Felipe Silva de Borba

Github: https://github.com/Felipe-Borba/prova2-dev-web

# Tarefa 1: Listagem de Requisitos

* RF01 Cada jogador pode pertencer a vários times
* RF02 Deve ser possível criar um jogador com atributos: Nome (string), Idade (number), Posição (string), Nacionalidade (string).
* RF03 Deve ser possível ler um jogador
* RF04 Deve ser possível atualizar um jogador
* RF05 Deve ser possível listar um jogador
* RF06 Deve ser possível deletar um jogador
* RF07 Deve ser possível criar um time com atributos Atributos: Nome (string), Ano de Fundação (number), País (string), Técnico (string).
* RF08 Deve ser possível ler um time
* RF09 Deve ser possível atualizar um time
* RF10 Deve ser possível listar um time
* RF11 Deve ser possível deletar um time
* RF12 Deve ser possível criar uma partida, com Atributos: Data (string), Local (string), Placar (string), Time da Casa (string), Time Visitante (string).
* RF13 Deve ser possível listar as partidas
* RF14 Deve ser possível encerrar uma partida
* RF12 Deve ser possível atualizar placar da partida
* RF 13 Deve ser possível adicionar um jogador em um time

# Tarefa 2: Implementação do Sistema

* RF01 Cada jogador pode pertencer a vários times

É possível ver a relação no arquivo schema.prisma

* RF02 Deve ser possível criar um jogador com atributos: Nome (string), Idade (number), Posição (string), Nacionalidade (string).

Rota: POST <http://localhost:8080/player>

Body: name (string), age (int), position (string), nationality(string)

Pontos: 10

* RF03 Deve ser possível ler um jogador

Rota: GET http://localhost:8080/player/:id

Param: id (jogador)

Pontos: 10

* RF04 Deve ser possível atualizar um jogador

Rota PUT <http://localhost:8080/player>

Body: id (string), name (string), age (int), position (string), nationality(string)

Pontos: 10

* RF05 Deve ser possível listar um jogador

Rota: GET <http://localhost:8080/player>

Pontos: 10

* RF06 Deve ser possível deletar um jogador

Rota: DELETE <http://localhost:8080/player/:id>

Param: id (string)

Pontos 10

* RF07 Deve ser possível criar um time com atributos Atributos: Nome (string), Ano de Fundação (number), País (string), Técnico (string).

Rota: POST <http://localhost:8080/soccerTeam>

Body: name (string), foundation (string date), coach (string), country (string)

Pontos 10

* RF08 Deve ser possível ler um time

Rota: GET <http://localhost:8080/soccerTeam/:id>

Param: id (string)

Pontos: 10

* RF09 Deve ser possível atualizar um time

Rota PUT <http://localhost:8080/soccerTeam>

Body: id(string), name (string), foundation (string date), coach (string), country (string)

Ponstos 10

* RF10 Deve ser possível listar um time

Rota: GET <http://localhost:8080/soccerTeam>

Pontos: 10

* RF11 Deve ser possível deletar um time

Rota: DELETE <http://localhost:8080/soccerTeam/:id>

Param: id (string)

Pontos 10

* RF12 Deve ser possível criar uma partida, com Atributos: Data (string), Local (string), Placar (string), Time da Casa (string), Time Visitante (string).

Rota: POST <http://localhost:8080/match>

Body: date (string date), place (string), score (string), homeTeam (string), visitTeam (string)

Ponstos 10

* RF13 Deve ser possível listar as partidas

Rota GET <http://localhost:8080/match>

Pontos 10

* RF14 Deve ser possível encerrar uma partida

Rotas: POST <http://localhost:8080/match/:matchId/finish>

Params: matchId (string)

Pontos 10

* RF12 Deve ser possível atualizar o placar da partida

Rotas: PUT <http://localhost:8080/match>

Body: id(string), date (string date), place (string), score (string), homeTeam (string), visitTeam (string)

Pontos 10

* RF 13 Deve ser possível adicionar um jogador em um time

Rotas POST <http://localhost:8080/soccerTeam/:soccerTeamId/add-player/:playerId>

Params: soccerTeamId (string), playerId (string)

Pontos 10