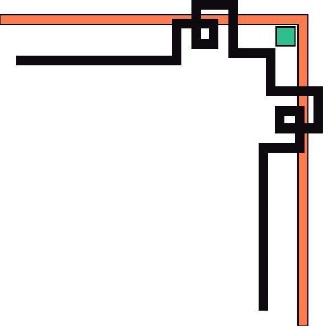
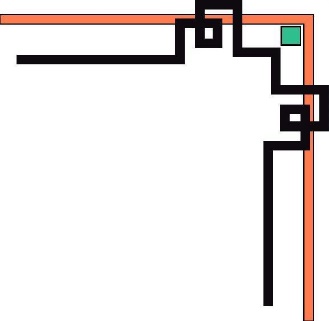
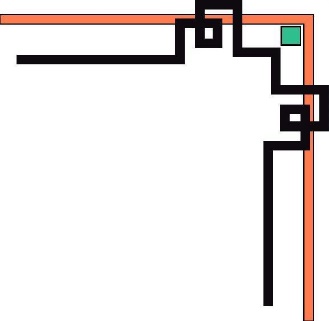
BỘ GIÁO DỤC VÀ ĐÀO TẠO



**TRƯỜNG ĐẠI HỌC NHA TRANG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÀI BÁO CÁO**

**PHÁT TRIỂN ỨNG DỤNG WEB 1**

**XÂY DỰNG WEB XEM PHIM FPT CLONE**

**Giảng viên hướng dẫn: Bùi Chí Thành**

**Sinh viên thực hiện: Nguyễn Xuân Đạt**

**Mã số sinh viên: 64130299**

**Lớp: 64.CNTT-CLC2**

Khánh Hòa, năm 2025

Mục Lục

[**I.** **Giới thiệu** 4](#_Toc187272566)

[**1.** **Mục tiêu của dự án** 4](#_Toc187272567)

[**2.** **Lý do chọn đề tài** 4](#_Toc187272568)

[**3.** **Giới hạn và phạm vi của ứng dụng** 4](#_Toc187272569)

[**II.** **Cơ sở lý thuyết liên quan** 6](#_Toc187272570)

[**1.** **Mô hình MVC trong ASP.NET** 6](#_Toc187272571)

[**2.** **Công nghệ sử dụng:** 7](#_Toc187272572)

[**2.1.** **ASP.NET MVC** 7](#_Toc187272573)

[**2.2.** **SQL Server** 7](#_Toc187272574)

[**2.3.** **Bootstrap** 8](#_Toc187272575)

[**2.4.** **Tailwind CSS** 9](#_Toc187272576)

[**2.5.** **JavaScript** 10](#_Toc187272577)

[**III.** **Khảo sát hiện trạng** 11](#_Toc187272578)

[**1.** **Phân tích các nền tảng xem phim hiện tại** 11](#_Toc187272579)

[**2.** **Ưu điểm và hạn chế của các nền tảng tham khảo** 11](#_Toc187272580)

[**3.** **Tóm tắt yêu cầu người dùng đối với ứng dụng xem phim** 12](#_Toc187272581)

[**IV.** **Phân tích hệ thống** 13](#_Toc187272582)

[**1.** **Mục tiêu của hệ thống** 13](#_Toc187272583)

[**2.** **Các yêu cầu cơ bản của hệ thống** 13](#_Toc187272584)

[**3.** **Yêu cầu chức năng** 13](#_Toc187272585)

[**4.** **Yêu cầu phi chức năng** 14](#_Toc187272586)

[**5.** **Quy trình hoạt động của hệ thống** 14](#_Toc187272587)

[**V.** **Thiết kế hệ thống** 16](#_Toc187272588)

[**1.** **Mô hình kiến trúc hệ thống** 16](#_Toc187272589)

[**2.** **Thiết kế và mô tả mô hình dữ liệu vật lý** 16](#_Toc187272590)

[**3.** **Thiết kế giao diện người dùng (UI/UX)** 19](#_Toc187272591)

[**VI.** **Cài đặt và triển khai ứng dụng** 20](#_Toc187272592)

[**1.** **Môi trường triển khai** 20](#_Toc187272593)

[**2.** **Sơ đồ giao diện phần mềm** 20](#_Toc187272594)

[**3.** **Các giao diện chính và diễn giải chức năng** 20](#_Toc187272595)

[**4.** **Giao diện đăng nhập, đăng kí** 20](#_Toc187272596)

[**5.** **Giao diện trang chủ** 21](#_Toc187272597)

[**6.** **Giao diện chi tiết phim** 21](#_Toc187272598)

[**7.** **Giao diện lọc phim** 21](#_Toc187272599)

[**8.** **Giao diện mua gói** 22](#_Toc187272600)

[**9.** **Giao diện đăng nhập trang quản trị** 22](#_Toc187272601)

[**10.** **Giao diện quản trị (Quản lý)** 22](#_Toc187272602)

[**11.** **Giao diện quản trị (Nhân Viên)** 23](#_Toc187272603)

[**VII.** **Kiểm thử ứng dụng** 24](#_Toc187272604)

[**1.** **Phương pháp kiểm thử** 24](#_Toc187272605)

[**2.** **Kết quả kiểm thử** 24](#_Toc187272606)

[**3.** **Tài khoản demo người quản lý** 24](#_Toc187272607)

[**VIII.** **Kết luận và đánh giá** 25](#_Toc187272608)

[**1.** **Nhận xét về ưu điểm của ứng dụng** 25](#_Toc187272609)

[**2.** **Hạn chế của ứng dụng** 25](#_Toc187272610)

[**3.** **Đề xuất và kiến nghị cho phát triển trong tương lai** 25](#_Toc187272611)

1. **Giới thiệu**
2. **Mục tiêu của dự án**

Dự án "Xây dựng Web Xem Phim FPTClone" nhằm tạo ra một nền tảng xem phim trực tuyến được lấy cảm hứng từ FPT Play, mang lại trải nghiệm người dùng tương tự nhưng được tối ưu hóa với các tính năng mới. Mục tiêu cụ thể của dự án bao gồm:

* Cung cấp trải nghiệm người dùng tốt: Thiết kế giao diện thân thiện, dễ sử dụng và tương thích với nhiều thiết bị.
* Hỗ trợ các chức năng cơ bản: Bao gồm tìm kiếm phim, xem chi tiết phim, phân loại theo thể loại và năm phát hành.
* Tích hợp quản lý hiệu quả: Xây dựng giao diện quản trị để quản lý dữ liệu phim, người dùng và danh mục.
* Đảm bảo tính ổn định và hiệu suất: Ứng dụng phải hoạt động nhanh, có tính bảo mật cao và khả năng mở rộng linh hoạt.

1. **Lý do chọn đề tài**

* Xu hướng phát triển: Với sự bùng nổ của các nền tảng xem phim trực tuyến như Netflix, FPT Play và VieON, nhu cầu giải trí qua internet ngày càng trở nên phổ biến.
* Ứng dụng thực tiễn: Dự án này là cơ hội để áp dụng các kiến thức về lập trình Fullstack trong ASP.NET MVC và học cách xây dựng một ứng dụng web hoàn chỉnh.
* Thị trường tiềm năng: Xây dựng một nền tảng giải trí tương tự FPT Play nhưng đơn giản hơn, nhằm phục vụ nhu cầu học tập và thực hành phát triển phần mềm.

1. **Giới hạn và phạm vi của ứng dụng**

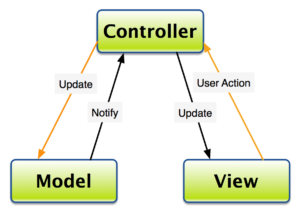
* **Giới hạn**:
  + Ứng dụng được xây dựng để phục vụ mục đích học tập, không mang tính thương mại.
  + Dữ liệu phim được lấy từ nguồn mở (phim.nguonc.com), không cập nhật nội dung độc quyền hoặc có bản quyền cao cấp.
  + Tính năng nâng cao như gợi ý thông minh, xem offline hoặc phát trực tiếp chưa được tích hợp.
* **Phạm vi:**
  + Người dùng thông thường: Có thể tìm kiếm phim, xem chi tiết phim và thanh toán để xem được những phim cần trả phí.
  + Quản trị viên: Được cung cấp giao diện quản lý để thêm, sửa, xóa, thống kê, báo cáo thông tin phim, danh mục và người dùng.
  + Môi trường triển khai: Chạy trên nền tảng web, sử dụng ASP.NET MVC, kết nối với cơ sở dữ liệu SQL Server và giao diện thân thiện nhờ Bootstrap và Tailwind.

1. **Cơ sở lý thuyết liên quan**
2. **Mô hình MVC trong ASP.NET**

* **Lý thuyết:**

MVC (Model-View-Controller) là một mẫu thiết kế phần mềm phổ biến trong phát triển ứng dụng web, giúp phân tách các thành phần của ứng dụng để dễ dàng quản lý và phát triển. Mẫu thiết kế này bao gồm ba thành phần chính:

* + Model: Quản lý dữ liệu và logic nghiệp vụ của ứng dụng. Model sẽ giao tiếp với cơ sở dữ liệu, thực hiện các thao tác như thêm, sửa, xóa dữ liệu.
  + View: Hiển thị giao diện người dùng. View nhận dữ liệu từ Model và hiển thị thông tin cho người dùng.
  + Controller: Là cầu nối giữa Model và View. Controller nhận yêu cầu từ người dùng (qua View), xử lý logic và truyền dữ liệu từ Model sang View để hiển thị.
* **Vai trò:**
  + Phân tách trách nhiệm: Giúp tách biệt rõ ràng giữa các phần của ứng dụng, giúp việc bảo trì và nâng cấp ứng dụng trở nên dễ dàng hơn.
  + Tái sử dụng mã nguồn: Mỗi thành phần có thể được tái sử dụng hoặc thay đổi mà không ảnh hưởng đến các thành phần khác của ứng dụng.
  + Quản lý dự án hiệu quả: Các nhóm phát triển có thể làm việc độc lập trên các thành phần khác nhau mà không bị ảnh hưởng lẫn nhau.
* **Hình ảnh minh họa:**



Ảnh 1: Mô hình MVC

1. **Công nghệ sử dụng:**
   1. **ASP.NET MVC**

ASP.NET MVC là một framework phát triển ứng dụng web mạnh mẽ của Microsoft, giúp xây dựng ứng dụng theo mô hình MVC (Model-View-Controller). Framework này được thiết kế để phát triển các ứng dụng web động, cung cấp sự linh hoạt trong việc xử lý các yêu cầu của người dùng và tương tác với cơ sở dữ liệu.

* **Ưu điểm trong dự án:**
  + Quản lý route linh hoạt: ASP.NET MVC cho phép định nghĩa các đường dẫn URL cho ứng dụng một cách dễ dàng, hỗ trợ routing theo cách thức dễ hiểu và mở rộng.
  + Hỗ trợ kiểm tra và xác thực dễ dàng: ASP.NET MVC cung cấp các công cụ tích hợp sẵn để thực hiện kiểm tra dữ liệu và xác thực yêu cầu từ người dùng.
  + Cộng đồng và tài liệu phong phú: Với sự hỗ trợ mạnh mẽ từ Microsoft và cộng đồng lập trình viên, ASP.NET MVC giúp việc phát triển ứng dụng trở nên nhanh chóng và hiệu quả.
* **Nhược điểm:**
  + Yêu cầu tài nguyên hệ thống cao: Các ứng dụng sử dụng ASP.NET MVC có thể yêu cầu tài nguyên hệ thống lớn hơn so với một số framework khác, điều này có thể ảnh hưởng đến hiệu suất trong môi trường có tài nguyên hạn chế.
  + Khó khăn khi triển khai trên các nền tảng không phải Windows: Mặc dù ASP.NET MVC ngày càng hỗ trợ tốt hơn trên các nền tảng không phải Windows, nhưng việc triển khai trên môi trường Linux hoặc macOS vẫn có thể gặp khó khăn và yêu cầu các cấu hình đặc biệt.
  + Cần có kiến thức về mô hình MVC: Nếu nhóm phát triển không quen thuộc với mô hình MVC, việc triển khai ban đầu có thể gặp phải một số khó khăn trong việc tổ chức mã nguồn.
  1. **SQL Server**

SQL Server là hệ quản trị cơ sở dữ liệu quan hệ (RDBMS) của Microsoft, được sử dụng để lưu trữ và quản lý dữ liệu trong ứng dụng. Dự án sử dụng SQL Server để lưu trữ thông tin phim, người dùng, các thể loại phim, và các dữ liệu liên quan khác.

* **Ưu điểm trong dự án:**
  + Lưu trữ và truy vấn dữ liệu hiệu quả: SQL Server giúp lưu trữ thông tin về phim, người dùng và các hoạt động liên quan trong cơ sở dữ liệu với khả năng truy vấn và tối ưu hóa dữ liệu nhanh chóng.
  + Tính bảo mật cao: SQL Server cung cấp các tính năng bảo mật mạnh mẽ, bao gồm mã hóa, phân quyền truy cập và kiểm soát người dùng.
  + Tích hợp tốt với ASP.NET MVC: Việc kết nối SQL Server với ASP.NET MVC dễ dàng thông qua ADO.NET hoặc Entity Framework.
* **Nhược điểm:**
  + Chi phí bản quyền cao: SQL Server có chi phí bản quyền khá cao so với các hệ quản trị cơ sở dữ liệu khác như MySQL hoặc PostgreSQL. Điều này có thể là một yếu tố hạn chế đối với các dự án nhỏ hoặc có ngân sách hạn chế.
  + Tính phức tạp trong quản lý: Việc cài đặt và quản lý SQL Server có thể phức tạp đối với những người không quen thuộc với các công cụ và kỹ thuật của Microsoft. Cần có đội ngũ quản trị cơ sở dữ liệu chuyên nghiệp để tối ưu hóa và duy trì hiệu suất hệ thống.
  1. **Bootstrap**

Bootstrap là một framework CSS phổ biến giúp xây dựng giao diện người dùng đẹp mắt và phản hồi nhanh (responsive) trên tất cả các thiết bị. Bootstrap cung cấp các thành phần giao diện có sẵn như nút, bảng, form, modal, carousel, và nhiều thành phần khác.

* **Ưu điểm trong dự án:**
  + Giao diện hiện đại và dễ sử dụng: Bootstrap cung cấp các lớp CSS sẵn có giúp tạo ra giao diện đẹp và thân thiện với người dùng mà không cần viết nhiều mã CSS.
  + Responsive Design: Các thành phần giao diện của Bootstrap sẽ tự động điều chỉnh để phù hợp với mọi kích thước màn hình, giúp ứng dụng hoạt động mượt mà trên các thiết bị di động và máy tính để bàn.
  + Dễ dàng mở rộng và tùy chỉnh: Bootstrap dễ dàng tích hợp với các công nghệ khác và có thể được tùy chỉnh thêm qua các file CSS và JavaScript.
* **Nhược điểm:**
  + Thiết kế giao diện giống nhau: Dù Bootstrap cung cấp giao diện đẹp và dễ sử dụng, nhưng nếu không tùy chỉnh đúng cách, nhiều trang web sử dụng Bootstrap có thể trông giống nhau, thiếu tính độc đáo và sáng tạo.
  + Tính năng nặng: Một số thành phần của Bootstrap có thể làm cho ứng dụng trở nên nặng nề, đặc biệt khi sử dụng quá nhiều thành phần CSS và JavaScript. Điều này có thể ảnh hưởng đến tốc độ tải trang.
  + Khả năng tùy chỉnh giới hạn: Dù Bootstrap dễ dàng tùy chỉnh, nhưng nếu bạn muốn tạo một giao diện hoàn toàn khác biệt, việc sử dụng Bootstrap có thể gặp phải một số hạn chế và đòi hỏi nhiều thời gian để chỉnh sửa CSS.
  1. **Tailwind CSS**

Tailwind CSS là một framework CSS tiện dụng, giúp tạo ra giao diện người dùng một cách nhanh chóng bằng cách sử dụng các lớp CSS tùy chỉnh (utility classes). Nó khác biệt với các framework như Bootstrap ở chỗ Tailwind không đi kèm với các thành phần giao diện sẵn có mà thay vào đó cho phép người dùng xây dựng giao diện bằng cách kết hợp các lớp CSS có sẵn.

* **Ưu điểm trong dự án:**
  + Tùy chỉnh linh hoạt: Tailwind CSS cho phép tùy chỉnh dễ dàng, giúp tạo ra giao diện độc đáo mà không phải thay đổi nhiều mã nguồn. Bạn có thể tạo ra các thiết kế phức tạp chỉ với các lớp CSS cơ bản.
  + Tối ưu hóa mã CSS: Tailwind CSS giúp giảm thiểu dung lượng CSS cuối cùng bằng cách chỉ giữ lại các lớp CSS được sử dụng trong dự án, giúp tăng tốc độ tải trang.
  + Thiết kế responsive dễ dàng: Tailwind CSS hỗ trợ tính năng responsive tốt, cho phép bạn dễ dàng thiết kế các giao diện thích ứng với nhiều kích thước màn hình khác nhau.
* **Nhược điểm:**
  + Dễ dẫn đến mã nguồn lộn xộn: Việc sử dụng quá nhiều lớp CSS trong HTML có thể khiến mã nguồn trở nên dài và khó duy trì, đặc biệt trong các dự án lớn.
  + Học curve steeper: Để sử dụng thành thạo Tailwind CSS, người lập trình viên cần phải hiểu rõ cách thức hoạt động của các lớp utility và cách cấu hình dự án để tận dụng tối đa các tính năng của Tailwind.
  + Không có thành phần giao diện sẵn có: Mặc dù sự linh hoạt cao, nhưng Tailwind không cung cấp sẵn các thành phần giao diện như nút, bảng, form, mà người dùng phải tự thiết kế từ các lớp cơ bản. Điều này có thể tốn thời gian cho những dự án yêu cầu giao diện phức tạp.
  1. **JavaScript**

JavaScript là ngôn ngữ lập trình được sử dụng để tạo ra các tính năng động và tương tác trên trang web. JavaScript giúp ứng dụng của bạn phản hồi nhanh chóng với người dùng mà không cần tải lại trang, như trong các tính năng tìm kiếm phim, thêm phim vào danh sách yêu thích, và chuyển đổi trang web.

* **Ưu điểm trong dự án:**
  + Tương tác động: JavaScript cho phép thực hiện các hành động trực tiếp trên giao diện người dùng mà không cần tải lại trang, giúp cải thiện trải nghiệm người dùng.
  + Khả năng tương thích cao: JavaScript được hỗ trợ trên tất cả các trình duyệt hiện đại, giúp ứng dụng của bạn chạy ổn định trên nhiều nền tảng khác nhau.
* **Nhược điểm:**
  + Tính bảo mật: JavaScript có thể trở thành mục tiêu của các cuộc tấn công bảo mật, đặc biệt là các cuộc tấn công Cross-Site Scripting (XSS). Do đó, cần cẩn thận khi xử lý dữ liệu từ người dùng.
  + Chạy không đồng bộ: Các chức năng trong JavaScript có thể chạy không đồng bộ, dẫn đến việc xử lý dữ liệu và các yêu cầu của người dùng không đồng bộ, có thể gây khó khăn trong việc đồng bộ hóa giao diện người dùng với dữ liệu thực tế.
  + Khó kiểm tra lỗi: Việc kiểm tra và xử lý lỗi trong JavaScript có thể gặp phải một số vấn đề do tính động của ngôn ngữ. Điều này có thể khiến việc gỡ lỗi trở nên phức tạp, đặc biệt đối với các ứng dụng lớn.

1. **Khảo sát hiện trạng**
2. **Phân tích các nền tảng xem phim hiện tại**

Các nền tảng xem phim hiện nay như Netflix, YouTube, Amazon Prime Video, FPT Play, và Zing TV, đều sử dụng mô hình ứng dụng web và di động để cung cấp dịch vụ xem phim trực tuyến. Những nền tảng này đều có những tính năng cơ bản như tìm kiếm, phân loại phim theo thể loại, bộ phim yêu thích, và khả năng đánh giá phim. Các dịch vụ này còn cung cấp tính năng phát trực tuyến với độ phân giải cao, hỗ trợ nhiều thiết bị truy cập và có giao diện người dùng tối ưu.

* Netflix: Dịch vụ phát video trực tuyến hàng đầu, cung cấp kho nội dung phong phú với các bộ phim gốc và bản quyền. Giao diện của Netflix được thiết kế đơn giản và thân thiện, dễ sử dụng trên cả điện thoại và máy tính.
* YouTube: Ngoài việc là nền tảng video phổ biến, YouTube cũng cung cấp các bộ phim và chương trình truyền hình có bản quyền dưới dạng cho thuê hoặc mua.
* FPT Play: Dịch vụ của FPT cung cấp nhiều thể loại phim, bao gồm phim chiếu rạp, truyền hình, thể thao và các chương trình truyền hình trực tiếp. Giao diện thân thiện, dễ dàng tìm kiếm.

1. **Ưu điểm và hạn chế của các nền tảng tham khảo**

* **Ưu điểm:**
  + Kho nội dung phong phú: Các nền tảng như Netflix và FPT Play có một thư viện phim lớn, bao gồm các bộ phim quốc tế, bản quyền, và các nội dung gốc.
  + Trải nghiệm người dùng mượt mà: Giao diện của các nền tảng thường được thiết kế rất dễ sử dụng, cho phép người dùng tìm kiếm và duyệt qua các bộ phim nhanh chóng.
  + Tính năng phát trực tuyến: Chất lượng video cao, khả năng hỗ trợ các thiết bị đa dạng từ điện thoại đến TV thông minh, giúp người dùng dễ dàng tiếp cận nội dung.
* **Hạn chế:**
  + Chi phí cao: Dịch vụ như Netflix và Amazon Prime Video yêu cầu người dùng phải trả phí thuê bao định kỳ, điều này có thể là rào cản đối với một bộ phận người dùng.
  + Thiếu tính tương tác: Các nền tảng hiện tại chủ yếu cung cấp dịch vụ xem phim một chiều, thiếu các tính năng tương tác như chia sẻ, xem chung với bạn bè, hay bình luận trực tiếp.
  + Chưa tối ưu cho một số thiết bị: Một số nền tảng có thể gặp khó khăn khi tương thích với một số dòng thiết bị hoặc hệ điều hành, gây khó khăn cho người dùng.

1. **Tóm tắt yêu cầu người dùng đối với ứng dụng xem phim**

Người dùng hiện nay yêu cầu các nền tảng xem phim phải đáp ứng những yêu cầu cơ bản và nâng cao để đảm bảo trải nghiệm tốt nhất:

* Trải nghiệm dễ dàng: Giao diện người dùng (UI) phải thân thiện, dễ sử dụng và dễ tìm kiếm phim, đặc biệt là với các bộ phim theo thể loại, năm phát hành, hoặc các đánh giá của người dùng.
* Chất lượng hình ảnh và âm thanh cao: Đảm bảo chất lượng video ổn định, đặc biệt là khi phát trực tuyến trên các thiết bị di động và TV.
* Tính năng đa dạng: Bao gồm khả năng lưu trữ bộ phim yêu thích, tạo danh sách theo dõi, đánh giá phim, xem lại lịch sử xem, và có các đề xuất phim dựa trên sở thích cá nhân.
* Khả năng tương tác xã hội: Người dùng mong muốn có thể chia sẻ bộ phim yêu thích, tham gia xem chung cùng bạn bè, và bình luận về các bộ phim hoặc tập phim.
* Hỗ trợ nhiều nền tảng và thiết bị: Người dùng muốn có thể truy cập và xem phim trên nhiều loại thiết bị khác nhau như điện thoại di động, máy tính bảng, máy tính cá nhân, và TV thông minh mà không gặp phải sự cố về tính tương thích.

Những yêu cầu này phản ánh xu hướng hiện tại của người dùng, nhấn mạnh vào sự tiện lợi, chất lượng và khả năng tương tác cao trong các nền tảng xem phim trực tuyến.

1. **Phân tích hệ thống**
2. **Mục tiêu của hệ thống**

Mục tiêu chính của hệ thống là xây dựng một ứng dụng web xem phim trực tuyến, dễ sử dụng, có khả năng phục vụ nhiều đối tượng người dùng khác nhau, từ người xem cá nhân đến nhóm bạn bè. Hệ thống phải hỗ trợ người dùng tìm kiếm và xem các bộ phim yêu thích, quản lý danh sách phim, và cung cấp các thông tin chi tiết về các bộ phim. Ngoài ra, hệ thống cũng cần có tính năng tương tác, cho phép người dùng đánh giá và chia sẻ phim với bạn bè.

1. **Các yêu cầu cơ bản của hệ thống**

* Tính dễ sử dụng: Giao diện người dùng phải dễ sử dụng, thân thiện với người dùng, cho phép người dùng dễ dàng tìm kiếm và duyệt qua các bộ phim.
* Tính tương thích cao: Hệ thống phải hỗ trợ nhiều thiết bị và nền tảng, bao gồm máy tính để bàn, điện thoại di động và TV thông minh.
* Tính năng linh hoạt: Hệ thống cần hỗ trợ các tính năng như đăng nhập, đăng ký tài khoản, quản lý danh sách yêu thích, và khả năng đánh giá phim.
* Quản lý nội dung phim hiệu quả: Hệ thống phải có khả năng tổ chức và quản lý kho nội dung phim một cách dễ dàng, bao gồm việc cập nhật các bộ phim mới, phân loại phim theo thể loại và cung cấp các đề xuất phim cho người dùng.
* Bảo mật thông tin người dùng: Các thông tin cá nhân của người dùng phải được bảo vệ và xử lý đúng cách, bao gồm mật khẩu và dữ liệu thanh toán.

1. **Yêu cầu chức năng**

* Đăng nhập và đăng ký tài khoản: Hệ thống phải cho phép người dùng đăng nhập và đăng ký tài khoản để cá nhân hóa trải nghiệm. Tài khoản sẽ lưu trữ thông tin về các bộ phim yêu thích, lịch sử xem phim và các đánh giá.
* Tìm kiếm và lọc phim: Người dùng có thể tìm kiếm phim theo tên, thể loại, năm phát hành hoặc quốc gia. Ngoài ra, hệ thống phải có chức năng lọc phim theo các tiêu chí như đánh giá cao nhất, mới nhất hoặc phổ biến nhất.
* Xem phim trực tuyến: Người dùng có thể xem các bộ phim trực tuyến với chất lượng video cao, hỗ trợ phát trực tiếp mà không bị gián đoạn.
* Đánh giá phim: Người dùng có thể đánh giá phim sau khi xem, tạo ra một cộng đồng người dùng có thể chia sẻ ý kiến về các bộ phim.
* Danh sách yêu thích: Người dùng có thể tạo và quản lý danh sách yêu thích để dễ dàng truy cập lại các bộ phim yêu thích hoặc chưa xem.
* Quản lý thông tin phim: Hệ thống phải hỗ trợ các quản trị viên thêm, sửa, xóa các bộ phim, bao gồm thông tin chi tiết như tên, mô tả, poster, trailer và thể loại phim.
* Phân quyền người dùng: Hệ thống phân quyền để quản lý quyền truy cập của người dùng, bao gồm:
* Nhân viên: Quản lý nội dung phim và nguồn phim.
* Quản trị viên: Quản lý toàn bộ hệ thống, bao gồm thông tin phim, người dùng và phân quyền.

1. **Yêu cầu phi chức năng**

* Hiệu suất: Hệ thống phải đảm bảo hiệu suất cao, có khả năng phục vụ nhiều người dùng cùng lúc mà không gặp phải vấn đề về độ trễ trong việc tải phim hoặc truy vấn dữ liệu.
* Bảo mật: Các thông tin người dùng, đặc biệt là dữ liệu đăng nhập và thông tin thanh toán, phải được bảo mật bằng các phương thức mã hóa mạnh mẽ và bảo vệ khỏi các mối đe dọa từ bên ngoài.
* Tính mở rộng: Hệ thống phải có khả năng mở rộng, cho phép dễ dàng thêm vào các tính năng mới hoặc mở rộng cơ sở dữ liệu khi cần thiết.
* Khả năng duy trì: Hệ thống cần phải dễ bảo trì, với mã nguồn rõ ràng và dễ hiểu, có thể dễ dàng sửa chữa hoặc nâng cấp khi có yêu cầu mới.
* Khả năng tương thích với nhiều nền tảng: Hệ thống phải có khả năng hoạt động tốt trên nhiều trình duyệt web khác nhau và trên các thiết bị di động, máy tính bảng, máy tính để bàn.

1. **Quy trình hoạt động của hệ thống**

Quy trình hoạt động của hệ thống bao gồm các bước sau:

* Người dùng truy cập trang web: Người dùng truy cập trang chủ của ứng dụng web xem phim, nơi họ có thể tìm kiếm và duyệt qua các bộ phim có sẵn.
* Đăng nhập/Đăng ký tài khoản: Nếu người dùng chưa có tài khoản, họ có thể đăng ký một tài khoản mới bằng cách cung cấp thông tin cá nhân như tên, email và mật khẩu. Nếu đã có tài khoản, họ có thể đăng nhập bằng tài khoản đã đăng ký.
* Tìm kiếm và lựa chọn phim: Sau khi đăng nhập, người dùng có thể tìm kiếm các bộ phim theo tên, thể loại, hoặc các tiêu chí khác. Hệ thống sẽ hiển thị các kết quả tìm kiếm theo các tiêu chí mà người dùng đã chọn.
* Xem phim: Khi người dùng chọn một bộ phim, hệ thống sẽ hiển thị trang chi tiết của bộ phim đó, bao gồm poster, mô tả, trailer và các thông tin liên quan. Người dùng có thể bắt đầu xem phim trực tuyến.
* Đánh giá và nhận xét: Sau khi xem phim, người dùng có thể đánh giá phim bằng cách chọn số sao Nhận xét sẽ được công khai cho các người dùng khác xem.
* Quản lý phim (dành cho quản trị viên): Quản trị viên có thể truy cập vào hệ thống quản lý phim để thêm mới, sửa đổi hoặc xóa các bộ phim, cập nhật thông tin liên quan đến phim.
* Kết thúc phiên làm việc: Người dùng có thể đăng xuất khỏi tài khoản khi không còn sử dụng ứng dụng. Hệ thống sẽ lưu lại các thay đổi hoặc hành động của người dùng trong phiên làm việc đó, như việc đánh giá phim hoặc cập nhật danh sách yêu thích.
* Phân quyền người dùng: Quản trị viên sẽ có khả năng phân quyền cho các người dùng khác nhau. Điều này bao gồm việc cấp quyền cho nhân viên, quản trị viên hoặc người dùng thông thường, mỗi nhóm có quyền hạn khác nhau trong hệ thống (quản lý nội dung phim, quản lý người dùng, v.v.).

Quy trình này đảm bảo tính mượt mà và liền mạch cho người dùng, từ khi truy cập trang chủ cho đến khi tương tác và sử dụng các tính năng của hệ thống.

1. **Thiết kế hệ thống**
2. **Mô hình kiến trúc hệ thống**

Hệ thống sử dụng mô hình kiến trúc MVC (Model-View-Controller) để phân tách các thành phần của hệ thống, giúp duy trì tính mở rộng, dễ bảo trì và dễ dàng kiểm thử.

* Model: Bao gồm các lớp và bảng cơ sở dữ liệu như User\_64130299, Movie\_64130299, Rating\_64130299, Favorite\_64130299, SubscriptionPlans\_64130299, và MemberSubscription\_64130299. Những bảng này chịu trách nhiệm quản lý dữ liệu người dùng, phim, đánh giá, danh sách yêu thích và các gói thành viên.
* View: Giao diện người dùng web sẽ hiển thị danh sách phim, trang chi tiết phim, các tính năng tìm kiếm, đánh giá, và danh sách yêu thích. Hệ thống quản trị viên sẽ có một giao diện riêng để quản lý phim, người dùng và các gói thành viên.
* Controller: Các controller sẽ xử lý các yêu cầu từ người dùng, ví dụ: đăng ký, đăng nhập, tìm kiếm phim, xem phim, đánh giá phim, và quản lý thông tin phim hoặc thành viên.

1. **Thiết kế và mô tả mô hình dữ liệu vật lý**

* **Bảng User\_64130299:**

|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Mô tả** |
| UserId | VARCHAR(100) | Mã người dùng (Khóa chính) |
| Email | NVARCHAR(256) | Email (Duy nhất, NOT NULL) |
| Password | NVARCHAR(256) | Mật khẩu (NOT NULL) |
| Username | NVARCHAR(50) | Tên người dùng (NOT NULL) |
| CreatedAt | DATETIME | Thời gian tạo tài khoản |
| UpdatedAt | DATETIME | Thời gian cập nhật tài khoản |

* **Bảng MovieUrls\_64130299:**

|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Mô tả** |
| MovieUrlId | VARCHAR(100) | Mã URL phim (Khóa chính) |
| MovieId | VARCHAR(100) | Mã phim (Khóa ngoại) |
| Url | NVARCHAR(512) | URL của phim |
| EpisodeNumber | INT | Số tập phim (dành cho phim bộ) |

* **Bảng Favorite\_64130299:**

|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Mô tả** |
| FavoriteId | VARCHAR(100) | Mã yêu thích (Khóa chính) |
| UserId | VARCHAR(100) | Mã người dùng (Khóa ngoại) |
| MovieId | VARCHAR(100) | Mã phim (Khóa ngoại) |
| CreatedAt | DATETIME | Thời gian thêm vào yêu thích |

* **Bảng Rating\_64130299:**

|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Mô tả** |
| RatingId | VARCHAR(100) | Mã đánh giá (Khóa chính) |
| UserId | VARCHAR(100) | Mã người dùng (Khóa ngoại) |
| MovieId | VARCHAR(100) | Mã phim (Khóa ngoại) |
| Rating | INT | Điểm đánh giá (1-5) |

* **Bảng AdminUsers\_64130299**

|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Mô tả** |
| Id | INT | Mã quản trị viên (Khóa chính) |
| Username | NVARCHAR(255) | Tên người dùng |
| PasswordHash | NVARCHAR(255) | Mã hóa mật khẩu |
| Role | NVARCHAR(50) | Vai trò người dùng (Admin) |

* **Bảng SubscriptionPlans\_64130299:**

|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Mô tả** |
| PlanId | INT | Mã gói (Khóa chính) |
| PlanName | NVARCHAR(50) | Tên gói (ví dụ: "1 tháng") |
| DurationMonths | INT | Thời gian sử dụng gói (tháng) |
| Price | DECIMAL(10, 2) | Giá gói thành viên |
| CreatedAt | DATETIME | Thời gian tạo gói |
| UpdatedAt | DATETIME | Thời gian cập nhật gói |

* **Bảng MemberSubscription\_64130299:**

|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Mô tả** |
| SubscriptionId | VARCHAR(100) | Mã đăng ký thành viên (Khóa chính) |
| UserId | VARCHAR(100) | Mã người dùng (Khóa ngoại) |
| PlanId | INT | Mã gói đăng ký (Khóa ngoại) |
| StartDate | DATETIME | Ngày bắt đầu đăng ký |
| ExpiryDate | DATETIME | Ngày hết hạn đăng ký |
| AccessLevel | NVARCHAR(50) | Mức độ truy cập (Premium/Free) |
| Status | NVARCHAR(50) | Trạng thái thẻ (Active/Expired) |
| PaymentMethod | NVARCHAR(50) | Phương thức thanh toán |
| AmountPaid | DECIMAL(10, 0) | Số tiền đã thanh toán |
| RenewalDate | DATETIME | Ngày gia hạn (nếu có) |
| CreatedAt | DATETIME | Thời gian tạo đăng ký |
| UpdatedAt | DATETIME | Thời gian cập nhật đăng ký |

1. **Thiết kế giao diện người dùng (UI/UX)**

* **Trang chủ:**
  + Hiển thị phim phổ biến, thể loại, và thanh tìm kiếm.
  + Danh sách thể loại phim và phim phổ biến theo thẻ.
* **Trang chi tiết phim:**
  + Hiển thị poster, trailer, mô tả, diễn viên, đánh giá.
  + Nút "Xem phim", "Đánh giá", "Yêu thích".
* **Trang quản lý phim (Admin):**
  + Thêm, sửa, xóa phim, cập nhật thông tin như poster, trailer, thể loại.
  + Trang quản lý người dùng (Admin):
  + Xem danh sách người dùng, phân quyền, sửa thông tin người dùng.
* **Trang quản lý gói thành viên (Admin):**
  + Quản lý các gói thành viên: tên, giá, thời gian sử dụng.
* **Trang đăng ký/đăng nhập:**
  + Đăng ký tài khoản mới hoặc đăng nhập vào tài khoản.
* **Trang người dùng (Sau khi đăng nhập):**
  + Quản lý yêu thích, đánh giá phim và gói thành viên.
* **Trang quản lý thành viên (Admin):**
  + Xem danh sách thành viên, in báo cáo, xóa thẻ thành viên.

1. **Cài đặt và triển khai ứng dụng**
2. **Môi trường triển khai**

* Web Server: IIS.
* Công nghệ: ASP.NET MVC cho backend, HTML, CSS, JavaScript cho frontend.
* Cơ sở dữ liệu: SQL Server.
* Môi trường phát triển: Visual Studio.

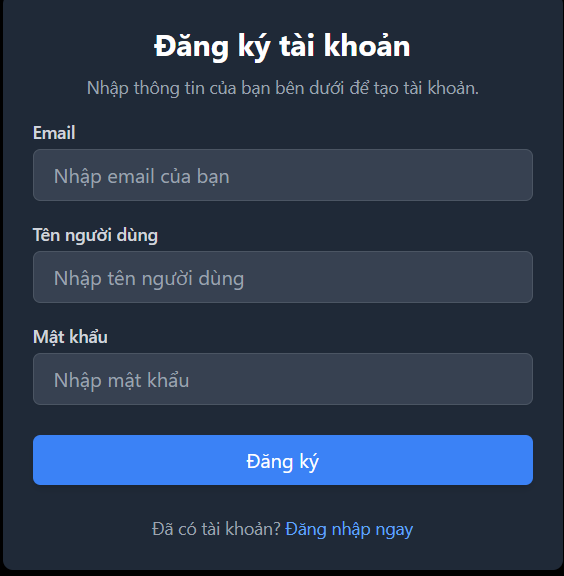
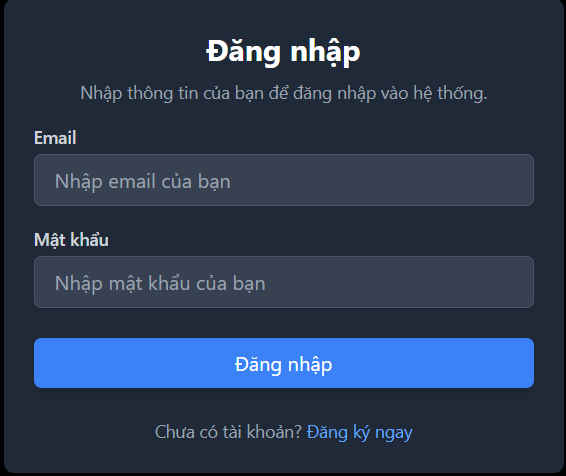
1. **Sơ đồ giao diện phần mềm**

* Giao diện người dùng: Trang chủ → Chi tiết phim → Trang người dùng (Quản lý yêu thích, đánh giá, đăng ký gói thành viên).
* Giao diện quản lý (Admin): Quản lý phim → Quản lý người dùng → Quản lý gói thành viên → Quản lý thành viên (Xem báo cáo, xóa thẻ).

1. **Các giao diện chính và diễn giải chức năng**

* Trang chủ: Hiển thị phim, thể loại, thanh tìm kiếm.
* Chi tiết phim: Thông tin chi tiết phim, trailer, dàn diễn viên, đánh giá.
* Trang người dùng: Quản lý yêu thích, đánh giá phim, đăng ký gói thành viên.
* Quản lý phim (Admin): Quản lý thông tin phim (thêm, sửa, xóa).
* Quản lý người dùng (Admin): Quản lý thông tin người dùng và phân quyền.
* Quản lý gói thành viên (Admin): Quản lý gói thành viên (tạo, sửa, xóa).
* Quản lý thành viên (Admin): Xem báo cáo và xóa thẻ thành viên.

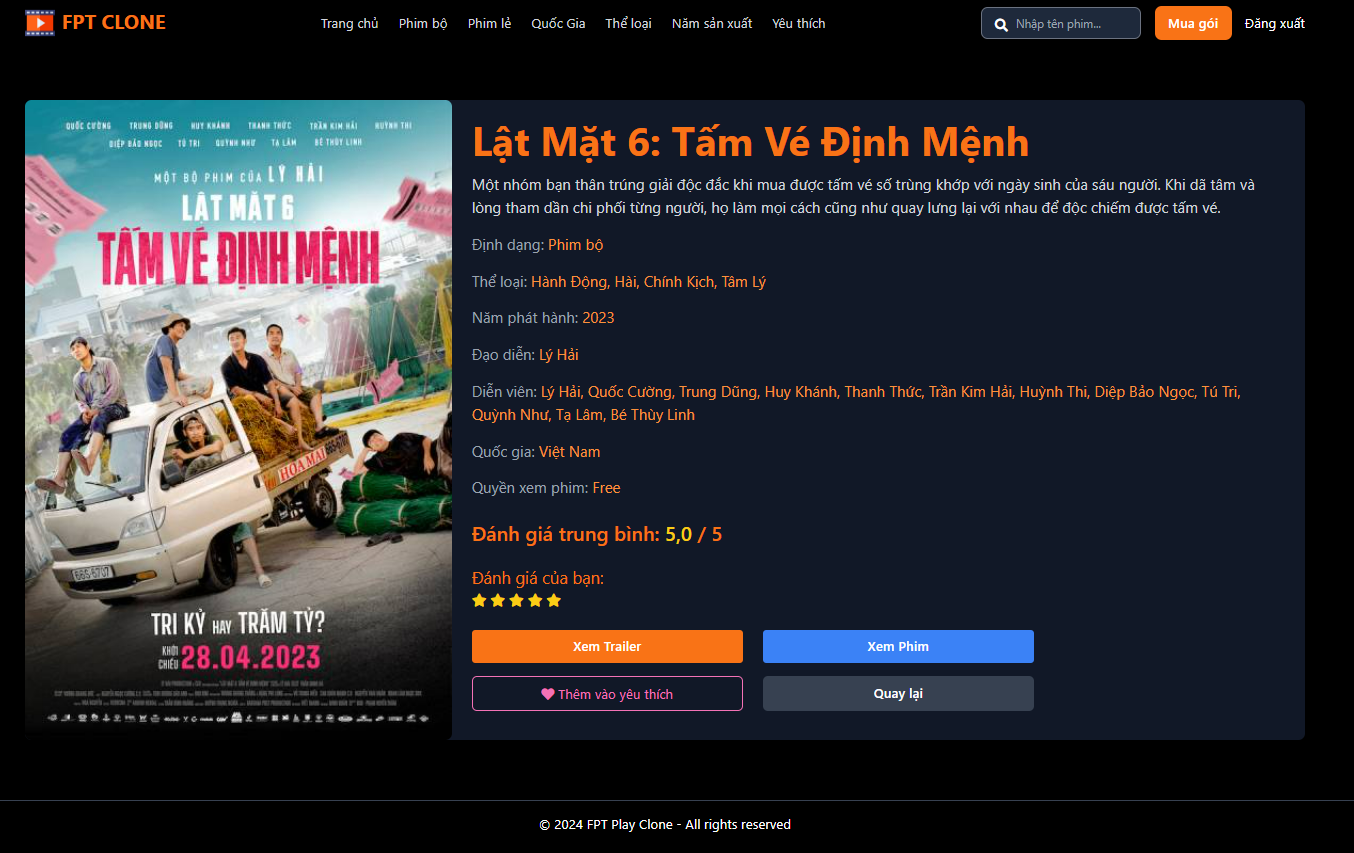
1. **Giao diện đăng nhập, đăng kí**

****

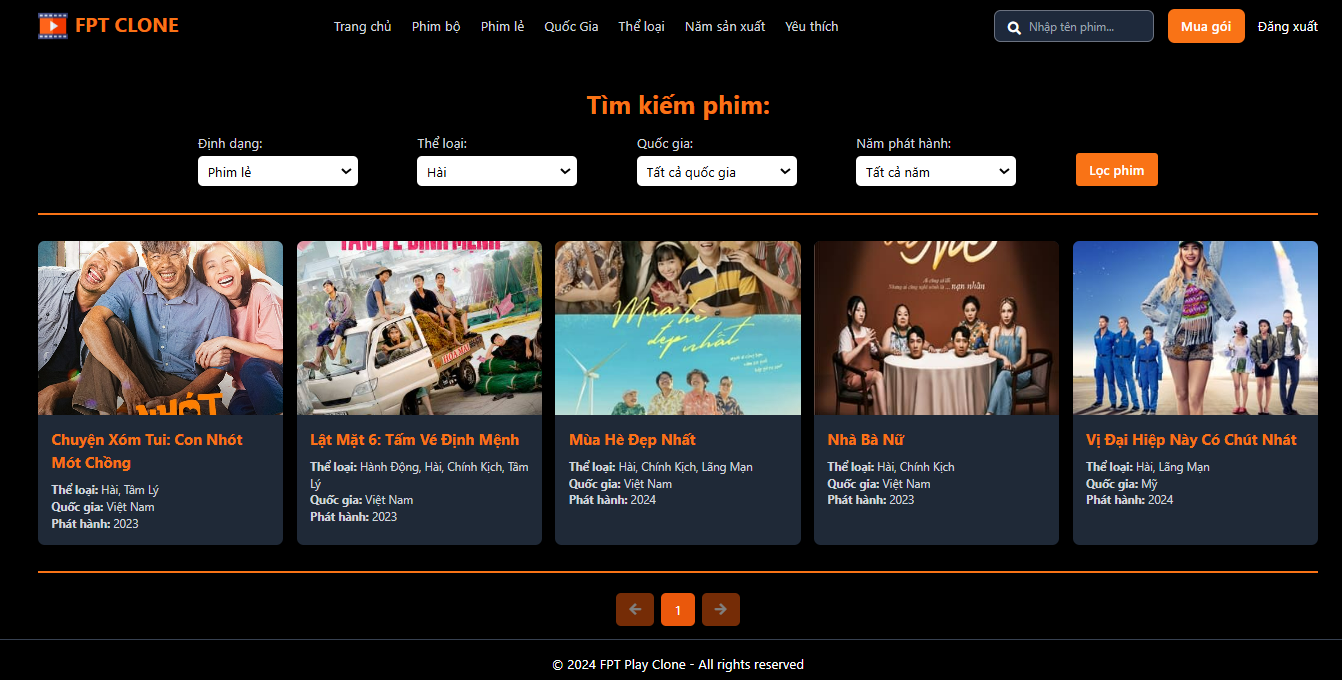
1. **Giao diện trang chủ**

****

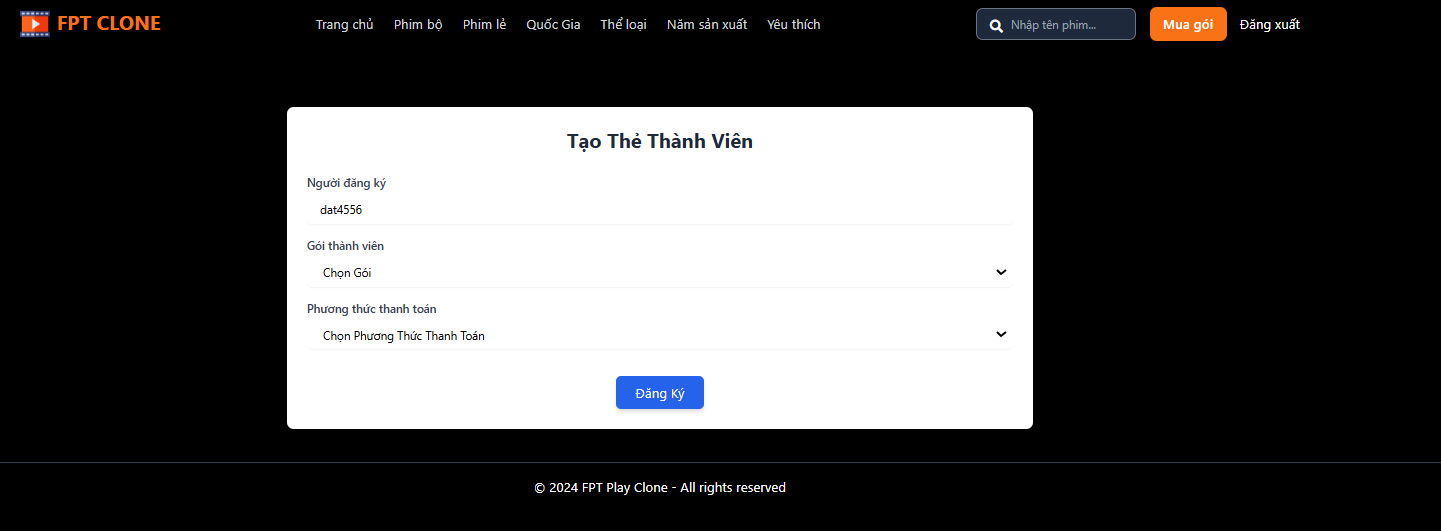
1. **Giao diện chi tiết phim**

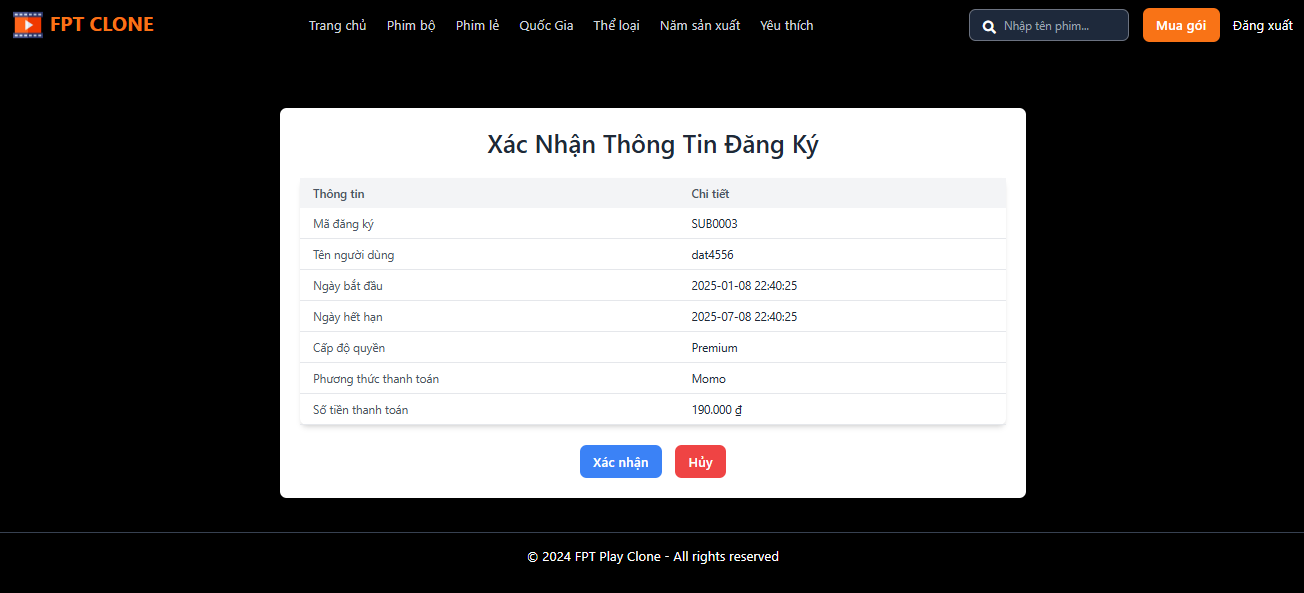
****

1. **Giao diện lọc phim**

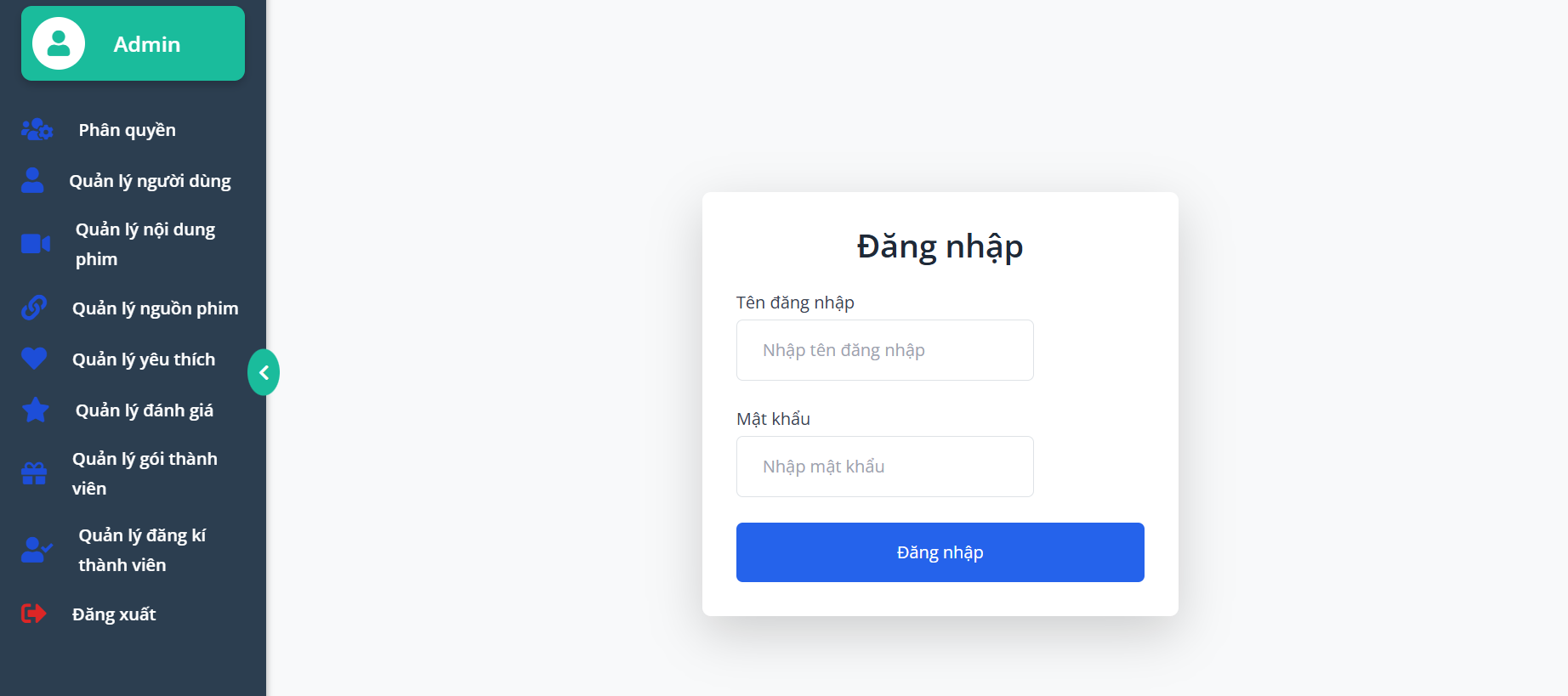
****

1. **Giao diện mua gói**

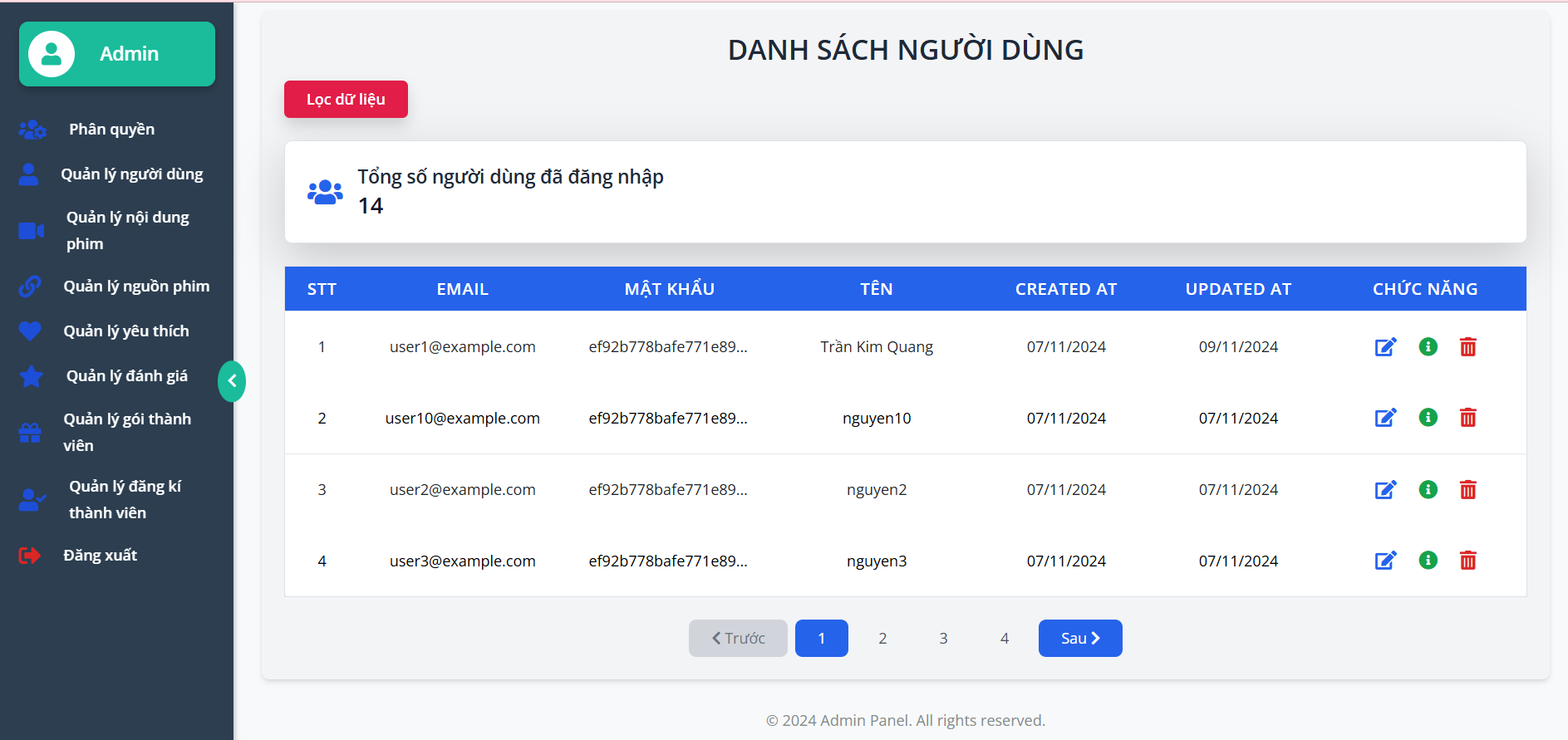
****

****

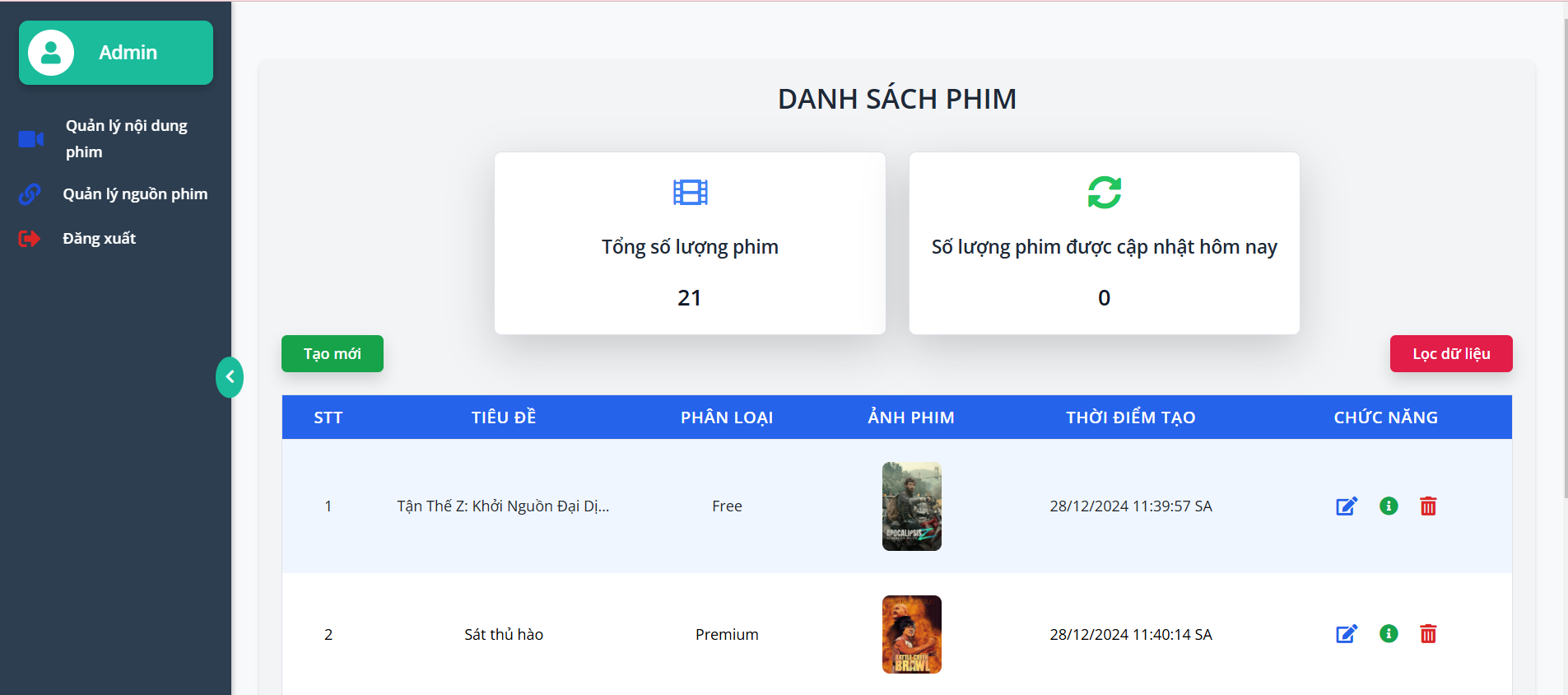
1. **Giao diện đăng nhập trang quản trị**

****

1. **Giao diện quản trị (Quản lý)**

****

1. **Giao diện quản trị (Nhân Viên)**

****

1. **Kiểm thử ứng dụng**
2. **Phương pháp kiểm thử**

* Kiểm thử chức năng: Kiểm tra các tính năng của ứng dụng như đăng nhập, quản lý người dùng, quản lý phim, và phân quyền.
* Kiểm thử giao diện: Đảm bảo giao diện người dùng (UI) thân thiện và dễ sử dụng, không có lỗi hiển thị.
* Kiểm thử hiệu suất: Kiểm tra tốc độ tải trang, độ phản hồi của các yêu cầu, và khả năng xử lý khi có lượng người dùng lớn.
* Kiểm thử bảo mật: Đảm bảo rằng các dữ liệu nhạy cảm được bảo vệ, và ứng dụng không dễ bị tấn công.

1. **Kết quả kiểm thử**

* Kết quả kiểm thử:
* Ứng dụng hoạt động ổn định, các chức năng cơ bản như quản lý người dùng, phim, và phân quyền đều thực hiện tốt.
* Giao diện trực quan và dễ sử dụng, không có lỗi hiển thị.
* Thời gian phản hồi nhanh và có thể xử lý được lượng người dùng cơ bản.
* Các yêu cầu bảo mật được đáp ứng, nhưng có thể cải thiện thêm về mã hóa và xác thực người dùng.

1. **Tài khoản demo người quản lý**

Username: admin

Password: XuanDat223@))

1. **Kết luận và đánh giá**
2. **Nhận xét về ưu điểm của ứng dụng**

* Ứng dụng dễ sử dụng với giao diện thân thiện và trực quan.
* Tính năng phân quyền linh hoạt, cho phép quản lý người dùng và nội dung hiệu quả.
* Đáp ứng được nhu cầu cơ bản của người dùng trong việc quản lý phim, nguồn phim, và thành viên.

1. **Hạn chế của ứng dụng**

* Chưa tối ưu hóa hiệu suất khi số lượng người dùng tăng cao.
* Các tính năng bảo mật có thể được cải thiện, đặc biệt là về xác thực và quyền truy cập.
* Còn thiếu một số tính năng nâng cao như thông báo hay phân tích dữ liệu người dùng.

1. **Đề xuất và kiến nghị cho phát triển trong tương lai**

* Tối ưu hóa mã nguồn và cơ sở dữ liệu để cải thiện hiệu suất.
* Thêm các tính năng bảo mật như mã hóa dữ liệu và xác thực đa yếu tố.
* Phát triển thêm các tính năng như thống kê người dùng và phân tích dữ liệu để nâng cao trải nghiệm người dùng.