

**BỘ THÔNG TIN VÀ TRUYỀN THÔNG
HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**



ĐỀ CƯƠNG DỰ ÁN

**ĐỀ TÀI
XÂY DỰNG WEBSITE ĐẶT VÉ XEM PHIM
TRỰC TUYẾN**

Giảng viên hướng dẫn : Kim Ngọc Bách

Sinh viên thực hiện: Trần Quang Nam

Mã sinh viên : B22DCDT208

Lớp : E22CQCN02-B

Mục lục

Chương 1: Tổng quan về đề tài	2
1.1. Lý do chọn đề tài	2
1.2. Mục tiêu	2
1.3. Phạm vi	2
1.4. Đối tượng sử dụng	2
1.5. Kết quả dự kiến	2
Chương 2: Công nghệ sử dụng	3
2.1. Figma	3
2.2. JavaScript	3
2.3. React	3
2.4. NodeJS	4
2.5. Express	4
2.6. MongoDB	4
Chương 3: Phân tích thiết kế hệ thống	5
3.1. Người sử dụng hệ thống	5
3.2. Yêu cầu của hệ thống và giao diện web	5
3.2.1. Yêu cầu của hệ thống	5
3.2.2. Yêu cầu về giao diện web	6
3.3. Các chức năng của hệ thống	6
3.3.1. Các chức năng của khách xem	6
3.3.2. Các chức năng của thành viên	7
3.3.3. Các chức năng của admin	7
Chương 4: Kế hoạch của dự án	7

Chương 1: Tổng quan về đề tài

1.1. Lý do chọn đề tài

Với sự đi lên cũng như áp dụng của công nghệ trong đời sống - xã hội hiện nay, công nghệ hóa mọi vấn đề trong cuộc sống dường như là một điều hiển nhiên. Công nghệ hóa chính là chìa khóa để mọi công việc có thể thực hiện dễ dàng, hiệu quả, nhanh chóng và an toàn. Chính vì vậy để bắt kịp xu thế, tiết kiệm thời gian và tiền bạc, mọi ngành nghề đều đã và đang dần chuyển sang sử dụng công nghệ thay vì sức người. Trong phạm vi này, tôi nghiên cứu và phát triển “Website đặt vé xem phim online” để nhằm đáp ứng nhu cầu cuộc sống, dễ dàng đặt vé mọi lúc mọi nơi, tiết kiệm sức lực và thời gian. Đồng thời để đuổi kịp và cạnh tranh trong thời kỳ công nghệ mới.

1.2. Mục tiêu

Xây dựng được một website đặt vé xem phim online hoàn chỉnh cả về chức năng lẫn trải nghiệm người dùng.

1.3. Phạm vi

Sau khi tìm hiểu và nghiên cứu sơ bộ về nhu cầu sử dụng của người dùng, tôi quyết định phạm vi thực hiện đề tài như sau:

- Phạm vi môi trường: triển khai sản phẩm đề tài trên môi trường Web
- Phạm vi chức năng:
 - + Đăng kí, đăng nhập.
 - + Quản lý thông tin người dùng.
 - + Quản lý phim với các thể loại phim khác nhau, khung giờ chiếu.
 - + Đặt vé.

1.4. Đối tượng sử dụng

- Người dùng có nhu cầu đặt vé xem phim ngoài rạp.
- Quản trị viên của website.

1.5. Kết quả dự kiến

- Nắm bắt và ứng dụng được các công nghệ mới để hoàn thiện sản phẩm đề tài.
- Hiểu rõ các nghiệp vụ, chức năng của một website đặt vé xem phim online.
- Ứng dụng được các kiến thức đã học về phân tích và thiết kế hệ thống phần mềm vào việc xây dựng và triển khai website sản phẩm đề tài.
- Có thể thay đổi giao diện một cách linh động và mở rộng thêm các chức năng mới cho website sản phẩm đề tài để phù hợp với nhu cầu thực tiễn trong tương lai.

Chương 2: Công nghệ sử dụng

2.1. Figma

- Figma là công cụ thiết kế giao diện người dùng (UI) và trải nghiệm người dùng (UX) dựa trên nền tảng đám mây.
- + Chức năng chính: Thiết kế giao diện, tạo prototype, hợp tác thời gian thực
- + Lý do sử dụng: Figma cho phép nhiều người cùng làm việc trên một thiết kế, lưu trữ trên đám mây, không cần cài đặt phần mềm phức tạp, và tích hợp tốt với quy trình phát triển web hiện đại.

2.2. JavaScript

- JavaScript là ngôn ngữ lập trình phía client phổ biến nhất cho web.
- + Chức năng chính: Xử lý tương tác người dùng, cập nhật nội dung động, xử lý sự kiện
- + Lý do sử dụng: JavaScript là ngôn ngữ chuẩn của web, chạy trên mọi trình duyệt, cho phép tạo trải nghiệm tương tác, và là nền tảng cho các framework hiện đại.

2.3. React

- React là thư viện JavaScript phát triển bởi Facebook để xây dựng giao diện người dùng.

- + Chức năng chính: Tạo UI động với components tái sử dụng, quản lý state, render hiệu quả với Virtual DOM
- + Lý do sử dụng: React giúp phát triển ứng dụng nhanh hơn với cấu trúc component, hiệu suất cao nhờ cơ chế cập nhật DOM thông minh, cộng đồng lớn và nhiều thư viện hỗ trợ.

2.4. NodeJS

- Node.js là môi trường chạy JavaScript phía máy chủ.
- + Chức năng chính: Xử lý yêu cầu HTTP, truy cập cơ sở dữ liệu, xử lý file
- + Lý do sử dụng: Node.js có hiệu suất cao với mô hình non-blocking I/O, cho phép dùng JavaScript ở cả frontend và backend (fullstack JavaScript), và có hệ sinh thái npm phong phú.

2.5. Express

- Express là framework web phổ biến nhất cho Node.js.
- + Chức năng chính: Định tuyến HTTP, middleware, xử lý yêu cầu và phản hồi
- + Lý do sử dụng: Express đơn giản, linh hoạt, hiệu suất cao, và dễ học với tài liệu phong phú.

2.6. MongoDB

- MongoDB là cơ sở dữ liệu NoSQL hướng tài liệu.
- + Chức năng chính: Lưu trữ dữ liệu dạng JSON (BSON), truy vấn linh hoạt, mở rộng dễ dàng
- + Lý do sử dụng: MongoDB phù hợp với JavaScript vì lưu trữ dữ liệu dạng JSON, linh hoạt với schema không cố định, mở rộng ngang tốt, và hiệu suất cao.

⇒ Kết hợp các công nghệ (MERN Stack + Figma)

Bộ công nghệ MERN Stack (MongoDB, Express, React, Node.js) kết hợp với Figma tạo thành một quy trình phát triển web hiện đại, toàn diện từ thiết kế đến triển khai:

- + Quy trình thiết kế: Figma cho phép thiết kế và prototype giao diện, với khả năng cộng tác thời gian thực giữa designers và developers.
- + Frontend phản hồi nhanh: React giúp xây dựng giao diện người dùng động

với components tái sử dụng, quản lý state hiệu quả.

- + Backend mạnh mẽ: NodeJs và Express cung cấp một nền tảng server linh hoạt, hiệu suất cao để xử lý yêu cầu từ client và giao tiếp với cơ sở dữ liệu.
- + Cơ sở dữ liệu linh hoạt: MongoDB lưu trữ dữ liệu dưới dạng JSON, phù hợp tự nhiên với JavaScript và dễ dàng mở rộng.
- + JavaScript xuyên suốt: Sử dụng JavaScript trên toàn bộ stack giúp đơn giản hóa quá trình phát triển, dễ dàng chia sẻ code giữa client và server.
- + Kiến trúc RESTful hoặc GraphQL: Dễ dàng xây dựng API hiện đại với Express để React frontend có thể giao tiếp với server.
- + Khả năng mở rộng: Toàn bộ stack có khả năng mở rộng tốt, phù hợp cho cả ứng dụng nhỏ lẫn lớn.

Chương 3: Phân tích thiết kế hệ thống

3.1. Người sử dụng hệ thống

- Có 3 đối tượng sử dụng hệ thống là: Nhà quản trị (Admin), thành viên và khách xem.
- + Admin: là thành viên quản trị của hệ thống, có các quyền và chức năng như: quản lý tài khoản, quản lý phim, xử lý đặt vé, quản lý tin tức, quản lý sự kiện,...
- + Khách xem: là thành viên chưa có tài khoản, có chức năng như: đăng ký tài khoản, xem và tìm kiếm thông tin phim. Không có chức năng đặt vé, nếu khách muốn đặt vé thì phải đăng nhập vào hệ thống.
- + Thành viên: là khách xem đã có tài khoản, có tất cả các chức năng của khách xem và có thêm chức năng khác như đặt vé.

3.2. Yêu cầu của hệ thống và giao diện web

3.2.1. Yêu cầu của hệ thống

- Hệ thống bán vé xem phim xây dựng trên cơ sở thỏa mãn các yêu cầu sau:

- + Hệ thống hướng tới nhiều đối tượng nên phải thiết kế đơn giản, đẹp mắt và dễ hiểu thuận tiện cho người mua để giảm bớt cảm giác phức tạp rắc rối cho họ. Tuy nhiên thiết kế giao diện người sử dụng (HCI) vẫn phải đảm bảo các quy tắc và tiêu chuẩn thiết kế, tạo sự thuận tiện và dễ dàng nhất cho người sử dụng.
- + Các chức năng phải được thực hiện một cách dễ dàng, nhanh chóng và chính xác. Đảm bảo các chức năng quan trọng nhất: đặt vé cho đối tượng muốn mua vé.
- + Việc tìm kiếm, tra cứu thông tin về phim, lịch chiếu, thông tin giảm giá theo một đặc điểm nào đó phải được thực hiện một cách nhanh chóng và chính xác.
- Sau khi tìm hiểu và khảo sát thực tế tại các rạp chiếu phim, tôi đã xác định các cá nhân sử dụng website như sau:
 - + Admin: Người quản lý hệ thống.
 - + Khách xem: Khách chỉ xem website mà không đăng ký thành viên tại website.
 - + Thành viên: Khách xem online và thực hiện đăng ký thành viên tại website thì trở thành thành viên.

3.2.2. Yêu cầu về giao diện web

- Giao diện người dùng.
- + Website có giao diện ưa nhìn, có sự bố trí và phối hợp hài hòa giữa các màu sắc và các thành phần trên website (Layout tốt).
- + Các chức năng dễ sử dụng, tiện dụng.
- + Có sử dụng các thư viện tạo hiệu ứng làm đẹp cho website.
- Giao diện nhà quản lý.
- + Giao diện đơn giản, dễ quản lý.
- + Hiện thị được tất cả các danh mục quản lý.
- + Phải được bảo vệ bằng tài khoản và mật khẩu riêng của nhà quản trị.

3.3. Các chức năng của hệ thống.

3.3.1. Các chức năng của khách xem.

- Xem danh sách phim đang chiếu, sắp chiếu.
- Tìm kiếm phim.
- Xem thông tin chi tiết phim (tóm tắt nội dung, diễn viên, thời lượng, lịch chiếu).
- Đăng ký tài khoản để trở thành thành viên.

3.3.2. Các chức năng của thành viên.

Có tất cả các chức năng của khách xem, thêm vào đó:

- Đăng nhập/Đăng xuất tài khoản.
- Đặt vé xem phim, chọn ghế, chọn suất chiếu, thanh toán vé.
- Xem lịch sử đặt vé.
- Chỉnh sửa thông tin cá nhân.
- Đổi mật khẩu, cập nhật hồ sơ.
- Đánh giá, bình luận về phim.

3.3.3. Các chức năng của admin.

- Quản lý người dùng
- + Xem danh sách người dùng
- + Khóa hoặc xóa tài khoản người dùng.
- Quản lý phim
- + Thêm, sửa, xóa thông tin phim.
- Quản lý đặt vé
- + Xem danh sách vé đã đặt.
- Quản lý bình luận
- + Xóa các bình luận vi phạm.
- Quản lý suất chiếu
- + Thêm, sửa, xóa suất chiếu.
- Quản lý rạp
- + Thêm, sửa, xóa thông tin rạp.
- Quản lý phòng chiếu
- + Thêm, sửa, xóa phòng chiếu.

Chương 4: Kế hoạch thực hiện dự án.

Giai đoạn 1: Thiết kế Figma

- + Nghiên cứu giao diện người dùng và trải nghiệm người dùng (UI/UX).

- + Thiết kế giao diện người dùng cho trang chủ, danh sách phim, chi tiết phim, đặt vé.
- + Thiết kế giao diện quản trị (Admin) cho quản lý phim, người dùng, suất chiếu.
- + Xây dựng prototype và điều chỉnh dựa trên phản hồi.

Giai đoạn 2: Thiết kế giao diện Frontend

- + Xây dựng giao diện người dùng bằng React.
- + Thiết kế trang chủ, danh sách phim, chi tiết phim, chức năng tìm kiếm.
- + Xây dựng giao diện đặt vé, chọn ghế, thanh toán.
- + Xây dựng giao diện Admin: quản lý phim, suất chiếu, người dùng.
- + Kiểm tra giao diện, tối ưu UI/UX.

Giai đoạn 3: Phát triển Backend

- + Thiết lập môi trường backend với Node.js và Express.
- + Xây dựng API cho quản lý người dùng, phim, suất chiếu, đặt vé.
- + Tích hợp cơ sở dữ liệu MongoDB.
- + Triển khai chức năng đăng nhập/đăng ký (JWT Authentication).
- + Kết nối frontend với backend, xử lý logic đặt vé.
- + Kiểm tra và sửa lỗi trong quá trình phát triển.

Giai đoạn 4: Kiểm thử, sửa lỗi và tối ưu hóa

- + Kiểm thử giao diện người dùng và hệ thống backend.
- + Kiểm thử chức năng đặt vé, thanh toán, quản lý phim.
- + Tối ưu hóa tốc độ tải trang, API, bảo mật.
- + Sửa lỗi và cải thiện trải nghiệm người dùng.

Giai đoạn 5: Báo cáo và hoàn thiện tài liệu

- + Viết báo cáo tổng kết dự án.