Đề tài: Web bán vé sự kiện

(Lấy ý tưởng từ Ticket box)

1. Lý do chọn đề tài

Hiện nay, nhu cầu giải trí và tham gia các sự kiện, cuộc vui của con người tăng cao. Tuy nhiên việc xếp hàng mua vé chắc chắn sẽ rất vất vả và tốn thời gian. Mặt khác nếu muốn tiết kiệm thời gian thì phải mua theo các nguồn không chính thống như chợ “đen” hay qua cá nhân nào đó và khả năng cao là không đảm bảo về chất lượng, có thể bị lừa tiền. Về phía người tổ chức sự kiện, có những trường hợp cơ quan tổ chức bị phá hoại vì không thể cung cấp đủ vé cho mọi người. Hiện tại, các nền tảng bán vé online cũng đã có tuy nhiên thường chỉ chuyên một sự kiện. Bản thân em cũng từng là người đi xếp hàng từ giữa đêm để đi mua vé một trận đấu bóng đá nên rất hiểu về sự bất tiện này. Nhằm giải quyết các vấn đề trên để có thể mua vé một cách dễ dàng và đảm bảo chất lượng nên bọn em đã chọn chủ đề này. Hệ thống không những giúp người dùng mua vé mà còn giúp người dùng và doanh nghiệp kết nối với nhau. Doanh nghiệp dễ dàng hơn quảng bá hoạt động của họ và người dùng sẽ dễ dàng tiếp cận hơn với các hoạt động sự kiện mà họ đang mong chờ.

2. Lý do chọn công nghệ

• Front-end: ReactJS

Ngày nay, ReactJS đã trở nên rất phổ biến bởi những tính năng linh hoạt và đơn giản với hơn 1,300 developer và hơn 94,000 trang web đang sử dụng ReactJS. ReactJS là một thư viện JavaScript mã nguồn mở được thiết kế bởi Facebook để tạo ra những ứng dụng web hấp dẫn, nhanh và hiệu quả với mã hóa tối thiểu. Mục đích cốt lõi của ReactJS không chỉ khiến cho trang web phải thật mượt mà còn phải nhanh, khả năng mở rộng cao và đơn giản. Mặt khác trong nhóm có một bạn có kinh nghiệm hơn 3 tháng làm việc với ReacJS tại doanh nghiệp.

• Back-end: JavaSpring

Spring là một trong những Framework nổi tiếng nhất để xây dựng ứng dụng dành cho doanh nghiệp. Trong thời gian gần đây, nó đã trở thành framework phổ biến nhất trong cộng đồng Java. Điều quan trọng là tất cả mọi thành viên của nhóm đều có nền tảng về lập trình hướng đối tượng nên dễ tiếp thu và học hỏi.

• Database: MySQL

Cơ sở dữ liệu MySQL đã trở thành cơ sở dữ liệu mã nguồn mở phổ biến nhất trên thế giới vì cơ chế xử lý nhanh và ổn định của nó, sự đáng tin cậy cao và dễ sử dụng. MySQL không chỉ là cơ sở dữ liệu mã nguồn mở phổ biến nhất trên thế giới, nó còn trở thành cơ sở dữ liệu được chọn cho thế hệ mới của các ứng dụng xây dựng trên nền Linux, Apache, MySQL, PHP/Perl/Python. MySQL chạy trên hơn 20 flatform bao gồm: Linux, Windows, OS/X, HP-UX, AIX, Netware, mang đến tính linh hoạt trong việc sử dụng.

3. Phương pháp coding và quản lý version

Nhóm chúng em quyết định sử dụng tích hợp giữa mô hình phát triển chữ V (V-model) và mô hình xoắn ốc để phát triển dự án. Bước đầu sẽ phân tích yêu cầu ở mức tương đối sau đó thực hiện coding và triển khai test UT để có thể tạo sản phẩm demo. Sau đó thực hiện phân tích và làm mịn yêu cầu của hệ thống, thực hiện nâng version của phần mềm và triển khai test UT lần 2. Tiếp theo thực hiện test IT, ST và UAT. Trong quá trình test sẽ thực hiện fix bug và update source code.

Nhóm chúng em công cụ github để quản lý version code và quản lý tiến độ dự án dựa trên web app Redmine.

Link github: https://github.com/thanhnh99/ticket\_web.git

4. Rủi ro để hoàn thành dự án

- Chưa xác định cụ thể scope của dự án dẫn đến dự án có thể mở rộng quá mức so với dự định ban đầu khiến không đủ thời gian hoàn thiện.

- Nhóm Front-end chỉ có 1 bạn nắm chắc công nghệ Reactjs, 2 bạn còn lại đang trong giai đoạn tìm hiểu.

- Sử dụng tên miền, server miễn phí nên bị giới hạn về băng thông và có thể chết server bất cứ lúc nào.

- PM chưa có nhiều kinh nghiệm thực tế về quản lý dự án dẫn tới có thể sai sót trong việc sắp xếp công việc.

- Khó khăn về việc kết hợp giữa code của front-end và back-end.

5. Những việc đã hoàn thành

- Dựng source base cả 2 phía back-end và front-end.

- Phân tích use case của hệ thống ở mức tương đối.

- Phân chia công việc cho các thành viên của dự án.

6. Những việc chưa hoàn thành

- Chưa thiết kế phần common của hệ thống.

- Chưa phân tích cụ thể số lượng màn hình của hệ thống.

- Một số ca sử dụng chưa phân tích cụ thể nghiệp vụ và luồng hệ thống.

- Chưa thống nhất API giữa front-end và back-end.