

Introdução

A avaliação da componente prática da disciplina de Programação na Internet será realizada com base na aplicação FOCA (FOotball Chelas Application), a ser desenvolvida ao longo do semestre. A aplicação fornece acesso, através de uma interface web (hipermédia), a algumas das funcionalidades disponibilizadas pelo sítio <https://www.football-data.org>, fazendo para esse efeito uso da **versão 2.0** da sua Web API:

<https://www.football-data.org/documentation/quickstart>.

O desenvolvimento será realizado de forma incremental, envolvendo necessariamente vários ciclos de refactoring do código produzido e, por isso, é fundamental que faça uso de técnicas Object-Oriented e de boas práticas de programação em geral, de modo a reduzir o esforço associado a cada ciclo.

O desenvolvimento da aplicação FOCA é faseado em três enunciados (Parte 1, Parte 2 e Parte 3). Em cada enunciado é definida a data limite de entrega da solução, um detalhe não negociável. A entrega é realizada através da criação da tag **ParteX** no repositório GitHub do grupo, onde X corresponde ao número da parte a entregar.

Para cada funcionalidade da aplicação FOCA deve ser definido o tipo de pedido HTTP, o respetivo Uri e o conteúdo da resposta. Esta documentação deve constar no wiki do repositório na forma de documentação da API (podem-se basear na documentação da Football Data API). No repositório deve ainda constar um ficheiro com o resultado da exportação da coleção Postman com os pedidos que validam a API.

Resumo dos artefactos a submeter na entrega:

- Directoria node.js com a aplicação FOCA
- Wiki com a documentação da API da aplicação FOCA
- Projecto Postman de validação da API da aplicação FOCA

Data limite de entrega: 18/11/2018-23h59.

Requisitos Funcionais

Desenvolver uma aplicação Web que disponibiliza uma Web API que segue os princípios REST, com respostas em formato Json e que suporta as seguintes funcionalidades:

- Obter a lista de todas as ligas
- Obter as equipas de uma determinada liga
- Gerir grupos de equipas favoritas
 - Criar grupo atribuindo-lhe um nome e descrição
 - Editar grupo, alterando o seu nome e descrição
 - Adicionar uma equipa a um grupo
 - Remover uma equipa de um grupo
 - Obter os jogos das equipas de um grupo entre duas datas, sendo essas datas parametrizáveis no pedido. Os jogos de todas as equipas do um grupo vêm ordenados por ordem cronológica.

Requisitos Não Funcionais

A aplicação devem ser desenvolvida com a tecnologia Node.js. Para o atendimento de pedidos HTTP deve ser usado o módulo [http](#). Para realização de pedidos, pode ser usado o módulo [http](#) ou em alternativa o módulo [request](#).

Os dados que são próprios da aplicação devem ser guardados no índice foca da base de dados Elasticsearch.

Qualquer um dos módulos base do Node.js pode ser usado. Além destes, nesta 1ª parte do trabalho, apenas podem ser usados os seguintes módulos:

- request – Realização de pedidos HTTP
- debug – Mensagens de debug
- mocha – Testes unitários

Qualquer outro módulo que pretenda usar, deve ser previamente debatido e autorizado pelo respetivo docente.

A aplicação servidora deve ser constituída por pelo menos 5 módulos Node:

- foca-server.js - ficheiro que constitui o ponto de entrada na aplicação servidora
- foca-web-api.js - implementação dos rotas HTTP que constituem a API REST da aplicação Web
- foca-services.js - implementação da lógica de cada uma das funcionalidades da aplicação
- football-data.js - acesso à API FootballData.
- foca-db.js - acesso à BD Elasticsearch.

As dependência entre estes módulos é a seguinte:

```
foca-server.js -> foca-web-api.js -> foca-services.js -> football-data.js  
                                     -> foca-db.js
```

A metodologia de desenvolvimento da aplicação servidora deve ser a seguinte e por esta ordem:

1. Desenhar e documentar as rotas da API (tipo de pedido HTTP + URL) e documentar no wiki do repositório do grupo
2. Criar uma coleção no Postman (exº Foca) para testar as rotas da API
3. No módulo `foca-web-api.js` implementar as rotas da API, uma a uma.
 - a. Para cada rota implementada criar um ou mais testes unitários que comprovem o correto funcionamento de cada rota.
 - b. Apenas passar à implementação da próxima rota quando a anterior estiver completamente implementada e testada.
 - c. Para cada rota criar um pedido na coleção do Postman que a valida.
 - d. Nesta fase da implementação do módulo `foca-web-api.js` **usar dados locais (mock do foca-service.js)**, ou seja, os testes deve ser realizados sem acesso à API do FootballData nem ao ElasticSearch.
4. Implementar os serviços da aplicação no módulo `foca-services.js`.
 - a. Seguir uma abordagem semelhante à utilizada em `foca-web-api.js` no desenvolvimento das funcionalidades deste módulo e respectivos testes unitários
 - b. À semelhança do módulo `foca-services.js` os testes unitários devem ser executados sem acesso à API do FootballData nem ao ElasticSearch.
5. Implementar em último lugar o módulo de entrada da aplicação servidora: `foca-server.js`. Para este módulo não é necessário criar testes unitários, uma vez que este não deve implementar qualquer lógica que não seja receber alguns argumentos da linha de comando (configuração), registar rotas e iniciar o servidor web.