



Universidade Federal
de São João del-Rei

**UNIVERSIDADE FEDERAL DE SÃO JOÃO DEL-REI
DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO
BANCO DE DADOS**

SISTEMA ESTATÍSTICO BRASFOOT

Matheus N Silva

São João del-Rei
2023

Lista de Figuras

| | | |
|---|---------------------------------------|---|
| 1 | Diagrama modelagem de dados | 6 |
|---|---------------------------------------|---|

Sumário

| | | |
|----------|---|----------|
| 1 | INTRODUÇÃO | 1 |
| 1.1 | Objetivo | 1 |
| 1.2 | Motivação | 1 |
| 2 | ABORDAGEM DO PROBLEMA | 1 |
| 3 | MODELAGEM DE DADOS | 2 |
| 3.1 | Diagrama da modelagem de dados | 6 |
| 4 | TECNOLOGIA EMRPEGADA | 6 |
| 5 | PROPOSTA DE MELHORIAS | 7 |
| 5.1 | Módulo de Gerenciamento de Partidas: | 7 |
| 5.2 | Funcionalidade de Transferência de Atletas: | 7 |
| 5.3 | Flexibilidade para Outros Campeonatos: | 8 |
| 5.4 | Aprimoramento da Interface e Usabilidade: | 8 |
| 5.5 | Segurança e Controle de Acesso: | 8 |
| 6 | CONCLUSÃO | 8 |
| | REFERÊNCIAS | 9 |

1 INTRODUÇÃO

Este é um trabalho prático da disciplina de Banco de dados no curso de Ciência da Computação na UFSJ, tendo como docente o professor Leonardo Rocha.

1.1 Objetivo

Neste trabalho, temos como objetivo desenvolver um sistema estatístico do campeonato brasileiro de futebol para uma importante empresa de mídia brasileira. Para tal o sistema deve ser dotado das seguintes funcionalidades: apresentar o artilheiro do campeonato, o atleta e equipe mais e menos violenta, simular jogos, apresentar o local em que as partidas serão realizadas bem como horário das mesmas, entre outros.

Assim, nosso objetivo é desenvolver uma aplicação abordando desde o desenvolvimento de uma plataforma de interação para o usuário, passando à uma plataforma de gerenciamento, e chegando à implementação do banco de dados utilizado, incluindo à modelagem de dados.

1.2 Motivação

Dada a descrição do problema, podemos concluir que ele se trata de um problema de armazenamento e processamento de dados. Esse tipo de problema tem bastante relevância não só na computação como no mundo, e a cada dia que passa torna-se fundamental o desenvolvimento de bancos que armazenam dados diversos, sejam dados de menor ou maior volume, e aplicações que possam manipula-los de forma a extrair o maximo de informações utilizáveis.

2 ABORDAGEM DO PROBLEMA

O primeiro passo que precisamos tomar para resolver esse problema é modelar de forma logica o banco de dados. Para isso devemos reunir quais informações serão necessárias para a criação de tal modelo.

Nesse problema a aplicação deve ser capaz de apresentar ao usuário quem é o atleta com mais gols, bem como o(s) atleta(s) e a(s) equipe(s) com menos e mais cartões, além de simular os jogos e informar o local onde ocorrerá partidas.

3 MODELAGEM DE DADOS

A modelagem de dados é um passo crucial no desenvolvimento de sistemas de informação, pois define como as informações serão organizadas e relacionadas entre si. No contexto do problema proposto, é essencial ter uma estrutura de dados bem definida para armazenar e gerenciar as informações sobre equipes, jogadores, partidas e estatísticas. A seguir apresentaremos uma modelagem de dados detalhada, destacando as entidades principais e seus atributos, além de explicar os relacionamentos entre elas. A compreensão dessa estrutura, será crucial para a construção um sistema robusto e eficiente para acompanhar e interagir com o campeonato de forma completa e organizada. Vamos explorar cada elemento da modelagem e seus respectivos vínculos para uma compreensão abrangente do sistema proposto.

Equipe

Atributos:

- **ID_Equipe (Chave primária):** Um identificador único para cada equipe.
- **Nome_Completo_Equipe:** O nome completo da equipe.
- **Nome_Equipe:** O nome principal da equipe.
- **Cidade:** A cidade de origem da equipe.
- **Estado:** O estado de origem da equipe.
- **Treinador:** O nome do treinador da equipe.
- **Estadio:** O estadio onde a equipe manda seus jogos.

A entidade "Equipe" representa as equipes de futebol que participam do campeonato. Ela armazena informações básicas sobre cada equipe, como seu nome, cidade, estado, treinador e estadio.

Atleta

Atributos:

- **ID_Atleta (Chave primária):** Um identificador único para cada atleta.
- **Nome_Atleta:** O nome do atleta.
- **Posicao:** A posição em que o atleta atua (atacante, meio-campista, etc.).
- **Data_Nasc:** A data de nascimento do atleta.
- **Nacionalidade:** A nacionalidade do atleta.
- **N_Camisa:** O número da camisa que o atleta utiliza durante os jogos.
- **ID_Equipe (Chave estrangeira):** Uma referência à equipe à qual o atleta pertence.

A entidade "Atleta" representa os atletas que fazem parte das equipes. Ela contém informações específicas sobre cada atleta, como seu nome, posição, data de nascimento, nacionalidade, número da camisa e a equipe à qual pertence.

Partida

Atributos:

- **ID_Partida (Chave primária):** Um identificador único para cada partida.
- **DataHora:** A data e o horario em que a partida será realizada.
- **Mandante (Chave estrangeira):** Uma referência à equipe que será a mandante na partida.
- **Visitante (Chave estrangeira):** Uma referência à equipe que será a visitante na partida.
- **Local:** O local onde a partida será disputada.
- **GolsMandante:** Campo onde sera armazenando os gols da equipe mandante.
- **GolsVisitante:** Campo onde sera armazenado os gols da equipe visitante.

A entidade "Partida" representa os jogos que ocorrerão no campeonato. Ela contém informações como a data e horario do jogo, as equipes que estão jogando (mandante e visitante), o local da partida e a quantidade de gols marcados por cada equipe.

Artilheiro

A entidade "Artilheiro" foi criada como uma view onde a mesma busca do banco de dados as informações de nome do atleta, equipe a que pertence e quantidade de gols marcados.

Atributos:

- ID_Atleta
- Nome_Atleta
- Nome_Equipe
- Gols

Cartões_Amarelos

A entidade "Cartões_Amarelos" foi criada como uma view onde a mesma busca do banco de dados as informações de nome do atleta, equipe a que pertence, quantidade de cartões amarelos recebidos. recebidos.

Atributos:

- ID_Atleta
- Nome_Atleta
- Nome_Equipe
- Quantidade

Cartões_Amarelos

A entidade "Cartões_Amarelos" foi criada como uma view onde a mesma busca do banco de dados as informações de nome do atleta, equipe a que pertence, quantidade de cartões amarelos recebidos. recebidos.

Atributos:

- ID_Atleta
- Nome_Atleta

- Nome_Equipe
- Quantidade

Relacionamentos

Equipe - atleta: Um atleta pertence a uma equipe.

Relacionamento: 1 para muitos (uma equipe pode ter muitos atletas, mas um atleta pertence a uma única equipe).

Partida - Equipe Uma partida tem 2 equipes (um mandante e um visitante).

Relacionamento: muitos para muitos. (uma partida tem "muitas" equipes e uma equipe pode jogar em várias partidas).

Estatísticas da Partida - Partida: As estatísticas de uma partida estão associadas a uma partida específica.

Relacionamento: Muitos para 1 (muitas estatísticas podem estar associadas a uma única partida, mas uma estatística pertence a uma única partida).

Artilheiro - atleta: O artilheiro está associado a varios atletas.

Relacionamento: 1 para muitos (tendo em vista que possa haver varios artilheiros o relacionamento pode ser um para muitos).

Cartões Amarelos - atleta: Os cartões amarelos estão associado a varios atletas.

Relacionamento: 1 para muitos (tendo em vista que varios atletas possam receber cartões amarelos o relacionamento pode ser um para muitos).

Cartões Vermelhos - atleta: Os cartões vermelhos estão associado a varios atletas.

Relacionamento: 1 para muitos (tendo em vista que varios atletas possam receber cartões vermelhos o relacionamento pode ser um para muitos).

3.1 Diagrama da modelagem de dados

Abaixo esta uma representação grafica de como foi construido o banco de dados.

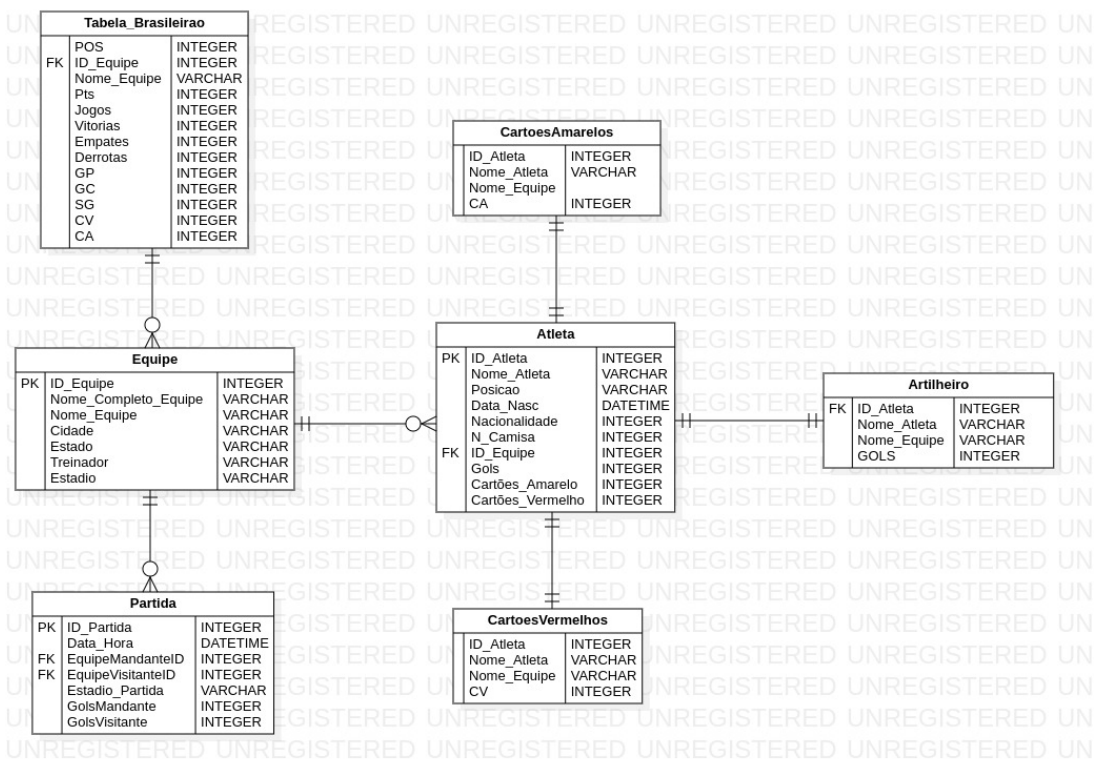


Figura 1: Diagrama modelagem de dados

4 TECNOLOGIA EMRPEGADA

O banco de dados foi desenvolvido utilizando a tecnologia MySQL, um sistema de gerenciamento de banco de dados relacional amplamente reconhecido e utilizado na indústria de software e desenvolvimento web. O MySQL oferece uma plataforma robusta e escalável para armazenar, organizar e recuperar dados de maneira eficiente e segura.

Esta escolha de tecnologia foi motivada pela confiabilidade, desempenho e facilidade de integração do MySQL em diferentes tipos de aplicações. Além disso, sua comunidade ativa e a vasta documentação disponível facilitam a implementação e manutenção do banco de dados ao longo do tempo.

O MySQL suporta uma ampla gama de funcionalidades, desde operações simples de inserção, atualização e consulta, até recursos avançados de modelagem de dados e otimização de consultas complexas. Isso o torna uma escolha sólida para projetos que exigem um sistema de gerenciamento de banco de dados confiável e poderoso.

Ademais, a compatibilidade do MySQL com diversas linguagens de programação e frameworks facilita a integração do banco de dados com outras partes do sistema, permitindo uma experiência de desenvolvimento mais fluida e eficiente.

Em resumo, a escolha do MySQL como a tecnologia central para o banco de dados deste projeto foi baseada em sua reputação sólida, desempenho confiável e suporte robusto da comunidade, o que contribui para a robustez e eficácia do sistema como um todo.

Para o desenvolvimento da aplicação web foi utilizado HTML/CSS como frontend e Python com Flask como backend. A utilização de Python, em conjunto com o framework Flask, é uma escolha popular para desenvolvimento web por diversas razões. Python é conhecido por sua legibilidade, vasta quantidade de bibliotecas e facilidade de manutenção. O Flask, sendo um microframework leve e flexível, oferece liberdade para os desenvolvedores escolherem ferramentas específicas para cada projeto.

Essa combinação proporciona vantagens como implementação ágil de APIs, escalabilidade, performance e segurança. A simplicidade do Flask permite interações eficientes com o frontend em HTML/CSS, enquanto a comunidade ativa e a extensa documentação garantem suporte aos desenvolvedores.

5 PROPOSTA DE MELHORIAS

Para aprimorar e expandir as funcionalidades do sistema atual, pode-se introduzir novas ferramentas que permitam uma gestão mais abrangente e flexível das atividades esportivas. Abaixo, estão algumas propostas de melhorias:

5.1 Módulo de Gerenciamento de Partidas:

- Oferecer a capacidade de modificar resultados de partidas anteriores, com a devida autorização e controle de acesso para garantir a integridade dos dados.

5.2 Funcionalidade de Transferência de Atletas:

- Desenvolver um recurso que possibilite a transferência de atletas entre equipes cadastradas no sistema.
- Garantir uma interface intuitiva para os gestores das equipes solicitarem e autorizarem transferências de jogadores.

5.3 Flexibilidade para Outros Campeonatos:

- Adaptar o sistema para suportar a gestão de múltiplos campeonatos ou ligas esportivas simultaneamente.
- Permitir a personalização de configurações para diferentes competições, como regras específicas, formatos de torneios e pontuações.

5.4 Aprimoramento da Interface e Usabilidade:

- Focar na usabilidade do sistema, garantindo uma interface amigável e de fácil navegação para os usuários, sejam eles gestores de equipes, atletas ou administradores.
- Incluir recursos visuais que ofereçam uma visão geral dos campeonatos, partidas agendadas e resultados de forma clara e organizada.

5.5 Segurança e Controle de Acesso:

- Assegurar que apenas usuários autorizados tenham acesso às funcionalidades de modificar resultados de partidas e realizar transferências de atletas.

6 CONCLUSÃO

Ao concluir este trabalho, é evidente o imenso desafio que enfrentamos ao nos aventurarmos em um campo praticamente desconhecido, como o desenvolvimento web e a manipulação de banco de dados. A falta de familiaridade com esses conceitos tornou o desafio ainda mais significativo. No entanto, a nossa determinação em aprender e superar obstáculos foi fundamental para alcançar um sistema funcional que atendesse às demandas iniciais.

Ao longo do processo de desenvolvimento, percebemos a necessidade de ajustar certos parâmetros inicialmente propostos para garantir o melhor desempenho do sistema. Enfrentamos inúmeras dificuldades na comunicação entre a aplicação web, o backend e o banco de dados, o que nos fez dedicar tempo e esforço na resolução desses problemas.

Esses obstáculos não foram apenas desafios a serem superados, mas também oportunidades de aprendizado e crescimento. Cada erro corrigido e cada alteração feita nos permitiu aprimorar nossas habilidades e compreensão do processo de desenvolvimento de

sistemas web. O entusiasmo e a persistência foram nossos principais aliados para não desistir da implementação e, como resultado, alcançar um produto final que atendeu, de maneira satisfatória, a maioria dos requisitos propostos inicialmente.

Este projeto não apenas nos fez sair da zona de conforto, mas também nos mostrou a importância de enfrentar desafios, buscar conhecimento em diversas fontes e não desistir diante das dificuldades.

Referências

[Foundation, 2023] Foundation, P. S. (2023). Python documentation. Acesso em: 05 de dezembro de 2023.

[Team, 2023] Team, P. (2023). Flask documentation. Acesso em: 5 de dezembro de 2023.