

Segundo Trabalho Prático

Objectivos: Prática com programação assíncrona; `CompletableFuture<T>`;

Data limite de entrega: 4 de Julho de 2016

NOTA: A solução entregue deve incluir **todos os testes unitários necessários** para validar o correto funcionamento das funcionalidades pedidas.

PARTE 1

Dada a API RESTfull <http://api.football-data.org/index> para acesso a informações de ligas de futebol, implementar uma aplicação Web que retorna páginas HTML com os dados obtidos da API RESTfull.

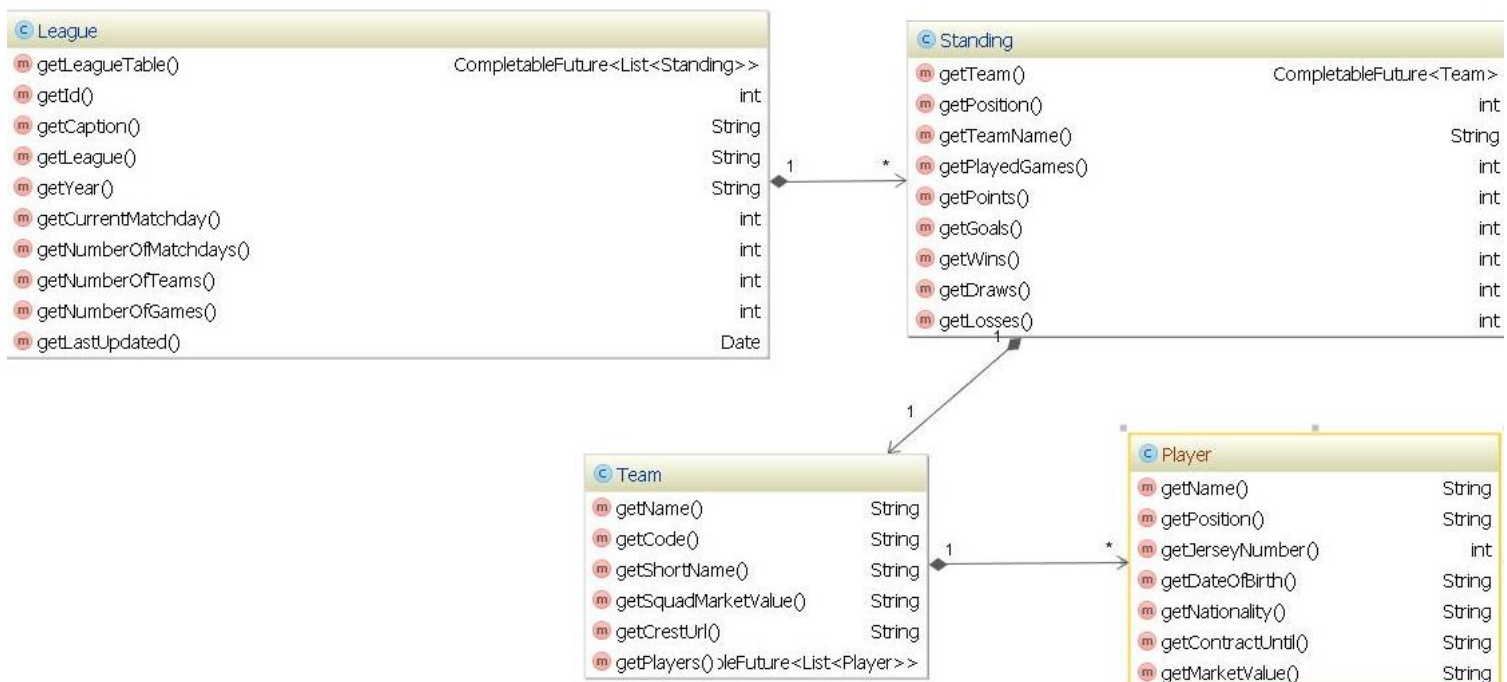
Devem ser considerados os 4 endpoints seguintes, como fontes de dados:

- <http://api.football-data.org/v1/soccerseasons>
- <http://api.football-data.org/v1/soccerseasons/{leagueId}/leagueTable>
- <http://api.football-data.org/v1/teams/{teamId}>
- <http://api.football-data.org/v1/teams/{teamId}/players>

O projecto <https://github.com/isel-leic-mpd/SoccerApp> disponibiliza uma API Java para acesso aos 4 endpoints anteriores.

NOTA: Deve registar-se em <http://api.football-data.org/index> e obter uma API Key que adicionará a `SoccerWebApi`.

Como base à construção da aplicação Web pode implementar um modelo de domínio com as relações apresentadas na figura seguinte:



A aplicação web deve disponibilizar os seguintes URIs:

- *soccerapp/leagues* – retorna uma listagem de todas as ligas obtidas de <http://api.football-data.org/v1/soccerseasons>
- *soccerapp/leagues/{id}* – sendo {id} o identificador da liga, retorna a tabela classificativa dessa liga. Por exemplo se {id} for 398, então para o URI *soccerapp/leagues/398* é retornada uma página HTML com os dados da tabela classificativa obtida a partir de <http://api.football-data.org/v1/soccerseasons/398/leagueTable>
- *soccerapp/teams/{id}* – sendo {id} o identificador da equipa devolve a informação detalhada dessa equipa. Por exemplo se {id} for 61, então para o URI *soccerapp/teams/61* é retornada uma página HTML com os dados da equipa obtida a partir de <http://api.football-data.org/v1/teams/61>
- *soccerapp/players/{id}* -- sendo {id} o identificador da equipa devolve a listagem dos jogadores dessa equipa. Por exemplo se {id} for 61, então para o URI *soccerapp/players/61* é retornada uma página HTML com a listagem dos jogadores obtidos de <http://api.football-data.org/v1/teams/61/players>

As páginas HTML devem incluir ligações (*links*) entre si.

As páginas HTML geradas devem ser gravadas em disco após a sua criação. Ou seja, em subsequentes execuções da aplicação web não devem ser geradas novamente as páginas que já tenham sido previamente criadas e gravadas.

Além disso a aplicação web deve manter em cache (memória) as páginas HTML. Ou seja, umas vez geradas, ou lida do disco a página HTML esta deve ser mantida em memória e reaproveitada em futuras respostas do mesmo URI. Implemente uma abstracção para aceder à cache. Na parte 2 serão introduzidos novos requisitos sobre o comportamento da cache.

As páginas das equipas e dos jogadores devem ser geradas assim que é pedido o URI da tabela classificativa. A criação dessas páginas deve ser feita de forma assíncrona de modo a que a obtenção da tabela classificativa não seja atrasada pela geração das consequentes páginas de cada equipa e seus jogadores. Por exemplo, o primeiro pedido de informação da Premier League (id 398) além da página com a tabela classificativa deve ainda diferir a criação de mais 40 páginas HTML: 20 para cada equipa e mais 20 para a listagem de jogadores de cada equipa. Se durante a geração dessas páginas existir um pedido para uma página que ainda não foi gerada, mas cuja geração está em curso, o respetivo pedido fica pendente, até que esta página esteja disponível e sem que exista um novo pedido à API.

Parte 2

A cache deve ser limitada na sua construção em número de entradas, e a política de remoção de páginas das cache é LRU (*Least Recently Used*).

Parte 3 (opcional)

Adicione páginas com informação detalhada de cada um dos jogadores. A página com a listagem dos jogadores deve ter ligações para a página de cada um dos seus jogadores.

Na página de detalhe do jogador inclua 5 fotos do jogador. Apresente as fotos com base no URL dado por uma API WEB de pesquisa de imagens, como por exemplo <https://datamarket.azure.com/dataset/bing/search>, ou outra.