

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

Khoa công nghệ thông tin

BÁO CÁO LẬP TRÌNH WED

Đề tài: Lập trình java spring ứng dụng quản lý hệ thống lịch thi đấu giải đua
motogp

Nhóm 1:

Nguyễn Xuân Hùng-B20DCCN298

Đoàn Quang Huy-B20DCCN310

Nguyễn Khánh Nam-B20DCCN454

CHƯƠNG 1: THIẾT KẾ VÀ TRIỂN KHAI THỰC NGHIỆM

1.1 Phân tích và thiết kế hệ thống

1.1.1 Phân tích yêu cầu

- Các chức năng cần có trong hệ thống:
 - Quản lý tay đua: cho phép quản lý (QL) thực hiện thêm, sửa, xóa thông tin tay đua
 - Đăng ký thi đấu: cho phép thêm mới, sửa đổi và xóa thông tin cuộc đua.
 - Tính điểm và xếp hạng cho các đội đua: tính điểm theo từng vòng đua, cập nhật điểm số và xếp hạng cho các đội đua sau mỗi vòng đua.
- Các đối tượng trong hệ thống:
 - Đội đua: lưu trữ thông tin về đội đua bao gồm tên, quốc gia, logo và các thông tin liên quan.
 - Cuộc đua: lưu trữ thông tin về cuộc đua bao gồm ngày, địa điểm, đường đua, số vòng và các thông tin liên quan.
 - Tay đua: lưu trữ thông tin về tay đua bao gồm tên, quốc tịch, số xe, đội đua và các thông tin liên quan.

- Điểm số: lưu trữ thông tin về điểm số của từng tay đua trong từng cuộc đua.

1.2.2 Thiết kế kiến trúc hệ thống

1.2.2.a) Xác định các thành phần chính của hệ thống:

- Ứng dụng web: là thành phần giao tiếp với người dùng, xử lý yêu cầu và hiển thị kết quả trả về.
- Cơ sở dữ liệu: là nơi lưu trữ thông tin về đường đua, đội đua, cuộc đua, tay đua và điểm số.
- RESTful API: là giao diện lập trình ứng dụng cho phép các ứng dụng khác giao tiếp với hệ thống của chúng ta.
- Thư viện hỗ trợ: trong hệ thống sẽ sử dụng các thư viện hỗ trợ của Spring Framework để xử lý các yêu cầu của người dùng và truy cập cơ sở dữ liệu.

1.2.2.b) Thiết kế giao diện

- trang đăng nhập cho người dùng. Người dùng cần phải đăng nhập để có thể truy cập các tính năng khác của hệ thống.
- Sau khi đăng nhập, hãy tạo một trang để đăng ký thi đấu cho các tay đua. Trong trang này, người dùng cần phải cung cấp thông tin về tên, độ tuổi, đội đua của họ và các thông tin khác liên quan đến việc thi đấu.
- Tạo một trang để cập nhật kết quả thi đấu. Trong trang này, người dùng cần phải cung cấp thông tin về tên tay đua, đội đua và thời gian hoàn thành của họ.
- Tạo một trang để xem bảng xếp hạng của các tay đua và đội đua. Trang này sẽ hiển thị thông tin về số điểm, số lần thắng và các thông tin khác liên quan đến việc đua xe.

- tạo một trang để quản lý các thông tin về tay đua và đội đua. Trong trang này, bạn có thể thêm, sửa đổi hoặc xóa thông tin về tay đua và đội đua.

1. Trang đăng ký tay đua:

- Form đăng ký với các thông tin cần thiết như tên, tuổi, giới tính, quê quán, số điện thoại, email, đội đua, v.v.
- Nút Đăng ký để hoàn tất quá trình đăng ký.

2. Trang quản lý tay đua:

- Danh sách tất cả các tay đua đã đăng ký với thông tin chi tiết của mỗi tay đua.
- Chức năng tìm kiếm tay đua bằng tên hoặc số điện thoại.
- Nút chỉnh sửa để chỉnh sửa thông tin của một tay đua.
- Nút xóa để xóa một tay đua khỏi danh sách.

3. Trang đăng ký thi đấu:

- Form đăng ký thi đấu với các thông tin cần thiết như tên cuộc thi, ngày thi đấu, địa điểm, v.v.
- Danh sách các tay đua đã đăng ký thi đấu cho cuộc thi đó.
- Chức năng tìm kiếm tay đua bằng tên hoặc số điện thoại.
- Nút đăng ký để hoàn tất quá trình đăng ký thi đấu.

4. Trang cập nhật kết quả thi đấu:

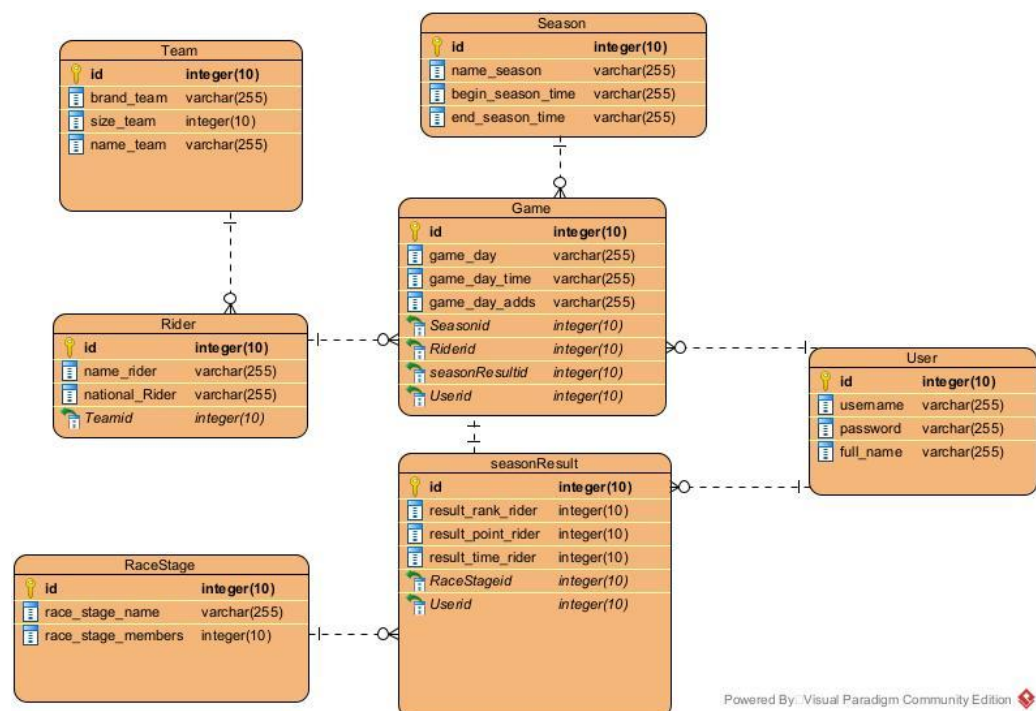
- Danh sách các cuộc thi đã diễn ra với thông tin chi tiết về cuộc thi đó.
- Danh sách các tay đua đã tham gia cuộc thi đó và kết quả của từng tay đua.

- Form cập nhật kết quả với các thông tin cần thiết như tên cuộc thi, ngày thi đấu, tên tay đua, v.v.
- Nút cập nhật để lưu kết quả đã cập nhật.

5. Trang xem bảng xếp hạng:

- Bảng xếp hạng của tất cả các tay đua theo điểm số.
- Chức năng sắp xếp tay đua theo thứ tự tăng hoặc giảm của điểm số.
- Bảng xếp hạng của các đội đua theo tổng điểm số của các tay đua trong đội.

1.2.2.c)Thiết kế cơ sở dữ liệu



1.2.2.c)Thiết kế kiến trúc hệ thống

- Kiến trúc hệ thống của một web API có thể được chia thành các thành phần sau:
 - Clients: Đây là các ứng dụng hoặc trang web sử dụng API của bạn để truy xuất và gửi dữ liệu.

- **API Gateway:** Đây là thành phần đầu vào duy nhất của hệ thống, nơi các yêu cầu được nhận và điều hướng đến các dịch vụ tương ứng.
- **Microservices:** Đây là các dịch vụ chính của hệ thống, mỗi dịch vụ chịu trách nhiệm xử lý một tập hợp các yêu cầu liên quan đến chức năng cụ thể.
- **Data Storage:** Đây là nơi lưu trữ dữ liệu cho các dịch vụ.
- **Authentication và Authorization:** Đây là các thành phần xác thực và phân quyền, đảm bảo rằng chỉ các người dùng được phép truy cập vào dữ liệu và chức năng mà họ có quyền truy cập.
- **Caching:** Đây là thành phần cho phép lưu trữ tạm thời kết quả yêu cầu trước đó để giảm thiểu thời gian xử lý yêu cầu mới.
- **Monitoring và Logging:** Đây là các công cụ cho phép theo dõi và ghi lại các hoạt động của hệ thống, giúp phát hiện và giải quyết các sự cố.
- **Load Balancer:** Đây là thành phần cho phép phân phối tải cho các dịch vụ để đảm bảo hệ thống hoạt động hiệu quả và ổn định.
- **Deployment và Configuration Management:** Đây là các công cụ quản lý và triển khai các phiên bản mới của hệ thống, đảm bảo rằng các phiên bản hoạt động ổn định và có hiệu quả.

2. phân công công việc

Đoàn Quang Huy(trưởng nhóm):thiết kế giao diện và code phần frontend

modul được phân chia:đăng nhập,

Nguyễn Xuân Hùng:phân tích và thiết kế database

modul được phân chia:quản lý tay đua

Nguyễn Khánh Nam:hostserver,test API,backend

moodul được phân chia:quản lý lịch thi đấu