



**Faculdade de Ciências e Tecnologia
Universidade de Coimbra**

Licenciatura em Engenharia Informática

Relatório Sistemas Distribuídos

scoreDEI: Resultados desportivos em direto

Trabalho realizado por:

**Miguel Ferreira | 2019214567 | PL5
Thomas Fresco | 2019219057 | PL5**

2021/2022

**Hugo Dinis Pereirinha da Silva Amaro
Filipe João Boavida Mendonça Machado de Araújo**

Índice

Arquitetura do software	3
Detalhes sobre a implementação	4
Distribuição das Tarefas	4
Testes Feitos à Plataforma	5
Referências Bibliográficas	7

1. Arquitetura do software

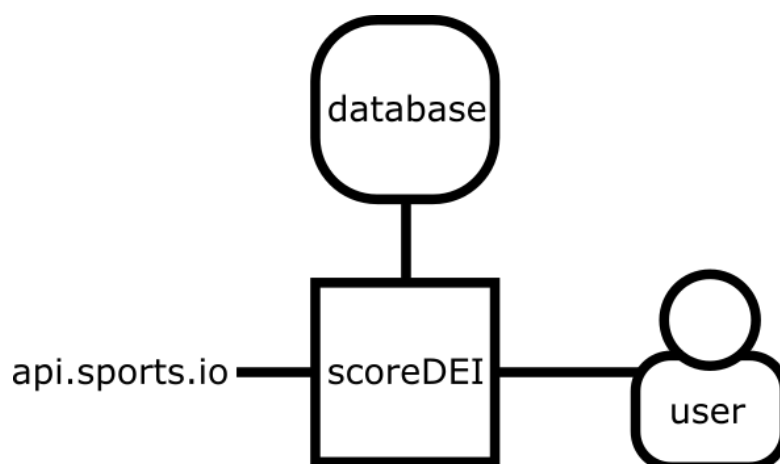


fig. 1 - Representação simples da Arquitectura scoreDEI

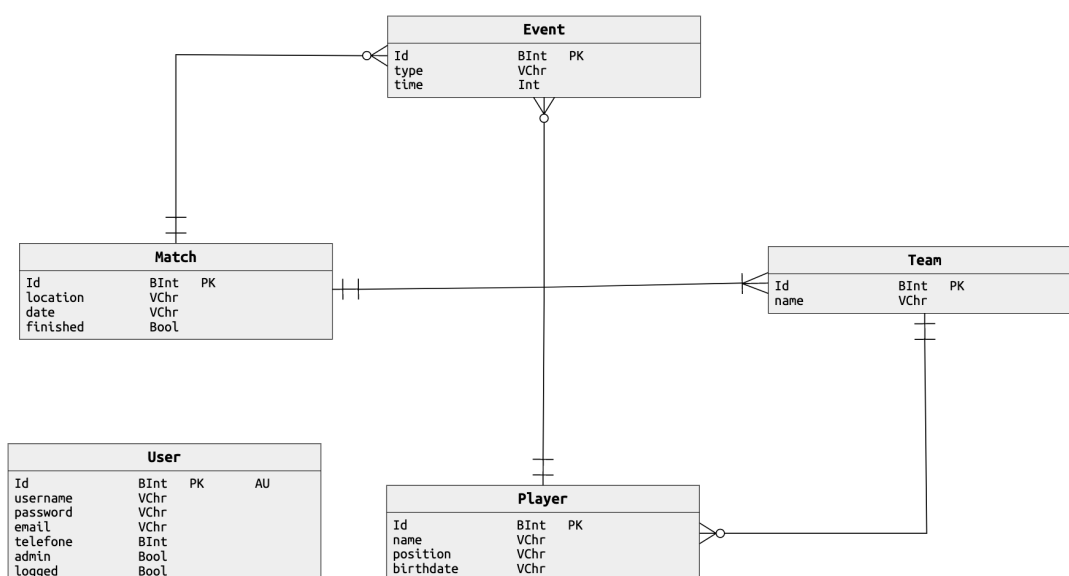


fig. 2 - Diagrama entidade-relacionamento scoreDEI (feito no ONDA)

A arquitetura desenvolvida no âmbito deste projeto gira à volta de cinco classes Java, cujos nomes são auto-explicativos: User, Player, Match, Team e Event. As relações entre elas e suas constituições são bem perceptíveis no diagrama entidade-relacionamento acima apresentado (fig. 2). Para cada uma destas classes, foram criadas 2 classes “auxiliares”, uma “service” e uma “repository”, que ajudaram na procura em tabelas da base de dados (pela facilitação de queries de PostgreSQL, por exemplo).

No que diz respeito a controladores, o ficheiro “DataController.java” de tudo trata, contendo todos os endpoints organizados, associados a mappings “GET” ou “POST” (ou “PUT”). Apresentam-se, como “GET”s, os pedidos de leitura das tabelas (como, por exemplo, as listas de Users, Players, Matches, Teams ou Events) e, como “POST”s, os pedidos de inserção e edição de tabelas (como, por exemplo, a adição de novos Users, Players, Matches, Teams ou Events).

No campo das vistas, para além da landing page (home), login e registo, desenharam-se páginas de visualização e edição/adição para cada uma das classes User, Player, Match, Team e Event. Além disso, criou-se, como pedido no enunciado, uma vista de estatísticas e uma vista de eventos por jogo.

2. Detalhes sobre a implementação

Pela presença de uma navbar “constante”, foram criadas vistas diferentes para os mesmos requests para ocultar funcionalidades dos utilizadores que, por uma questão de segurança e de posse de cargos não adequados, não tinham permissão para tais ações (mediante necessidade). Na mesma linha de pensamentos, o botão de “edit” presente em cada linhas das tabelas de jogos, jogadores e equipas fica oculto para os utilizadores registados e não registados (apenas aparece para os administradores) e o botão de adição de evento num jogo fica oculto para os utilizadores não registados (apenas aparece para os registados e para os administradores). Esta proteção baseou-se na constante verificação da variável “admin” (boolean) presente na classe “User”, para o utilizador autenticado no momento.

Em relação à API externa, talvez a tarefa mais difícil, foi gerada uma *key* através do link fornecido no enunciado, a qual foi usada juntamente com o URL e o endpoint para obter uma resposta em formato JSON, que por si foi tratada selecionando os dados relevantes (nome, equipa, data de nascimento) sobre equipas e jogadores usados como base para a plataforma.

3. Distribuição das Tarefas

- **Miguel Ferreira** - especial atenção à API externa e endpoints.
- **Thomas Fresco** - especial atenção ao login e parte visual da plataforma.

A maior parte do projeto foi desenvolvida em conjunto, com recurso a pair programming e a um repositório partilhado no GitHub.

4. Testes Feitos à Plataforma

NOTA: O login é bloqueado a apenas um utilizador de cada vez.

Teste	Comp. Observado	pass / fail
Login com credenciais válidas (utilizador registado ou admin)	Landing page com a mensagem "Welcome to scoreDEI," seguido do nome do utilizador acabado de autenticar.	
Login com palavra-passe inválida	Retorno à página de login	
Login com username não registado	Retorno à página de login	
Clique em "Matches" como utilizador não registado	Página com a lista de jogos	
Clique em "Teams" como utilizador não registado	Página com a lista de equipas	
Clique em "Players" como utilizador não registado	Página com a lista de jogadores	
Clique em "Statistics" como utilizador não registado	Página com as estatísticas sobre jogos, vitórias, derrotas e empates	
Clique em "Login" como utilizador não registado	Página de login	
Clique em "Matches" como utilizador registado	Página com a lista de jogos	
Clique em "Teams" como utilizador registado	Página com a lista de equipas	
Clique em "Players" como utilizador registado	Página com a lista de jogadores	
Clique em "Statistics" como utilizador registado	Página com as estatísticas sobre jogos, vitórias, derrotas e empates	
Clique em "Logout" como utilizador registado	Landing page com a mensagem "Welcome to scoreDEI, Stanger!"	
Clique em "events" numa linha da páginas de jogos	Páginas com os eventos desse jogo	
Clique no botão "Add event"	página com formulário de criação de novo evento	

Clique em “Criar Evento”	Evento adicionado com sucesso ao respetivo jogo	
Clique em “Matches” como administrador	Página com a lista de jogos, com um botão para adicionar um jogo	
Clique em “Teams” como administrador	Página com a lista de equipas, com um botão para adicionar uma equipa	
Clique em “Players” como administrador	Página com a lista de jogadores, com um botão para adicionar um jogador	
Clique em “Statistics” como administrador	Página com as estatísticas sobre jogos, vitórias, derrotas e empates	
Clique em “Logout” como administrador	Landing page com a mensagem “Welcome to scoreDEI, Stanger!”	
Clique em “Users” como administrador	Página com a lista de utilizadores, com um botão para registar um novo utilizador	
Clique em “Regist New User”	página com formulário de criação de novo utilizador	
Submissão do formulário de criação de novo utilizador	Utilizador adicionado com sucesso à tabela de utilizadores	
Aparecer apenas a lista de jogadores das equipas presentes no jogo em questão, ao adicionar um evento	Aparecem todos os jogadores de todas as equipas, em vez de só as que interessam	

5. Referências Bibliográficas

- Saliev, Y., consultado a 28 de maio em: <https://stackoverflow.com/questions/51686944/org-springframework-expression-spel-spel-evaluationexception>
- Ngo G., consultado a 28 de maio em: <https://hellokoding.com/registration-and-login-example-with-spring-xml-configuration-maven-jsp-and-mysql/>
- Rai, A., consultado a 29 de maio em: <https://www.websparrow.org/spring/spring-data-derived-findby-query-methods-example>
- w3schools, consultado a 31 de maio em: <https://www.w3schools.com/bootstrap/default.asp>
- RapidAPI Staff, consultado a 1 de maio em: <https://rapidapi.com/blog/how-to-use-an-api-with-java/>
- redocly, consultado a 1 de maio em: <https://api-sports.io/documentation/football/v3>

Ferramentas externas utilizadas:

- ONDA: <http://onda.dei.uc.pt/v3/>