Danh sách phim	27
3.2.14.	Trang Sửa thông tin danh sách phim.....	29
3.2.15.	Trang Đổi mật khẩu.....	29
3.2.16.	Trang Admin	30
TÀI LIỆU THAM KHẢO.....		32
PHỤ LỤC 1: GIỚI THIỆU (DEMO) KẾT QUẢ		33
PHỤ LỤC 2: DOCSTRING.....		44

MỤC LỤC BẢNG

Bảng 2.1: Chức năng của các trang trong website Bình luận phim.....	8
Bảng 2.2: URL của các trang con trong website Bình Luận Phim.....	11
Bảng 2.3: URL và phương thức để thực hiện một số tác vụ khác trong website	11
Bảng 2.4: Các thành phần thiết lập lên các trang con trong website	12

MỤC LỤC HÌNH ẢNH

Hình 1.1: Sitemap của website.....	6
Hình 2.1: Mô hình UML cho cơ sở dữ liệu BinhLuanPhim.....	7
Hình 3.1: Database BinhLuanPhim hiển thị trên pgAdmin 4.....	14
Hình 3.2: Giao diện trang chủ.....	15
Hình 3.3: Giao diện trang Duyệt tìm phim mặc định	16
Hình 3.4: Giao diện trang Duyệt tìm phim sau khi lọc kết quả	16
Hình 3.5: Giao diện trang Chi tiết phim – phần Thông tin phim.....	17
Hình 3.6: Giao diện trang Chi tiết phim – phần Đánh giá	18
Hình 3.7: Giao diện trang Chi tiết phim – Xác nhận xóa đánh giá.....	18
Hình 3.8: Giao diện trang Sửa đánh giá phim.....	19
Hình 3.9: Giao diện trang Đăng nhập	20
Hình 3.10: Giao diện trang Đăng ký	21
Hình 3.11: Giao diện trang Khởi động lại mật khẩu.....	22
Hình 3.12: Giao diện trang Xác nhận đã gửi mail khởi động mật khẩu	23
Hình 3.13: Email người dùng nhận được khi xác nhận lại mật khẩu	23
Hình 3.14: Giao diện trang Nhập lại mật khẩu mới.....	24
Hình 3.15: Giao diện trang Hoàn thành khởi động lại mật khẩu	25
Hình 3.16: Giao diện trang Tài khoản cá nhân – thông tin người dùng và bộ sưu tập phim... 25	
Hình 3.17: Giao diện trang Tài khoản cá nhân – Lịch sử đánh giá	26
Hình 3.18: Giao diện trang Tạo danh sách phim mới.....	27
Hình 3.19: Giao diện trang Chi tiết Danh sách phim – phần thông tin Danh sách phim	28
Hình 3.20: Giao diện trang Chi tiết Danh sách phim – phần điều chỉnh danh sách phim.....	28
Hình 3.21: Giao diện trang Sửa thông tin danh sách phim	29

Hình 3.22: Giao diện trang Đổi mật khẩu	30
Hình 3.23: Giao diện trang Admin.....	31

Chương 1:

GIỚI THIỆU ĐỒ ÁN

1.1. Tên Website:

WEBSITE BÌNH LUẬN PHIM

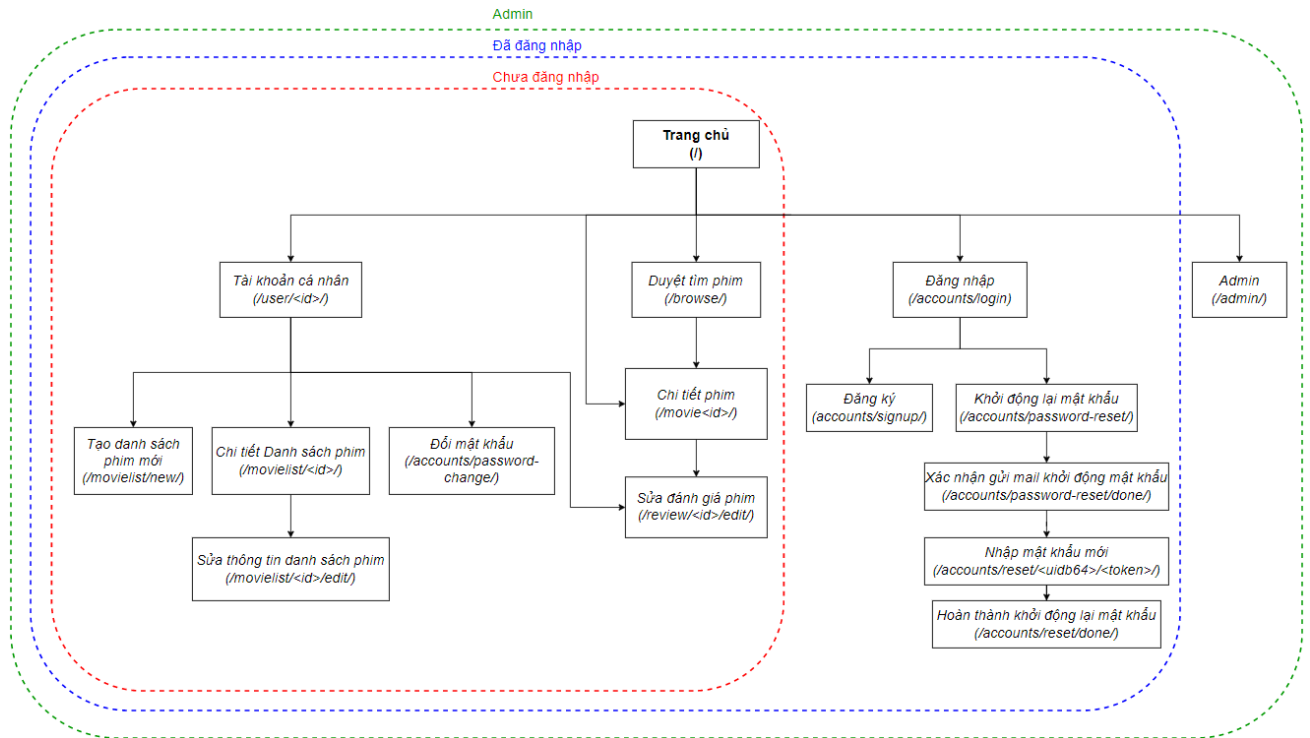
1.2. Mô tả Website:

- Tạo ra ứng dụng Web Bình Luận Phim để người dùng có thể đăng ký tài khoản, duyệt tìm thông tin phim, viết bài đánh giá cho các phim, tạo danh sách phim cá nhân.
- Website có các chức năng:
 - + Duyệt tìm thông tin, lọc các phim theo tiêu đề, thể loại, năm phát hành, thời lượng
 - + Tạo tài khoản, đăng ký, đăng nhập, đăng xuất, khởi động lại mật khẩu, đổi mật khẩu
 - + Viết các bài đánh giá phim và cho điểm. Cho phép chỉnh sửa, xóa bài đánh giá
 - + Lưu lại / bỏ lưu các bộ phim yêu thích
 - + Tạo danh sách các bộ phim. Chỉnh sửa tên và mô tả danh sách. Tìm, thêm và xóa phim khỏi các danh sách đó

1.3. Lý do chọn thiết kế Website:

- Mong muốn tạo ra một trang web phục vụ cho việc ứng dụng những nội dung đã học vào xây dựng một trang web thực tế.
- Tạo ra một trang web nơi người dùng có thể lưu lại và chia sẻ những trải nghiệm, suy nghĩ của mình sau khi thưởng thức các bộ phim, sưu tầm và lưu trữ những bộ phim mà mình yêu thích, lập ra các danh sách phim phục vụ các chủ đề theo mục đích cá nhân.

1.1. Sơ đồ sitemap của Website:



Hình 1.1: Sitemap của website

1.4. Công nghệ sử dụng:

- Web Framework Django, URL: <https://www.djangoproject.com/>
- Lưu trữ cơ sở dữ liệu trên PostgreSQL, URL: <https://www.postgresql.org/>
- Sử dụng library Bootstrap 5, URL: <https://getbootstrap.com/>
- Deploy cơ sở dữ liệu và dịch vụ web trên render, URL: <https://render.com/>

1.5. Công cụ hỗ trợ:

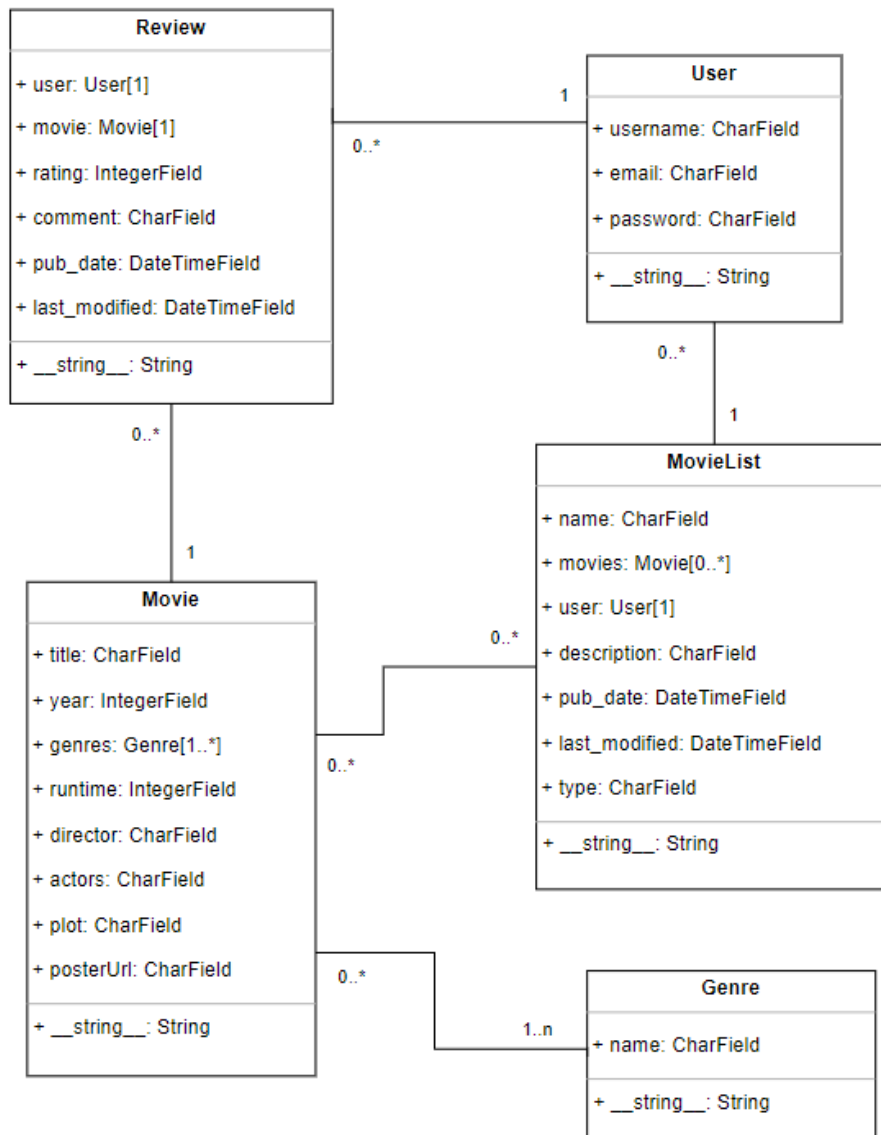
- IDE: PyCharm
- Công cụ quản lý PostgreSQL: pgAdmin 4
- Vẽ các sơ đồ: draw.io

Chương 2:

TÓM TẮT QUÁ TRÌNH THỰC HIỆN

2.1. Xây dựng model

- Các Model được sử dụng gồm:
 - + Movie: chi tiết thông tin phim
 - + Genre: thể loại phim
 - + Review: bài đánh giá
 - + MovieList: danh sách phim
 - + User (từ django.contrib.auth.models): người dùng



Hình 2.1: Mô hình UML cho cơ sở dữ liệu BinhLuanPhim

- Khởi tạo project Django. Thiết lập cơ sở dữ liệu là PostgreSQL trong file settings.py. Thiết lập các model trong file model.py, tạo các migration và tiến hành migrate lên cơ sở dữ liệu

2.2. Thiết kế, lập trình, kết nối view và template.

- Dựa trên chức năng, xác định các trang web cần tạo

Bảng 2.1: Chức năng của các trang trong website Bình luận phim

Trang	Chức năng
Trang chủ	Trang đầu tiên mà người dùng truy cập Liệt kê các phim mới ra mắt Cho phép truy cập vào trang Đăng nhập, trang Duyệt tìm phim
Admin	Cho phép Admin và những người liên quan có thể truy cập và tiến hành xem, thêm, xóa, sửa các đối tượng Movie, Genre, Review, MovieList, User. Tiến hành phân quyền
Duyệt tìm phim	Cho phép tìm và lọc các phim theo các thuộc tính: genre, year, runtime, title. Hiển thị kết quả tìm kiếm / lọc phim theo phân trang (pagination). Mặc định sẽ hiển thị tất cả các phim. Cho phép truy cập vào trang Chi tiết phim
Chi tiết phim	Hiển thị thông tin của bộ phim liên quan Hiển thị các bài đánh giá (Review) của tất cả người dùng Cho phép thêm phim vào danh sách yêu thích (khi đã đăng nhập) Cho phép thêm phim vào danh sách phim (khi đã đăng nhập) Cho phép đăng bình luận phim (yêu cầu đăng nhập) Cho phép truy cập trang chỉnh sửa đánh giá (đối với đánh giá do người dùng đó viết) Cho phép xóa đánh giá (đối với đánh giá do người dùng đó viết). Hiển thị modal xác nhận đồng ý xóa. Sau khi thực hiện xóa, website sẽ tải lại trang kèm thông báo
Sửa đánh giá phim	Chứa form cho phép chỉnh sửa các trường rating và comment Chỉnh sửa thành công sẽ quay về trang Chi tiết phim

	Chỉnh sửa không thành công sẽ tải lại trang và thông báo lỗi
Đăng nhập	<p>Hiển thị form đăng nhập, nhập các trường username, password, ghi nhớ đăng nhập</p> <p>Cho phép truy cập trang Đăng ký để đăng ký tài khoản mới</p> <p>Cho phép truy cập trang Khởi động lại mật khẩu trong trường hợp quên mật khẩu và muốn khởi động lại</p>
Đăng ký	<p>Hiển thị form đăng ký tài khoản mới, nhập thông tin các trường username, email, password. Nhập lại mật khẩu để xác nhận mật khẩu</p> <p>Người dùng đăng ký thành công sẽ đi trên trang chủ</p> <p>Đăng ký thất bại sẽ tải lại trang Đăng ký và thông báo lỗi</p>
Khởi động lại mật khẩu	<p>Hiển thị form nhập email để khởi động lại mật khẩu.</p> <p>Người dùng nhập email và nhấn nút Khởi động lại mật khẩu. Nếu tìm thấy User có email tương ứng, website sẽ được chuyển qua trang Xác nhận đã gửi mail khởi động mật khẩu</p> <p>Nếu không tìm thấy người dùng có Email hay form không hợp lệ sẽ tải lại trang và thông báo lỗi</p>
Xác nhận gửi mail khởi động mật khẩu	Hiển thị thông tin đã gửi mail chứa đường link khởi động lại mật khẩu
Nhập mật khẩu mới	<p>Hiển thị form nhập mật khẩu mới, nhập trường password. Điền mục Mật khẩu mới và Nhập lại mật khẩu mới</p> <p>Người dùng gửi form thành công sẽ được thông báo và website sẽ chuyển qua trang Hoàn thành khởi động lại mật khẩu</p> <p>Nếu form không hợp lệ sẽ tải lại trang và thông báo lỗi</p>
Hoàn thành khởi động lại mật khẩu	<p>Hiển thị thông tin đã Hoàn thành khởi động lại mật khẩu</p> <p>Cho phép truy cập vào trang Đăng nhập</p>
Tài khoản cá nhân	<p>Hiển thị thông tin email và ngày đăng ký, số lượng bài đánh giá, số lượng danh sách phim người dùng đã tạo</p> <p>Cho phép thực hiện đăng xuất (đối với trang cá nhân của người dùng đó)</p>

	<p>Cho phép đi đến trang Đổi mật khẩu (đối với trang cá nhân của người dùng đó)</p> <p>Hiển thị các Danh sách phim của người dùng đó</p> <p>Cho phép đi đến trang Chi tiết danh sách phim cho từng Danh sách phim</p> <p>Cho phép đi đến trang tạo Danh sách phim mới (đối với trang cá nhân của người dùng đó)</p> <p>Hiển thị các đánh giá mà người dùng đó đã đăng</p> <p>Cho phép đi đến trang chỉnh sửa đánh giá (đối với trang cá nhân của người dùng đó)</p> <p>Cho phép xóa bài đánh giá (đối với trang cá nhân của người dùng đó)</p>
Tạo danh sách phim mới	<p>Hiển thị form tạo Danh sách phim mới, nhập các trường: name, description</p> <p>Khi form được gửi thành công, website sẽ chuyển qua trang Chi tiết Danh sách phim cho Danh sách vừa được tạo kèm thông báo đã tạo thành công</p> <p>Nếu form không hợp lệ sẽ tải lại trang và thông báo lỗi</p>
Chi tiết Danh sách phim	<p>Hiển thị thông tin của Danh sách phim, gồm tên, số lượng phim, mô tả</p> <p>Cho phép đi đến trang chỉnh sửa Danh sách phim</p> <p>Cho phép xóa Danh sách phim. Hiển thị modal xác nhận xóa trước khi thực hiện. Sau khi xóa thành công website sẽ chuyển qua trang Tài khoản cá nhân</p> <p>Hiển thị các phim trong Danh sách phim</p> <p>Cho phép tìm kiếm, thêm, xóa phim</p>
Sửa thông tin danh sách phim	<p>Hiển thị form Sửa thông tin danh sách phim, chỉnh sửa các trường: name, description</p> <p>Khi form được gửi thành công, website sẽ chuyển qua trang Chi tiết Danh sách phim và thông báo đã cập nhật thành công</p> <p>Nếu form không hợp lệ sẽ tải lại trang và thông báo lỗi</p>
Đổi mật khẩu	<p>Hiển thị form Đổi mật khẩu, chỉnh sửa trường password. Yêu cầu nhập mật khẩu cũ, mật khẩu mới, nhập lại mật khẩu mới</p>

	Khi form được gửi thành công, website sẽ chuyển qua trang Tài khoản cá nhân và thông báo đã cập nhật thành công Nếu form không hợp lệ sẽ tải lại trang và thông báo lỗi
--	--

- Tạo các template chung:
 - + header_footer.html: Header, Thanh điều hướng, Footer
 - + message.html: hiển thị thông báo cho người dùng
 - + modal_delete_confirm: hiển thị modal delete (xác nhận trước khi xóa)
- Tạo url, form, template, view cho từng trang.

Bảng 2.2: URL của các trang con trong website Bình Luận Phim

Trang	Url
Trang chủ	/
Admin	/admin/
Duyệt tìm phim	/browse/
Chi tiết phim	/movie/<movie_id>/
Sửa đánh giá phim	/review/<review_id>/edit/
Đăng nhập	/accounts/login/
Đăng ký	/accounts/signup/
Khởi động lại mật khẩu	/accounts/password-reset/
Xác nhận gửi mail khởi động mật khẩu	/accounts/password-reset/done/
Nhập mật khẩu mới	/accounts/reset/<uidb64>/<token>
Hoàn thành khởi động lại mật khẩu	/accounts/reset/done/
Tài khoản cá nhân	/user/<user_username>/
Tạo danh sách phim mới	/movielist/new/
Chi tiết Danh sách phim	movielist/<movielist_id>/
Sửa thông tin danh sách phim	/movielist/<movielist_id>/edit/
Đổi mật khẩu	/accounts/password-change/

Bảng 2.3: URL và phương thức để thực hiện một số tác vụ khác trong website

Tác vụ	Url	Method
--------	-----	--------

Gửi đánh giá	/movie/<movie_id>/review/	send_review
Xóa đánh giá	/review/<review_id>/delete/	delete_review

Bảng 2.4: Các thành phần thiết lập lên các trang con trong website

Trang	Class/Method	Form	Template
Trang chủ	IndexView		index.html
Admin	MovieAdmin		base_site.html
Duyệt tìm phim	browse_view		browse.html
Chi tiết phim	DetailView		detail.html
Sửa đánh giá phim	ReviewUpdateView	ReviewUpdate Form	review_update_form.html
Đăng nhập	login_request	UpdatedLogin Form	login.html
Đăng ký	signup_request	NewUserForm	signup.html
Khởi động lại mật khẩu	password_reset_request		password_reset_request.html
Xác nhận gửi mail khởi động mật khẩu	PasswordResetDoneView		password_reset_done.html
Nhập mật khẩu mới	PasswordResetConfirmView		password_reset_confirm.html
Hoàn thành khởi động lại mật khẩu	PasswordResetCompleteView		password_reset_complete.html
Tài khoản cá nhân	profile_view		profile.html
Tạo danh sách phim mới	MovielistCreateView	MovieListForm	movielist_form.html
Chi tiết Danh sách phim	MovielistDetailView		movielist_detail.html

Sửa thông tin danh sách phim	MovielistUpdateView	MovieListForm	movielist_form.html
Đổi mật khẩu	password_change	PasswordChange Form	password_change_form.html

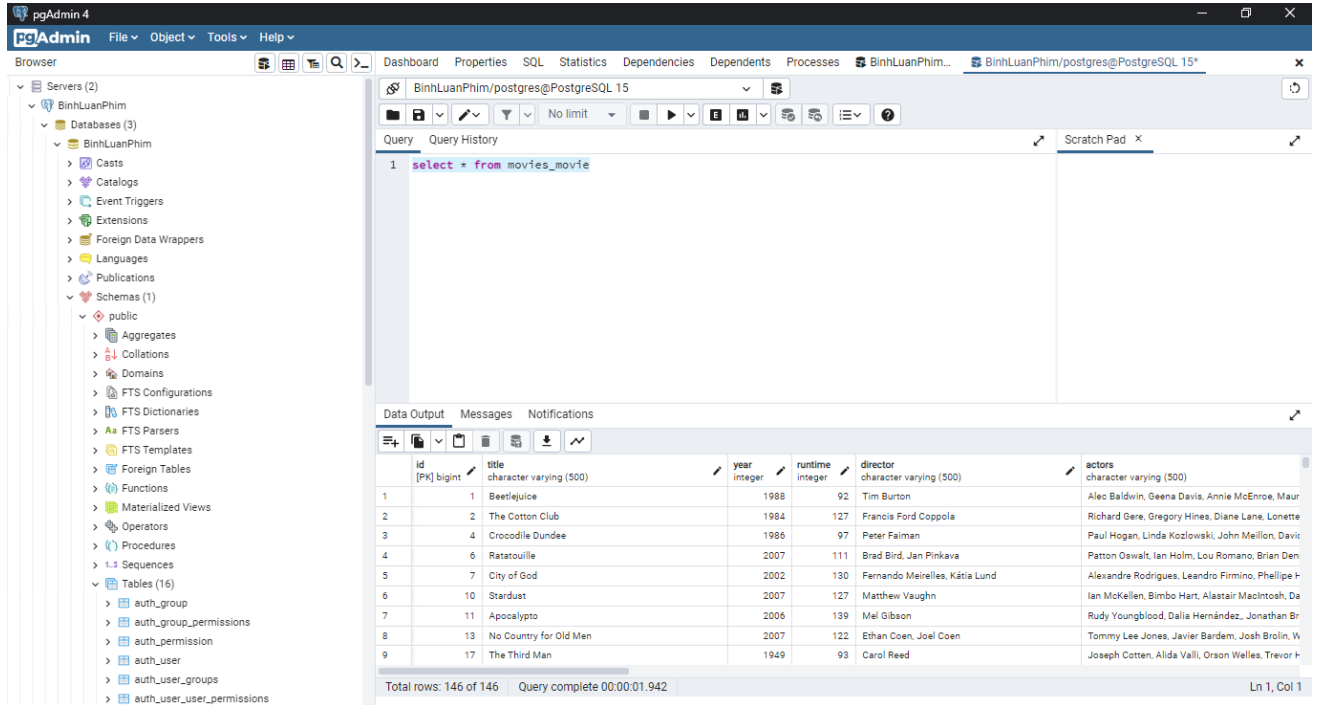
2.3. Tiến hành thiết lập settings và deploy trang web

- Tạo tài khoản, thiết lập cơ sở dữ liệu PostgreSQL trên render.com
- Tiến hành nhập dữ liệu trên PostgreSQL
- Thiết lập settings.py để kết nối ứng dụng với cơ sở dữ liệu và chuẩn bị cho deploy
- Đẩy ứng dụng lên github. Kết nối repo với tính năng web service của render.com
- Điền các thông số environment trên render.com và tiến hành deploy.

Chương 3:

KẾT QUẢ ĐẠT ĐƯỢC

3.1. Tạo cơ sở dữ liệu PostgresDB



The screenshot shows the pgAdmin 4 interface. On the left, the 'BinhLuanPhim' database is expanded, showing various objects like Casts, Catalogs, Event Triggers, Extensions, Foreign Data Wrappers, Languages, Publications, Schemas, and Tables. The 'public' schema is selected, and the 'movies_movie' table is highlighted. The main pane shows the query 'select * from movies_movie' and its results. The results are displayed in a table with columns: id, title, year, runtime, director, and actors. The table contains 146 rows of data.

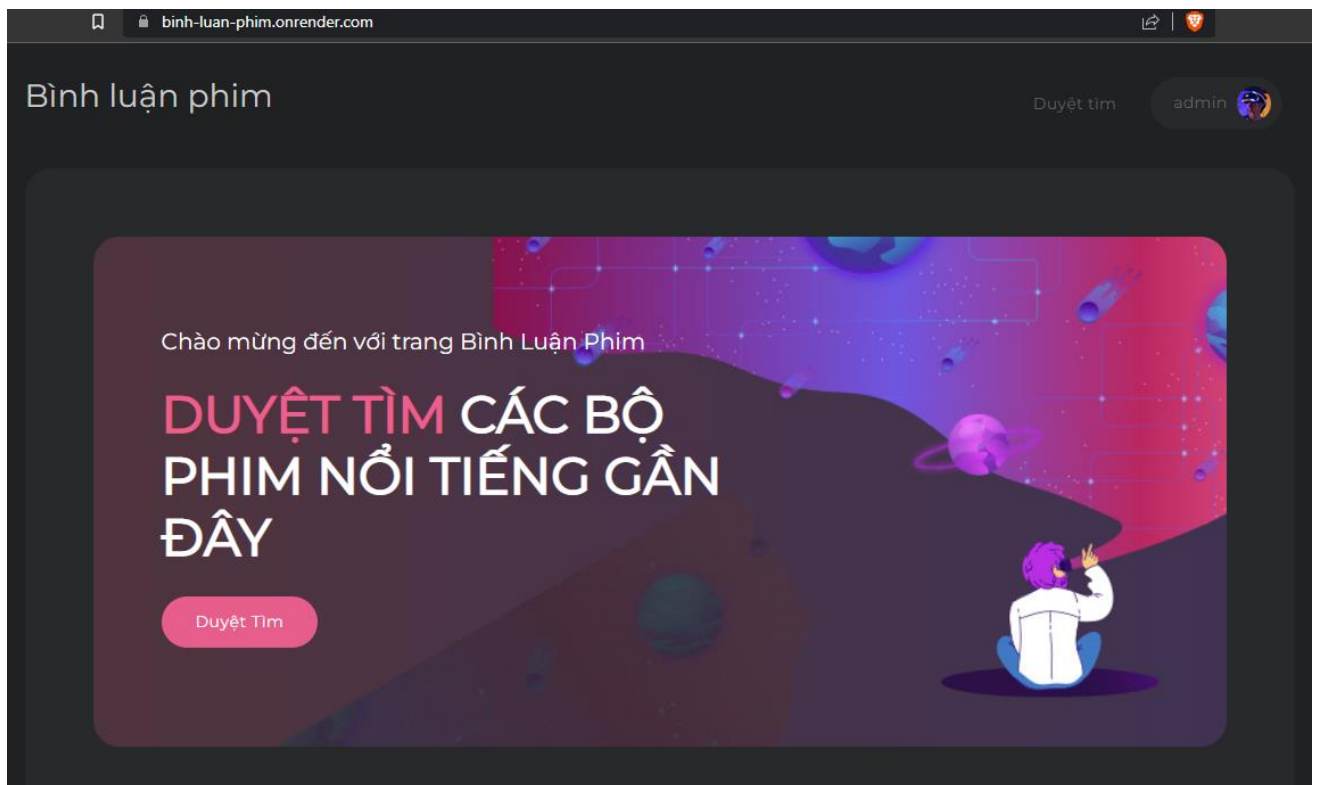
id	title	year	runtime	director	actors
1	Beetlejuice	1988	92	Tim Burton	Alec Baldwin, Geena Davis, Annie McEnroe, Maur...
2	The Cotton Club	1984	127	Francois Ford Coppola	Richard Gere, Gregory Hines, Diane Lane, Lonette
3	Crocodile Dundee	1986	97	Peter Faiman	Paul Hogan, Linda Kozlowski, John Meillon, Davi...
4	Ratatouille	2007	111	Brad Bird, Jan Pinkava	Patton Oswalt, Ian Holm, Lou Romano, Brian Den...
5	City of God	2002	130	Fernando Meirelles, Kátia Lund	Alexandre Rodrigues, Leandro Firmino, Phillippe F...
6	Stardust	2007	127	Matthew Vaughn	Ian McKellen, Bimbo Hart, Alastair MacIntosh, Da...
7	Apocalypse	2006	139	Mel Gibson	Rudy Youngblood, Dalia Hernandez, Jonathan Br...
8	No Country for Old Men	2007	122	Ethan Coen, Joel Coen	Tommy Lee Jones, Javier Bardem, Josh Brolin, W...
9	The Third Man	1949	93	Carol Reed	Joseph Cotten, Alida Valli, Orson Welles, Trevor F...

Hình 3.1: Database BinhLuanPhim hiển thị trên pgAdmin 4

3.2. Ứng dụng Web Bình Luận Phim

3.2.1. Trang chủ

a. Giao diện



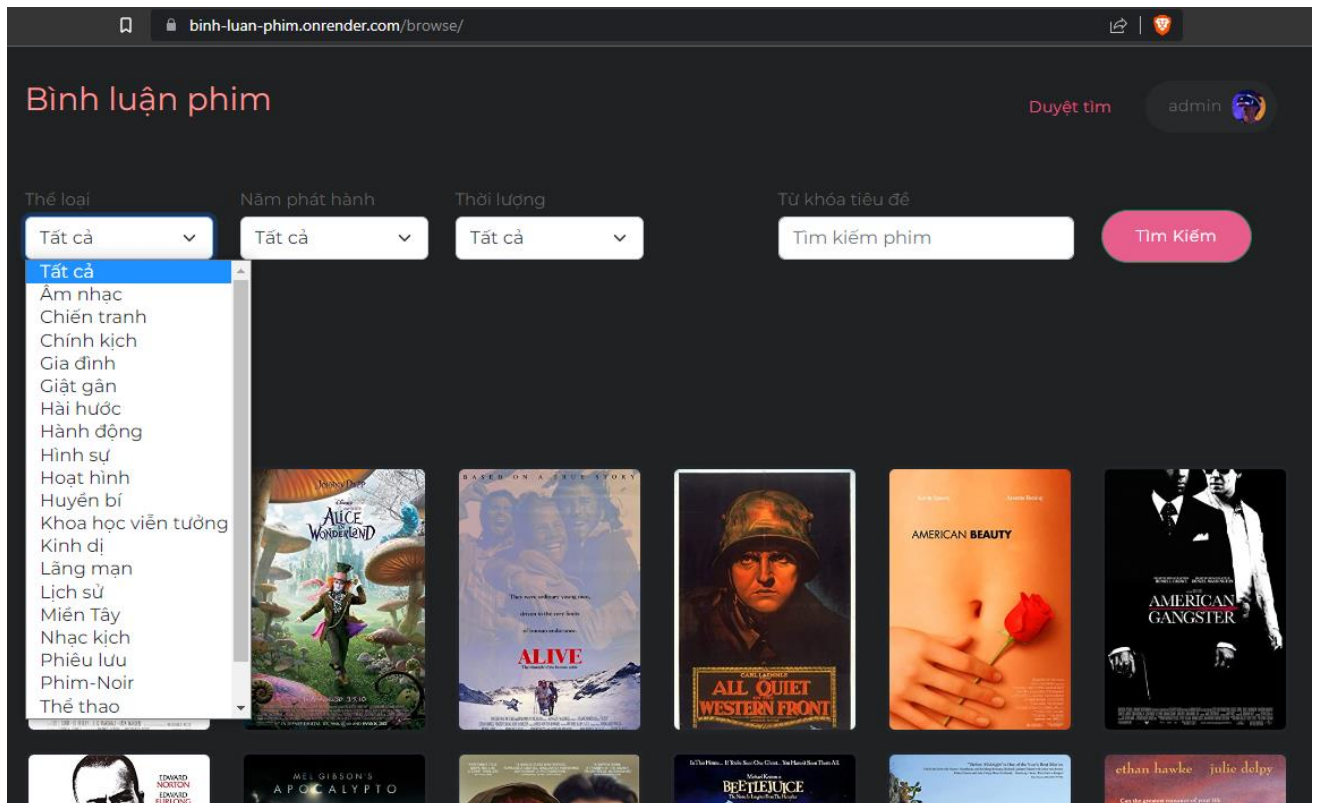
Hình 3.2: Giao diện trang chủ

b. Mô tả

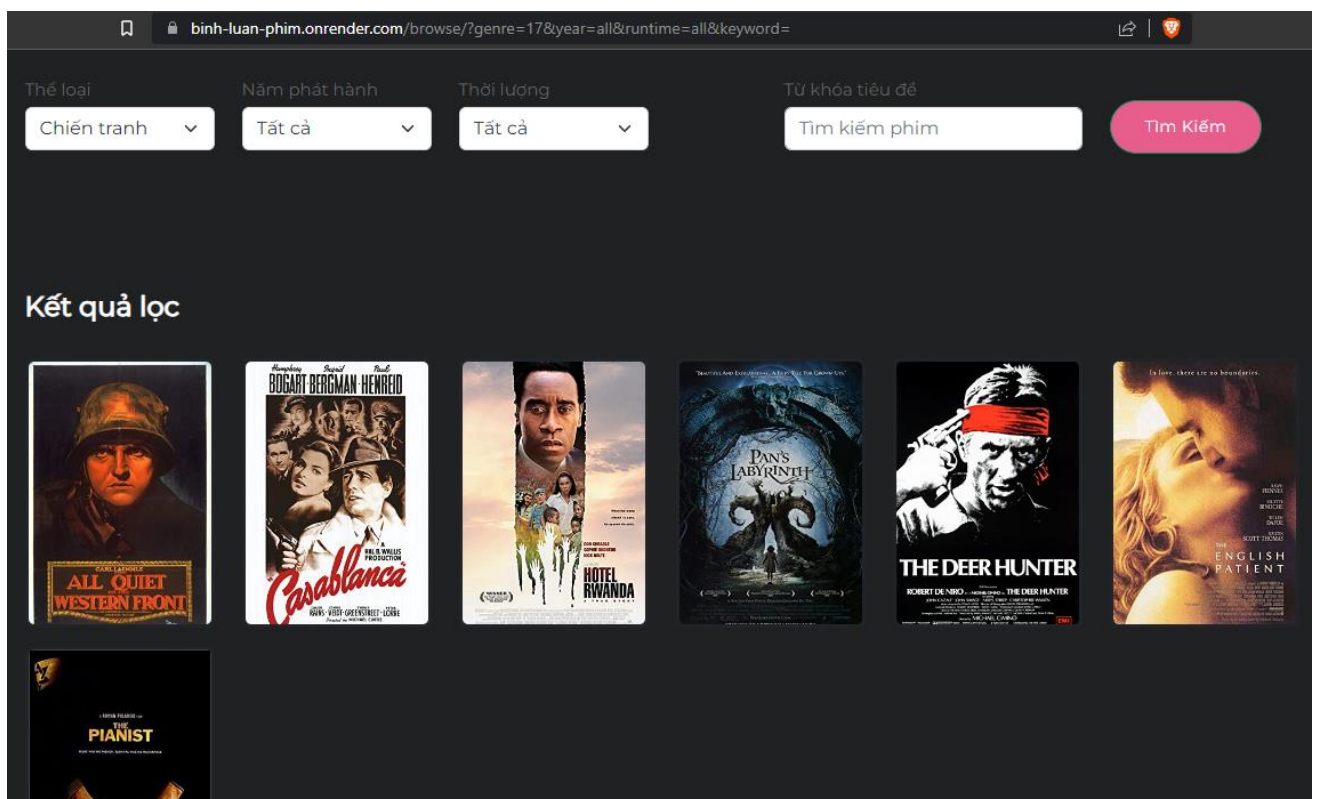
- Chức năng: là trang đầu tiên người dùng truy cập
- Liệt kê các phim mới ra mắt
- Nhấn vào nút Đăng nhập/Đăng ký để đến trang Đăng nhập
- Nhấn vào nút Duyệt tìm để đến trang Duyệt tìm phim

3.2.2. Duyệt tìm phim

a. Giao diện



Hình 3.3: Giao diện trang Duyệt tìm phim mặc định



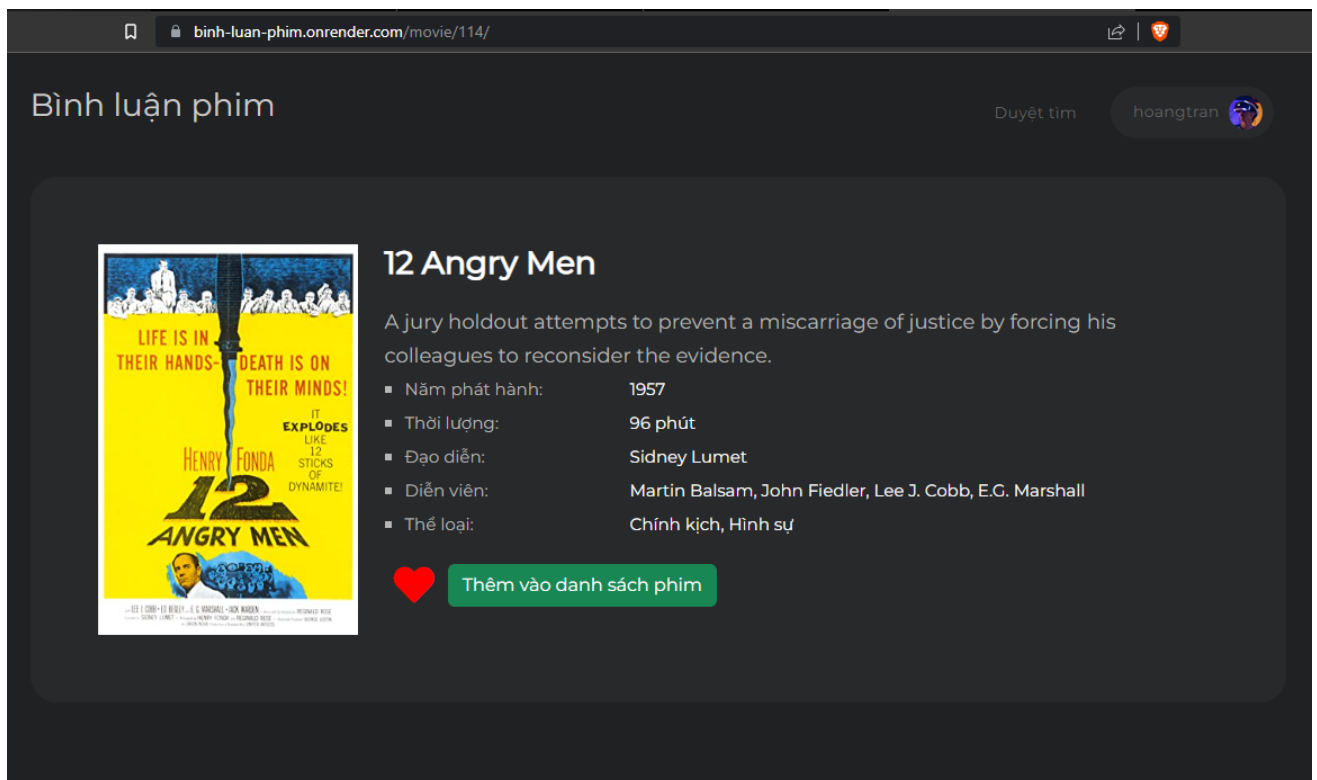
Hình 3.4: Giao diện trang Duyệt tìm phim sau khi lọc kết quả

b. Mô tả

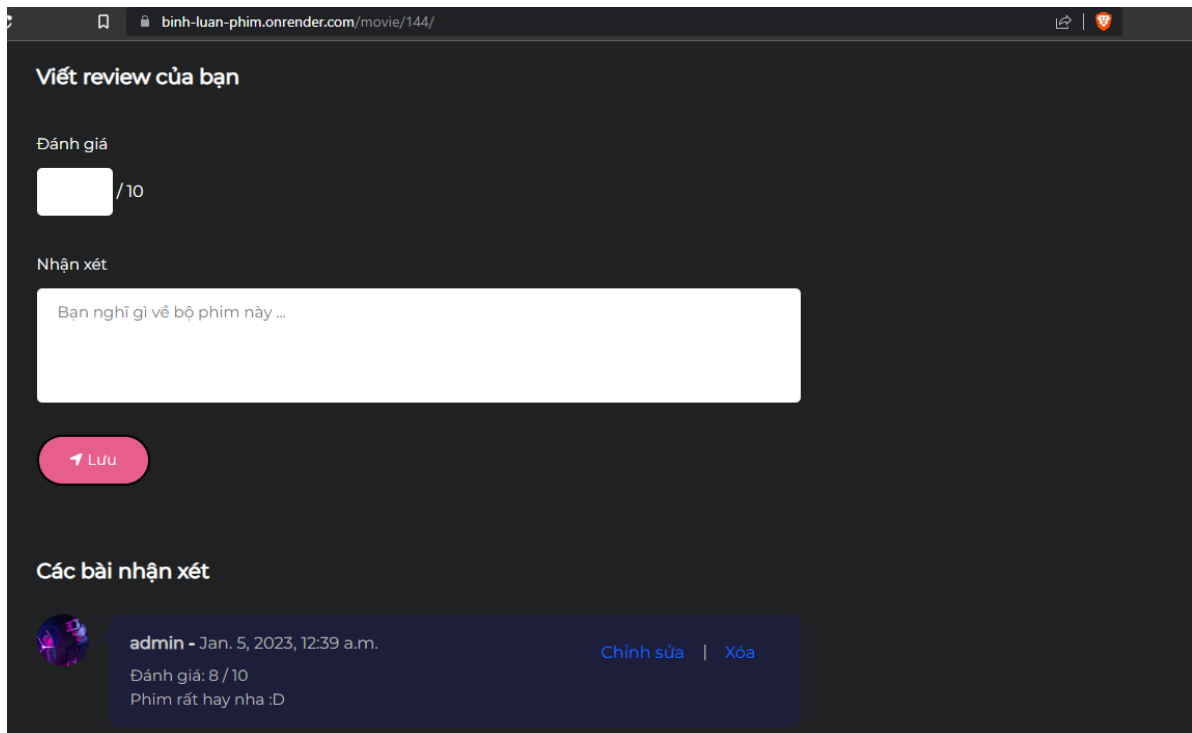
- Chức năng: tìm kiếm/lọc hiển thị phim
- Cho phép tìm và lọc các phim theo các thuộc tính: genre, year, runtime, title. Nhấn nút Tìm kiếm để áp dụng bộ lọc
- Hiển thị kết quả tìm kiếm / lọc phim theo phân trang (pagination). Mặc định sẽ hiển thị tất cả các phim.
- Nhấn vào từng kết quả tìm kiếm để truy cập vào trang Chi tiết phim của phim đó

3.2.3. Trang Chi tiết phim

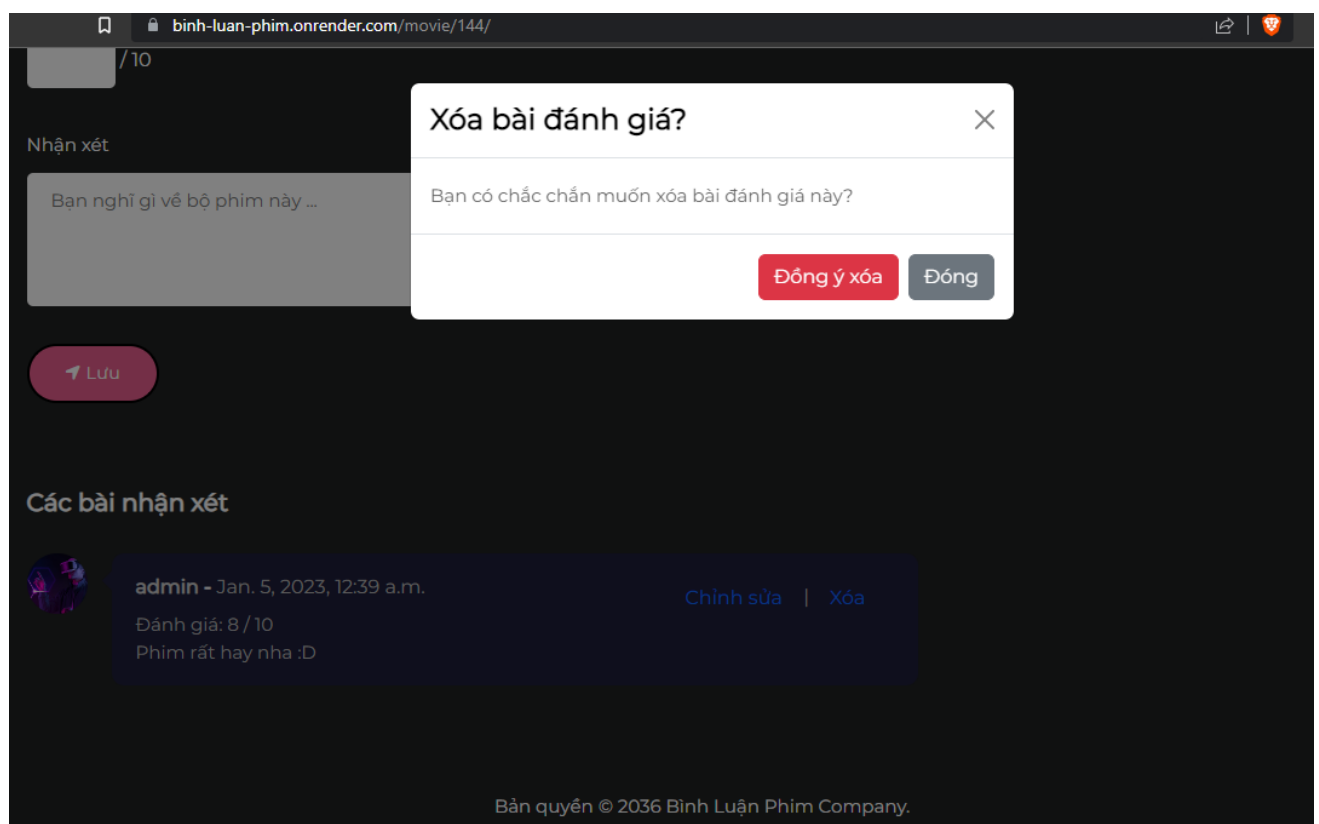
a. Giao diện



Hình 3.5: Giao diện trang Chi tiết phim – phần Thông tin phim



Hình 3.6: Giao diện trang Chi tiết phim – phần Đánh giá



Hình 3.7: Giao diện trang Chi tiết phim – Xác nhận xóa đánh giá

b. Mô tả

- Chức năng: Hiển thị thông tin của bộ phim liên quan
- Hiển thị các bài đánh giá (Review) của tất cả người dùng
- Nhấn vào biểu tượng hình tìm để thêm phim vào danh sách yêu thích (khi đã đăng nhập)
- Nhấn vào nút “Thêm vào danh sách phim”, modal sẽ hiển thị liệt kê các danh sách phim đang có
- Điền vào form bình luận và nhấn nút lưu để đăng bình luận phim (yêu cầu đăng nhập)
- Ở mỗi đánh giá, nhấn vào nút “Chỉnh sửa” (nếu có) để tiến hành chỉnh sửa đánh giá đó (đối với đánh giá do người dùng đó viết)
- Nhấn vào nút “Xóa” ở mỗi đánh giá (đối với đánh giá do người dùng đó viết) để xóa đánh giá. Modal sẽ hiển thị yêu cầu xác nhận đồng ý xóa. Nếu đồng ý, hệ thống thực hiện xóa đánh giá, website sẽ tải lại trang kèm thông báo đã xóa đánh giá thành công

3.2.4. Trang Sửa đánh giá phim

a. Giao diện

Hình 3.8: Giao diện trang Sửa đánh giá phim

b. Mô tả

- Hiển thị thông tin phim muốn sửa đánh giá
- Chỉnh sửa bài đánh giá trong form. Nhấn “Cập nhật” để lưu các thay đổi. Chỉnh sửa thành công sẽ quay về trang Chi tiết phim

- Chính sửa không thành công sẽ tải lại trang và thông báo lỗi.

3.2.5. Trang Đăng nhập

a. Giao diện

Hình 3.9: Giao diện trang Đăng nhập

b. Mô tả

- Hiện thị form đăng nhập. Nhập các thông tin Tên người dùng, Mật khẩu, Lưu đăng nhập. Nhấn nút “Đăng nhập” để tiến hành đăng nhập
- Nhấn “Tạo tài khoản mới” để đến trang Đăng ký
- Nhấn “Tôi không nhớ mật khẩu” để truy cập trang Khởi động lại mật khẩu trong trường hợp quên mật khẩu và muốn khởi động lại

3.2.6. Trang Đăng ký

a. Giao diện

binh-luan-phim.onrender.com/accounts/signup/

Đăng Ký

Tên người dùng*

Tên người dùng ít hơn 150 ký tự. Chỉ dùng chữ, số và các ký tự @/./+/-/_

Email*

Mật khẩu*

Mật khẩu mới không được quá giống với mật khẩu cũ, có ít nhất 8 ký tự, có tính bảo mật, và không được chứa toàn số

Nhập lại mật khẩu*

Nhập lại mật khẩu như trên để xác minh

Đăng Ký

Hình 3.10: Giao diện trang Đăng ký

b. Mô tả

- Tại trang này người dùng có thể đăng ký thông tin của mình để tạo tài khoản
- Điền các thông tin vào form theo yêu cầu và nhấn Đăng ký. Nếu đăng ký thành công, website sẽ đăng nhập với tài khoản người dùng vừa tạo và đi đến trang chủ
- Đăng ký thất bại sẽ tải lại trang Đăng ký và thông báo lỗi

3.2.7. Trang Khởi động lại mật khẩu

a. Giao diện

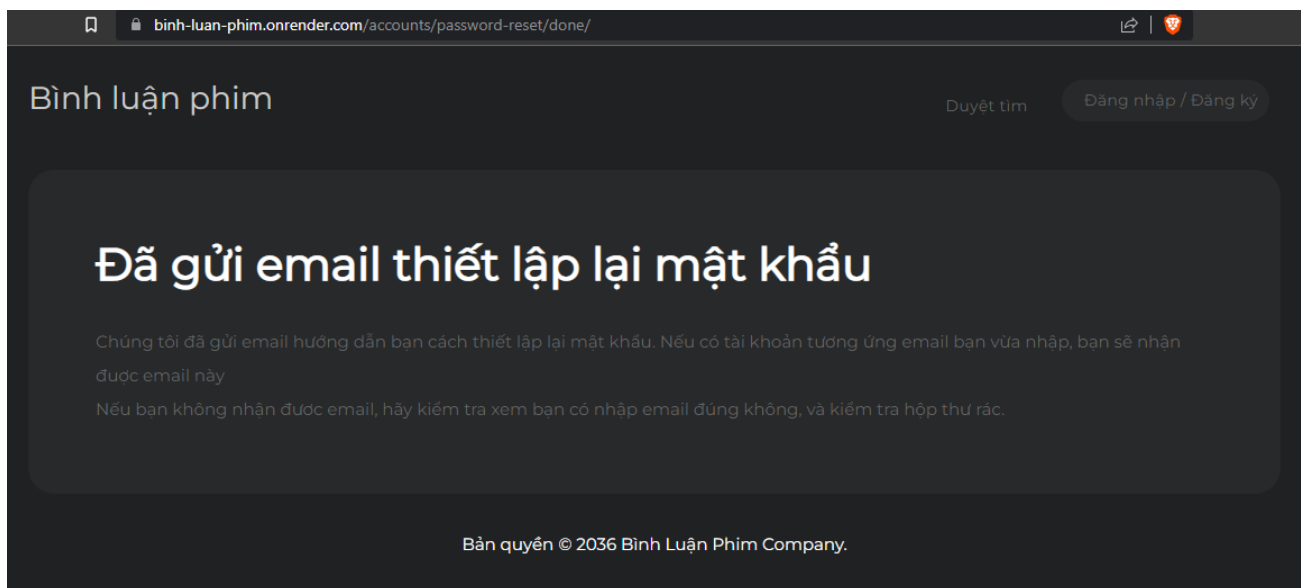
Hình 3.11: Giao diện trang Khởi động lại mật khẩu

b. Mô tả

- Hiện thị form để người dùng nhập email, tiến hành khởi động lại mật khẩu.
- Người dùng nhập email và nhấn nút “Gửi link qua Email”. Nếu tìm thấy User có email tương ứng, website sẽ được chuyển qua trang Xác nhận đã gửi mail khởi động mật khẩu
- Nếu không tìm thấy người dùng có Email hay form không hợp lệ sẽ tải lại trang và thông báo lỗi

3.2.8. Trang Xác nhận gửi mail khởi động mật khẩu

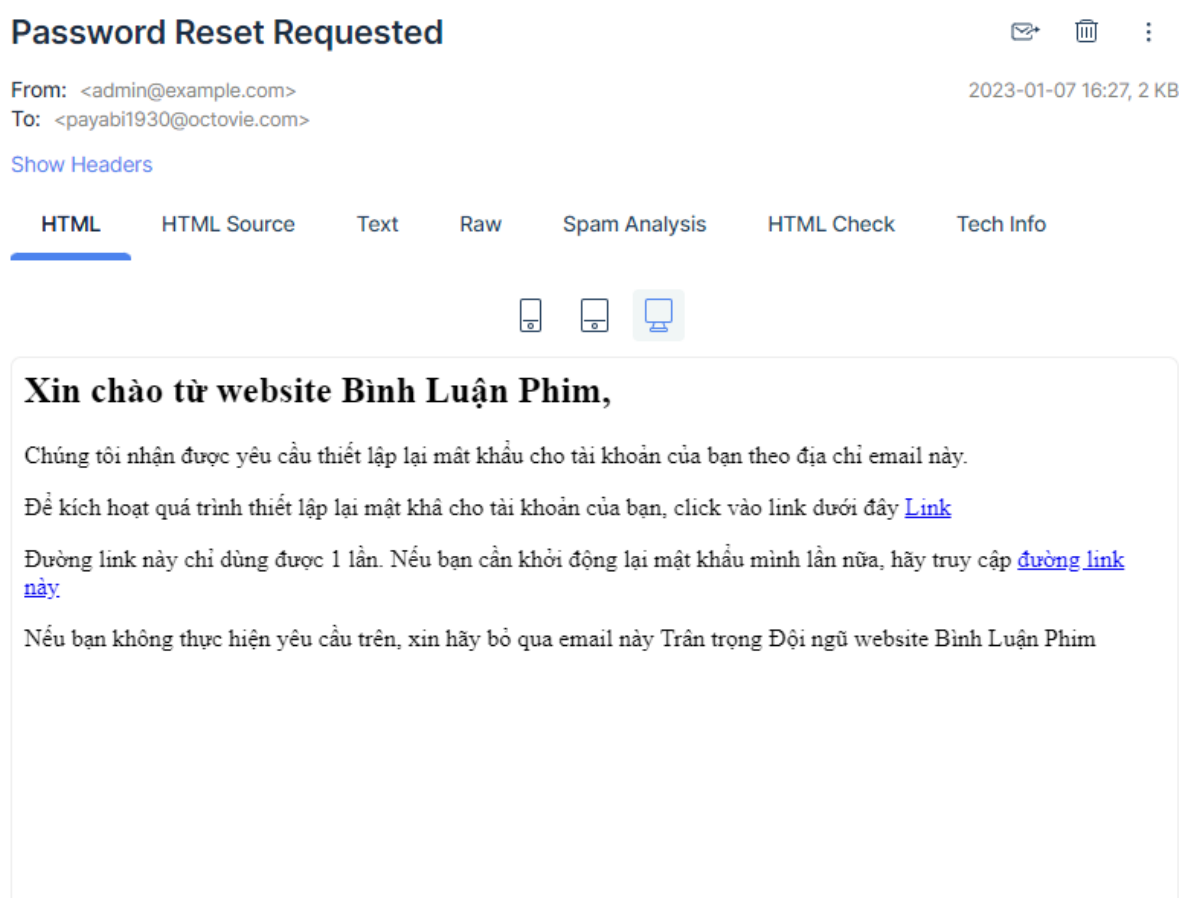
a. Giao diện



Hình 3.12: Giao diện trang Xác nhận đã gửi mail khởi động mật khẩu

b. Mô tả

- Hiển thị thông tin đã gửi mail chứa đường link khởi động lại mật khẩu



Hình 3.13: Email người dùng nhận được khi xác nhận lại mật khẩu

3.2.9. Trang Nhập mật khẩu mới

a. Giao diện

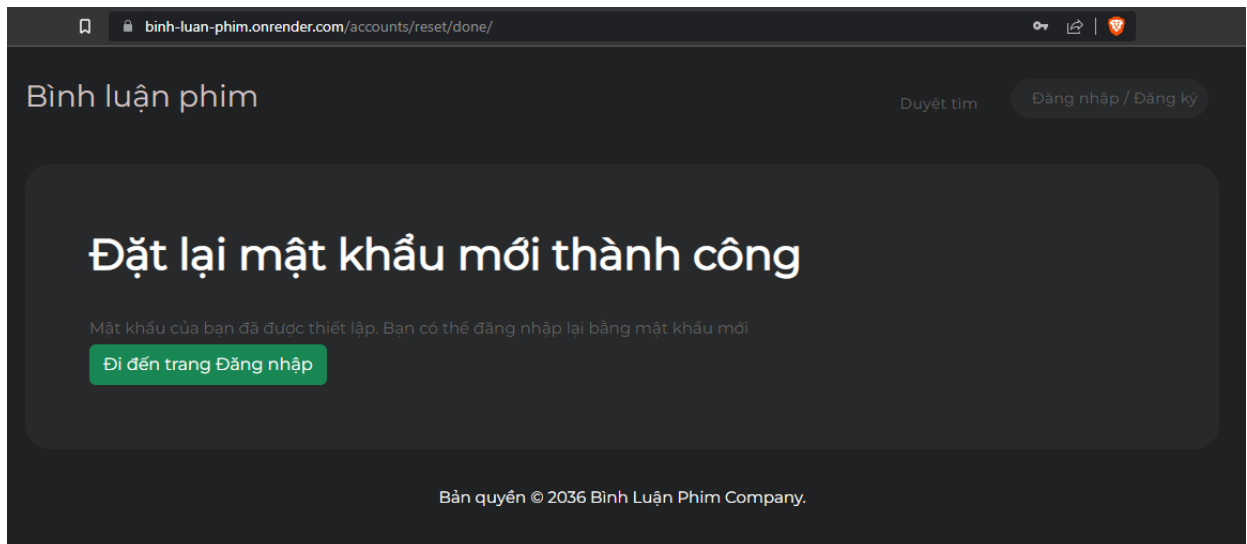
Hình 3.14: Giao diện trang Nhập lại mật khẩu mới

b. Mô tả

- Hiển thị form nhập mật khẩu mới. Điền mục Mật khẩu mới và Nhập lại mật khẩu mới, sau đó nhấn nút “Đặt lại mật khẩu”
- Người dùng gửi form thành công sẽ được thông báo và website sẽ chuyển qua trang Hoàn thành khởi động lại mật khẩu
- Nếu form không hợp lệ sẽ tại lại trang và thông báo lỗi

3.2.10. Trang Hoàn thành khởi động lại mật khẩu

a. Giao diện



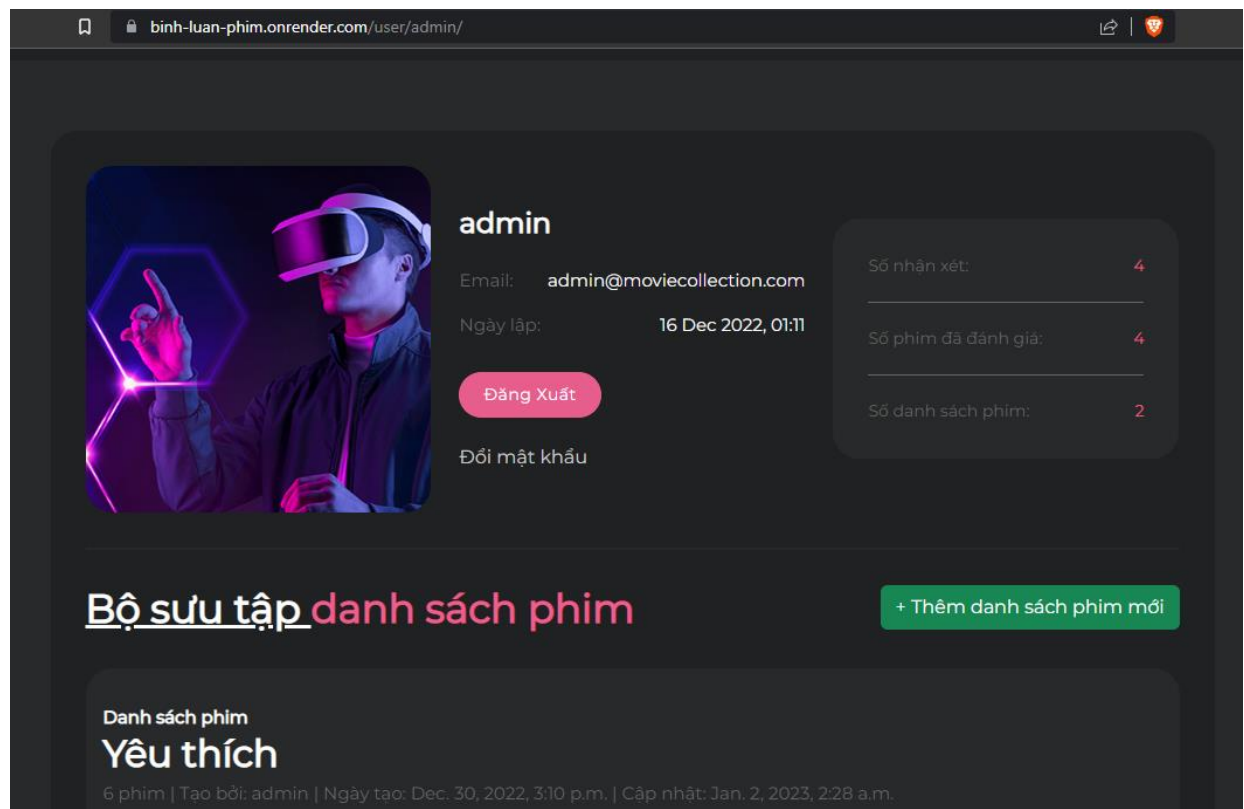
Hình 3.15: Giao diện trang Hoàn thành khởi động lại mật khẩu

b. Mô tả

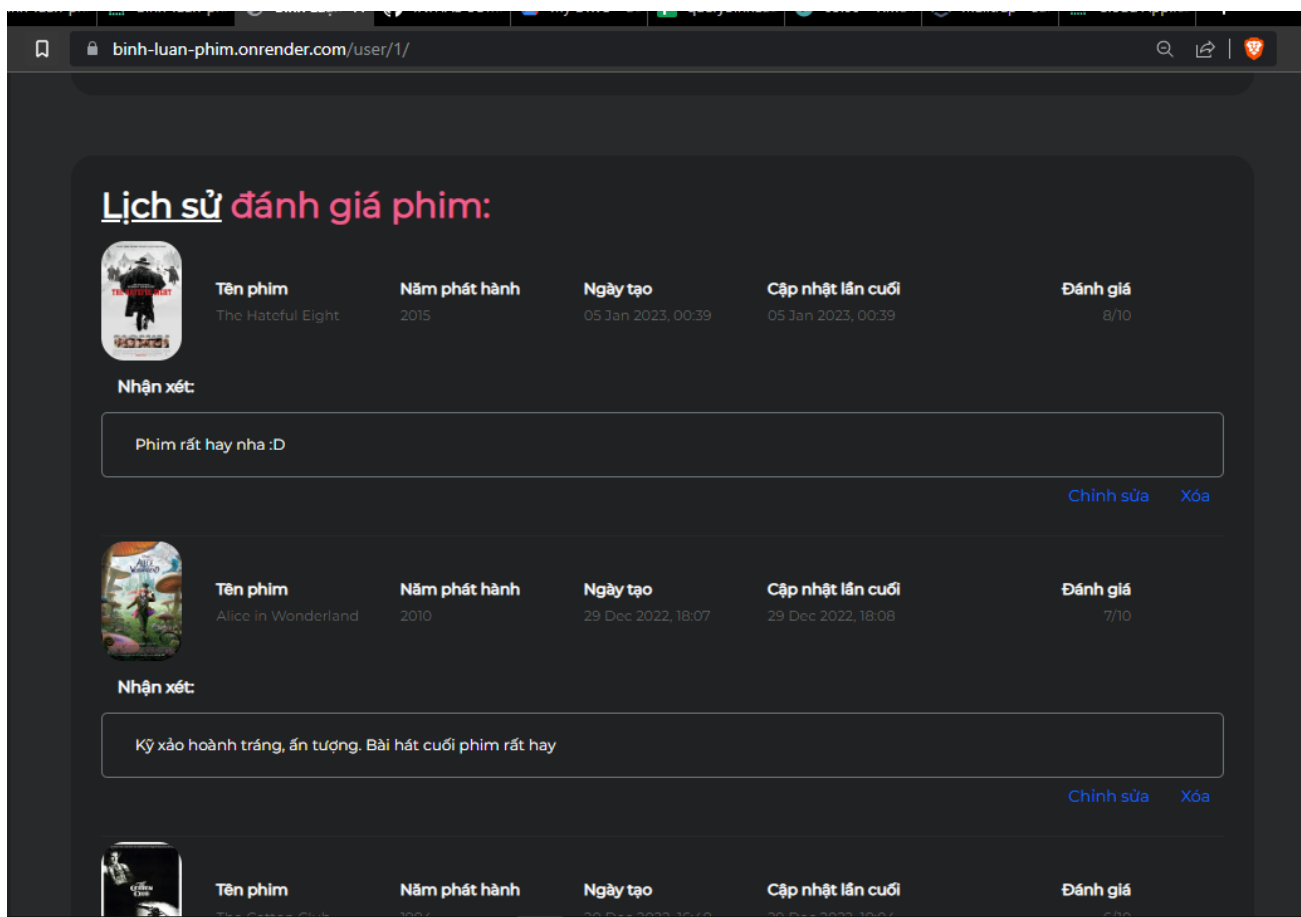
- Hiển thị thông tin đã Hoàn thành khởi động lại mật khẩu
- Nhấn vào nút “Trở lại trang Đăng nhập” để đi đến trang Đăng nhập

3.2.11. Trang Tài khoản cá nhân

a. Giao diện



Hình 3.16: Giao diện trang Tài khoản cá nhân – thông tin người dùng và bộ sưu tập phim



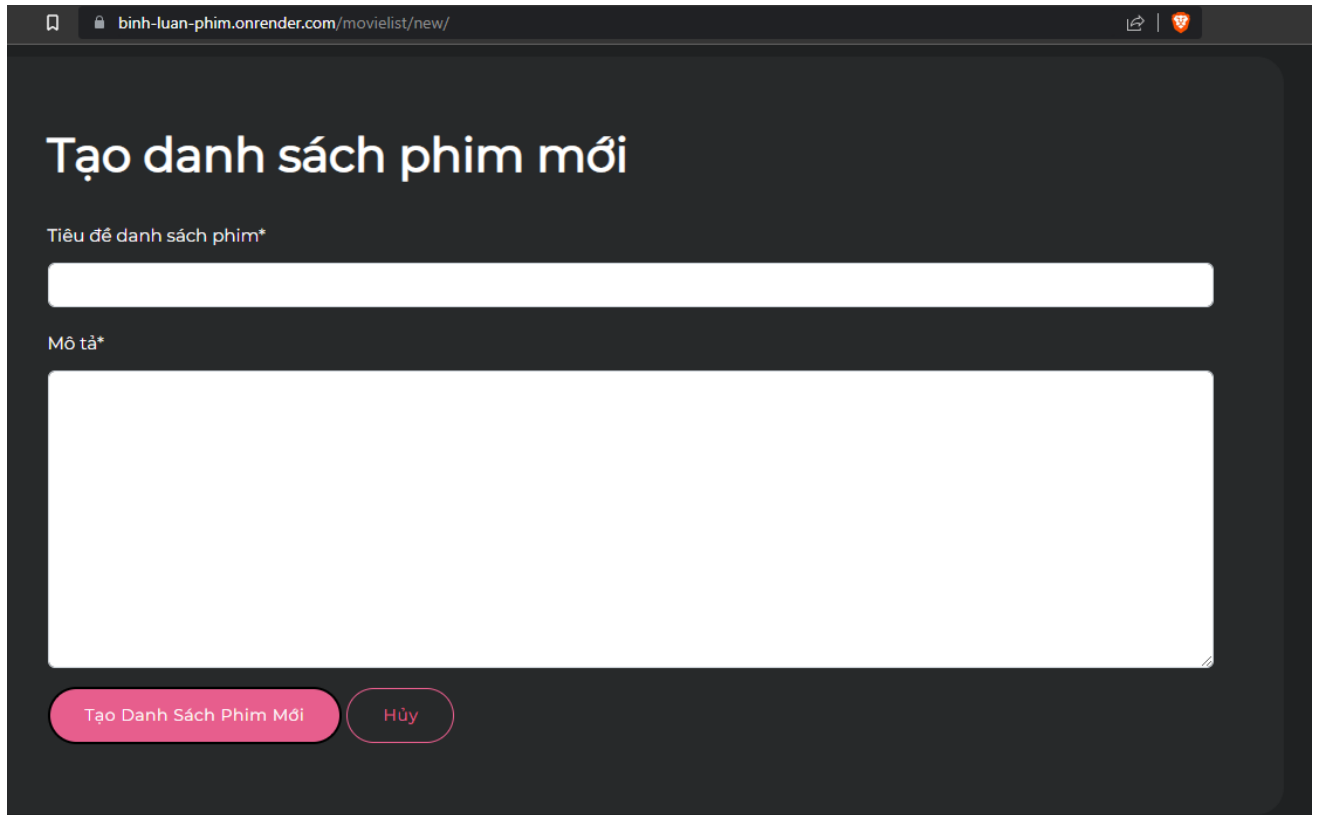
Hình 3.17: Giao diện trang Tài khoản cá nhân – Lịch sử đánh giá

b. Mô tả

- Hiển thị thông tin email và ngày đăng ký, số lượng bài đánh giá, số lượng danh sách phim người dùng đã tạo
- Nhấn nút “Đăng xuất” thực hiện đăng xuất (đối với trang cá nhân của người dùng đó)
- Nhấn nút “Đổi mật khẩu” để đi đến trang Đổi mật khẩu (đối với trang cá nhân của người dùng đó)
- Hiển thị các Danh sách phim của người dùng đó
- Nhấn vào nút “Chi tiết” của Danh sách phim để đi đến trang Chi tiết Danh sách phim tương ứng
- Nhấn vào nút “Tạo danh sách phim mới” để đến trang tạo Danh sách phim mới (đối với trang cá nhân của người dùng đó)
- Hiển thị các đánh giá mà người dùng đó đã đăng
- Với mỗi đánh giá, nhấn “Chỉnh sửa” để tiến hành chỉnh sửa đánh giá, nhấn nút “Xóa” để xóa bài đánh giá đó (đối với trang cá nhân của người dùng đó)

3.2.12. Trang Tạo danh sách phim mới

a. Giao diện



The screenshot shows a web browser window with the URL `binh-luan-phim.onrender.com/movielist/new/`. The page title is "Tạo danh sách phim mới". Below the title, there is a form with two input fields: "Tiêu đề danh sách phim*" (Movie list title) and "Mô tả*" (Description). At the bottom of the form, there are two buttons: "Tạo Danh Sách Phim Mới" (Create New Movie List) and "Hủy" (Cancel).

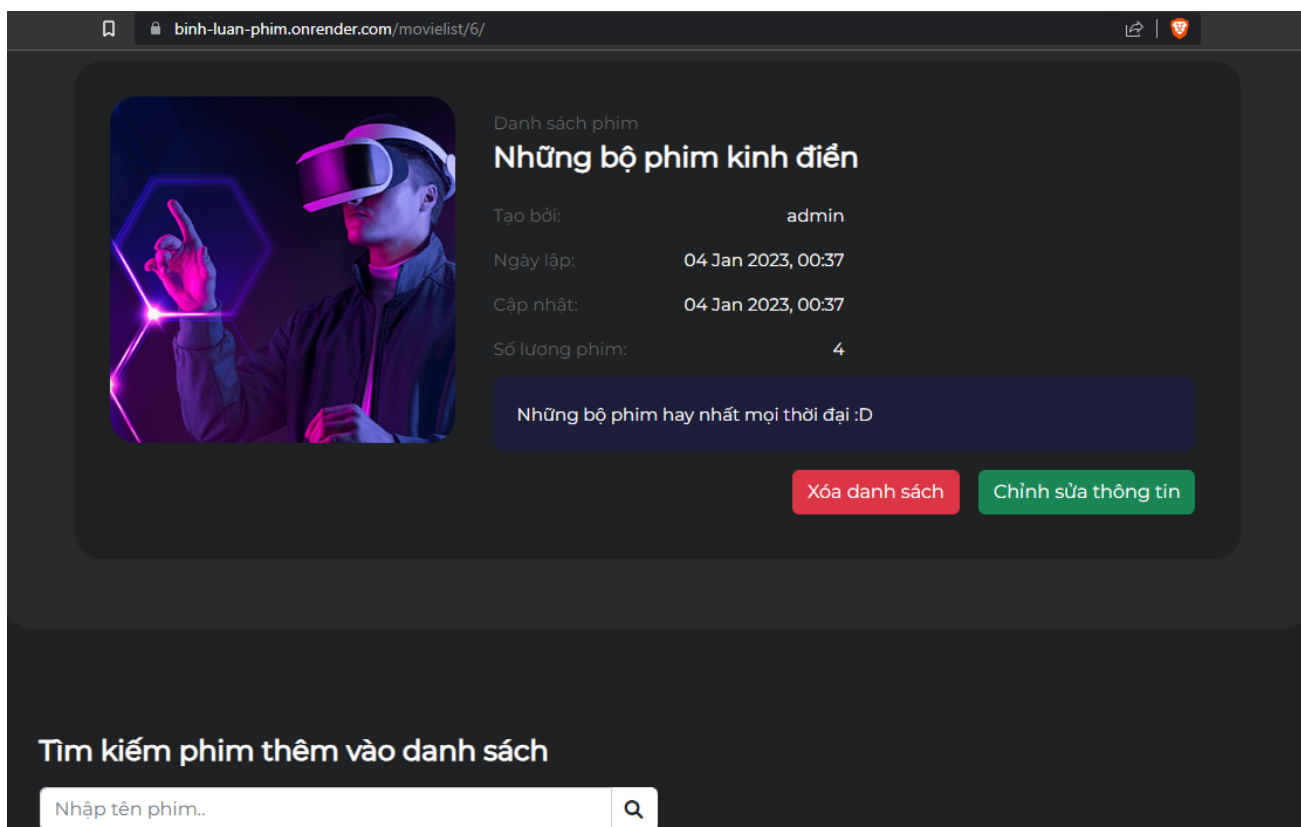
Hình 3.18: Giao diện trang Tạo danh sách phim mới

b. Mô tả

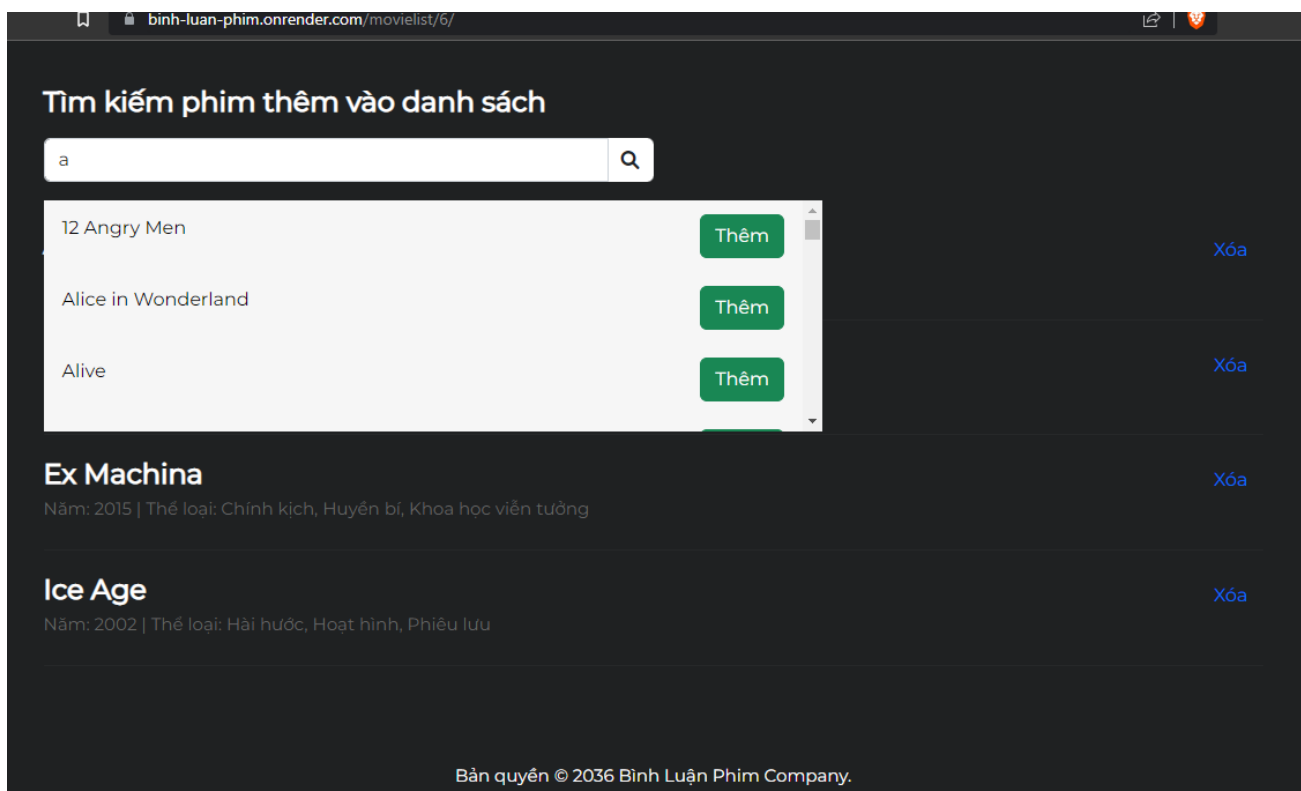
- Hiện thị form tạo Danh sách phim mới. Nhập các mục Tiêu đề danh sách phim, Mô tả, rồi nhấn nút “Tạo danh sách mới”
- Khi form được gửi thành công, website sẽ tạo Danh sách phim mới dựa trên form vừa nhập, chuyển qua trang Chi tiết Danh sách phim cho Danh sách vừa được tạo kèm thông báo đã tạo thành công
- Nếu form không hợp lệ sẽ tải lại trang và thông báo lỗi

3.2.13. Trang Chi tiết Danh sách phim

a. Giao diện



Hình 3.19: Giao diện trang Chi tiết Danh sách phim – phần thông tin Danh sách phim



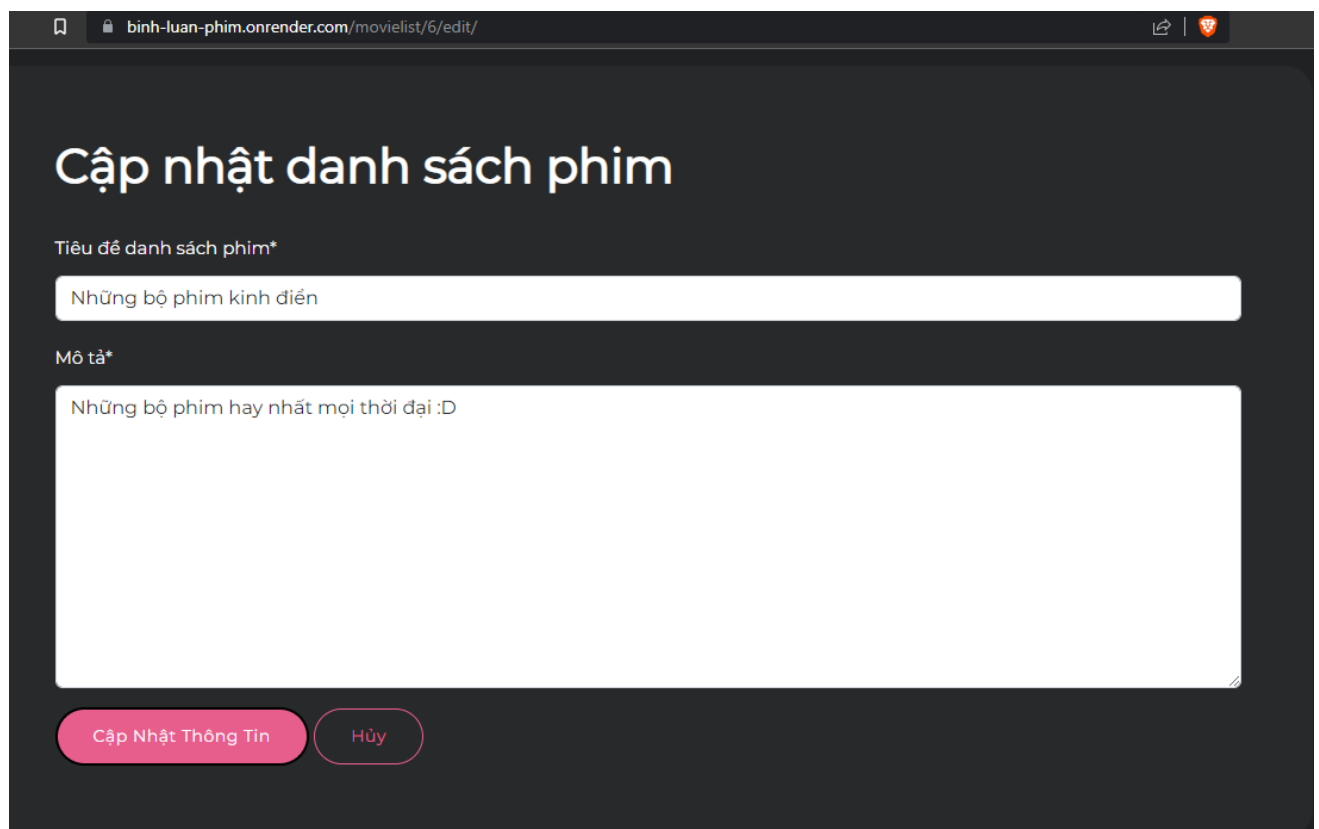
Hình 3.20: Giao diện trang Chi tiết Danh sách phim – phần điều chỉnh danh sách phim

b. Mô tả

- Hiển thị thông tin của Danh sách phim, gồm tên, số lượng phim, mô tả
- Cho phép đi đến trang chỉnh sửa Danh sách phim
- Cho phép xóa Danh sách phim. Hiển thị modal xác nhận xóa trước khi thực hiện. Sau khi xóa thành công website sẽ chuyển qua trang Tài khoản cá nhân
- Hiển thị các phim trong Danh sách phim
- Cho phép tìm kiếm, thêm, xóa phim

3.2.14. Trang Sửa thông tin danh sách phim

a. Giao diện



The screenshot shows a web browser window with the URL `binh-luan-phim.onrender.com/movielist/6/edit/`. The page title is 'Cập nhật danh sách phim'. Below the title, there are two input fields. The first field is labeled 'Tiêu đề danh sách phim*' and contains the text 'Những bộ phim kinh điển'. The second field is labeled 'Mô tả*' and contains the text 'Những bộ phim hay nhất mọi thời đại :D'. At the bottom of the form, there are two buttons: 'Cập Nhật Thông Tin' (Update Information) and 'Hủy' (Cancel).

Hình 3.21: Giao diện trang Sửa thông tin danh sách phim

b. Mô tả

- Hiển thị form Sửa thông tin danh sách phim, chỉnh sửa các trường: name, description
- Khi form được gửi thành công, website sẽ chuyển qua trang Chi tiết Danh sách phim và thông báo đã cập nhật thành công
- Nếu form không hợp lệ sẽ tải lại trang và thông báo lỗi

3.2.15. Trang Đổi mật khẩu

a. Giao diện

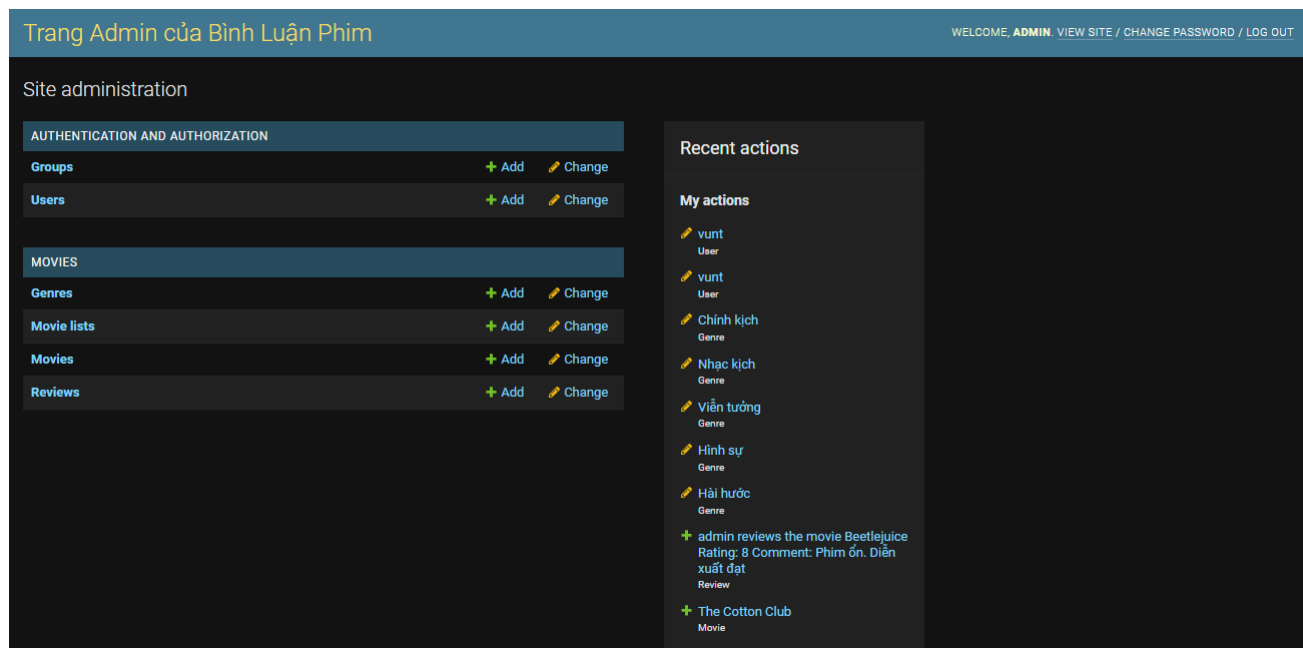
Hình 3.22: Giao diện trang Đổi mật khẩu

b. Mô tả

- Hiện thị form Đổi mật khẩu. Nhập thông tin vào các mục Mật khẩu cũ, Mật khẩu mới, Nhập lại mật khẩu mới
- Khi form được gửi thành công, website sẽ chuyển qua trang Tài khoản cá nhân và thông báo đã cập nhật thành công
- Nếu form không hợp lệ sẽ tải lại trang và thông báo lỗi

3.2.16. Trang Admin

a. Giao diện



Hình 3.23: Giao diện trang Admin

b. Mô tả

- Trang dành cho người dùng là admin hoặc quản trị web
- Cho phép Admin và những người liên quan có thể truy cập và tiến hành xem, thêm, xóa, sửa các đối tượng Movie, Genre, Review, MovieList, User.
- Tiến hành phân quyền

TÀI LIỆU THAM KHẢO

[1] Django Documentation: <https://docs.djangoproject.com/en/4.1/>

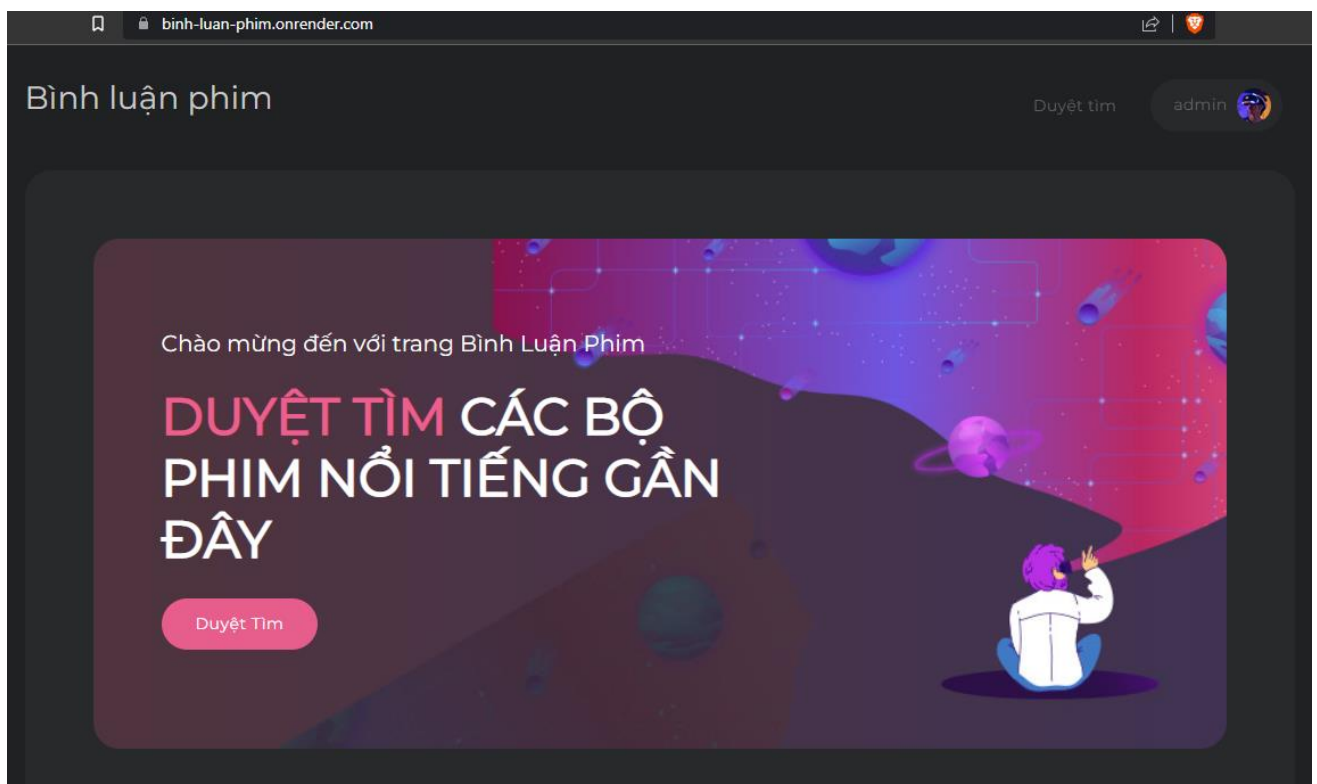
PHỤ LỤC 1: GIỚI THIỆU (DEMO) KẾT QUẢ

GitHub của đồ án: <https://github.com/vunguyent17/binh-luan-phim>

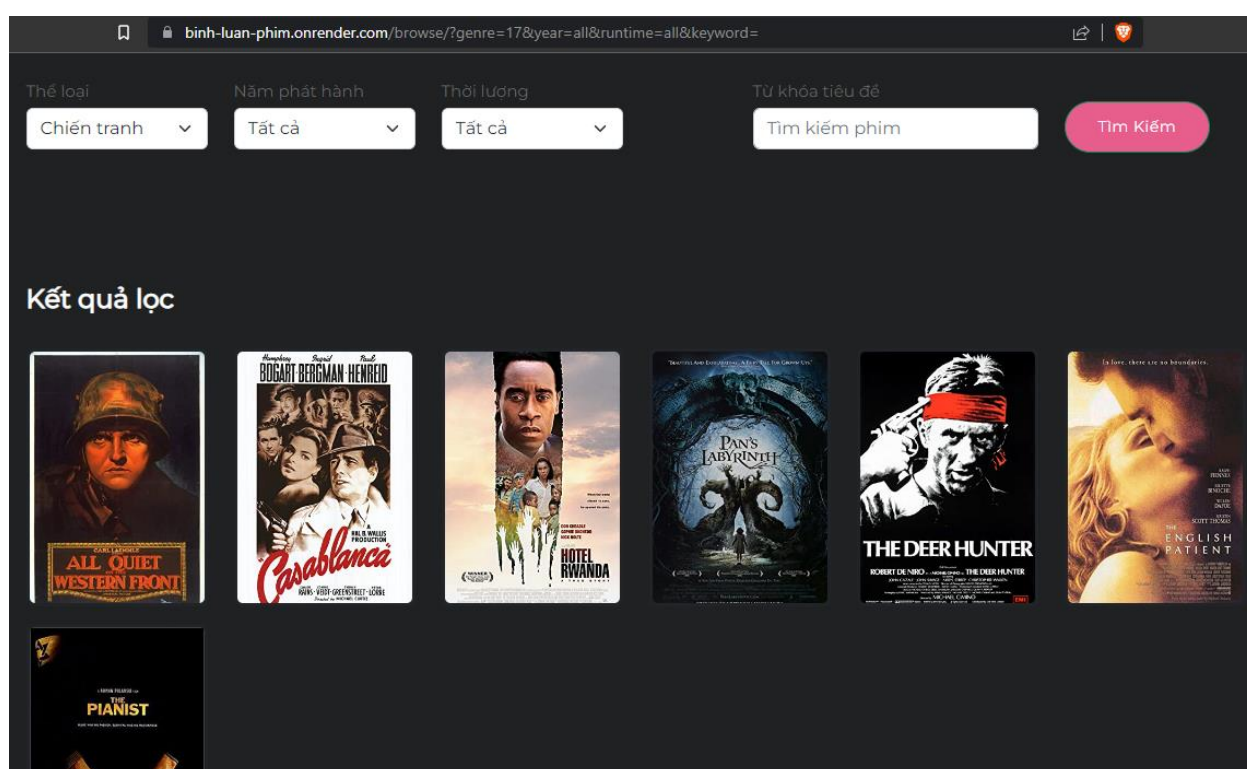
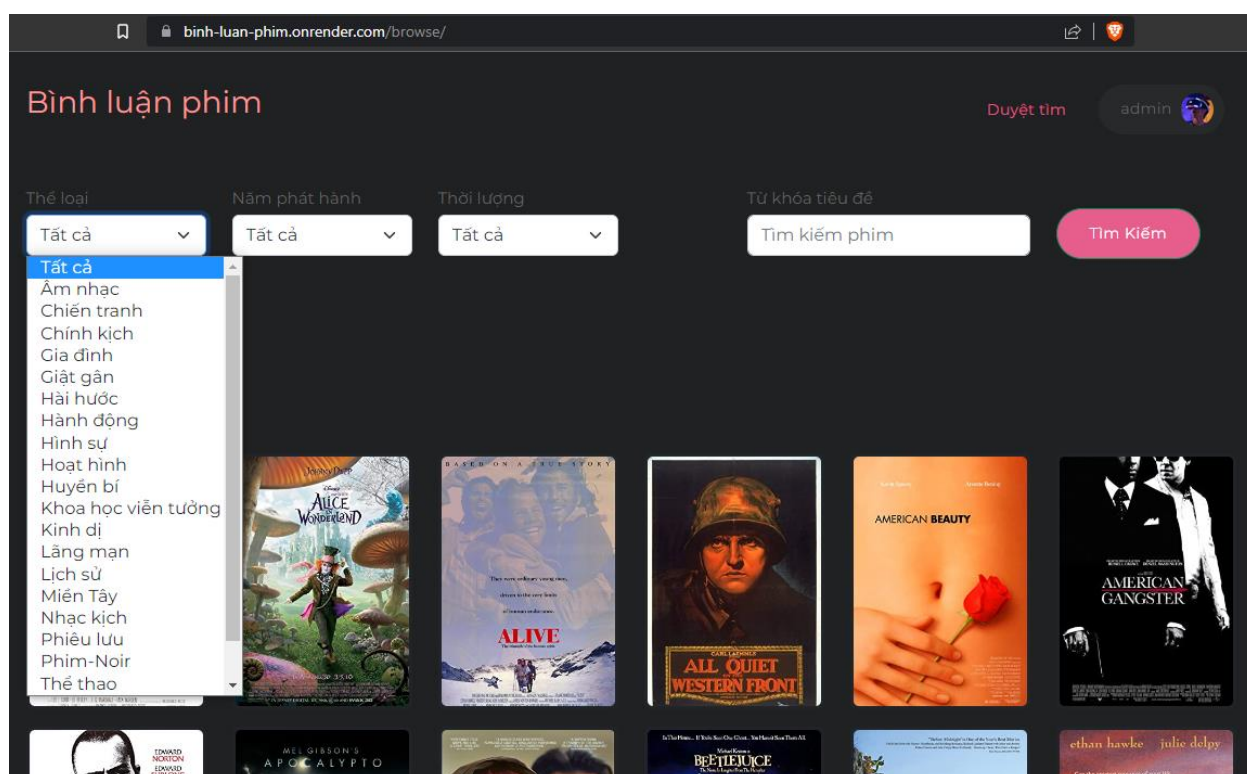
Website demo: <https://binh-luan-phim.onrender.com/>

Đăng nhập admin:

- Username: admin
- Mật khẩu: admin012
- Trang chủ



- Trang Duyệt tìm phim




- Trang Chi tiết phim

binh-luan-phim.onrender.com/movie/114/

Bình luận phim

Duyệt tìm

hoangtran



LIFE IS IN THEIR HANDS- DEATH IS ON THEIR MINDS!

IT EXPLODES LIKE 12 STICKS OF DYNAMITE!


HENRY FORDA

12 ANGRY MEN

12 Angry Men

A jury holdout attempts to prevent a miscarriage of justice by forcing his colleagues to reconsider the evidence.

- Năm phát hành: 1957
- Thời lượng: 96 phút
- Đạo diễn: Sidney Lumet
- Diễn viên: Martin Balsam, John Fiedler, Lee J. Cobb, E.G. Marshall
- Thể loại: Chính kịch, Hình sự

 [Thêm vào danh sách phim](#)

binh-luan-phim.onrender.com/movie/144/

Viết review của bạn

Đánh giá


/10

Nhận xét

Bạn nghĩ gì về bộ phim này ...

Lưu

Các bài nhận xét



admin - Jan. 5, 2023, 12:39 a.m.

Đánh giá: 8 / 10

Phim rất hay nha :D

[Chỉnh sửa](#) | [Xóa](#)

- Trang Sửa đánh giá phim

35

Sửa đánh giá

Thông tin phim

- Tên phim: Alice in Wonderland
- Năm phát hành: 2010
- Thể loại: Gia đình, Phiêu lưu, Viễn tưởng

Đánh giá*

7

Trên thang điểm 10

Nhận xét*

Kỹ xảo hoành tráng, ấn tượng. Bài hát cuối phim rất hay

Cập Nhật Hủy

- Trang Đăng nhập

binh-luan-phim.onrender.com/accounts/login/ Đăng nhập / Đăng ký

Bình luận phim

Đăng nhập

Tên người dùng*

Mật khẩu*

☐ Lưu đăng nhập

Đăng Nhập

Bạn chưa có tài khoản? [Tạo tài khoản mới.](#)

[Tôi không nhớ mật khẩu.](#)

- Trang Đăng ký

binh-luan-phim.onrender.com/accounts/signup/

Đăng Ký

Tên người dùng*

Tên người dùng ít hơn 150 ký tự. Chỉ dùng chữ, số và các ký tự @/./+/_/./

Email*

Mật khẩu*

Mật khẩu mới không được quá giống với mật khẩu cũ, có ít nhất 8 ký tự, có tính bảo mật, và không được chứa toàn số

Nhập lại mật khẩu*

Nhập lại mật khẩu như trên để xác minh

Đăng Ký

- Trang Khởi động lại mật khẩu

Bình luận phim

Duyệt tìm

Đăng nhập / Đăng ký

Khởi động lại mật khẩu

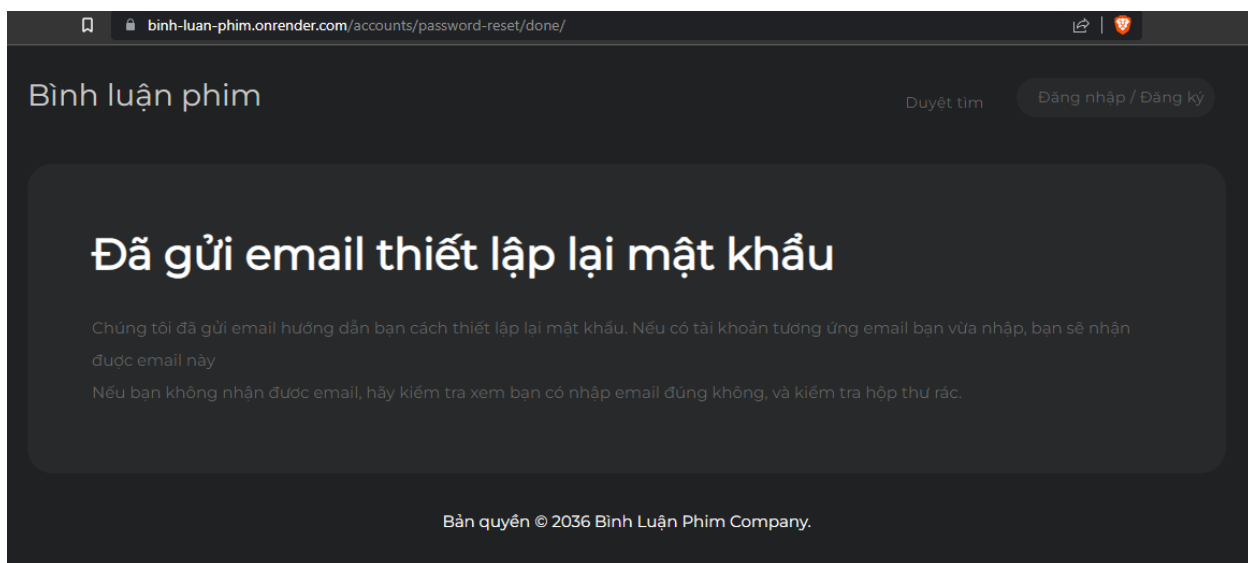
Nhập email tương ứng với tài khoản mà bạn muốn khởi động mật khẩu, chúng tôi sẽ gửi email hướng dẫn cách thiết lập.

Email*

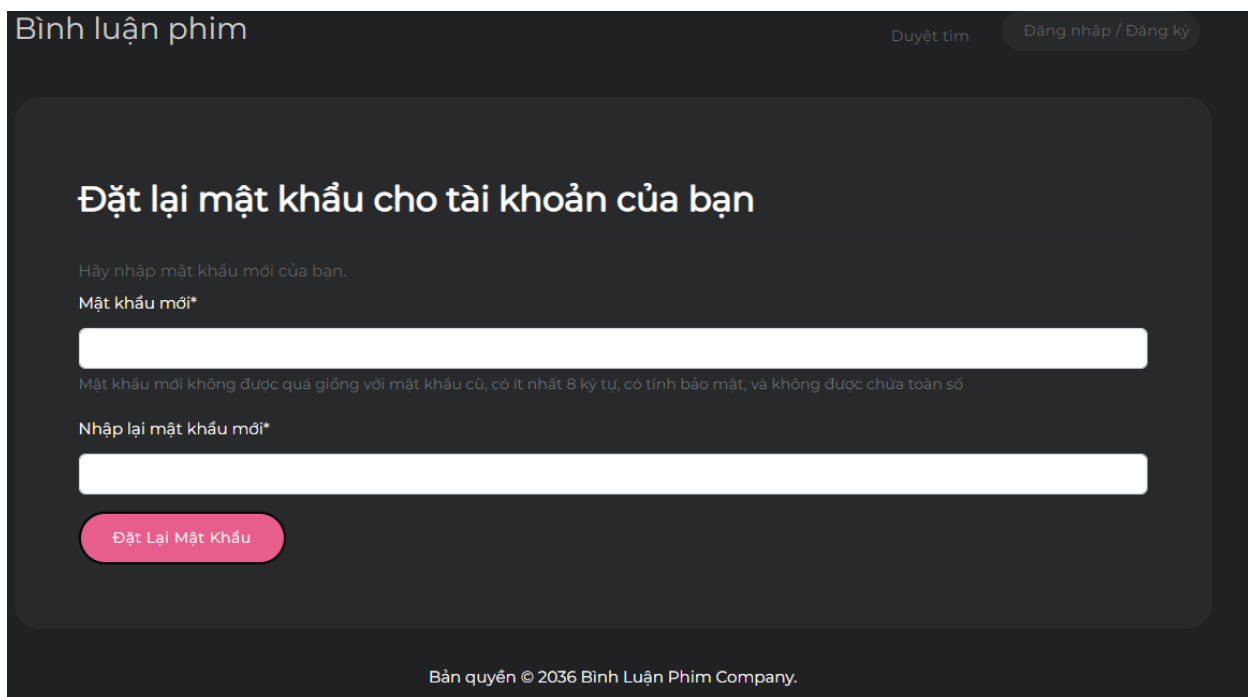
Gửi Link Qua Email

Bản quyền © 2036 Bình Luận Phim Company.

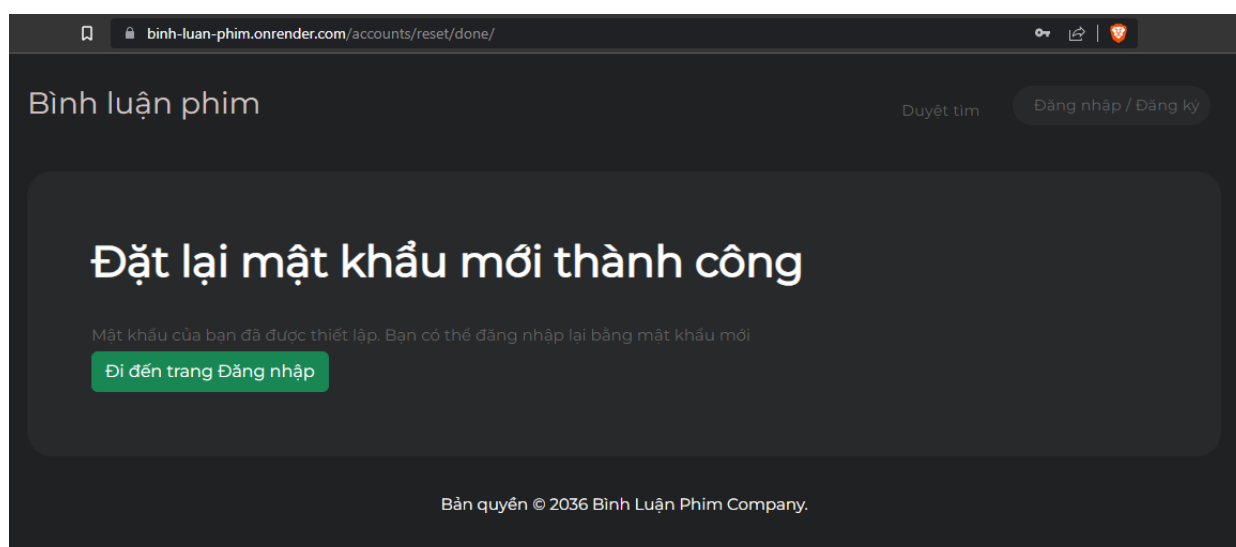
- Trang Xác nhận đã gửi email chứa link đặt lại mật khẩu



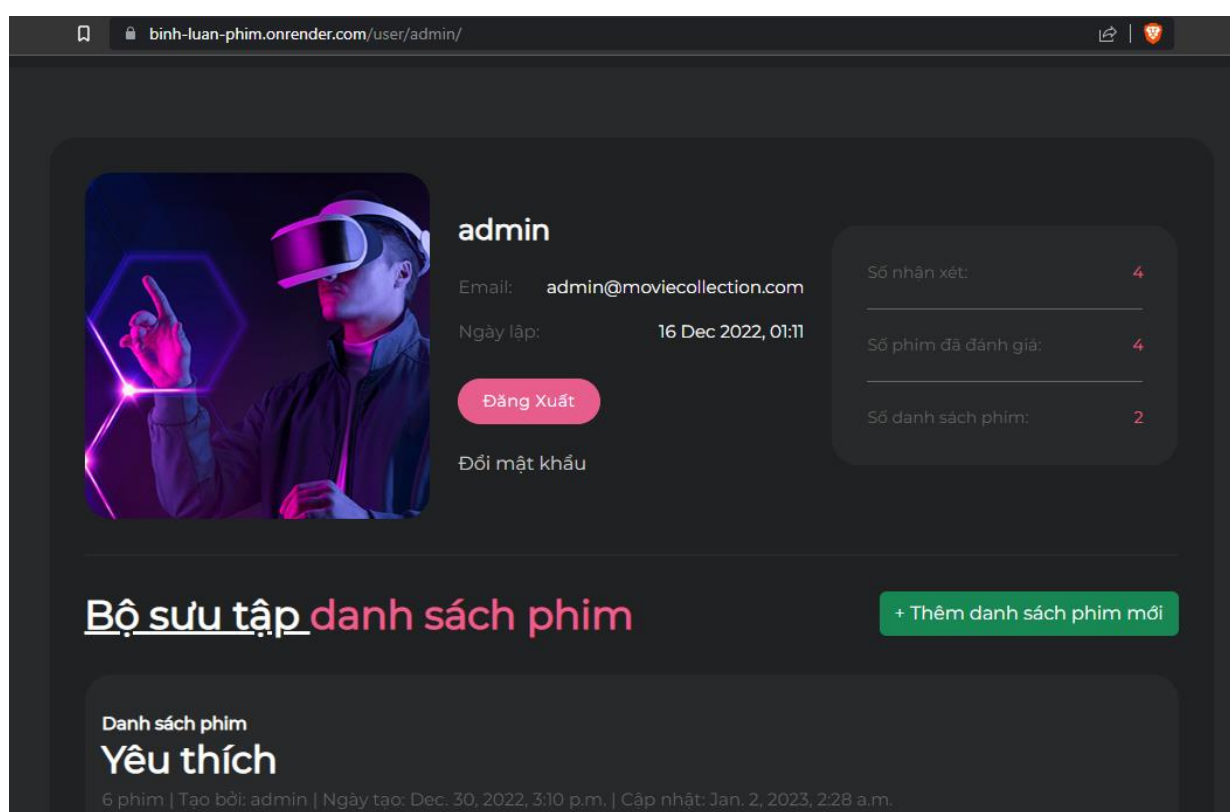
- Trang Nhập lại mật khẩu mới

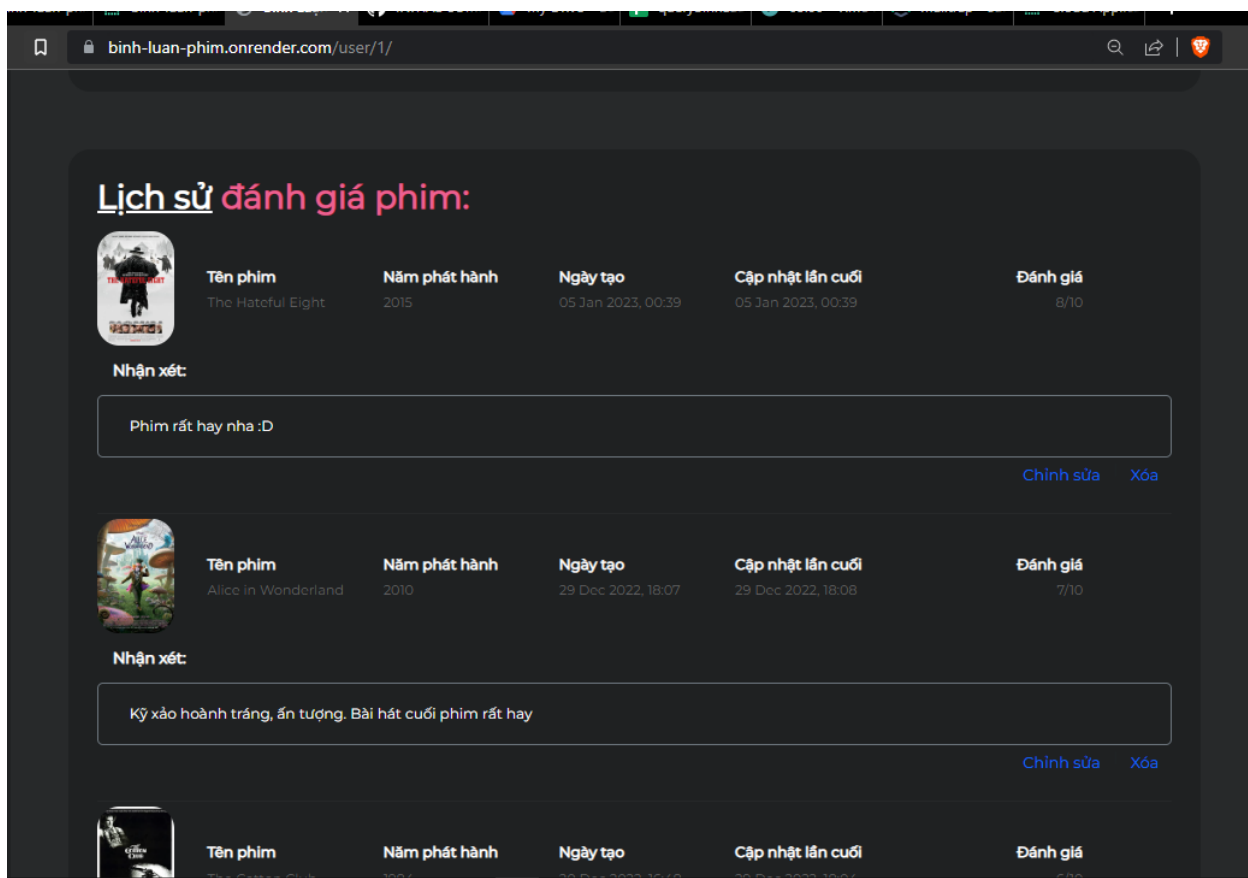


- Trang Hoàn thành quá trình khởi động lại mật khẩu

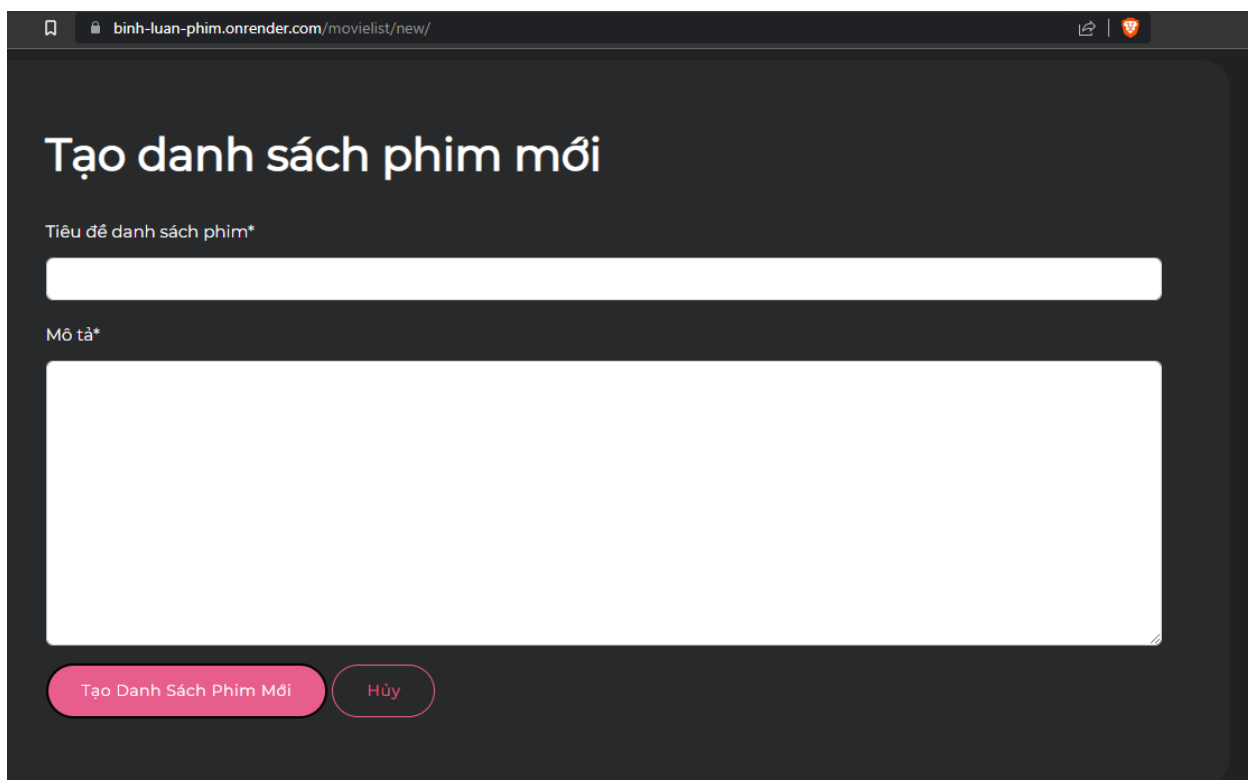


- Trang Tài khoản cá nhân






- Trang Tạo danh sách phim mới



- Trang Chi tiết Danh sách phim



Danh sách phim

Những bộ phim kinh điển

Tạo bởi: admin

Ngày lập: 04 Jan 2023, 00:37

Cập nhật: 04 Jan 2023, 00:37

Số lượng phim: 4

Những bộ phim hay nhất mọi thời đại :D

Xóa danh sách

Chỉnh sửa thông tin

Tìm kiếm phim thêm vào danh sách

Nhập tên phim..

Q

Tìm kiếm phim thêm vào danh sách

a

Q

12 Angry Men

Thêm

Xóa

Alice in Wonderland

Thêm

Xóa

Alive

Thêm

Xóa

Ex Machina

Xóa

Ice Age

Xóa

Bản quyền © 2036 Bình Luận Phim Company.

- Trang Sửa thông tin Danh sách phim

🔖 binh-luan-phim.onrender.com/movielist/6/edit/ 🔗 🔒

Cập nhật danh sách phim

Tiêu đề danh sách phim*

Mô tả*

Cập Nhật Thông Tin Hủy

- Trang Đổi mật khẩu

🔖 binh-luan-phim.onrender.com/accounts/password-change/ 🔗 🔒

Đổi mật khẩu

Mật khẩu hiện tại*

Mật khẩu mới*

Mật khẩu mới không được quá giống với mật khẩu cũ, có ít nhất 8 ký tự, có tính bảo mật, và không được chứa toàn số

Nhập lại mật khẩu mới*

Đổi Mật Khẩu

Bản quyền © 2036 Binh Luận Phim Company.

- Trang Admin

Site administration

AUTHENTICATION AND AUTHORIZATION

Groups [+ Add](#) [Change](#)

Users [+ Add](#) [Change](#)

MOVIES

Genres [+ Add](#) [Change](#)

Movie lists [+ Add](#) [Change](#)

Movies [+ Add](#) [Change](#)

Reviews [+ Add](#) [Change](#)

Recent actions

My actions

[vunt](#)

User

[vunt](#)

User

[Chính kịch](#)

Genre

[Nhạc kịch](#)

Genre

[Viễn tưởng](#)

Genre

[Hình sự](#)

Genre

[Hài hước](#)

Genre

[+ admin reviews the movie Beetlejuice](#)

Rating: 8 Comment: Phim ổn. Diễn

xuất đạt

Review

[+ The Cotton Club](#)

Movie

[Beetlejuice](#)

PHỤ LỤC 2: DOCSTRING

Class IndexView

Class IndexView (generic.ListView)	Màn hình trang chủ, kế thừa từ ListView có sẵn của Django - Trong trang chủ có hiển thị danh sách phim mới phát hành
get_queryset(self)	Lấy thông tin các phim ra mắt gần đây nhất để hiển thị trên trang chủ - Input: không có - Output: 12 phim ra mắt gần đây nhất xếp theo năm phát hành

Class DetailView

class DetailView (generic.DetailView)	Màn hình trang Chi tiết thông tin phim, kế thừa từ DetailView - Hiển thị đầy đủ thông tin phim - Hiển thị nút yêu thích (Thêm vào Danh sách yêu thích) - Hiển thị form viết bình luận - Hiển thị các bình luận phim từ người dùng
get_context_data (self, **kwargs)	Bổ sung context data cho trang: thêm giá trị favorite_list chứa các phim yêu thích của người dùng (trong trường hợp đã đăng nhập) để xác định người dùng đã thêm phim hiện tại vào Danh Sách yêu thích chưa, thêm giá trị reviews để sắp xếp kết quả lọc phim theo ngày đăng lùi dần và nếu đã đăng nhập thì ưu tiên bài đăng của người đó lên đầu - Input: Thông tin user từ request - Output: Context data có chứa giá trị favorite_list (trong trường hợp đã đăng nhập) và reviews
get(self, request, *args, **kwargs)	Bổ sung xử lý GET request từ trang, thêm chức năng lấy thông tin Danh sách phim - Input: Request từ trang Chi tiết Danh sách phim - Output:

	<p>+ Trường hợp là request có operation là "get_movielists_from_username" gửi bằng AJAX thì xử lý và trả Response có context dạng json gồm các giá trị:</p> <ul style="list-style-type: none"> . Danh sách các Danh sách phim (movielists) . Số lượng phim trong danh sách đó (movielistscount) <p>+ Ngược lại thì theo phương thức get mặc định của DetailView</p>
post (self, request, pk)	<p>Xử lý POST request thêm phim vào Danh sách Yêu thích / Danh sách phim của người dùng</p> <ul style="list-style-type: none"> - Input: Thông tin user từ request, id phim từ URL, id của Danh sách phim (trong trường hợp thêm danh sách phim) - Output: response dạng json gồm: <p>+ Giá trị added thông báo đã thực hiện thêm vào danh sách thành công,</p> <p>+ Giá trị movie_id trả về id của phim đã thêm vào</p>
send_review (request, movie_id)	<p>Xử lý request gửi form đánh giá phim của người dùng</p> <p>Yêu cầu người dùng đã đăng nhập mới có thể đăng đánh giá</p> <ul style="list-style-type: none"> - Input: <p>+ Giá trị rating từ form</p> <p>+ Giá trị comment từ form</p> <p>+ Giá trị id phim được đánh giá, lấy từ tham số URL</p> <ul style="list-style-type: none"> - Output: <p>+ Trường hợp người dùng viết thiếu mục thì trả về trang Chi tiết phim kèm thông báo lỗi</p> <p>+ Trường hợp viết đủ mục thì lưu thông tin bài đánh giá và trả về trang Chi tiết phim kèm thông báo đã gửi bài đánh giá thành công</p>
delete_review (request, review_id=None)	<p>Xử lý request xóa bài đánh giá phim của người dùng</p> <p>Yêu cầu người dùng đã đăng nhập mới có thể xóa bài đánh giá</p> <ul style="list-style-type: none"> - Input: <p>+ Request</p> <p>+ Giá trị id của bài đánh giá, lấy từ tham số URL</p>

	<ul style="list-style-type: none"> - Output: + Thông báo kết quả xóa bài đánh giá + Điều hướng qua trang Chi tiết phim của phim bị xóa
--	---

Class ReviewUpdateView

class ReviewUpdateView (UpdateView)	Trang Cập nhật Đánh giá phim, được kế thừa từ UpdateView của django
get(self, request, *args, **kwargs)	<p>Kiểm tra người dùng có phải chủ nhân bài đánh giá để tiến hành chỉnh sửa</p> <ul style="list-style-type: none"> - Input: request - Output: <p>Trường hợp không phải là do người dùng viết sẽ thông báo chỉ có thể sửa bài đăng của chính mình và điều hướng về trang Chi tiết phim</p> <p>Trường hợp đúng là người viết yêu cầu thì trả response hiển thị form chỉnh sửa</p>
form_valid(self, form)	<p>Thêm thông tin cho form instance được gửi đến trước khi lưu vào cơ sở dữ liệu</p> <ul style="list-style-type: none"> - Input: kết quả form cập nhật Bình luận phim - Output: Phương thức mặc định form_valid mà ReviewUpdateView được kế thừa từ UpdateView (sau khi tiến hành thêm thông tin cho form)
get_success_url(self)	<p>Xử lý và điều hướng trang khi lưu dữ liệu thành công</p> <ul style="list-style-type: none"> - Input: không có - Output: Trả về trang Chi tiết phim (tham số url chứa id phim có đánh giá phim đã được cập nhật)

Class MovielistCreateView

class MovielistCreateView (CreateView)	Trang tạo Danh sách phim mới, được kế thừa từ CreateView của django.views.generic.edit
---	--

get_context_data (self, **kwargs)	Bổ sung context data cho trang: thêm giá trị submitButtonLabel, sectionTitle để chỉnh sửa nhãn cho giao diện - Input: Không có - Output: Context data có chứa giá trị submitButtonLabel, sectionTitle
form_valid(self, form)	Thêm thông tin cho form instance được gửi đến trước khi lưu vào cơ sở dữ liệu - Input: kết quả form tạo Danh sách phim mới - Output: Phương thức mặc định form_valid mà MovielistCreateView được kế thừa từ CreateView (sau khi tiến hành thêm thông tin cho form)
get_success_url(self)	Xử lý và điều hướng trang khi lưu dữ liệu thành công - Input: không có - Output: Trả về trang Chi tiết Danh sách phim vừa tạo (tham số url chứa id của Danh sách phim mới)
get_success_url(self)	Xử lý và điều hướng trang khi lưu dữ liệu thành công - Input: không có - Output: Trả về trang Chi tiết phim (tham số url chứa id phim có đánh giá phim đã được cập nhật)

Class MovielistDetailView

class MovielistDetailView (generic.DetailView)	Trang Chi tiết Danh sách phim, được kế thừa từ DetailView của django
get(self, request, *args, **kwargs)	Bổ sung xử lý GET request từ trang, dùng cho chức năng tìm kiếm phim trực tiếp (live search) để tìm kiếm và thêm phim vào Danh sách phim - Input: Request từ trang Chi tiết Danh sách phim - Output: + Trường hợp là request có operation là "live_search_movies" gửi từ AJAX thì xử lý và trả Response có context đang json gồm các giá trị:

	<ul style="list-style-type: none"> . Danh sách Kết quả tìm kiếm phim (movies_live_result) . Số lượng phim trong danh sách đó(moviescount) . Từ khóa tìm kiếm (searchtext) + Ngược lại thì theo phương thức get mặc định của MovielistDetailView
post(self, request, pk)	<p>Xử lý POST request từ trang</p> <ul style="list-style-type: none"> - Input: <ul style="list-style-type: none"> + Request từ trang Chi tiết Danh sách phim + id của Danh sách phim, lấy từ tham số url (pk) - Output: Thực hiện một trong 3 chức năng: <ul style="list-style-type: none"> + Xóa phim khỏi danh sách phim(phương thức delete_movie_in_list) + Xóa danh sách phim(phương thức delete_movielist) + Thêm phim danh sách phim(phương thức add_movie_to_this_list)
delete_movie_in_list(self, request, pk)	<p>Thực hiện chức năng Xóa phim khỏi danh sách phim</p> <ul style="list-style-type: none"> - Input: <ul style="list-style-type: none"> + Request từ trang Chi tiết Danh sách phim + Id của Danh sách phim, lấy từ tham số url (pk) - Output: <ul style="list-style-type: none"> + Trường hợp giá trị operation nhận từ AJAX POST request là "delete_movie_in_list" thì thực hiện xoá phim khỏi danh sách phim trên cơ sở dữ liệu, trả về response có context chứa thông tin thông báo kết quả xóa, id của phim bị xóa, id của danh sách phim liên quan + Ngược lại thì không làm gì
delete_movielist(self, request, pk)	<p>Thực hiện chức năng Xóa danh sách phim</p> <ul style="list-style-type: none"> - Input: <ul style="list-style-type: none"> + Request từ trang Chi tiết Danh sách phim + id của Danh sách phim, lấy từ tham số url (pk)

	<p>- Output:</p> <p>+ Trường hợp giá trị operation nhận từ AJAX POST request là "delete_movielist" thì thực hiện xóa danh sách phim trên cơ sở dữ liệu, thông báo kết quả xóa (thành công / không thành công)</p> <p>trả về response có context chứa thông tin link điều hướng về trang tài khoản cá nhân của người dùng</p> <p>+ Ngược lại thì không làm gì</p>
add_movie_to_this_list(self, request, pk):	<p>Thực hiện chức năng Thêm phim vào danh sách phim</p> <p>- Input:</p> <p>+ Request từ trang Chi tiết Danh sách phim</p> <p>+ id của Danh sách phim, lấy từ tham số url (pk)</p> <p>- Output:</p> <p>+ Trường hợp giá trị operation nhận từ AJAX POST request là "add_movie_to_this_list_submit" thì thực hiện thêm phim vào danh sách phim trên cơ sở dữ liệu, trả về response có context chứa thông tin</p> <p>kết quả thêm phim vào danh sách, thông tin của phim được thêm vào, id của danh sách phim liên quan</p> <p>+ Ngược lại thì không làm gì</p>

Class MovielistDetailView

class MovielistUpdateView (UpdateView)	Trang Cập nhật Danh sách phim, được kế thừa từ UpdateView của django
get_context_data(self, **kwargs)	<p>Bổ sung context data cho trang: thêm giá trị submitButtonLabel, sectionTitle để chỉnh sửa nhãn cho giao diện</p> <p>- Input: Không có</p> <p>- Output: Context data có chứa giá trị submitButtonLabel, sectionTitle</p>

form_valid(self, form)	<p>Thêm thông tin cho form instance được gửi đến trước khi lưu vào cơ sở dữ liệu</p> <ul style="list-style-type: none"> - Input: kết quả form cập nhật Danh sách phim - Output: Hàm mặc định form_valid mà MovielistUpdateView được kế thừa từ UpdateView (sau khi tiến hành thêm thông tin cho form)
get_success_url(self)	<p>Xử lý và điều hướng trang khi lưu dữ liệu thành công</p> <ul style="list-style-type: none"> - Input: không có - Output: Trả về trang Chi tiết Danh sách phim (tham số url chứa id của Danh sách phim đã được cập nhật)

Class NewUserForm

Class NewUserForm (UserCreationForm)	Thiết lập form đăng ký cho website, chỉnh sửa nhãn, thêm trường email
__init__(self, *args, **kwargs)	<p>Chỉnh sửa nhãn cho form</p> <ul style="list-style-type: none"> - Input: không có - Output: form có nhãn và help text được chỉnh sửa
save(self, commit=True)	<p>Bổ sung thêm thông tin email từ form trước khi lưu dữ liệu User</p> <ul style="list-style-type: none"> - Input: không có - Output: user có chứa thông tin email

Class LoginForm

Class LoginForm (AuthenticationForm)	Thiết lập form đăng nhập cho website, chỉnh sửa nhãn, thêm trường remember_me
__init__(self, *args, **kwargs)	<p>Sửa lại nhãn cho các trường trong form đăng nhập</p> <ul style="list-style-type: none"> - Input: không có - Output: form có nhãn được chỉnh sửa

Class UpdatedLoginForm

class UpdatedLoginForm (LoginForm)	Kế thừa từ LoginForm và thêm tính năng ghi nhớ đăng nhập
form_valid(self, form)	Xử lý tính năng Ghi nhớ đăng nhập - Input: form đăng nhập - Output: Khi gửi form bổ sung thêm bước Nếu không có remember me thì thiết lập session cookie sẽ hết hạn khi đóng trình duyệt.

class PasswordChangeForm

class PasswordChangeForm (PasswordChangeForm)	Chỉnh sửa các nhãn
__init__(self, *args, **kwargs)	Chỉnh sửa các nhãn - Input: không có - Output: form có nhãn và help text được chỉnh sửa

class SetPasswordForm

class SetPasswordForm (SetPasswordForm)	Chỉnh sửa các nhãn
__init__(self, *args, **kwargs)	Chỉnh sửa các nhãn - Input: không có - Output: form có nhãn và help text được chỉnh sửa

Class ReviewUpdateForm

Class ReviewUpdateForm (forms.ModelForm)	Chỉnh sửa các nhãn, thay đổi loại input của comment là textarea
---	---

Class MovieListForm

Class MovieListForm (forms.ModelForm)	Chỉnh sửa các nhãn, thay đổi loại input của description là textarea
--	---

Class MovieAdmin

Class MovieAdmin (admin.ModelAdmin)	Thiết lập giao diện trang admin và các mục cần chỉnh sửa. Dựa trên <code>django.contrib.admin</code> của Django
--	---

Các phương thức khác được sử dụng

<code>browse_view(request)</code>	Trang duyệt tìm phim - Input: Dựa trên Query string từ GET request, gồm các giá trị <code>genre</code> , <code>year</code> , <code>runtime</code> , <code>keyword</code> , <code>page</code> (dùng để phân trang) - Output: Trang duyệt tìm chứa context gồm: + Các tùy chọn trong <code>genres</code> , <code>year</code> , <code>runtime</code> + <code>filtered_movies</code> : chứa các bộ phim đã lọc theo query string + <code>url</code> : chứa GET request của trang đã yêu cầu (dùng để phân trang)
<code>display_filter_fields()</code>	Hỗ trợ hiển thị trang Duyệt tìm(browse), tạo giá trị cho các trường chọn - Input: Không có - Output: dict chứa các giá trị <code>genres</code> , <code>year</code> , <code>runtime</code>
<code>profile_view</code> <code>(request, user_id)</code>	Trang tài khoản cá nhân - Input: <code>user_username</code> của tài khoản cá nhân, dựa trên tham số URL - Output: Trang tài khoản cá nhân chứa context gồm: + Thông tin chủ trang tài khoản cá nhân đó (<code>user</code>) + Các bài đánh giá mà tài khoản đó đã tạo (<code>reviews</code>) + Số lượng các bài đánh giá (<code>reviews_count</code>) + Số lượng các bộ phim đã đánh giá (<code>reviews_count</code>) + Các danh sách phim mà tài khoản đó đã tạo (<code>movielists</code>)
<code>signup_request(request)</code>	Xử lý request liên quan đến trang Đăng Ký (tạo tài khoản) - Input: request từ người dùng

	<p>- Output:</p> <p>+ Nếu request POST và form hợp lệ thì tiến hành lưu thông tin và đăng nhập bằng tài khoản vừa tạo, điều hướng về trang chủ</p> <p>+ Nếu request POST và form không hợp lệ thì thông báo tạo tài khoản thất bại và đi đến trang đăng ký (có chứa thông tin chứa hợp lệ người dùng đã điền trước đó)</p> <p>+ Nếu request GET thì trả về trang đăng ký chứa form đăng ký trống</p>
login_request(request)	<p>Xử lý request liên quan đến trang Đăng Nhập</p> <p>- Input: request từ người dùng</p> <p>- Output:</p> <p>+ Nếu request POST và form hợp lệ thì tiến hành xác thực</p> <p>. Nếu xác thực đúng thì tiến hành đăng nhập, thông báo đã đăng nhập thành công, điều hướng về trang chủ</p> <p>. Nếu xác thực sai thì thông báo Tên người dùng hay Mật khẩu không hợp lệ, đi đến trang đăng nhập</p> <p>+ Nếu request POST và form không hợp lệ thì thông báo Bạn cần điền đầy đủ thông tin, đi đến trang đăng nhập</p> <p>+ Nếu request GET thì đi đến trang đăng nhập</p>
logout_request(request)	<p>Xử lý request đăng xuất</p> <p>- Input: request từ người dùng</p> <p>- Output:</p> <p>+ Thực hiện đăng xuất, thông báo đã đăng xuất thành công,</p> <p>+ Điều hướng đến trang đăng nhập</p>
password_change(request)	<p>Xử lý request liên quan đến trang Đổi mật khẩu</p> <p>- Input: request từ người dùng</p> <p>- Output:</p> <p>+ Nếu request POST và form hợp lệ thì:</p> <p>. Lưu thông tin mật khẩu mới, thông báo đổi mật khẩu thành công</p>

	<p>. Điều hướng đến trang Đăng nhập</p> <p>+ Nếu request POST và form không hợp lệ thì thông báo lỗi và đi đến trang Đổi mật khẩu</p> <p>+ Nếu request GET thì đi đến trang Đổi mật khẩu</p>
password_reset_request (request)	<p>Xử lý request liên quan đến trang Khởi động lại mật khẩu</p> <p>- Input: request từ người dùng</p> <p>- Output:</p> <p>+ Nếu request POST và form hợp lệ thì tiến hành lấy email đối chiếu với email của người dùng trong cơ sở dữ liệu, nếu có thì tiến hành gửi thư đến email đó để thực hiện Khởi động lại mật khẩu. Sau khi gửi xong đi đến trang /accounts/password-reset/done/</p> <p>+ Nếu request POST và form không hợp lệ hoặc không tìm thấy người dùng khớp thì thông báo lỗi và tải trang Khởi động mật khẩu</p> <p>+ Nếu request GET thì đi đến trang Khởi động mật khẩu</p>