1. **Xác định Yêu cầu Người dùng (User Stories) dựa trên FPT Play**

**A. Khách truy cập (Chưa đăng ký)**

* **Xem trang chủ và danh sách phim**: Hiển thị danh sách phim nổi bật, mới ra mắt, hoặc đang thịnh hành.
* **Tìm kiếm phim theo tên và thể loại**: Chỉ cần hỗ trợ tìm kiếm đơn giản theo tên phim và thể loại, không cần lọc theo năm phát hành hoặc diễn viên.
* **Xem trailer hoặc nội dung miễn phí**: Chỉ cần cung cấp một số trailer hoặc video giới thiệu phim miễn phí.
* **Đăng ký tài khoản**: Giữ lại tính năng đăng ký, nhưng chỉ yêu cầu email và mật khẩu (bỏ qua đăng ký qua Google/Facebook để đơn giản hóa).

**B. Người dùng đã đăng ký**

* **Đăng nhập và cá nhân hóa trải nghiệm**: Tính năng này được giữ nguyên.
* **Xem phim**: Người dùng có thể xem phim trên thiết bị mà họ đang sử dụng (không cần tích hợp nhiều thiết bị đồng bộ trong giai đoạn này).
* **Lưu phim yêu thích hoặc xem sau**: Người dùng có thể thêm phim vào danh sách yêu thích.
* **Lịch sử xem phim**: Hỗ trợ tính năng tiếp tục xem từ vị trí đã dừng nhưng không cần lưu trữ quá nhiều thông tin lịch sử.
* **Đánh giá phim**: Thay vì cả hệ thống đánh giá phức tạp, bạn có thể chỉ cần hỗ trợ người dùng đánh giá bằng số sao hoặc nút "Like".

**C. Quản trị viên**

* **Quản lý nội dung phim**: Tính năng thêm, sửa, xóa phim vẫn giữ lại, nhưng không cần quản lý chi tiết phức tạp.
* **Duyệt bình luận**: Hỗ trợ quản trị viên duyệt hoặc xóa bình luận của người dùng.
* **Quản lý quảng cáo**: Không cần tích hợp quảng cáo trong giai đoạn này, có thể thêm vào sau.

**Tính năng đột phá nhưng đơn giản**

**1. Phát video trực tiếp (Live Streaming)**

* **Tính năng**: Cung cấp tính năng phát trực tiếp các chương trình sự kiện đặc biệt, như liveshow hoặc bóng đá. Có thể sử dụng các dịch vụ live streaming hiện có như **YouTube Live** hoặc **Twitch** để tích hợp nhanh chóng, giảm tải cho hệ thống.

**2. Tính năng xem chung (Watch Party)**

* **Tính năng**: Cho phép người dùng tạo "phòng xem chung", nơi bạn bè có thể cùng xem phim và trò chuyện theo thời gian thực. Dùng **SignalR** để tạo một hệ thống chat đơn giản kết hợp phát video đồng bộ.

**3. Gợi ý phim thông minh**

* **Tính năng**: Sử dụng thuật toán đơn giản để gợi ý phim dựa trên những gì người dùng đã xem hoặc thích. Bạn có thể sử dụng các thuật toán gợi ý cơ bản mà không cần học máy (machine learning).

**Kế hoạch thực hiện trong 2 tháng**

**Tháng 1:**

* **Tuần 1**: Thiết kế cơ sở dữ liệu và giao diện cơ bản (trang chủ, trang phim).
* **Tuần 2**: Phát triển tính năng tìm kiếm và danh sách phim, chức năng xem trailer.
* **Tuần 3**: Xây dựng tính năng đăng ký/đăng nhập, xem phim cơ bản.
* **Tuần 4**: Phát triển quản trị viên cơ bản (thêm/sửa/xóa phim).

**Tháng 2:**

* **Tuần 1**: Hoàn thiện tính năng lịch sử xem phim và danh sách yêu thích.
* **Tuần 2**: Thêm tính năng đánh giá phim và quản lý bình luận.
* **Tuần 3**: Phát triển tính năng đột phá (Xem chung hoặc phát trực tiếp).
* **Tuần 4**: Kiểm thử, sửa lỗi, và triển khai.

Công việc tháng 1:

Tuần 1:

* + - **Thiết kế cơ sở dữ liệu:**

1. Tạo Cơ sở dữ liệu và Các Bảng

-- **Tạo cơ sở dữ liệu**

CREATE DATABASE MovieDatabase;

GO

-- **Chọn cơ sở dữ liệu để làm việc**

USE MovieDatabase;

GO

-- **Tạo bảng Users**

CREATE TABLE Users (

UserId INT IDENTITY(1,1) PRIMARY KEY,

Email NVARCHAR(256) UNIQUE NOT NULL,

Password NVARCHAR(256) NOT NULL,

CreatedAt DATETIME DEFAULT GETDATE(),

UpdatedAt DATETIME DEFAULT GETDATE()

);

GO

-- **Tạo bảng Movies**

CREATE TABLE Movies (

MovieId INT IDENTITY(1,1) PRIMARY KEY,

Title NVARCHAR(256) NOT NULL,

Description NVARCHAR(MAX),

ReleaseDate DATE,

Genre NVARCHAR(100),

PosterUrl NVARCHAR(512),

TrailerUrl NVARCHAR(512),

CreatedAt DATETIME DEFAULT GETDATE(),

UpdatedAt DATETIME DEFAULT GETDATE()

);

GO

-- **Tạo bảng Favorites**

CREATE TABLE Favorites (

FavoriteId INT IDENTITY(1,1) PRIMARY KEY,

UserId INT FOREIGN KEY REFERENCES Users(UserId),

MovieId INT FOREIGN KEY REFERENCES Movies(MovieId),

CreatedAt DATETIME DEFAULT GETDATE()

);

GO

-- **Tạo bảng WatchHistory**

CREATE TABLE WatchHistory (

HistoryId INT IDENTITY(1,1) PRIMARY KEY,

UserId INT FOREIGN KEY REFERENCES Users(UserId),

MovieId INT FOREIGN KEY REFERENCES Movies(MovieId),

LastWatchedTime DATETIME DEFAULT GETDATE(),

CreatedAt DATETIME DEFAULT GETDATE()

);

GO

-- **Tạo bảng Ratings**

CREATE TABLE Ratings (

RatingId INT IDENTITY(1,1) PRIMARY KEY,

UserId INT FOREIGN KEY REFERENCES Users(UserId),

MovieId INT FOREIGN KEY REFERENCES Movies(MovieId),

Rating INT CHECK (Rating BETWEEN 1 AND 5),

CreatedAt DATETIME DEFAULT GETDATE()

);

GO

-- **Tạo bảng Comments**

CREATE TABLE Comments (

CommentId INT IDENTITY(1,1) PRIMARY KEY,

UserId INT FOREIGN KEY REFERENCES Users(UserId),

MovieId INT FOREIGN KEY REFERENCES Movies(MovieId),

Content NVARCHAR(MAX),

CreatedAt DATETIME DEFAULT GETDATE()

);

GO

1. **Chèn dữ liệu mẫu**

**-- Chèn dữ liệu mẫu vào bảng Users**

**INSERT INTO Users (Email, Password)**

**VALUES ('user1@example.com', 'hashedpassword1'),**

**('user2@example.com', 'hashedpassword2');**

**GO**

**-- Chèn dữ liệu mẫu vào bảng Movies**

**INSERT INTO Movies (Title, Description, ReleaseDate, Genre, PosterUrl, TrailerUrl)**

**VALUES ('Movie Title 1', 'Description of movie 1', '2024-01-01', 'Action', 'http://example.com/poster1.jpg', 'http://example.com/trailer1.mp4'),**

**('Movie Title 2', 'Description of movie 2', '2024-02-01', 'Drama', 'http://example.com/poster2.jpg', 'http://example.com/trailer2.mp4');**

**GO**

**-- Chèn dữ liệu mẫu vào bảng Favorites**

**INSERT INTO Favorites (UserId, MovieId)**

**VALUES (1, 1),**

**(2, 2);**

**GO**

**-- Chèn dữ liệu mẫu vào bảng WatchHistory**

**INSERT INTO WatchHistory (UserId, MovieId, LastWatchedTime)**

**VALUES (1, 1, '2024-01-10 10:00:00'),**

**(2, 2, '2024-02-15 12:00:00');**

**GO**

**-- Chèn dữ liệu mẫu vào bảng Ratings**

**INSERT INTO Ratings (UserId, MovieId, Rating)**

**VALUES (1, 1, 5),**

**(2, 2, 4);**

**GO**

**-- Chèn dữ liệu mẫu vào bảng Comments**

**INSERT INTO Comments (UserId, MovieId, Content)**

**VALUES (1, 1, 'Great movie!'),**

**(2, 2, 'Not bad, but could be better.');**

**GO**

### 3. Truy vấn Dữ liệu

**a. Truy vấn danh sách phim:**

sql

Sao chép mã

SELECT \* FROM Movies;

GO

**b. Tìm kiếm phim theo tiêu đề hoặc thể loại:**

sql

Sao chép mã

SELECT \* FROM Movies

WHERE Title LIKE '%action%'

OR Genre = 'Action';

GO

**c. Lịch sử xem của người dùng:**

sql

Sao chép mã

SELECT m.Title, wh.LastWatchedTime

FROM WatchHistory wh

JOIN Movies m ON wh.MovieId = m.MovieId

WHERE wh.UserId = 1;

GO

**d. Danh sách yêu thích của người dùng:**

sql

Sao chép mã

SELECT m.Title

FROM Favorites f

JOIN Movies m ON f.MovieId = m.MovieId

WHERE f.UserId = 1;

GO

**e. Đánh giá phim của người dùng:**

sql

Sao chép mã

SELECT m.Title, r.Rating

FROM Ratings r

JOIN Movies m ON r.MovieId = m.MovieId

WHERE r.UserId = 1;

GO

**f. Bình luận về phim:**

sql

Sao chép mã

SELECT m.Title, c.Content

FROM Comments c

JOIN Movies m ON c.MovieId = m.MovieId

WHERE c.UserId = 1;

GO