

**Recolha de Informação de Eventos  
Desportivos usando Técnicas de Mineração de  
Dados:  
Investigação, Desenho e Implementação de uma  
Solução Web baseada na Nuvem**

**Fábio Alexandre Teodoro Mota**

Dissertação para obtenção do Grau de Mestre em  
**Engenharia Informática**  
(2º ciclo de estudos)

Orientador: Prof. Doutor Lúcio Miguel Studer Ferreira  
Co-orientador: Prof. Doutor Sebastião Augusto Rodrigues Figueiredo Pais

**Covilhã, junho de 2020**



# Agradecimentos

Começo pelo meu orientador, Prof. Doutor Lúcio Miguel Studer Ferreira, docente do Departamento de Informática da Universidade da Beira Interior, por ter aceite a orientação desta dissertação e estar sempre disponível para dar os melhores conselhos em relação ao projeto que foi desenvolvido.

Ao meu co-orientador, Prof. Doutor Sebastião Augusto Rodrigues Figueiredo Pais, docente do Departamento de Informática da Universidade da Beira Interior, por ter estado presente ao longo do desenvolvimento da dissertação e por partilhar os seus conhecimentos.

À minha família, pelo esforço, apoio e confiança deposita em mim ao longo do meu percurso académico. Em especial aos meus pais, Paula Cristina Vaz Teodoro Mota e Hélder Gil Mota, com quem pude contar sempre e estiveram presentes em todas as alturas da minha vida.

Ao Prof. Doutor Bruno Filipe Rama Travassos, docente do Departamento de Desporto da Universidade da Beira Interior, e ao Dário Emanuel Proença Gaspar, treinador principal de Futsal do GD Mata/AAUBI, pela partilha de conhecimentos e pelos comentários acerca da importância da aplicação da tecnologia na área do desporto.

A todos os meus colegas de curso, pelo espírito de entreajuda e cooperação demonstrado ao longo dos anos.

A todos os meus amigos, que estiveram sempre presentes e foram importantíssimos para esta conquista.

Por último, à Universidade da Beira Interior.

A todos um enorme obrigado!



## Resumo

Esta dissertação consistiu no desenvolvimento de uma aplicação Web de Recolha de Informação de Eventos Desportivos usando Técnicas de Mineração de Dados: Investigação, Desenho e Implementação de uma Solução Web baseada na Nuvem para a Mixed Creations, um projeto de design desportivo desenvolvido por mim em 2017 que promove atletas, essencialmente de futebol e futsal, acrescentando-lhes valor através da imagem desportiva que transmitem para o público. O desporto possui uma grande quantidade de dados associados, o grande desafio passa por sintetizá-los o mais rápido possível, de forma a que estes possam ser utilizados pelos utilizadores. Ao observarmos outras empresas ou marcas que façam o mesmo trabalho que a Mixed Creations, rapidamente chegamos à conclusão que a criação de uma aplicação Web que permita a obtenção de dados relacionados com desporto (jogadores, equipas e jogos) de forma seletiva e à qual é ainda possível associar cartazes é algo único e que não existe no mercado. Assim sendo, a Mixed Creations consegue diferenciar-se dos seus concorrentes ao introduzir uma solução Web que implemente um modelo automático e não supervisionado. A mineração de dados segue o modelo ETL, a extração dos dados é feita através da recuperação de dados contidos na fonte de informação. Em seguida, é efetuado o processo de tratamento dos dados, terminando com o respetivo carregamento na base de dados. O utilizador quando efetuar pedidos de dados, estes serão recebidos pela aplicação que através das camadas *Model*, *View* e *Controller* do padrão MVC implementado vão efetuar as respetivas solicitações e respostas. No final de todo o processo, o utilizador vai conseguir visualizar os dados pedidos através de tabelas.

Com a criação da aplicação passou a ser possível a adição, edição e remoção de vários tipos de dados como equipas, jogadores, jogos e fotografias dos clientes da Mixed Creations à base de dados. O facto das fotos serem disponibilizadas por links permite o acesso a todo o histórico de fotos elaboradas para um determinado jogador que participou num determinado jogo sem ocupar memória da base de dados com o tamanho das imagens. A aplicação foi lançada na nuvem na página <http://mixedcreations.herokuapp.com/> [1]. Os trabalhos finais são inseridos na aplicação após serem carregados para a nuvem e depois apenas é necessário inserir o link do produto final. Os objetivos foram alcançados com sucesso, sendo criada uma ferramenta inovadora que reduz o tempo dispendido na procura da informação, antes demorava aproximadamente 8 horas semanalmente enquanto agora demora 2 minutos, e cria todas as condições para o arranque da época desportiva 2020/21.

## Palavras-Chaves

Mixed Creations; Marketing desportivo; Aplicação Web; Mineração de dados; Mineração na Web; Nuvem; Solução Web; Automático; Não supervisionado;



# Abstract

This dissertation consisted in the development of a Web application for collecting information of sport events using data mining techniques: research, design and implementation of a cloud-based web solution for Mixed Creations, a sport design project developed by me in 2017 which promotes athletes, essentially football and futsal ones, adding value to them through their sport image, that they transmit to the public. Sport has a large amount of associated data, so the big challenge is to synthesize them as quickly as possible, so that can be used by users. When looking at other companies or brands that do the same job as Mixed Creations, we quickly came to the conclusion that the creation of a web application that allows the obtaining of sports-related data (players, teams and games) selectively and which makes possible to associate posters is something unique and does not exist in the market yet. Therefore, Mixed Creations is able to differentiate itself from its competitors by introducing a Web solution that implements an automatic and unsupervised model. The data mining follows the ETL model. Data extraction is done by retrieving data contained in the information source. Then, the data processing process is carried out, ending with the respective loading in the database. When the user requests for data, that will be received by the application which through the layers textit Model, textit View and textit Controller of the implemented MVC standard will make the respective requests and responses. At the end of the entire process, the user will be able to view the requested data through tables.

With the creation of the application, it became possible to add, edit and remove various types of data such as teams, players, games and photographs of Mixed Creations customers to the database. The fact that the photos are made available through links allows access to the entire history of photos made for a particular player who participated in a particular game without occupying the database's memory with the size of the images. The application was launched in the cloud on the page <http://mixedcreations.herokuapp.com/> [1]. The final works are inserted in the application after being uploaded to the cloud and then it is only necessary to insert the link of the final product. The objectives were successfully achieved, creating an innovative tool that reduces the time spent (before I usually needed about 8 hours a week, while now it takes 2 minutes only) looking for information and creates all the conditions for the start of the 2020/21 sports season.

## Keywords

Mixed Creations; Sports Marketing; Web application; Data mining; Web mining; Cloud; Web solution; Automatic; Unsupervised.



# Conteúdo

<b>1</b>	<b>Introdução</b>	<b>1</b>
1.1	Enquadramento . . . . .	1
1.2	Motivação . . . . .	1
1.3	Objetivos . . . . .	3
1.4	Contributos . . . . .	3
1.5	Organização do Documento . . . . .	4
<b>2</b>	<b>Estado da arte</b>	<b>7</b>
2.1	Aplicações e Serviços associados a eventos desportivos . . . . .	7
2.1.1	Introdução . . . . .	7
2.1.2	Zerozero . . . . .	8
2.1.3	Sportradar . . . . .	8
2.1.4	Wyscout . . . . .	9
2.1.5	Outras aplicações com dados de eventos desportivos . . . . .	10
2.2	Marketing Desportivo . . . . .	10
2.2.1	Introdução . . . . .	10
2.2.2	Mixed Creations . . . . .	10
2.3	Aplicações Web . . . . .	14
2.4	Mineração de dados . . . . .	16
2.5	Artigos de referência . . . . .	18
2.6	Conclusões . . . . .	20
<b>3</b>	<b>Engenharia de Software</b>	<b>21</b>
3.1	Análise de Requisitos . . . . .	21
3.1.1	Requisitos Funcionais . . . . .	21
3.1.2	Requisitos Não Funcionais . . . . .	23
3.2	Casos de Uso . . . . .	24
3.3	Diagrama de Atividades . . . . .	26
3.3.1	Diagrama de atividades: Adicionar dados . . . . .	26
3.3.2	Diagrama de atividades: Atualizar dados . . . . .	26
3.3.3	Diagrama de atividades: Eliminar dados . . . . .	27
3.3.4	Diagrama de atividades: Visualizar dados . . . . .	27
3.4	Modelo de Base de Dados . . . . .	28
3.4.1	Modelo Entidade Relacionamento . . . . .	28
3.4.2	Modelo Físico . . . . .	30
3.5	Modos de visualização e Sumarização de dados . . . . .	32
3.6	Conclusões . . . . .	35

<b>4</b>	<b>Modelo e Implementação</b>	<b>37</b>
4.1	Arquitetura . . . . .	37
4.2	Perímetros tecnológicos da implementação . . . . .	41
4.3	Implementação do modelo de mineração . . . . .	43
4.3.1	Extração dos dados . . . . .	43
4.3.2	Tratamento dos dados . . . . .	45
4.3.3	Carregamento dos dados . . . . .	46
4.4	Implementação da aplicação Web . . . . .	47
4.4.1	Criação da base de dados . . . . .	47
4.4.2	Implementação do padrão MVC . . . . .	48
4.4.3	Estrutura das páginas e Funcionalidades . . . . .	50
4.5	Lançamento na nuvem . . . . .	59
4.6	Produto Final . . . . .	60
4.7	Conclusões . . . . .	62
<b>5</b>	<b>Testes e Análise de Resultados</b>	<b>65</b>
5.1	Testes . . . . .	65
5.1.1	Testes e Resultados Previstos . . . . .	65
5.1.2	Resultados Alcançados . . . . .	67
5.2	Análise de Resultados . . . . .	67
<b>6</b>	<b>Conclusões e Trabalho Futuro</b>	<b>71</b>
6.1	Conclusões Principais . . . . .	71
6.2	Trabalho Futuro . . . . .	72
	<b>Bibliografia</b>	<b>73</b>

# Lista de figuras

2.1	Captura de tela da página principal do Zerozero.pt . . . . .	8
2.2	Captura de tela dos dados da Sportradar . . . . .	9
2.3	Captura de tela dos dados da Wyscout . . . . .	9
2.4	Marketing desportivo nas redes sociais de um jogador . . . . .	11
2.5	Página de Facebook da Mixed Creations . . . . .	12
2.6	Página de Instagram da Mixed Creations . . . . .	12
2.7	Fotografias dos jogadores - Google Drive da Mixed Creations . . . . .	13
2.8	Google Drive da Mixed Creations . . . . .	13
2.9	Serviço Web . . . . .	14
2.10	MVC container . . . . .	16
2.11	Procedimento da Extração, Tratamento e Carregamento dos dados . . . . .	17
3.1	Diagrama de casos de uso . . . . .	25
3.2	Diagrama de atividades - Adicionar . . . . .	26
3.3	Diagrama de atividades - Atualizar . . . . .	27
3.4	Diagrama de atividades - Eliminar . . . . .	27
3.5	Diagrama de atividades - Visualizar . . . . .	28
3.6	Modelo Entidade Relacionamento . . . . .	30
3.7	Modelo Físico . . . . .	32
3.8	Modo de visualização da Página Jogadores . . . . .	33
3.9	Modo de visualização da Página Equipas . . . . .	34
3.10	Modo de visualização da Página Jogos . . . . .	34
3.11	Modo de visualização da Página Fotos . . . . .	34
3.12	Modo de visualização da Página Jogo Equipa . . . . .	35
4.1	Arquitetura da plataforma de um modelo automático e não supervisionado . . . . .	38
4.2	Perímetros tecnológicos da implementação da arquitetura . . . . .	43
4.3	Excerto de visualização do HTML do Zerozero . . . . .	44
4.4	Extração dos dados . . . . .	45
4.5	Tratamento dos dados . . . . .	45
4.6	Processo de Web Scraping para a obtenção de informação de um Jogo . . . . .	47
4.7	Criação do modelo da base de dados . . . . .	48
4.8	Implementação do padrão MVC . . . . .	49
4.9	Rota da página de edição de um jogador . . . . .	49
4.10	Métodos Get/Post . . . . .	50
4.11	Estrutura das Páginas . . . . .	50
4.12	Visualização da página da equipa SL Benfica no Zerozero . . . . .	51
4.13	Inserir a equipa na aplicação Web . . . . .	52
4.14	Visualização das equipas na aplicação Web . . . . .	52
4.15	Visualização da página do jogador Franco Cervi no Zerozero . . . . .	53

4.16	Inserir o jogador na aplicação Web . . . . .	53
4.17	Visualização dos jogadores na aplicação Web . . . . .	54
4.18	Visualização, após a inserção, de todos os próximos jogos de todas as equipas	55
4.19	Inserção dos próximos jogos do SC Braga . . . . .	55
4.20	Visualização, após a inserção dos jogos do SC Braga, de todos os próximos jogos de todas as equipas . . . . .	56
4.21	Escolha do jogador na inserção de fotos a um determinado jogo . . . . .	57
4.22	Visualização de todas as fotos existentes na aplicação Web . . . . .	57
4.23	Visualização de todas as fotos do Rodrigo Battaglia existentes na aplicação Web . . . . .	58
4.24	Inserção de forma manual de uma equipa . . . . .	58
4.25	Edição dos dados de uma equipa . . . . .	59
4.26	Aplicação Web a correr na nuvem (Heroku) . . . . .	60
4.27	Produto final da promoção de um jogo feita para o jogador Franco Cervi .	61
4.28	Registo do cartaz desenvolvido na aplicação Web . . . . .	62
4.29	Histórico de todos os cartazes realizados . . . . .	62
5.1	Tempo médio dispendido na obtenção da informação de eventos desportivos	69

# Lista de tabelas

3.1	Caraterísticas - Modelo Entidade Relacionamento . . . . .	28
3.2	Caraterísticas - Modelo Físico . . . . .	30
5.1	Testes e Resultados Previstos . . . . .	66
5.2	Resultados Alcançados . . . . .	67



# Acrónimos

- API** Application Programming Interface
- ETL** Extract Transform Load
- FPF** Federação Portuguesa de Futebol
- HTML** HyperText Markup Language
- HTTP** Hypertext Transfer Protocol
- HTTPS** Hyper Text Transfer Protocol Secure
- JSON** JavaScript Object Notation
- MVC** Model-View-Controller
- ORM** Object-relational mapping
- PaaS** Platform-as-a-Service
- PLN** Processamento de Linguagem Natural
- SGBD** Sistema de Gestão de Base de Dados
- UML** Unified Modeling Language
- URL** Uniform Resource Locator
- WWW** World Wide Web



# Capítulo 1

## Introdução

### 1.1 Enquadramento

Este projeto foi elaborado visando a obtenção do grau de mestre em Engenharia Informática no ano letivo de 2019/20. Encontra-se no seguimento do meu projeto final de Licenciatura em Informática Web [2], desenvolvido no ano letivo 2017/18, que consistiu no desenvolvimento de uma plataforma Web que permitisse a gestão da imagem desportiva de vários atletas. Esta plataforma encontra-se disponível em <https://mixedcreations2017.wixsite.com/mixedcreations> [3]. Com esta tese foi dada uma continuidade à sofisticação de um projeto pessoal que foi criado por mim, a Mixed Creations.

A importância dada ao desporto e a quantidade da informação na Web é cada vez maior, pelo que é muito importante para ajudar na planificação e otimização dos resultados de uma empresa, uma ferramenta que permita a obtenção dos dados que queremos em tempo real e consiga auxiliar a organização do trabalho. Se conseguirmos aliar as tecnologias de ponta às metodologias de trabalho, podemos obter excelentes resultados no que diz respeito à diferenciação perante as marcas concorrentes e à eficiência em termos do tempo dispendido na execução de uma determinada tarefa.

### 1.2 Motivação

Nos dias que correm existem com cada vez maior frequência eventos desportivos. O surgimento de novos desportos e novos escalões dentro de cada desporto, assim como o aumento significativo de participantes tem trazido uma nova realidade ao desporto, tornando-o mais atrativo e também mais gerador de riqueza, atraindo mais investimento. Além disso, muitos desportos que outrora eram praticados apenas de forma amadora, são agora feitos de forma profissional, o que obriga a um olhar muito mais estratégico para a profissão que um desportista pratica e como deve promover a sua imagem. Há sempre uma grande quantidade de informação associada a cada evento e desde o final da década de 90 que a Internet e principalmente as redes sociais mais tarde, se tornaram formas fundamentais de promoção do negócio das organizações desportivas e também da gestão de imagem dos próprios atletas. Costa (2014), em [4], aborda o facto de as redes sociais conseguirem criar comunidades online de grande dimensão e interligar as pessoas de uma forma mais eficaz, tornando-as o meio ideal de comunicar com consumidores e apoiantes, tendo em vista uma lógica de aumento das suas iniciativas de relações públicas com os seus seguidores e apoiantes. Para os clubes, estas ferramentas permitem-lhes desenvolver a sua imagem e reputação e aos jogadores acontece exatamente a mesma coisa,

já que têm ao seu dispor ferramentas que lhes permitem mostrar e promover o seu trabalho e criar mais interesse à volta dele. Para uma organização desportiva e para um jogador, sendo que o sucesso de ambos se baseia nos fãs e na sua paixão, torna-se muito mais atrativo investir num clube ou num jogador que tenha uma base sólida de media e uma comunicação cuidada e pensada nas redes sociais, atraindo melhores oportunidades e patrocínios. Por todos estes fatores, o cuidado com a imagem desportiva que o jogador transmite de si próprio para as restantes pessoas na Internet começou a ser uma preocupação e a ser visto como um negócio, já que pode influenciar toda a carreira do jogador, assim como a promoção do clube que representa para o exterior.

Uma das principais formas de comunicação dos jogadores nas redes sociais está na promoção dos próximos eventos que vão ter, assim como anúncios de contratação ou renovação de contrato no começo de uma temporada, já que são notícias que têm muito impacto para o público alvo. É neste contexto que empresas de *design* desportivo como a Mixed Creations, que apostam em realizar este tipo de conteúdos para jogadores e desportistas têm um papel preponderante para criar e acrescentar valor à comunicação dos jogadores, já que a estética e a clareza da mensagem passada têm cada vez mais importância, o que é demonstrado pela procura crescente destes serviços, tanto por parte de jogadores de forma individual, como por parte dos próprios clubes que também já querem ter uma linha de comunicação online pensada, estratégica e principalmente uniforme, utilizando um *design* idêntico em todas as suas publicações, para que crie uma identidade online vincada e facilmente reconhecível. Se as ações de promoção e *marketing* forem bem executadas, tanto jogadores como clubes poderão atingir resultados ótimos, chamando à atenção do público-alvo que definirem como prioritário. Além disso, devido ao forte crescimento das redes sociais atualmente e que não se prevê abrandar mas sim aumentar cada vez mais e ganhar cada vez mais dimensão e escala, é natural que estas sejam hoje em dia uma das principais formas exploradas para recrutar novos jogadores por parte dos agentes e dirigentes dos clubes. Se outrora se via com maus olhos contratar jogadores sem os ter visto a jogar presencialmente antes, hoje que mesmo os jogos das provas amadoras começam a ser gravados, o aumento do material audiovisual da performance dos jogadores é notório, e há uma grande aposta em contratação baseada nesse tipo de materiais. E mesmo quando os clubes não querem apostar em jogadores sem antes verem a sua capacidade de forma presencial, serve ainda assim como um ótima ferramenta de triagem inicial para saber que jogadores parecem mais interessantes para observar ao vivo, permitindo logo fazer uma pré-seleção inicial mais coerente e acertada do que acontecia antigamente. Esta possibilidade de mostrar o seu trabalho mais frequentemente e com mais qualidade, fez com que projetos como a Mixed Creations comessem a ser procurados por atletas de todos os escalões e níveis de profissionalismo, sendo que isso se prova pela crescente procura de jogadores de formação, que querem desde cedo ter uma aposta séria neste tipo de promoção online. Assim, a Mixed Creations tem aumentado o seu raio de observação e de análise de dados, procurando por vários sítios da Web relativos a várias competições diferentes e de diferentes desportos, para pesquisar as informações dos respetivos jogos por equipa e por campeonato. Assim e tendo como

incentivo o trabalho desenvolvido no projeto final de licenciatura, surgiu a ideia de desenvolver mais uma ferramenta, desta feita que fosse capaz de aumentar a eficiência e os tempos de resposta dados aos jogadores (elemento fundamental dado que o desporto necessita de respostas rápidas já que muitas vezes existem dois eventos desportivos por semana que cada jogador necessita de promover e que não podem esperar). A criação de uma aplicação Web que fizesse uma seleção específica dos dados dos clientes da Mixed Creations em tempo real é de facto uma mais valia para a organização semanal da mesma e uma aposta que tinha de acontecer para tornar o projeto viável a longo prazo e passível de crescimento constante.

### **1.3 Objetivos**

O principal objetivo deste trabalho é a investigação, desenho e implementação de uma solução Web baseada na nuvem que aplique técnicas de mineração de dados para agregar informação referente a eventos desportivos de forma automática e transparente ao utilizador. O tipo de informação a ser agregada serão todos os dados necessários dos jogadores, equipas e jogos onde participarão os clientes da Mixed Creations, a incluir num cartaz promocional desse determinado evento desportivo (equipa casa, equipa fora, data, hora, local, competição e jornada), proporcionando assim uma melhoria na organização e na eficiência dos tempos de resposta. É proposto um sistema de protótipo totalmente automático e não supervisionado, isto é, não haverá incoerências porque estamos a falar do uso de uma fonte de informação que é uma referência mundial. O modelo implementado extrai e integra os dados por trás do formulário de pesquisa na base de dados. Outro objetivo a ser implementado é o da aplicação Web permitir a associação de fotografias a jogos e jogadores, assim num futuro deverá ser possível visualizar os cartazes que foram realizados para um determinado jogo ou então ver mesmo todos os cartazes que foram realizados para um determinado jogador. O facto de não existir nenhuma aplicação Web que conjuga todas as funcionalidades anteriormente descritas, torna a aplicação Web desenvolvida uma ferramenta muito poderosa e de uso bastante eficiente. Em termos científicos ao ser usado o *web mining* faz com que o processo de adição de novos dados seja efetuado de forma totalmente automática e proporciona um acesso à informação de forma muito mais rápida.

### **1.4 Contributos**

A aplicação Web desenvolvida que se encontra disponível em <http://mixedcreations.herokuapp.com/> [1] é uma grande contribuição para a Mixed Creations no que diz respeito à melhoria na gestão do trabalho semanal e conseqüentemente aumentará a capacidade dos tempos de resposta da mesma. Trata-se de uma ferramenta muito robusta que obtém os dados em tempo real e as mais recentes informações dos eventos desportivos agendados. Para além de ser um grande contributo para a agenda semanal e para a eficiência dos tempos de resposta da Mixed Creations, é também uma grande mais valia para o ar-

mazenamento de trabalhos realizados, isto é, a aplicação Web ao permitir o registo de fotografias através de links faz com que seja possível aceder a todos os trabalhos que foram efetuados no passado filtrando a pesquisa por jogo ou por jogador. Após efetuar testes com e sem o uso da aplicação Web, os resultados são bastante positivos após o desenvolvimento da mesma. É possível concluir que o tempo que era despendido na obtenção das informações passa a ser tempo disponível para outro tipo de trabalhos, assim como desassociar o crescimento da carteira de clientes com um aumento exponencial da carga de trabalho, dois fatores que por si só já não valiam mais. Outro grande contributo e fator diferenciador tem a ver com a parte financeira, o modelo que implemento evita gastos na aquisição de APIs que forneçam a informação necessária para o correto funcionamento da aplicação Web.

*”No desporto, fruto da grande quantidade de dados disponíveis na comunicação social, nas plataformas de análise de rendimento, ou mesmo dos dados recolhidos pela equipa técnica, o grande desafio que se coloca passa pela capacidade de sintetizar e compreender esses dados de forma simples e útil para o treinador e equipa técnica. Esta plataforma, desenvolvida pelo Fábio, cria as bases para a integração de informação de diferentes plataformas, disponibilizando-as para qualquer agente desportivo que a possa vir a utilizar.”*

*Bruno Travassos,  
docente do Departamento de Ciências do Desporto da Universidade da Beira Interior*

*”Parece-me que a passos largos o mundo tecnológico caminha lado a lado com a vertente desportiva naquilo que possa ser a sintetização, facilidade e rapidez no acesso a dados capazes de ajudar a tomar decisões no mundo do desporto. A forma como o Fábio pensou e construiu este produto pode ser uma alavanca para ajudar na gestão dos clubes, na aquisição e seleção de jogadores e ser uma mais valia para a equipa técnica com o objetivo de aumentar o rendimento desportivo dos jogadores/equipa.”*

*Dário Gaspar,  
treinador de Futsal do GD Mata/AAUBI*

## **1.5 Organização do Documento**

De modo a refletir o trabalho que foi feito, este documento encontra-se estruturado da seguinte forma:

1. O Capítulo 1 – **Introdução** – apresenta o projeto, a motivação para a sua escolha, o enquadramento para o mesmo, os seus objetivos e a respetiva organização do documento.
2. O Capítulo 2 – **Estado da arte** – apresenta os conceitos que servem de referência e o que já existe em torno do assunto investigado.

3. O Capítulo 3 – **Engenharia de Software** – descreve os requisitos funcionais e não funcionais, assim como os casos de uso resultantes destes, os diagramas de atividades, os modelos de base de dados e também, os modos de visualização e sumariação.
4. O Capítulo 4 – **Modelo e implementação** – descreve a arquitetura, a implementação, o modelo de mineração de dados, a implementação da aplicação Web, o lançamento na nuvem e ilustra o exemplo de um produto final.
5. O Capítulo 5 – **Testes e Análise de Resultados** – aborda os testes efetuados e faz uma análise comparativa das principais diferenças nos tempos de resposta no antes e pós desenvolvimento da aplicação Web.
6. O Capítulo 6 – **Conclusões e Trabalho Futuro** – descreve as conclusões tiradas após a implementação do projeto e as propostas de trabalho futuro.



# Capítulo 2

## Estado da arte

O segundo Capítulo apresenta o estado da arte. Na Secção 2.1 são apresentados vários exemplos de aplicações e serviços associados a eventos desportivos que existem no mercado. A Secção 2.2 faz um enquadramento da importância do *marketing* desportivo e o quão bons resultados este pode atingir quando é bem trabalhado. São apresentados vários exemplos de aplicação e são especificados os serviços que a Mixed Creations oferece. Na Secção 2.3 contém alguns conceitos relacionados com as aplicações Web, onde são abordados os serviços Web, a computação na nuvem e o *Model View Controller* (MVC). Na Secção 2.4 é falado sobre a mineração de dados, sendo descritos alguns conceitos como o processamento de linguagem natural, o uso de expressões regulares e o procedimento *Extract Transform Load* (ETL). A Secção 2.5 aborda alguns dos artigos de referência que foram tidos em conta no desenvolvimento desta dissertação. Para finalizar o Capítulo, seguem-se umas pequenas conclusões na Secção 2.6.

### 2.1 Aplicações e Serviços associados a eventos desportivos

#### 2.1.1 Introdução

Atualmente o mundo do desporto está cada vez mais ligado às tecnologias de ponta, o aparecimento da Internet fez com que houvesse uma explosão em termos da visibilidade do desporto no mundo. Há cerca de 25 anos, antes da explosão da Internet, o acesso às informações dos eventos desportivos era feito por via dos jornais. O aparecimento dos *websites* desportivos fez com que houvesse uma maior facilidade de acesso à informação. Atualmente, podemos aceder às informações de determinadas competições a qualquer hora e qualquer lugar. Existem vários *websites* que disponibilizam a informação dos eventos desportivos que se vão realizar (data, hora, local, competição, etc...), bem como dos que já se realizaram (resultado, marcadores, ficha de jogo, etc...). Um exemplo disso é o zerozero que será o mais explorado durante esta tese. A alimentação desses dados nos *websites* pode ser feita de duas maneiras, manualmente ou através de uma API. A introdução dos dados de forma manual envolve mão humana, pessoas que trabalham nesses *websites*, estando constantemente a atualizar os dados, neste caso é natural que o acesso aos dados em tempo real possa sofrer alguns atrasos, como por exemplo quando é realizado um sorteio de uma determinada competição. As APIs conseguem oferecer uma maior rapidez de resposta, como é o caso da *Sportradar* que emprega milhares de pessoas. No caso dos resultados ao vivo, a *Sportradar* têm várias pessoas a cobrir um determinado jogo. Há medida que acontece alguma situação relevante, esta é imediatamente monitorizada e distribuídas pelas diversas aplicações e *websites* (sites desportivos, casas de apostas, entre outros) que possuem os serviços da *Sportradar*.

### 2.1.2 Zerozero

O Zerozero [5] é um site dedicado ao desporto, sendo considerado a maior base de dados do mundo desportivo de acesso livre. São disponibilizadas informações sobre competições nacionais e internacionais em cada época, apresentando estatísticas, resultados e classificações. O Zerozero conta com imensos colaboradores e trata essencialmente as informações de Basquetebol, Andebol, Hóquei em Patins, Voleibol, Basquetebol, Futebol, Futebol de Praia e Futsal. No que diz respeito aos dados das modalidades, são vários os colaboradores que trabalham na introdução dos dados após a definição das datas e horários pelas entidades responsáveis das modalidades em causa, a sua vasta equipa de trabalho permite-lhe estar sempre em cima das principais ocorrências. Em relação ao futebol e com a existência de serviços que disponibilizam as informações em tempo real, o Zerozero possui algoritmos que alimentam a base de dados de forma mais rápida à semelhança do que será feito na aplicação Web realizada nesta dissertação. Na figura 2.1. podemos observar uma captura de tela da página zerozero.pt onde é visível as divisórias da página pelas diversas modalidades.



Figura 2.1: Captura de tela da página principal do Zerozero.pt

### 2.1.3 Sportradar

A Sportradar [6] é líder global no entendimento e no aproveitamento dos conteúdos digitais e dados desportivos para os clientes em todo o mundo. Fornece serviços para casas de apostas, federações desportivas nacionais e internacionais e disponibiliza uma vasta gama de serviços de ponta para empresas de mídia. É uma organização internacional, emprega mais de um milhar de pessoas espalhadas por mais de 30 locais em todo o mundo. A inovação tecnológica e um profundo entendimento das necessidades dos negócios levou a um rápido crescimento. A Sportradar trabalha para mais de 600 empresas em mais de 80 países. A distribuição de desportos da Sportradar cobre mais de 325.000 eventos desportivos e alimenta clientes como Google e Fox Sports. A Sportradar também fornece

dados para o *Facebook Sports Stadium*. Na Figura 2.2 está ilustrado um exemplo de um computador que possui vários tipos de dados, em tempo real, de um determinado evento. No exemplo apresentado, são esses dados que vão ser enviados através das APIs diretamente para as casas de apostas.



Figura 2.2: Captura de tela dos dados da Sportradar

#### 2.1.4 Wyscout

A *Wyscout* [7] é uma plataforma digital paga que disponibiliza serviços vocacionados para os agentes desportivos e analistas de jogos. Além de permitir a análise de vídeo de um determinado jogo, possui uma forte base dados com todos os dados das performances das equipas, jogadores e treinadores nos vários jogos. A *Wyscout* envia os dados através de uma API de sistema padrão em formato *JavaScript Object Notation* (JSON). Na Figura 2.3 está ilustrado o exemplo de uma visualização da quantidade de dados de um determinado jogador que está armazenada na base de dados da plataforma.

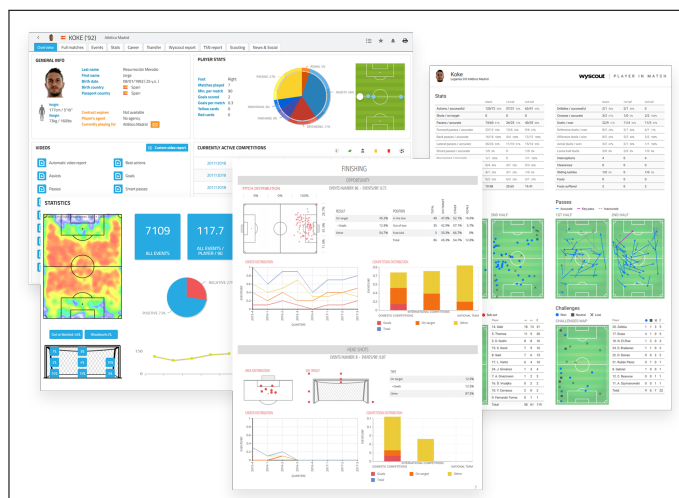


Figura 2.3: Captura de tela dos dados da Wyscout

### **2.1.5 Outras aplicações com dados de eventos desportivos**

Para além do Zerozero existem inúmeras aplicações que fornecem informações relativamente a eventos desportivos antes e ao vivo. Alguns exemplos disso são por exemplo o *flashscore* [8], onde é possível ver os eventos desportivos de maior relevância de cada desporto que irão decorrer nos próximos 7 dias e os resultados dos que ocorreram nos últimos 7 dias, para isso basta selecionar o desporto e o dia em que queremos pesquisar os eventos que ocorreram ou irão ocorrer. Alguns exemplos semelhantes são também o do *sofascore* [9], *livescore* [10], *besoccer* [11], entre outros. Existem também *websites* de jornais que disponibilizam informações relativamente a eventos desportivos.

## **2.2 Marketing Desportivo**

### **2.2.1 Introdução**

O *marketing* desportivo nos dias de hoje é cada vez mais importante na comunicação com os consumidores da informação, bem como na reputação de um clube ou de um determinado jogador. A preocupação com as redes sociais e da imagem que é transmitida subiu de tal ordem que atualmente, é um meio importantíssimo na atração de novas oportunidades e patrocinadores. Uma das principais formas dos jogadores comunicarem com os seus apoiantes está na promoção dos eventos desportivos que vão ter pois as notícias têm muito impacto para o público alvo. Seguindo essa ordem de ideias, é fácil de perceber que as empresas que efetuem trabalhos de *design* para jogadores e clubes, como o caso da Mixed Creations, necessitam sempre das informações corretas e com a maior brevidade possível de forma a poderem organizar o seu trabalho. O mesmo se aplica a agentes desportivos que queiram acompanhar semanalmente os seus jogadores ou até mesmo para observadores que pretendam acompanhar o desempenho de um jogador num determinado jogo.

### **2.2.2 Mixed Creations**

A Mixed Creations [1] é um projeto desenvolvido por mim, que teve início em 2017 e que consiste numa marca de design com foco no *marketing* desportivo. O principal objetivo é o tratamento da imagem desportiva tendo em vista a promoção dos jogos de várias equipas e de vários jogadores. Na Figura 2.4, tal como falado na Secção 2.2.1, podemos observar as redes sociais de um jogador da Mixed Creations, onde podemos visualizar uma linha padrão de *design* que é seguida na interação do jogador com os seus seguidores. A imagem que este passa para o exterior é importantíssima para a sua carreira e pode possibilitar a obtenção de melhores contratos futuros.

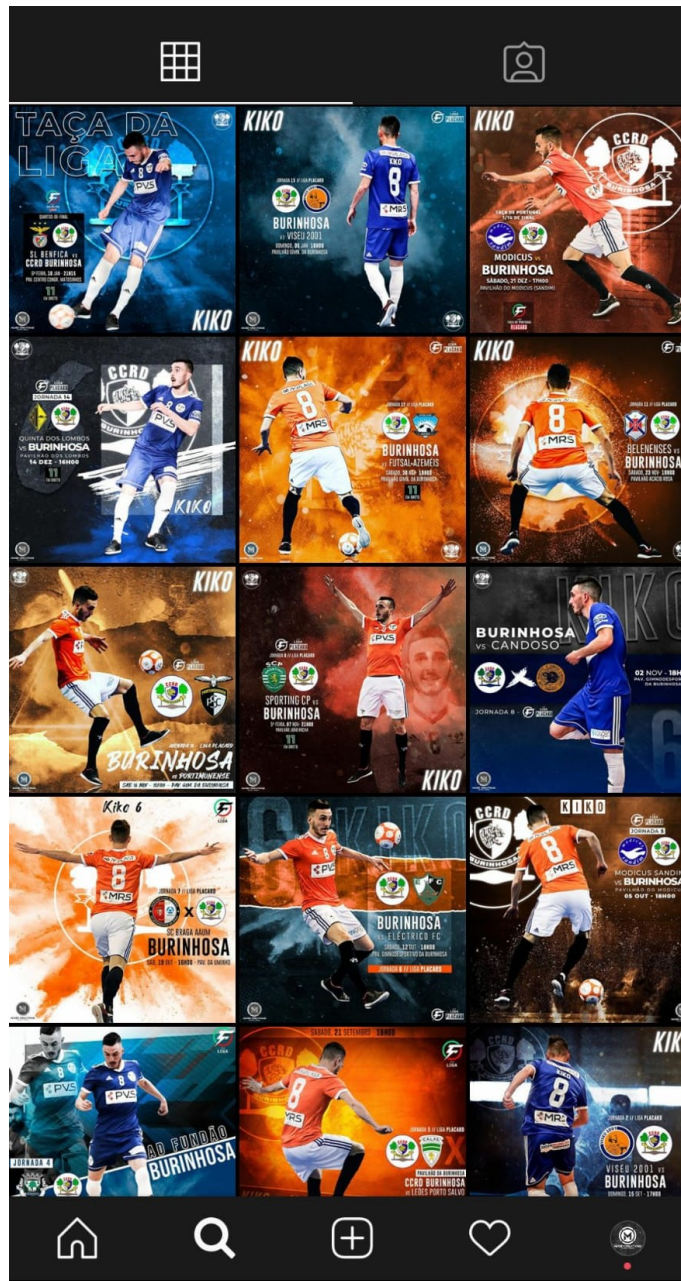


Figura 2.4: Marketing desportivo nas redes sociais de um jogador

As redes sociais da Mixed Creations servem como um meio de comunicação com os seus seguidores, bem como da publicação dos trabalhos desenvolvidos. As campanhas de *marketing* são um fator muito importante para o sucesso da marca e para o aumento de vendas. Assim sendo, a maioria das despesas envolvidas são direcionadas sobretudo às redes sociais, nomeadamente ao *Facebook* [12] e ao *Instagram* [13], como meios de atrair novos clientes e manter os atuais. Nas Figuras 2.5 e 2.6 estão capturas de ecrã que demonstram o aspeto das páginas.

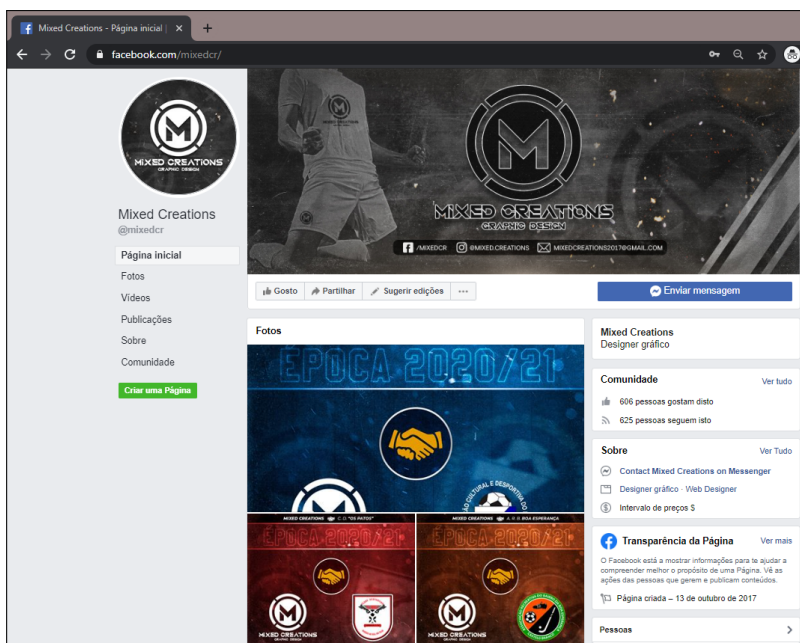


Figura 2.5: Página de Facebook da Mixed Creations

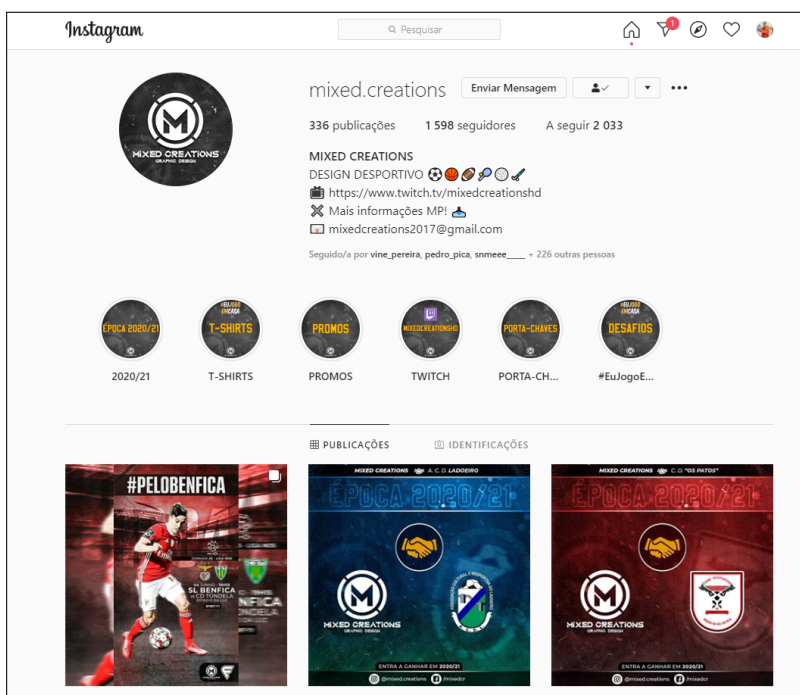


Figura 2.6: Página de Instagram da Mixed Creations

O *Google Drive* é um serviço de armazenamento na nuvem que lida com uma grande quantidade de dados. A Mixed Creations utiliza bastante este serviço para armazenar fotografias e ficheiros relevantes à organização da mesma. Na Figura 2.7 é possível observar no canto inferior esquerdo a quantidade de memória que é ocupada na *Google Drive* da Mixed Creations, neste caso até ao momento estariam a ser usados 5,8 GB de dados. Ao lado direito é possível ver as pastas correspondentes aos clubes, onde lá dentro estão todas imagens de cada jogador.

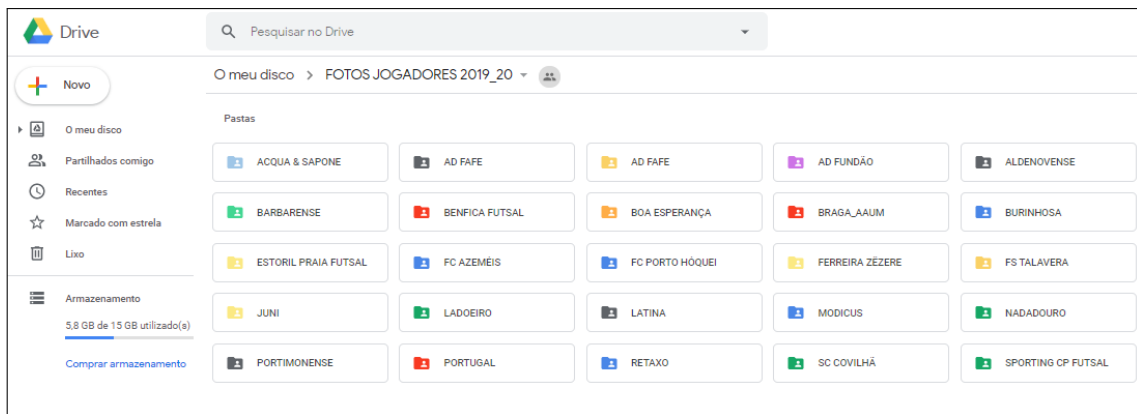


Figura 2.7: Fotografias dos jogadores - Google Drive da Mixed Creations

Na Figura 2.8 podemos observar o exemplo de um documento presente na *Google Drive* da Mixed Creations onde são ilustrados os comentários em tempo real e a edição do documento por mais que um utilizador ao mesmo tempo, possibilitando assim o trabalho em equipa.

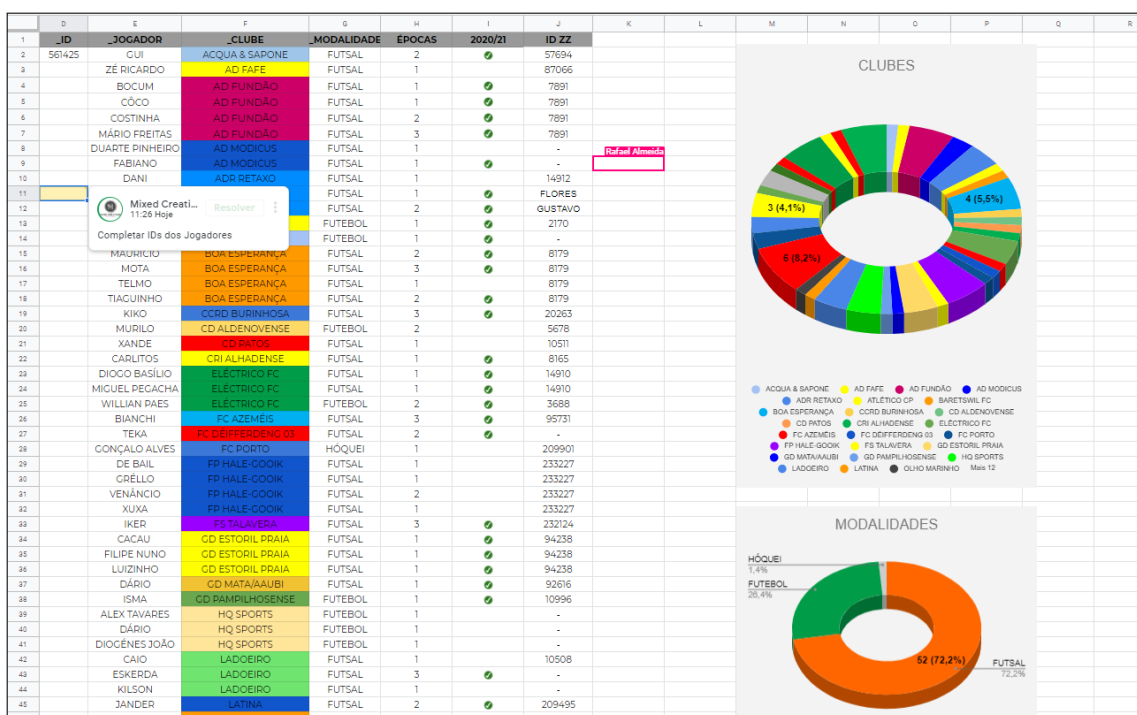


Figura 2.8: Google Drive da Mixed Creations

Como podemos observar a implementação da tecnologia na área do desporto é uma mais valia e serve como uma grande motivação para o desenvolvimento desta dissertação. O setor desportivo envolve muitas pessoas a trabalhar na área e há cada vez mais equipas a investir dinheiro neste tipo de tecnologias. Uma das principais lacunas é o facto de não existir APIs que forneçam as informações de todas as modalidades praticadas em Portugal e as que são mais completas em termos de quantidade de dados são todas pagas. Assim, a aplicação Web a ser desenvolvida será uma mais valia porque virá dar resposta a esse problema comum, que é o da procura de informações o mais atualizadas possíveis, sem

ter qualquer custo acrescido.

## 2.3 Aplicações Web

A contínua evolução da era digital leva o *marketing* digital para um patamar de extrema importância na vida das empresas. Este permite garantir uma forte presença na Internet, construir uma audiência qualificada e permite ainda monitorizar os resultados de diversas campanhas publicitárias com precisão. A Internet é uma rede que suporta vários tipos de protocolos de comunicação e disponibiliza vários tipos de serviços. Um deles são os *websites* e muitos deles implementam serviços. Guruge (2004), em [14], define os serviços Web como soluções utilizadas na integração de sistemas bem como na comunicação efetuada entre diferentes aplicações, permitindo a receção e envio de dados em *websites* com linguagem *HyperText Markup Language* (HTML). Na Figura 2.9 temos um exemplo do funcionamento de um *web service*. Do lado direito temos uma página Web que acede a um *Uniform Resource Locator* (URL) que tem um caminho, denominado de serviço Web que vai consultar à base de dados os dados que foram pedidos através da aplicação.



Figura 2.9: Serviço Web

O *Hyper Text Transfer Protocol* (HTTP) é o protocolo de comunicação responsável por essa transferência de dados na *World Wide Web* (WWW). Um *interface* de programação de aplicações, denominado de *Application Programming Interface* (API) faz a conexão de dois programas, efetuando a comunicação entre eles e específica a forma de interação dos *softwares*. As aplicações são um *software* de um projeto para aplicar, utilizar ou executar uma determinada função ou tarefa para satisfazer as necessidades do utilizador. As aplicações Web são um sistema informático projetado para a sua utilização através de um navegador na Internet. Estas podem ser executadas a partir do dispositivo do utilizador (local) ou através de um servidor HTTP.

A computação na nuvem é um processo cada vez mais importante para as empresas que absorvem, analisam e processam diversos tipos de dados todos os dias. Os métodos de gestão de informações que não são centrados na tecnologia tendem a ser mais lentos, terão um custo mais elevado e são mais suscetíveis a erros de mão humana. Para colmatar estas fragilidades, surge a computação na nuvem. Os serviços e aplicações Web na nuvem possuem enormes benefícios para as empresas. Destaca-se a excelente escalabilidade, isto é, a capacidade do sistema estar preparado para aumentar o número de recursos assim que seja exigido, o que faz com que desapareça a necessidade de otimização dos servidores internos. A nuvem oferece isso independentemente do processamento que seja necessário

naquele determinado momento. A nuvem está em todo o lado, o que garante uma excelente mobilidade, é possível aceder aos arquivos em qualquer momento e em qualquer lugar, aumentando assim a produtividade. Uma outra grande vantagem é a possibilidade de trabalhar em equipa num mesmo documento, existindo um histórico de alterações e sendo possível a comunicação através de *chats* e comentários. Mais vantagens são o custo/benefício. Os custos de armazenamento na nuvem trazem melhor custo/benefício do que o investimento que teria de ser feito para o armazenamento interno. Por último e não menos importante, a segurança é muito maior. O armazenamento na nuvem é protegido de erros físicos, assim como de ameaças virtuais. Basta colocarmos a hipótese de que se armazenarmos os dados internamente e ocorrer uma falha elétrica, podemos não conseguir guardar os ficheiros e conseqüentemente perdê-los, enquanto que na nuvem não temos esse tipo de problema pois os dados são gravados automaticamente. Para além disso, é usada criptografia e existem medidas de segurança que excluem possíveis ameaças de acesso indevido aos dados.

O *Model View Controller* (MVC) é um padrão de arquitetura de software que divide a aplicação em três diferentes camadas:

- A camada de manipulação dos dados (*model*), são classes que representam os dados e a validação lógica destes.
- A camada de interação com o utilizador (*view*), são *templates* usados dinamicamente para gerar respostas HTML.
- A camada de controlo (*controller*), recebe solicitações por parte do utilizador, obtém os dados a partir dos modelos e especifica os *templates* de visualização que retornarão como resposta ao navegador.

Os *request* são solicitações ao *controller* que invocam rotas URL e acedem às ações do *controller*. Tal como ilustrado na Figura 2.10 o utilizador acede a uma página Web que efetua solicitações através de pedidos HTTP. Na arquitetura MVC, as solicitações HTTP chegam ao *controller* que por sua vez irá efetuar pedidos de dados ao *model*. Este último faz uma conexão à base de dados, solicita informação e retorna os dados corretos ao *controller*. Com os dados prontos a ser exibidos, a *view* possui os templates de visualização que vão permitir ao utilizador visualizar a informação.

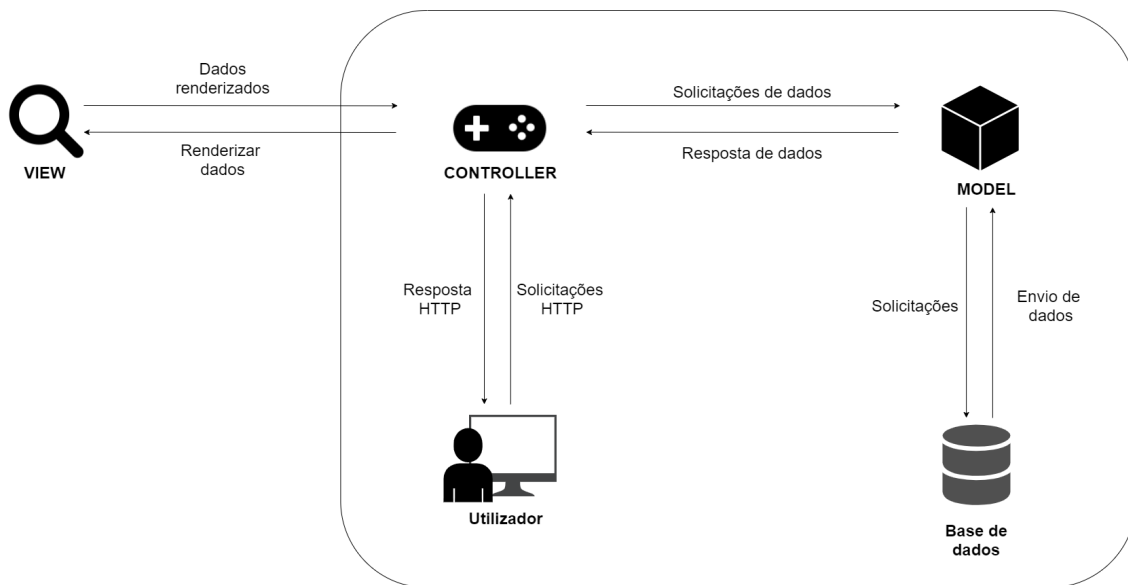


Figura 2.10: MVC container

## 2.4 Mineração de dados

O Processamento de Linguagem Natural (PLN) é a aplicação de uma teoria computacional da linguagem humana. O principal objetivo passa pela criação de sistemas capazes de comunicar com as pessoas através da linguagem humana e de forma natural. As ciências da computação, inteligência artificial e a linguística são áreas bastante importantes para o PLN. Um dos principais motivos para que exista uma necessidade da análise e tratamento da informação é que a Web neste momento é uma forma primordial de comunicação humana, existe uma crescente informação textual e a comunicação mediada por máquinas também está em expansão.

As expressões regulares são uma forma de notação para descrever a língua (conjunto de *strings* produzidas), sendo usadas por diversos editores de texto e linguagens de programação para procurar linguagens de *scripting*, manipulando o texto com base em padrões. Outros dos usos são a validação de formatos de texto, a implementação interna de um sistema de sintaxe e até mesmo a filtragem da informação em bases de dados textuais.

A mineração de dados é um conjunto de dados, estatísticas e padrões onde vamos ter dois objetos diferentes, a estrutura de mineração e o modelo de mineração. A estrutura de mineração vai armazenar as informações que definem a fonte dos dados enquanto que, o modelo de mineração vai armazenar as informações que resultam do processamento dos dados, tal como os padrões como o resultado da análise.

Para a definição do modelo de mineração de dados devem ser tidas em conta as seguintes etapas:

- Criação da estrutura de mineração que inclui as *tags* de dados que possam ser necessários;
- Seleção do algoritmo mais adequado para a tarefa analítica;

- Escolha das *tags* que vão ser usadas no modelo e especificação de qual *tag* contém o resultado desejável.

O modelo de mineração encontra-se vazio até que os dados que vão ser fornecidos pela estrutura de mineração sejam processados e analisados. Quando o modelo é processado, os dados armazenados pela estrutura de mineração são passados através de um filtro e analisados pelo algoritmo. O algoritmo irá computar um conjunto de estatísticas que descrevem os dados, identificam as regras e os padrões para preencher o modelo.

O procedimento *Extract Transform Load* (ETL) é usado para interligar vários tipos de dados que podem ser movidos e oriundos de várias fontes de informação. O procedimento passa por três fases:

- Extração de dados;
- Tratamento dos dados;
- Carregamento dos dados.

Resumidamente, os dados são extraídos de uma ou mais fontes externas, seguindo-se o tratamento dos mesmos e por fim são carregados na base de dados. A extração de dados pode ser feita usando vários tipos de ferramentas, bem como o seu respetivo tratamento. Atualmente, existem várias soluções e uma grande diversidade de tecnologias com capacidade para dar resposta ao problema. Na Figura 2.11 é ilustrado o processo descrito anteriormente.

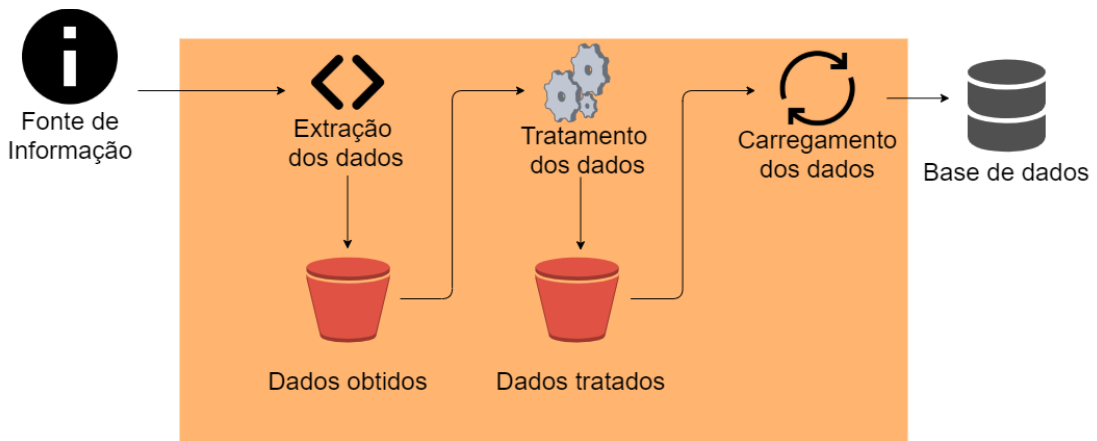


Figura 2.11: Procedimento da Extração, Tratamento e Carregamento dos dados

Na aplicação Web a ser desenvolvida irá ser efetuado *web mining* através da implementação de ferramentas de mineração de dados para extração de conteúdo das páginas ou relacionados com a estrutura (*links*) da própria página Web. A mineração de dados da aplicação Web é feita através do uso de ferramentas que acedem às fontes de informação para recuperação da informação, isto é, é efetuado *text mining* e a informação fica guardada em repositórios temporários. O processo é feito através da exploração de grandes quantidades de dados na procura de detetar padrões, sequências ou relacionamentos entre as variáveis. No caso da aplicação Web a desenvolver, o objetivo será o de utilizar essas ferramentas como meio de conseguir captar as informações relevantes de eventos desportivos

(equipas, data, hora, local, competição, etc...), de equipas (nome, estádio, etc...) e de jogadores (nome, clube, data de nascimento, etc...). Após a recuperação da informação, a informação extraída deve ser tratada com a ajuda de uma ferramenta que implemente técnicas de mineração de dados ou através do uso de expressões regulares. Por fim, a informação que até então estava contida em repositórios temporários estará pronta para ser carregada diretamente na base de dados.

## 2.5 Artigos de referência

A aplicação Web a ser desenvolvida apresenta várias tecnologias e ferramentas que ao serem interligadas formam uma ferramenta poderosa. Será implementada uma solução Web baseada na nuvem de recolha de informações de eventos desportivos através do uso de técnicas de mineração de dados. Nesta Secção serão abordados artigos importantes que abordem a temática do desporto, da mineração de dados e do *web scraping*, servindo como base para o trabalho a desenvolver.

Wang e Kuzmanovic (2018), em [15], abordam matéria bastante interessantes relativamente à obtenção de resultado ao vivo e em tempo real. É usada uma metodologia para estimar atrasos absolutos através de medições de atrasos relativos obtidos através de vídeo. Aborda também o papel das diferentes entidades na distribuição de dados desportivos, é ainda demonstrada a existência de uma discrepância de ordem de magnitude em atraso e precisão entre os provedores atuais. Apesar do foco desta tese não ser a obtenção dos resultados ao vivo, o artigo referenciado é um bom exemplo de como pode ser extraída a informação através de outras formas, neste caso através do recurso a vídeo que como desvantagem tem os *delays* que podem levar a uma menor precisão do *timing* em que o resultado é atualizado.

Costa (2014) , em [4], aborda na sua dissertação o sucesso das organizações desportivas e do jogador, sendo que a conjugação dos dois fatores atrai muitos fãs e acaba por se tornar muito mais atrativo para investir num clube ou num jogador podendo originar melhores oportunidades e conseqüentemente patrocínios. Por todos os fatores mencionados na dissertação rapidamente chegamos à conclusão que a imagem desportiva é muito importante para a comunidade, podendo influenciar a carreira de um jogador, assim como a publicidade que o clube passa para o exterior.

Thomas e Mathur (2019), em [16], falam sobre temáticas muito semelhantes às desenvolvidas nesta dissertação relativamente ao *web scraping*, os principais resultados do artigo foram o desenvolvimento de uma interface de pesquisa, indexação, processamento de consultas e técnicas eficazes de extração de dados com base na estrutura da Web. A técnica utilizada na implementação é a mesma de Matho e Singh (2016), em [17], que usa a biblioteca *BeautifulSoup* do *Python* tal como vai ser feito por mim no desenvolvimento desta dissertação. As informações na Web são organizadas com a ajuda de um bot e são definidas três fases principais. Inicialmente o *web scraper* seleciona os links desejados na Web, em seguida será feita uma extração para obter os dados contidos nos links de origem e por fim irão ser armazenados os dados obtidos num ficheiro csv. As ferramentas

escolhidas para a realização destes artigos são muito poderosas e de fácil utilização, o que facilitou a sua implementação.

Sundaramoorthy et al. (2017), em [18], falam sobre uma plataforma que agrega todas as atualizações de notícias mais recentes de vários recursos nacionais e internacionais (*NewsOne*). O *web scraping* é usado como forma para extrair o conteúdo das várias notícias. Este artigo é mais um bom exemplo da grande diversidade de áreas de aplicação do *web scraping* (jornalismo, saúde, desporto, entre outros). Este artigo tem um objetivo semelhante ao desta dissertação em relação ao tempo que é dispendido para obter uma determinada informação. Apesar de se tratarem de diferentes áreas, o objetivo principal do artigo é o de fornecer informações valiosas ao utilizador, evitando que o mesmo tenha de perder o seu tempo na leitura de todo o conteúdo de uma determinada notícia para obter a informação desejada.

Maududie (2018), em [19], baseia-se em *websites* de notícias, descrevendo uma abordagem que fornece o título, data de publicação, autor, artigo de texto limpo e endereço URL da página deixando de haver a necessidade de efetuar o processo de copiar e colar de forma manual. Essa abordagem passa por três etapas: análise da estrutura do *website*, construção de um padrão de *RegEx* e implementação de padrões seguindo um determinado conjunto de regras aquando do *web mining*. O artigo usa uma metodologia bastante prática, destacando-se pela positiva ao conseguir implementar uma solução que obtivesse os dados sem ter de copiar e colar de forma manual. As técnicas utilizadas neste artigo assemelham-se ao que será desenvolvido nesta dissertação, o *RegEx* como forma de tratamento da informação é uma mais valia.

Diouf et al. (2019), em [20], falam também sobre o *web scraping* no qual o principal objetivo é extrair informações de um ou vários *websites* e organizá-las em estruturas simples, como por exemplo em bases de dados ou arquivos csv. O artigo aborda as diferentes formas existentes de uso do *web scraping* e as suas áreas de aplicação. Tal como é feito na dissertação, o artigo aborda os tempos de resposta aquando de uma procura dos dados de forma manual e compara-os com a utilização de formas automáticas. Na procura manual, além de se tratar de uma tarefa muito complicada o tempo dispendido é muito superior, o uso de soluções automáticas aumenta bastante a eficiência do processo. Nesse sentido, o artigo é claramente uma excelente referência pois consegue apresentar resultados que vão de encontro aos que foram obtidos ao longo desta dissertação.

Marinho e Girardi, em [21], falam sobre as técnicas de mineração de dados e sobre a obtenção de informações na Internet de forma automática. Este artigo foi bastante importante para a minha dissertação pois foi tomada em conta a excelente esquematização de todas as fases a ser seguidas no processo de mineração dos dados. Com base nas principais fases descritas e através da utilização técnicas de mineração de dados, o artigo serviu de auxílio para o desenho do modelo de mineração de dados da aplicação Web.

Minewiskan (2018), em [22], fala sobre os modelos de mineração de dados e de que forma é que eles podem ser usados, fala sobre a arquitetura dos mesmos e sobre as propriedades que estes devem possuir. Este é um excelente artigo em termos de informação teórica de como deve ser construído um modelo de mineração. É uma excelente referên-

cia aquando do desenvolvimento do modelo de mineração a ser desenvolvida pois fornece informações técnicas bastante relevantes e que devem ser tidas em conta aquando da criação do mesmo.

Fernandes et al. (2020), em [23], abordam a metodologia de uma implementação ETL baseada na nuvem para dispositivos móveis. O artigo tem uma metodologia bastante bem trabalhada o que serviu de inspiração e base, a par com o MVC, para o desenvolvimento da arquitetura da minha aplicação Web que também implementa uma solução baseada na nuvem.

## **2.6 Conclusões**

Após o desenvolvimento deste Capítulo, é possível concluir que apesar da tecnologia já estar enraizada no mundo do desporto e de já existirem várias fontes de informação que possuem as informações organizadas por campeonatos como o caso do Zerozero, ou até que alimentem casas de apostas como o caso da *sportradar*, não há nenhuma plataforma que permita a obtenção dos dados de forma seletiva no ponto em que é possível seleccionarmos os nossos próprios jogadores e cada um deles esteja associado a uma determinada equipa podendo ainda serem adicionadas fotografias realizadas a cada evento desportivo. Nesse sentido, a aplicação Web desenvolvida pode ser muito boa para a organização do trabalho da Mixed Creations mas poderá também ser repensada como uma mais valia para muitos agentes ou agências de comunicação que queiram acompanhar os jogos futuros dos seus vários jogadores.

# Capítulo 3

## Engenharia de Software

O terceiro Capítulo é o da engenharia de software e está dividido em 6 secções. A Secção 3.1 contém a análise de requisitos, onde são definidos os requisitos funcionais e não funcionais. A Secção 3.2 contém o diagrama de casos de uso e a Secção 3.3 possui os vários diagramas de atividades. Os modelos físico e entidade relacionamento da base de dados são abordados na Secção 3.4. São propostos os modelos de visualização e sumarização dos dados na Secção 3.5 e para finalizar temos as conclusões na Secção 3.6.

### 3.1 Análise de Requisitos

A análise de requisitos é o resultado da leitura do enunciado do projeto, bem como de conversas com envolvidos neste. É composta por requisitos funcionais e não funcionais que especificam de forma clara as funcionalidades requeridas do Protótipo.

#### 3.1.1 Requisitos Funcionais

A pesquisa dos dados na aplicação Web baseia-se nas cinco principais tabelas da base de dados (Jogador, Equipa, Jogo, JogoEquipa e JogoFotoJogador) e vai conter as seguintes páginas:

- Página Inicial;
- Equipas;
- Jogadores;
- Jogos;
- Fotos;
- Jogo Equipa;
- Sobre.

Há 5 páginas que vão corresponder às tabelas da base de dados e em que mostram a informação registada em tabelas. Vão existir ainda mais duas páginas, uma Página Inicial e uma Página Sobre. Na Página Inicial será exibida uma foto publicitária da Mixed Creations seguindo-se uma lista com os nomes dos jogadores, as suas equipas e respetivas modalidades. Por sua vez, a página Sobre apenas irá conter um simples texto que aborda o enquadramento da aplicação Web.

Seguem-se os **requisitos funcionais** do sistema.

**RF1** A página dos jogos deverá:

- **RF1.1** permitir ao utilizador a visualização de uma tabela com um limite de exibição de 10 linhas de cada vez, com todos os jogos da base de dados que irão decorrer nos próximos 10 dias;
- **RF1.2** permitir ao utilizador a visualização de uma tabela com todos os jogos existentes na base de dados;
- **RF1.3** conter a informação detalhada sobre os jogos (Equipas, Data, Hora, Local, Competição, Jornada ou Fase da competição);
- **RF1.4** permitir que o utilizador escolha a forma como visualiza os dados da tabela, isto é, por ordem cronológica, por ordem alfabética de nomes e deve ainda permitir consultas à base de dados através de um campo de pesquisa;
- **RF1.5** permitir a atualização, edição e remoção dos jogos da base de dados;
- **RF1.6** permitir a adição de jogos à base de dados através de três possibilidades. A primeira é através da seleção de uma determinada equipa, onde o programa através do *web scraping* irá fornecer os dados dos próximos jogos, se houver, dessa determinada equipa e consequentemente adicioná-los à base de dados. A segunda é de forma igual à primeira mas em vez de adicionar de uma só equipa irá adicionar de todas as equipas existentes na base de dados da aplicação Web. A terceira é de forma manual, onde o utilizador, a partir de um formulário irá fornecer ao programa todos os dados em detalhe de um determinado jogo, esta possibilidade visa colmatar qualquer eventual falha existente na plataforma base Zerozero, constituindo por isso uma importante redundância do software.

**RF2** A página das equipas deverá:

- **RF2.1** permitir ao utilizador a visualização de uma tabela com um limite de exibição de 10 linhas de cada vez, com todas as equipas presentes na base de dados;
- **RF2.2** permitir a atualização, edição e remoção das equipas da base de dados;
- **RF2.3** permitir a adição de equipas à base de dados através de duas possibilidades. A primeira é através da introdução do id do zerozero para essa respetiva equipa e o programa através do *web scraping* irá fornecer os dados dessa equipa, desde o nome, local e a modalidade. Consequentemente esses dados irão servir para adicionar essa equipa à base de dados. A segunda é de forma manual, onde o utilizador, a partir de um formulário terá de especificar todos esses dados para uma determinada equipa.

**RF3** A página dos jogadores deverá:

- **RF3.1** permitir ao utilizador a visualização de uma tabela com um limite de exibição de 10 linhas de cada vez, com todos os jogadores presentes na base de dados;
- **RF3.2** permitir a atualização, edição e remoção dos jogadores da base de dados;

- **RF3.3** permitir a adição de jogadores à base de dados através de duas possibilidades. A primeira é através da introdução do id do zerozero para esse respetivo jogador e o programa através do *web scraping* irá fornecer os dados desse jogador (nome, data nascimento, nacionalidade e equipa). Consequentemente esses dados irão servir para adicionar essa equipa à base de dados. A segunda é de forma manual, onde o utilizador, a partir de um formulário terá de especificar todos esses dados para uma determinada equipa;
- **RF3.4** permitir que o campo seleção seja opcional e editável, como as convocatórias à seleção estão constantemente a mudar, não poderemos partir do pressuposto que um jogador faz parte dela mas sim a vai representar em determinada altura, assim o utilizador pode configurar este campo conforme pretender.

**RF4** A página das fotos deverá:

- **RF4.1** permitir ao utilizador a visualização de uma tabela com um limite de exibição de 10 linhas de cada vez, com todos as fotos presentes na base de dados;
- **R4.2** permitir ao utilizador a visualização das fotos desenvolvidas para um jogador de um determinado jogo;
- **RF4.3** permitir a edição e remoção das fotos da base de dados;
- **RF4.4** permitir a adição das fotos à base de dados através de links.

**RF5** A página dos Jogos Equipa deverá:

- **RF5.1** permitir ao utilizador a visualização de uma tabela com um limite de exibição de 10 linhas de cada vez, com os ids das equipas que participam num determinado jogo.

**RF6** A página inicial deverá:

- **RF6.1** permitir ao utilizador a visualização de uma tabela com um limite de exibição de 10 linhas de cada vez, com todos os jogadores, as suas equipas e respetivas modalidades presentes na base de dados;
- **RF6.2** exibir uma imagem publicitária que contém todos os contatos da Mixed Creations.

**RF7** A página sobre deverá:

- **RF7.1** apresentar um texto que faça o enquadramento do desenvolvimento da aplicação Web.

### 3.1.2 Requisitos Não Funcionais

Os **requisitos não funcionais** definem essencialmente as propriedades do sistema e as suas restrições. Listam-se em baixo os **requisitos não funcionais**, as propriedades que se esperam do sistema.

### **RNF1 Tecnologia**

- Para o desenvolvimento da aplicação Web iremos usar a linguagem de programação *Python* com recurso ao *BeautifulSoup*, que é uma biblioteca para captura de informações de páginas da Web. O *Flask* será o padrão de desenvolvimento. A plataforma para aplicações na nuvem (*Heroku*) será o local onde poderá ser executada a aplicação Web.

### **RNF2 Compatibilidade**

- O acesso à aplicação Web é realizado através de um computador, sendo necessário haver conexão à Internet.
- Deve ser compatível com os vários navegadores existentes.

### **RNF3 Consistência**

- Os conteúdo apresentado deve manter a sua aparência em diferentes navegadores.

### **RNF4 Usabilidade**

- Deve ser prático, funcional e de fácil utilização.

### **RNF5 Segurança**

- O sistema deverá ser seguro permitindo apenas ao utilizador final ver as informações.

### **RNF6 Disponibilidade**

- O nosso sistema funciona 24 horas por dia, as informações são dadas em tempo real.

### **RNF7 Eficiência**

- Devido ao grande volume de dados da base de dados, devem ser disponibilizadas soluções mais rápidas para atualizar ou pesquisar um determinado jogo de uma equipa.

## **3.2 Casos de Uso**

O *Unified Modeling Language* (UML) é uma linguagem de modelagem de um *software* ou de uma aplicação. O UML gera diagramas que auxiliam na modelagem do *software*. Os diagramas servem para o entendimento da aplicação Web tanto pela parte dos desenvolvedores como do utilizador final. O diagrama de casos de uso tem o objetivo de demonstrar as funcionalidades do sistema, quem executa e qual a sua interação com as outras funcionalidades. O diagrama de casos de usos está representado na Figura 3.1.



### 3.3 Diagrama de Atividades

O diagrama de atividades foca-se na representação das sequências das atividades que ocorrem na aplicação Web. Este preocupa-se com as ações do utilizador, da aplicação e do servidor. O diagrama de atividades está também diretamente relacionado com os casos de uso.

O diagrama é composto por quatro atividades principais: Adicionar dados, Atualizar dados, Eliminar dados e o Visualizar dados.

O utilizador inicia a aplicação Web no ponto de entrada que é um círculo preenchido de preto e termina à mesma num círculo preenchido de preto também mas com uma circunferência com bordas pretas ao redor. As caixas são as atividades enquanto que as setas representam o fluxo entre as atividades. Os losangos são os diamantes de decisão, onde pode ocorrer uma alteração do fluxo consoante cada situação específica.

#### 3.3.1 Diagrama de atividades: Adicionar dados

O diagrama de atividades referente à adição de dados (Jogadores, Equipas, Jogos ou Fotos) à aplicação Web está representada na Figura 3.2.

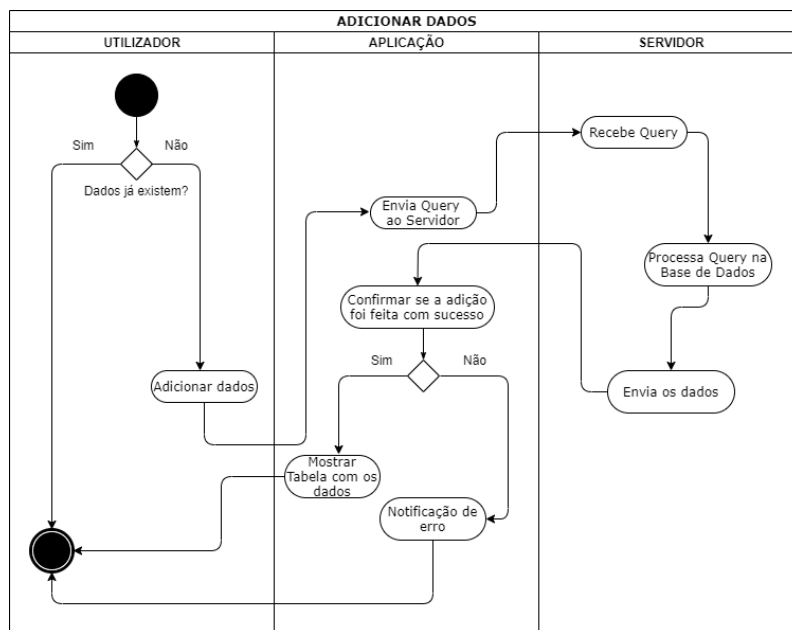


Figura 3.2: Diagrama de atividades - Adicionar

#### 3.3.2 Diagrama de atividades: Atualizar dados

O diagrama de atividades referente à atualização de dados (Jogadores, Equipas, Jogos ou Fotos) à aplicação Web está representada na Figura 3.3.

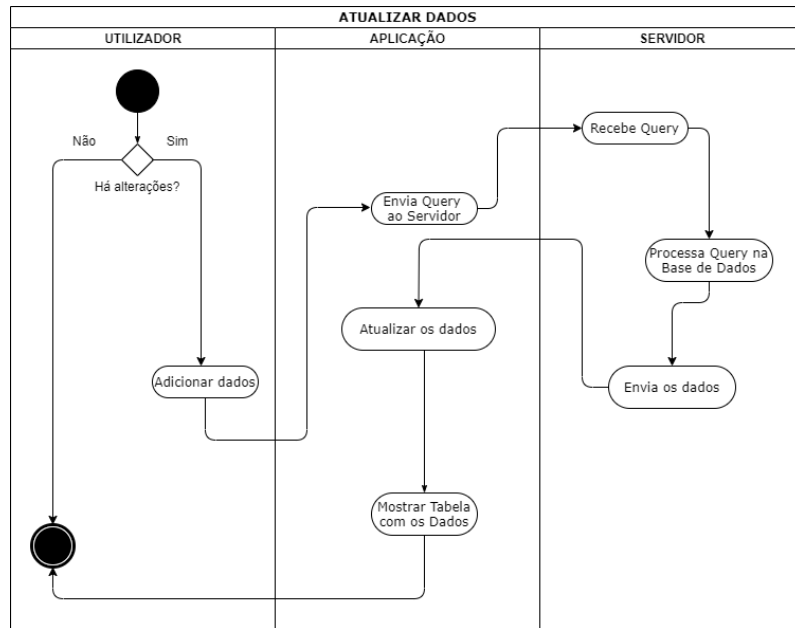


Figura 3.3: Diagrama de atividades - Atualizar

### 3.3.3 Diagrama de atividades: Eliminar dados

O diagrama de atividades referente à eliminação de dados (Jogadores, Equipas, Jogos ou Fotos) à aplicação Web está representada na Figura 3.4.

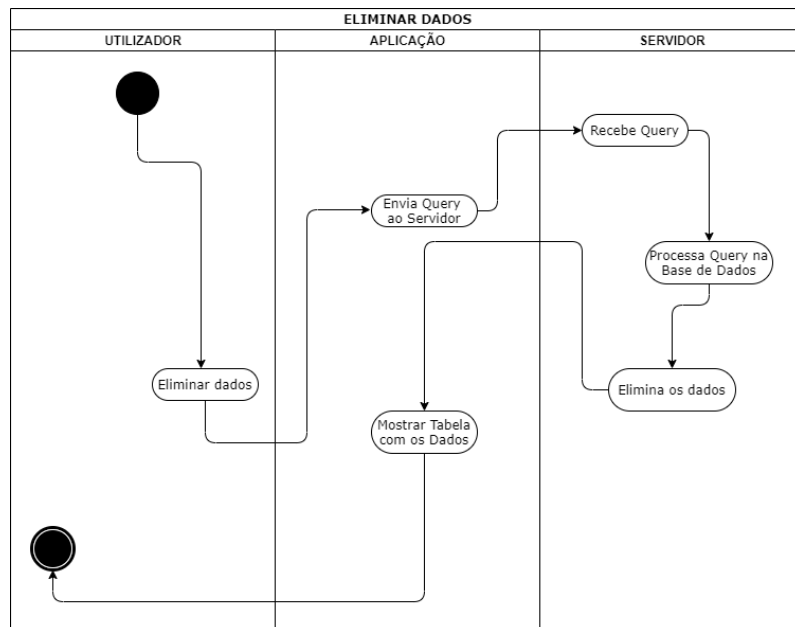


Figura 3.4: Diagrama de atividades - Eliminar

### 3.3.4 Diagrama de atividades: Visualizar dados

O diagrama de atividades referente à visualização de dados (Jogadores, Equipas, Jogos ou Fotos) à aplicação Web está representada na Figura 3.5.

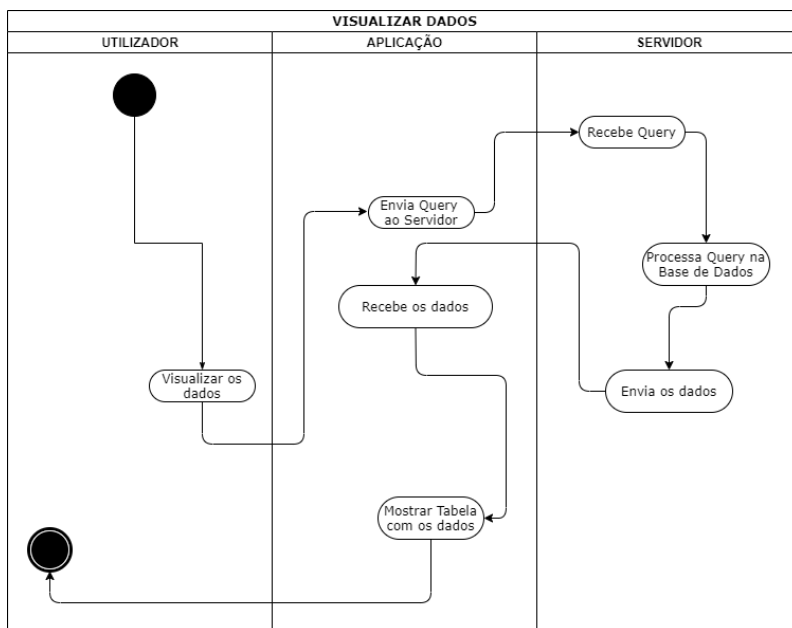


Figura 3.5: Diagrama de atividades - Visualizar

## 3.4 Modelo de Base de Dados

Os modelos de base de dados são extremamente importantes para a organização dos dados. A informação a apresentar ao utilizador vai passar por um processo de tratamento de dados pelo que a criação da base de dados teve de ser pensada de forma a que consiga interligar o sistema todo e apresentar todos os dados necessários. Na Secção 3.4.1 será mostrado o modelo entidade relacionamento e na Secção 3.4.2 é apresentado o modelo físico.

### 3.4.1 Modelo Entidade Relacionamento

O modelo entidade relacionamento tem como objetivo a criação de um modelo gráfico da base de dados, onde estarão representadas todos os relacionamentos e as entidades. Neste modelo os detalhes não são especificados, pelo que costuma ser o primeiro modelo a ser desenhado aquando da criação de uma base de dados. Na Tabela 3.2 são exibidas as características que são apresentadas no modelo entidade relacionamento.

Tabela 3.1: Características - Modelo Entidade Relacionamento

<b>Caraterísticas</b>
Nome da entidade
Relacionamentos da entidade
Atributos

O modelo entidade relacionamento desenhado para a aplicação Web está presente na Figura 3.6 onde podemos ver que existem 5 entidades (Jogador, Equipa, Jogo, JogoFoto-Jogador, JogoEquipa). As entidades presentes no modelo entidade relacionamento são:

- **Jogador:** A entidade *Jogador* relaciona-se com as entidades *Equipa* e *JogoFotoJogador*. A relação com a *Equipa* é de N:1 porque uma equipa pode ter vários jogadores. A relação com a entidade *JogoFotoJogador* é de 1:N pois um jogador pode ter várias fotos em vários jogos.
- **Equipa:** A entidade *Equipa* relaciona-se com as entidades *Jogador* e *JogoEquipa*. A relação com o *Jogador* é de 1:N porque, como já referido, uma equipa pode ter um plantel composto por vários jogadores. A relação com a entidade *JogoEquipa* é de N:1 devido a num evento desportivo participarem sempre duas equipas, isto é, um jogo vai ter sempre mais que uma equipa associada.
- **JogoEquipa:** A entidade *JogoEquipa* relaciona-se com as entidades *Equipa* e *Jogo*. A relação com a *Equipa* é de 1:N pelos motivos referidos no tópico anterior. A relação com a entidade *Jogo* é de 1:1 porque apenas será adicionada uma linha à base de dados da entidade *JogoEquipa* por cada jogo que houver.
- **Jogo:** A entidade *Jogo* relaciona-se com as entidades *JogoEquipa* e *JogoFotoJogador*. A relação com a entidade *JogoEquipa* é de 1:1 pelos motivos referidos no tópico anterior. A relação com a entidade *JogoFotoJogador* é de 1:N porque um determinado jogo pode ter várias fotos associadas.
- **JogoFotoJogador:** A entidade *JogoFotoJogador* relaciona-se com as entidades *Jogo* e *Jogador*. Ambas as relações já foram explicadas nos tópicos anteriores, um jogo e um jogador podem ter várias fotos associadas pelo que as relações da entidade *JogoFotoJogador* são ambas de N:1.

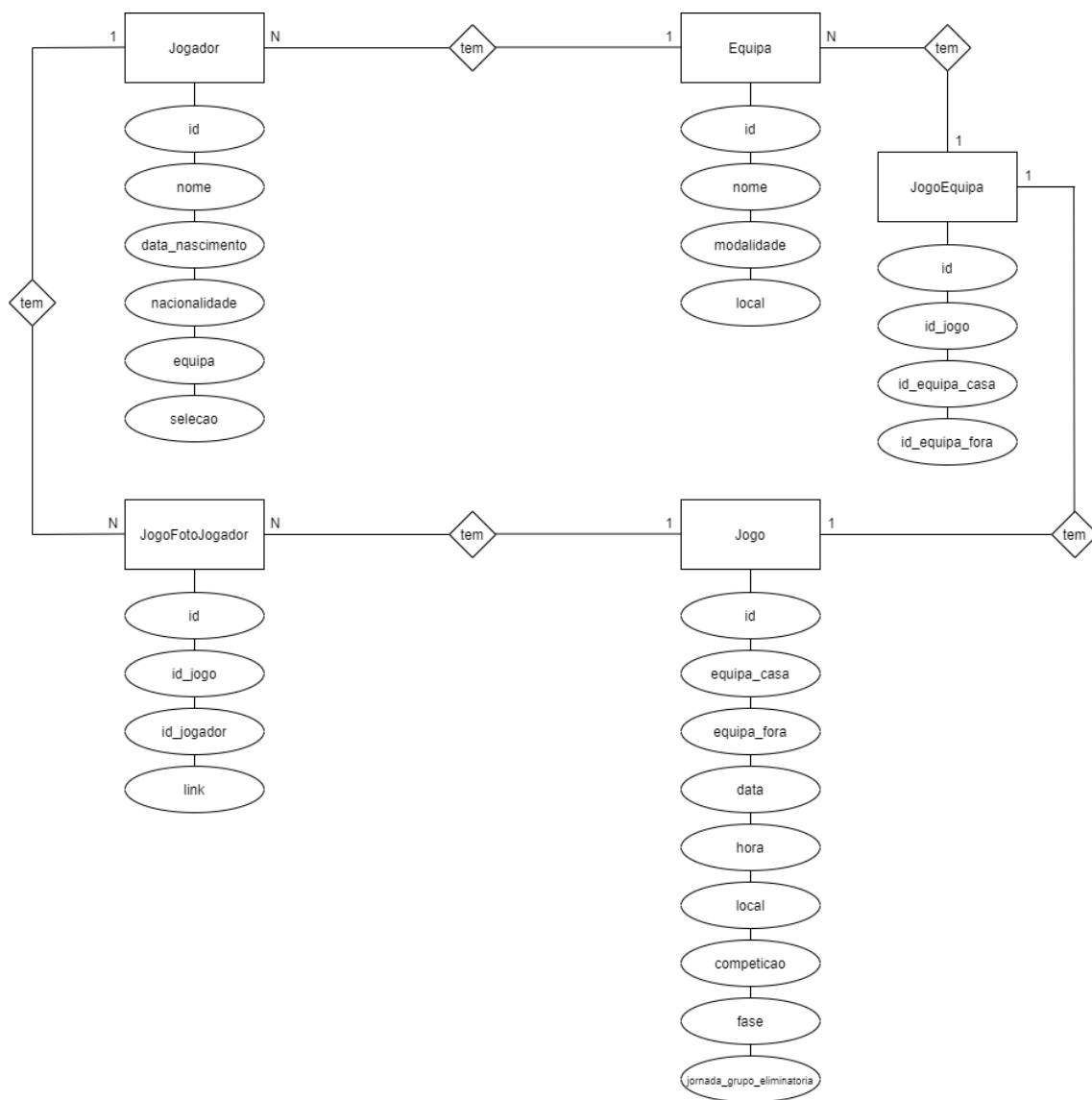


Figura 3.6: Modelo Entidade Relacionamento

### 3.4.2 Modelo Físico

O modelo físico define com maior detalhe toda a base de dados. Todas as informações das tabelas e colunas são apresentadas com o máximo detalhe, os tipos de chaves dos atributos também são especificados. A Tabela 3.2 mostram as características que são apresentadas no modelo físico.

Tabela 3.2: Características - Modelo Físico

<b>Caraterísticas</b>
Chave primária
Chave estrangeira
Nome das tabelas
Nome das colunas
Tipo das colunas

Na Figura 3.7 pode ser observado o desenho do Modelo Físico da minha base de dados.

As entidades presentes no modelo físico são:

- **Jogador:** A entidade *Jogador* possui 6 atributos (id, nome, data nascimento, nacionalidade, equipa, selecao). O id é a chave primária, sendo única para cada jogador. O nome corresponde ao nome do jogador. A nacionalidade ao seu país de origem, a seleção pode ser adicionada manualmente no caso do jogador ser convocado para uma determinada competição e a equipa é uma chave estrangeira que corresponde ao id da atual equipa do jogador.
- **Equipa:** A entidade *Equipa* possui 4 atributos (id, nome, modalidade, local). O id é a chave primária, sendo única para cada equipa. O nome corresponde ao nome da equipa. A modalidade corresponde ao tipo de desporto que aquela determinada equipa pratica. O local é o nome do pavilhão ou estádio onde a equipa joga, a designação do mesmo depende também da modalidade em causa (Pavilhão ou Estádio).
- **JogoEquipa:** A entidade *JogoEquipa* possui 4 atributos (id, id jogo, id equipa casa, id equipa fora). O id é a chave primária, sendo única para cada evento associado à tabela *JogoEquipa*. O id jogo é uma chave estrangeira correspondente a um determinado jogo. A equipa casa e equipa fora são também, chaves estrangeiras que possuem o id das equipas que vão participar nesse determinado jogo, isto é, correspondem à equipa da casa e à equipa de fora respetivamente.
- **Jogo:** A entidade *Jogo* possui 9 atributos (id, equipa casa, equipa fora, data, hora, local, competicao, fase, jornada grupo eliminatória). O id é a chave primária, sendo única para cada jogo. A equipa da casa é o nome da equipa que vai receber o jogo em sua casa. A equipa fora é o nome da equipa visitante, a que se vai deslocar até ao terreno de outra para a defrontar. A data corresponde ao dia, mês e ano em que o evento irá ocorrer. A hora corresponde ao horário do jogo. O local é o nome do pavilhão ou estádio onde o jogo decorrerá. A competição, fase e jornada grupo eliminatória correspondem à própria competição em si que podem variar duma competição para a outra consoante os moldes em que elas decorrem.
- **JogoFotoJogador:** A entidade *JogoFotoJogador* possui 4 atributos (id, id jogo, id jogador, link). O id é a chave primária, sendo única para cada fotografia adicionada à tabela *JogoFotoJogador*. O id jogo é uma chave estrangeira e corresponde a um determinado evento desportivo. O id jogador é uma chave estrangeira é uma chave estrangeira e corresponde a um determinado jogador. O link é o caminho para fazer download de uma determinada fotografia.

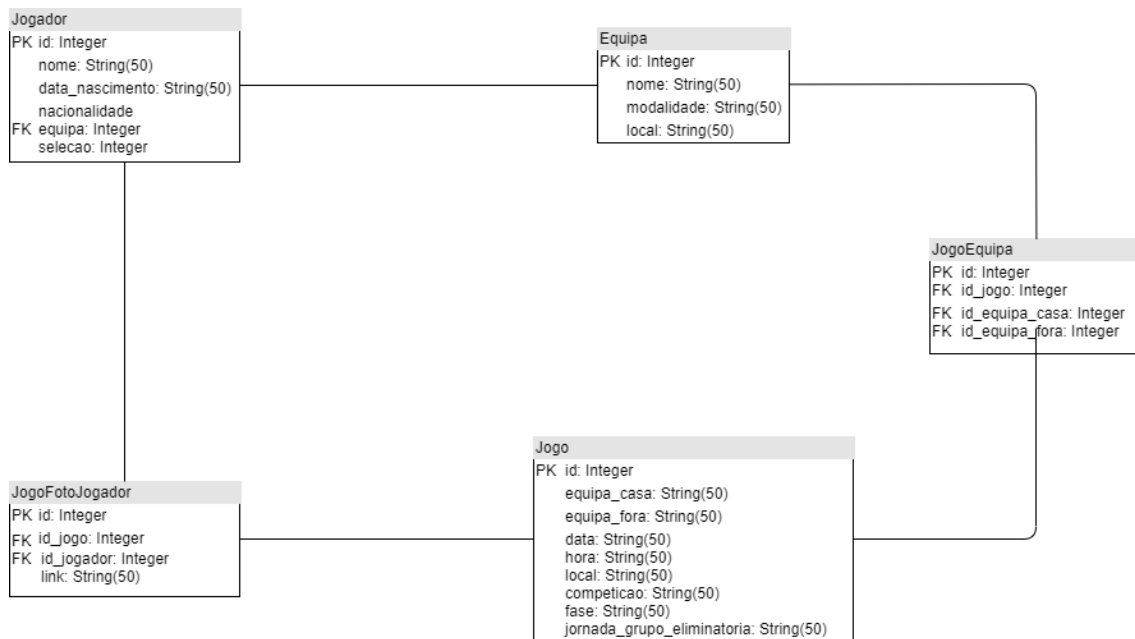


Figura 3.7: Modelo Físico

### 3.5 Modos de visualização e Sumarização de dados

São propostos um conjunto de visualizações e sumarização de dados, apresentados nas Figuras 3.8 a 3.12.

A pesquisa dos dados na aplicação Web vai estar dividida através das cinco entidades da base de dados (Jogo, Equipa, Jogador, JogoFotoJogador e JogoEquipa).

A página Jogadores vai permitir a exibição dos dados pessoais e mais relevantes de cada jogador como o nome, a sua data de nascimento e a sua nacionalidade. O Jogador terá também associado a ele uma equipa e poderá ter também um campo que corresponde à seleção no caso de este ser convocado para a mesma, esta pode ser introduzida manualmente consoante as convocatórias. O protótipo da página dos Jogadores está na Figura 3.8.

A página Equipas vai permitir ao utilizador visualizar uma tabela com os dados relevantes das equipas associadas à Mixed Creations, isto é, o nome da equipa, a modalidade que corresponde ao desporto praticado pela mesma (Futebol, Futsal, etc...) e o local que é o sítio onde são acolhidos os eventos por parte dessa equipa aquando de jogar em casa. O protótipo da página relativa às Equipas pode ser observado na Figura 3.9.

A informação relativa aos eventos desportivos está contida na página dos Jogos, onde o utilizador tem também acesso a duas tabelas. A primeira apresenta todos os jogos que irão decorrer nos próximos 10 dias enquanto que a segunda apresenta todos os jogos existentes na base de dados. Ambas as tabelas possuem todas as informações necessárias para a realização de um cartaz promocional de um determinado evento desportivo. Assim sendo, é fácil de perceber que a tabela Jogo é a que contém um maior número de atributos devido à quantidade de informações que é necessário especificar. Os jogos decorrem com uma equipa a defrontar uma outra, pelo que teremos um atributo designado Equipa Casa e um

outro campo chamado Equipa Fora. Tal como em todos os eventos, existe uma data, um horário e um local pelo que teremos os atributos Data, Hora e Local também representados. Foram definidos também os campos Competição, Fase e JornadaGrupoEliminatória para sabermos qual é a competição e em que fase esta se encontra. Por exemplo, numa jornada normal desta época do escalão máximo do futebol em Portugal teríamos na Competição designado a Liga Nos 2019/20, na Fase o Campeonato e na JornadaGrupoEliminatória o número da mesma jornada. Se ao invés de uma jornada normal tivéssemos uma fase a eliminar da Liga dos Campeões, por exemplo a primeira mão de uma meia-final, na Competição teríamos o nome Uefa Champions League 2019/20, na Fase iríamos ter Meia-Final e na JornadaGrupoEliminatória iríamos ter 1ª mão. Através de um formulário de pesquisa, o utilizador deve conseguir pesquisar nas tabelas os jogos correspondentes à equipa que ele quiser ou até mesmo pesquisar os jogos que ocorreram num determinado local desde que esses dados introduzidos no formulário de pesquisa existam na base de dados. O protótipo da página dos Jogos é exibido na Figura 3.10.

Cada jogo permite a adição de fotografias relativas a esse evento desportivo e a elas estão sempre associadas um determinado jogador, pelo que é possível aceder às fotografias por jogos ou por jogadores. As fotografias são adicionadas através da atribuição de links Web. O protótipo da página Fotos é exibido na Figura 3.11.

A página Jogo Equipa permite a visualização dos ids dos jogos e das equipas que participaram num determinado jogo por questões de organização e interligação de todos os dados da base de dados. O protótipo é bem menos complexo e é exibido na Figura 3.12.



Figura 3.8: Modo de visualização da Página Jogadores

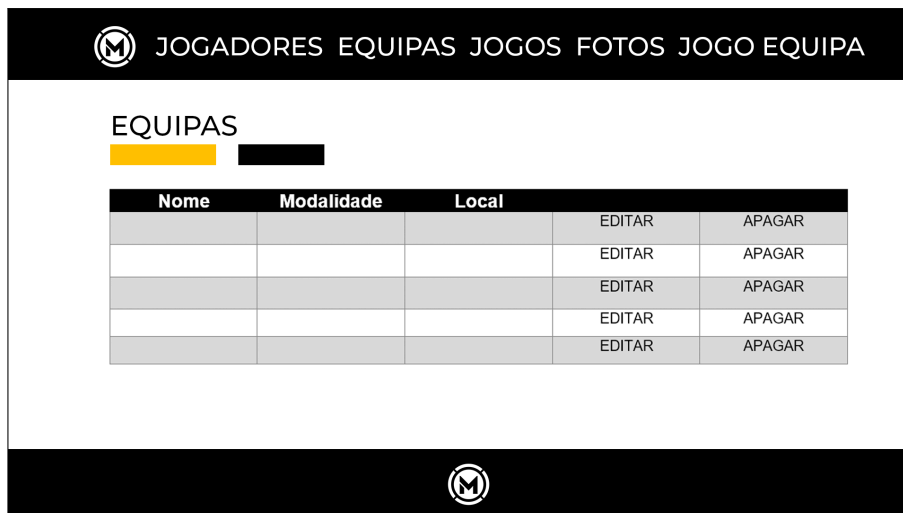


Figura 3.9: Modo de visualização da Página Equipas

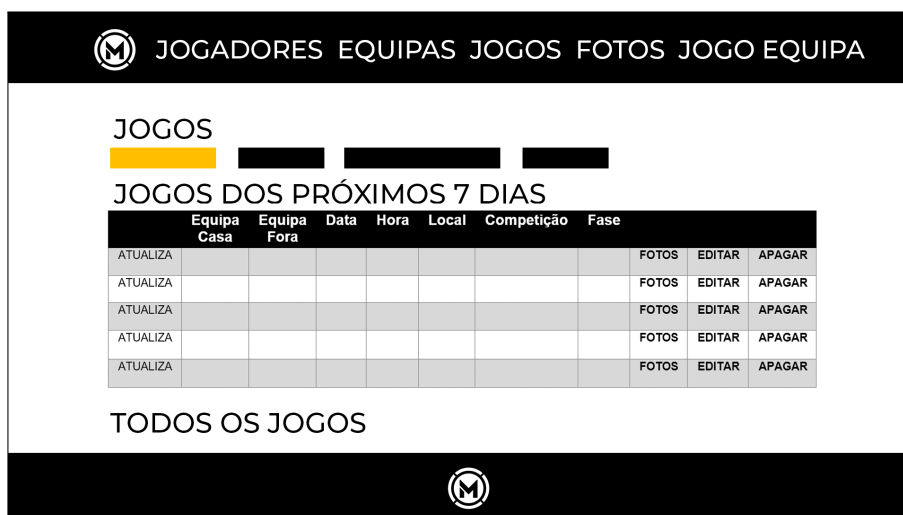


Figura 3.10: Modo de visualização da Página Jogos



Figura 3.11: Modo de visualização da Página Fotos



Figura 3.12: Modo de visualização da Página Jogo Equipa

### 3.6 Conclusões

O Capítulo 3 é um dos mais importantes no que diz respeito à forma como irá ser desenvolvida a aplicação Web, a definição dos requisitos tem de ser muito bem calculada e estruturada para que o sistema possa cumprir com tudo o que foi descrito. De forma a organizar o processo de implementação, os requisitos foram enumerados consoante as funcionalidades que tinham de implementar numa determinada página da aplicação Web. Essa organização foi uma ideia que permitiu encarar o problema de forma simples e direta. O diagrama de casos de uso indica-nos todos os cenários possíveis de encontrar que foram implementados no sistema. A organização da base de dados também foi feita de forma simples e inteligente, todas as entidades acabam por se ligar umas através das outras e assim permitem a obtenção de dados de forma interna sem ter de ser necessário recorrer a *scripts* para saber por exemplo se a equipa de um determinado jogador já existe na base de dados ou não. A implementação foi pensada para que esse tipo de respostas fosse possível e fáceis de obter. No final do Capítulo e não menos importante, foram exibidos os *layouts* que permitem uma visão muito mais ampla do que será a aplicação Web a funcionar e facilita o processo de raciocínio aquando da implementação.



# Capítulo 4

## Modelo e Implementação

O quarto Capítulo é o do modelo e implementação estando dividido em 6 secções. A Secção 4.1 contém toda a arquitetura implementada no sistema através da descrição de todos os seus componentes e fluxos. Na Secção 4.2 é abordada a implementação num modo mais geral, sendo numa fase inicial apresentadas todas as tecnologias envolvidas e ferramentas utilizadas no desenvolvimento de toda a implementação. Em seguida, é apresentado o perímetro tecnológico da implementação com a descrição do processo. Nas Secções 4.3 e 4.4 são detalhados os módulos de mineração de dados e da aplicação Web mais ao detalhe. Na mineração é abordada a extração dos dados, o tratamento dos mesmos e é ilustrado com um exemplo de um carregamento de dados para a base de dados usando um modelo automático e não supervisionado. Na parte da aplicação Web é apresentada o tipo de base de dados usado, a implementação do padrão MVC e acaba com várias ilustrações do funcionamento de toda a aplicação. É exibida a estruturação das páginas da aplicação Web e as várias funcionalidades presentes, desde a adição, atualização, edição e eliminação dos dados. Na Secção 4.5 temos o lançamento da aplicação Web na nuvem. A Secção 4.6 exhibe um exemplo de um produto final desenvolvido pela Mixed Creations após a obtenção dos dados usando a aplicação Web e por fim, seguem-se as conclusões finais na Secção 4.7.

### 4.1 Arquitetura

Antes de apresentar os módulos, é importante referir que em cada campeonato desportivo são regularmente agendados novos jogos, existindo vários repositórios que agregam estas informações online. Os utilizadores pretendem saber quando são os jogos, bem como estar a par das suas atualizações. Para tal, foi desenhado um módulo de mineração de dados, que é baseado no processo *Extract Transform Load* (ETL), e uma aplicação Web para mostrar as informações aos utilizadores, que segue o modelo *Model View Controller* (MVC), tal como é ilustrado na Figura 4.1. De um lado está o ETL que transforma os dados naquilo que é necessário para a aplicação Web, do outro lado MVC que é excelente para fazer o desenvolvimento de aplicações e software. A combinação do poder do MVC para criar aplicações com o poder do ETL para a transformação dos dados em formas facilmente utilizáveis, permite aos desenvolvedores de software conseguirem expandir o alcance, melhorar a qualidade das suas aplicações e chegar a uma maior diversidade de mercados.

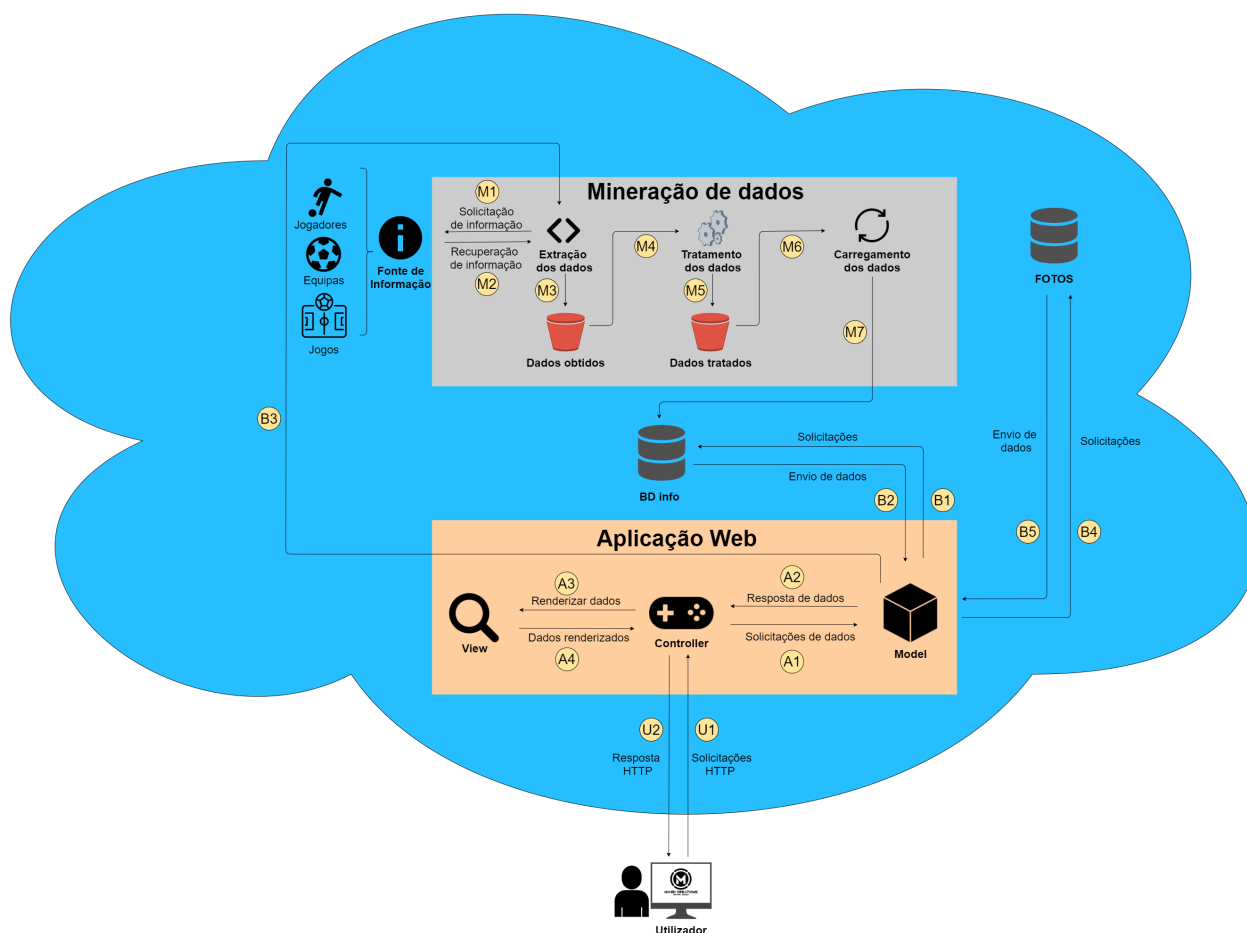


Figura 4.1: Arquitetura da plataforma de um modelo automático e não supervisionado

A arquitetura primeiro explica-se pela descrição dos seus componentes e depois pela descrição dos seus fluxos. De seguida é feita a descrição dos componentes que constituem a arquitetura:

- Fonte da informação;
- Módulo da mineração de dados;
- Base de dados;
- Fotos;
- Utilizador;
- Módulo da aplicação Web.

A **fonte de informação** contém dados sobre jogadores, equipas e jogos. Esta pode ser uma API ou um *website* com as informações online.

O **módulo de mineração de dados**, baseado no processo ETL, pode ser descrito em através dos seus três componentes:

- O **módulo de extração** é responsável por pedir um conjunto de páginas à fonte de informação e são recuperados um conjunto específico de dados que serão guardados num repositório temporário de dados extraídos.
- O **módulo de transformação** recebe os dados obtidos e transforma-os de forma a corresponderem à informação que se pretende, no formato adequado para a base de dados.
- O **módulo de carregamento de dados** recebe os dados tratados e encarrega-se de carregar os dados na base de dados nas tabelas correspondentes.

A **base de dados** é quem armazena todos os dados prontos a exibir para o utilizador. Esta pode ser alimentada através de processo automático e não supervisionado ou então de forma manual pelo utilizador. A base de dados vai receber solicitações por parte do *model* e retornará como resposta os dados que foram pedidos, caso existam.

As **fotos** que serão os produtos finais do trabalho desenvolvido pela Mixed Creations, estão presentes num repositório de fotografias *online* podendo ser introduzidas na aplicação através da adição dos links da mesmas à base de dados.

O **utilizador** interage com a aplicação Web através de um navegador, enviando solicitações HTTP que devolverão como resposta a visualização de dados. No caso de serem adicionados, atualizados ou removidos dados da base de dados será accionado um método POST ao qual se segue sempre um novo método GET para retornar a página já com os novos dados.

O **módulo da aplicação Web** baseia-se no padrão MVC e é usado como o padrão de software da aplicação Web que vai utilizar esses dados. Este módulo pode ser descrito em cinco componentes:

- O **controller** é quem assume o controlo da aplicação, este retorna uma visualização pedida, com base nos dados recebidos do *model* e na visualização que foi construída com base nestes dados.
- o **model** é a componente principal da arquitetura, modela os dados. É a camada lógica da aplicação que gere o comportamento dos dados através de funções. As suas funções estão à espera de serem chamadas para recolher, gravar e exibir dados. É o coração da execução responsável por tudo o que a aplicação vai fazer a partir dos comandos do *controller*.
- A **view** é responsável pela visualização dos dados, possui os *templates* que serão exibidos como resposta aos pedidos feitos pelo utilizador.

Para uma melhor explicação dos fluxos de processamento, foi seguido o padrão de trabalho (*work pattern*), isto é, o fluxo será descrito em baixo com letras e números que fazem correspondência com a Figura 4.1. As etapas do utilizador foram descritas com a letra U, as do módulo da aplicação Web com a letra A, as que implicam solicitações de dados à base de dados ou à fonte de informação com a letra B e as do módulo da mineração de dados com a letra M, as da mineração De seguida faz-se a descrição dos fluxos de processamento, ilustrados na Figura 4.1:

- **U1:** o utilizador interage com a aplicação Web e vai pedindo os dados de jogadores, equipas, jogos ou fotos (solicitações HTTP). Essa solicitação é feita através de um método GET que contém a informação no URL e segue para o *controller*.
- **A1:** o *controller*, em primeiro lugar, vai fazer solicitações de dados ao *model*.
- **B1:** o *model* vai fazer pedidos à base de dados para extrair os dados pretendidos.
- **B2:** a base de dados vai verificar se contém os dados que foram solicitados e envia-os como resposta para o *model*.
- **B3:** no caso da base de dados não conter os dados que foram solicitados, que podem ser informações de jogadores, equipas ou jogos, o *model* irá despoletar o módulo de mineração de dados ao efetuar solicitações de informações, como por exemplo o nome de uma determinada equipa, modalidade e estádio, ao extrator de dados.
- **M1:** o módulo de extração de dados vai aceder e pedir um conjunto de páginas à fonte de informação que é quem possui as informações pedidas no fluxo anterior pelo *model*.
- **M2:** o módulo de extração de dados irá efetuar a recuperação da informação, através do uso de técnicas de mineração na Web, como resposta do seu acesso à fonte de informação.
- **M3:** o módulo de extração de dados com a informação extraída, vai guardar os dados obtidos em repositórios temporários.
- **M4:** o módulo de transformação de dados vai receber esses dados obtidos para proceder ao respetivo tratamento usando técnicas de mineração de dados, uso de expressões regulares, entre outras soluções.
- **M5:** o módulo de transformação de dados com a informação tratada, vai guardar os dados tratados em repositórios temporários.
- **M6:** o módulo de carregamento dos dados vai receber os dados tratados.
- **M7:** o módulo de carregamento dos dados com os dados tratados, encarrega-se de carregar os dados na base de dados nas tabelas correspondentes, ficando estes disponíveis para serem exibidos aquando das solicitações efetuadas pelo *model*.
- **B4:** o *model* pode fazer ainda outros pedidos de acesso a repositórios que armazenam outro tipo de informação, como por exemplo *links de imagens*.
- **B5:** o repositório de informação vai responder com o envio dos dados solicitados diretamente para o *model*.
- **A2:** o *model* após ter resposta para os dados, que foram pedidos pelo utilizador (jogadores, equipas, jogos ou fotos), vai enviá-los para o *controller*.

- **A3:** o *controller* vai renderizar os dados através da ligação com a *view* que contém os templates para a visualização dos dados.
- **A4:** a *view* vai enviar os dados renderizados para o *controller*.
- **U2:** o *controller* vai enviar respostas HTTP. Será exibido um método POST se no fluxo anteriormente descrito foram adicionados, atualizados ou removidos dados da base de dados, seguindo-se sempre um GET com a informação para a exibição presente no URL. A resposta HTTP vai concluir todo este fluxo, permitindo ao utilizador a visualização de todas as informações que foram solicitadas durante o uso da aplicação Web.

## 4.2 Perímetros tecnológicos da implementação

Nesta Secção estão expostas as tecnologias e ferramentas que permitiram o desenvolvimento do protótipo. Em primeiro lugar, são apresentadas todas as tecnologias envolvidas neste projeto, com uma breve descrição das mesmas:

- O **Python** [24] é uma linguagem de programação interpretada de alto nível. Esta permite aos programadores escrever código lógico de fácil leitura. Escolhi esta linguagem dado a simplicidade da mesma e também pela possibilidade de importar bibliotecas que efetuem o tratamento dos dados.
- O **BeautifulSoup** [25] é uma biblioteca para captura de informações de páginas da Web, é um pacote do *Python* que serve para trabalhar com HTML. O *BeautifulSoup* funciona como um analisador do HTML, oferece suporte à deteção de codificação, podendo assim fornecer melhores resultados para as páginas HTML.
- O **Regex** [26] importa o módulo *re* do *Python* e é uma sequência de caracteres que podem formar um padrão de pesquisa. Pode ser usado através de uma pesquisa que irá verificar a existência de uma determinada sequência. A partir do detetor de padrões que o *Regex* implementa, podemos tratar a informação da forma que queremos, isto é, na aplicação Web desenvolvida este é muito útil na eliminação de sequências irrelevantes e no tratamento de sequências de texto relevantes.
- O **SQLAlchemy** [27] possui um vasto conjunto de ferramentas *Python SQL* com a capacidade de mapear objetos relacionais, o conhecido *Object Relational Mapper* (ORM). O ORM permite à aplicação um alto desempenho e um acesso eficiente à base de dados através do uso de entidades de alto nível, como classes, objetos e métodos, em vez de tabelas e SQ que fornece um vasto poder de flexibilidade do SQL. O ORM possibilita o mapeamento entre tabelas de uma base de dados relacional e objetos, onde cada tabela tem uma classe associada.
- O **Flask** [28] é um *web application framework* para desenvolvimento de aplicações Web na linguagem de programação *Python*. Após o *import*, o *Flask* é usado para o

desenvolvimento de aplicações Web. Na aplicação Web a ser desenvolvida o uso do *Flask* irá permitir à aplicação Web as seguintes operações:

- Roteamento a partir de URL;
  - Criação de formulários que manipulam e validam os dados;
  - Associação com HTML;
  - Ligação com a base de dados e manipulação de dados através dum ORM.
- O **Heroku** [29] é uma plataforma do tipo *Platform-as-a-Service* (PaaS) e suporta um vasto leque de linguagens de programação. Este permite aos desenvolvedores criar, executar e lançar as aplicações desenvolvidas na nuvem.

Em seguida, são apresentadas as ferramentas utilizadas para a realização deste projeto, com uma breve descrição das mesmas e o motivo da sua utilização:

- O **Visual Studio Code** [30] foi lançado em 2015 pela Microsoft, é um software livre e de código aberto que suporta um grande número de linguagens de programação. Possui ainda um vasto conjunto de recursos que podem ser integrados na linguagem escolhida para o desenvolvimento de um determinado projeto.
- O **Adobe Photoshop** [31] é um software de edição de imagens bidimensionais do tipo *raster*, possuindo também capacidades típicas de alguns editores vetoriais. Desenvolvido pela *Adobe Systems* para *macOS* e *Windows*, é considerado líder no mercado dos editores de imagem profissionais, bem como o programa de eleição para edição profissional de imagens digitais e trabalhos de pré-impressão. Ele oferece um vasto leque de recursos, que são úteis e cumprem bem a função que se propõem. Os recursos funcionam muito bem e proporcionam uma edição de alta qualidade tanto para fotos captadas com câmaras básicas como profissionais. É um programa muito utilizado para a elaboração de *layouts*, publicidades e *webdesign*. Devido a todos estes fatores e também ao facto de conhecer muito bem a ferramenta através dos trabalhos desenvolvidos pela Mixed Creations, considerei o *Photoshop* o programa ideal para o desenho dos *layouts* da aplicação Web.

A implementação da solução Web proposta para recolha de informação de eventos desportivos foi desenvolvida segundo duas componentes, primeiro sobre um modelo de mineração de dados, que segue o processo ETL como referência, e depois sobre o módulo da própria aplicação Web que implementa o conhecido padrão MVC. O modelo de implementação e todos os processos efetuados até a informação chegar ao utilizador final podem ser observados na Figura 4.2. O modelo de mineração de dados, que implementa um modelo automático e não supervisionado, efetua a recuperação da informação que é pedida pelo utilizador na utilização da aplicação Web, o respetivo tratamento e posteriormente o seu carregamento. Na utilização da aplicação Web, temos um utilizador que acede através de um navegador à página Web da aplicação e são feitos *HTTP Request* ao *controller*, que vai controlar a aplicação Web por completo e interagir com o modelo de mineração, onde vão ser usadas as ferramentas utilizadas na recuperação, extração e

tratamento da informação. O utilizador ao interagir com a página, através do uso de formulários, vai fazendo pedidos de dados à aplicação Web. Esta vai recorrer ao modelo de mineração que é quem possui um extrator de dados, neste caso o *BeautifulSoup*, que vai implementar técnicas de *web scraping* com o objetivo de extrair os dados necessários da fonte de informação, o Zerozero. Após a recolha dos dados, estes serão tratados através do *RegEx* e do uso de expressões que regulares. Por fim, os dados tratados serão carregados e adicionados à base de dados. O *Python* é a linguagem padrão que permite a integração de todas estas tecnologias e possibilita a ocorrência de todos estes procedimentos. Graças ao suporte que o *Flask* oferece a toda a aplicação Web, os dados que estão gravados na base de dados são apresentados numa página Web e estão prontos para ser exibidos no ecrã do utilizador. A solução Web baseada na nuvem que irá armazenar a aplicação Web e dar suporte a todas as tecnologias utilizadas será o *Heroku*.

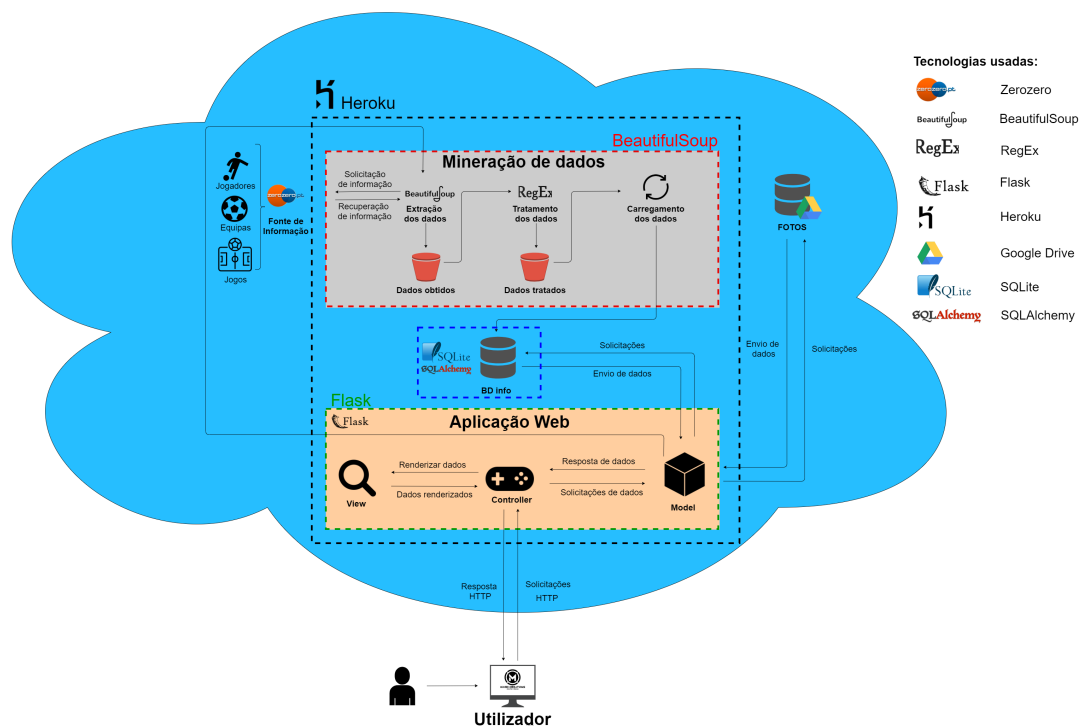


Figura 4.2: Perímetros tecnológicos da implementação da arquitetura

## 4.3 Implementação do modelo de mineração

### 4.3.1 Extração dos dados

Tal como descrito na Secção 4.1 são seguidos os vários fluxos na implementação da arquitetura. Os dados a serem armazenados na base de dados criada para a aplicação Web terão como fonte principal o Zerozero.pt, que é das maior base de dados do mundo no que diz respeito ao futebol e às modalidades. Uma outra alternativa será a introdução dos dados à mão mas esta apenas foi adicionada para o caso de haver uma situação excecional em que um determinado evento desportivo não esteja presente na Internet, uma situação



```

574  for u in urls:
575      try:
576          abre_link = 'https://www.zerozero.pt'+u
577          lista_ids = []
578          jogo = Jogo.query.all()
579          for j in jogo:
580              lista_ids.append(j.id)
581          for i in range(0,len(lista_ids),1):
582              lista_ids[i] = '/jogo.php?id=' + str(lista_ids[i])
583
584
585          result_link = requests.get(abre_link)
586          src_link = result_link.content
587          soup2_link = BeautifulSoup(src_link, 'lxml')
588
589          _idjogo = u
590          _idjogofinal = _idjogo[13:]
591
592          urls_teams = []
593
594          for tag in soup2_link.find_all('div', class_='logo'):
595              a_tag = tag.find('a')
596              urls_teams.append(a_tag.attrs['href'])
597              if(len(urls_teams)==2):
598                  break
599

```

Figura 4.4: Extração dos dados

### 4.3.2 Tratamento dos dados

Após a extração da informação contida no HTML da página correspondente a um determinado evento desportivo e como explicado na Secção 4.1 é altura de tratar os dados, ou seja, será necessário identificar as regras e detetar padrões. Isso será possível através da criação de um filtro na biblioteca *BeautifulSoup* para a obtenção da estrutura HTML da forma desejável. Na Figura 4.5 temos exemplificado a obtenção dos dados dia e hora para o preenchimento da tabela Jogo da nossa base de dados.

```

612  info = soup2_link.find_all('div', id='match_data')
613  dia_hora = remove_html_tags(str(info))
614  dia_hora = dia_hora[1:]
615  dia_hora = dia_hora.split(" - ", 8)
616  _dia = dia_hora[0]
617  _hora = dia_hora[1]
618  if len(_hora) != 5:
619      _hora = ''
620  _dia = _dia.split(" ", 4)
621
622
623  _mes=_dia[2]
624
625  if(_dia[2]=='Janeiro'):
626      _mes= '01'
627  elif(_dia[2]=='Fevereiro'):
628      _mes= '02'
629  elif(_dia[2]=='Março'):
630      _mes= '03'
631  elif(_dia[2]=='Abril'):
632      _mes= '04'
633  elif(_dia[2]=='Maio'):
634      _mes= '05'
635  elif(_dia[2]=='Junho'):
636      _mes= '06'
637  elif(_dia[2]=='Julho'):
638      _mes= '07'
639  elif(_dia[2]=='Agosto'):
640      _mes= '08'

```

Figura 4.5: Tratamento dos dados

Em primeiro lugar é criado um filtro para a *div* que contém os dados do jogo através do *BeautifulSoup* (linha 695). Após o filtro da informação aplicado sobre o código HTML da página, para ficarmos com os dados completamente tratados e prontos a adicionar na base dados será adicionada à *string* info uma função que irá remover as *tags* HTML através do *RegEx* (linha 613). Em seguida, através de outras funções implementadas no *Python* iremos ficar com a informação tratada (linha 614 a 623). No caso específico do dia, ficaríamos com o mês escrito por extenso mas de forma a possibilitar a consulta de *queries* à base dados que organize os dados de forma cronologicamente optei por transformar os meses escritos por extensos em números como podemos observar na imagem (linhas 625 a 640). O processo para a obtenção do nome das equipas, nome do pavilhão, competição, fase, jornada segue as mesmas metodologias mas tem ajustado consoante as *tags* e ids da estrutura HTML, tendo o *RegEx* de ser ajustado de forma individual para cada campo consoante a estrutura HTML. Para finalizar o processo de mineração na Web, os dados passarão por uma análise e respetiva validação, se estes forem válidos serão adicionados à base de dados e estão prontos para serem visualizados na aplicação Web, caso contrário será criada uma exceção e nesta situação específica o evento desportivo não será adicionado.

### 4.3.3 Carregamento dos dados

Nas Secções 4.3.1 e 4.3.2 foram descritos o processo de extração e tratamento dos dados. Uma questão que se levanta após a explicação de todo o processo é o da aplicação Web desenvolvida estar dependente da estrutura HTML do *website* escolhido como referência para obtenção dos dados. A aplicação Web usa o Zerozero na implementação mas ela foi desenvolvida de forma a que facilmente seja ajustável a outras diferentes fontes de informação, não estando assim totalmente dependente do Zerozero. A única coisa que será necessário redefinir será a implementação do *web mining* em função da fonte de informação que seja escolhida. Para um melhor entendimento, encontra-se ilustrado na Figura 4.6 a forma como são obtidas e guardadas as informações para um determinado jogo na base de dados. Cada jogo tem vários atributos que necessitam de dados, foram definidas variáveis correspondentes a cada atributo que se queira obter para a consequente adição dos dados à base de dados. Para efetuar este processo e como já mencionado anteriormente é usado o módulo de mineração de dados, através da biblioteca *BeautifulSoup* do *Python*, que vai usar técnicas de mineração na Web para recuperar os dados relevantes contidos no HTML da fonte de informação escolhida. Após a obtenção desses dados, estes vão ser guardados em repositórios temporários (variáveis) e o *RegEx* vai efetuar o tratamento dos dados. Ao fim da remoção das *tags*, os dados estão prontos para serem carregados na base de dados através da atribuição de cada variável ao seu respetivo atributo da tabela Jogo. Assim, é terminado o processo efetuado no módulo de mineração de dados, indo de encontro ao que foi detalhado na Secção 4.1.

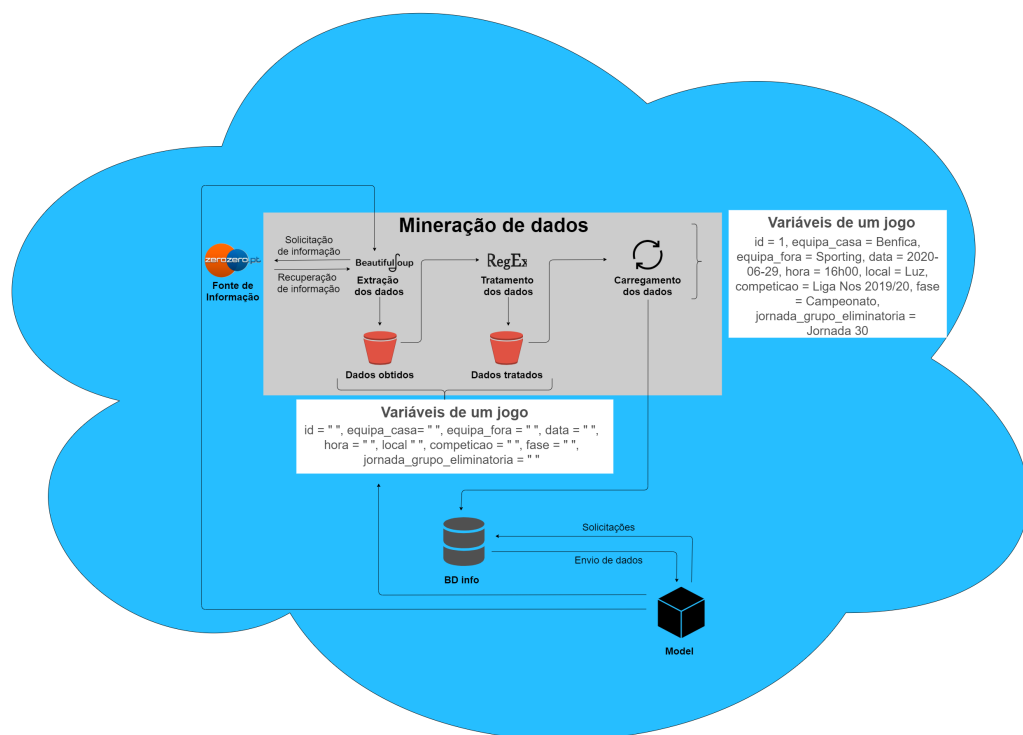


Figura 4.6: Processo de Web Scraping para a obtenção de informação de um Jogo

No caso de querer optar por outro *website* que não o Zerozero é nas variáveis que teria de ser refinado a implementação, isto é, teria de ser tido em conta o *website* escolhido para definir as informações que queria obter. Outra possibilidade e que viria a facilitar todo o processo era o uso de uma API de desporto que alimentasse diretamente a aplicação Web, assim todo este processo de mineração na Web deixaria de ser a principal fonte de informação porque a API alimentaria todo o sistema. O único problema das APIs de desporto é que são bastante limitadas no que diz respeito às informações das modalidades ou então as que possuem essa informação são todas pagas. Devido a estes motivos, foi descartada a hipótese de usar uma API que alimentasse a base de dados da aplicação Web desenvolvida.

## 4.4 Implementação da aplicação Web

### 4.4.1 Criação da base de dados

O modelo de base de dados utilizado implementa o *SQLAlchemy* que tem a capacidade de mapear objetos relacionais através dos ORM, que possibilita a interação com uma base de dados *SQLite*. Os dados em servidores do Sistema de gestão de base de dados (SGBD) estão guardados em tabelas. Na Figura 4.7 é ilustrado um excerto do código utilizado para a criação do modelo da base de dados, com a definição das respetivas entidades e atributos.

```
app > models.py > Jogo
C:\Users\35196\Desktop\APP_TESE_MC_UBI_FINAL_TESTE_HEROKU\app
2
3 class Equipa(db.Model):
4     id = db.Column(db.Integer, primary_key=True)
5     nome = db.Column(db.String(50))
6     modalidade = db.Column(db.String(50))
7     local = db.Column(db.String(50))
8
9     def __repr__(self):
10        return '<Equipa {}>'.format(self.id)
11
12
13 class Jogador(db.Model):
14     id = db.Column(db.Integer, primary_key=True)
15     nome = db.Column(db.String(50))
16     data_nascimento = db.Column(db.String(50))
17     nacionalidade = db.Column(db.String(50))
18     equipa = db.Column(db.Integer, db.ForeignKey('equipa.id'))
19     selecao = db.Column(db.Integer)
20
21     def __repr__(self):
22        return '<Jogador {}>'.format(self.id)
23
24 class Jogo(db.Model):
25     id = db.Column(db.Integer, primary_key=True)
26     equipa_casa = db.Column(db.Integer, db.ForeignKey('equipa.id'))
27     equipa_fora = db.Column(db.Integer, db.ForeignKey('equipa.id'))
28     data = db.Column(db.String(50))
29     hora = db.Column(db.String(50))
30     local = db.Column(db.String(50))
31     competicao = db.Column(db.String(50))
32     fase = db.Column(db.String(50))
33     jornada_grupo_eliminatoria = db.Column(db.String(50))
```

Figura 4.7: Criação do modelo da base de dados

#### 4.4.2 Implementação do padrão MVC

A arquitetura deste projeto é do tipo cliente-servidor baseado no padrão, onde o cliente é a aplicação Web que corre no *browser* e o servidor é o local onde se encontram armazenados todos os dados necessários para a criação das tabelas de visualização para o utilizador. De forma a possibilitar todas as respostas aos pedidos que são efetuados foi implementada uma arquitetura MVC. A comunicação é efetuada graças ao *Hyper Text Protocol Secure* (HTTPS). A interação com o utilizador faz-se através de links e métodos GET/POST que permitem enviar informação ou indicar pedidos de informação. Os métodos GET e POST são mensagens enviadas pelo utilizador para o controller, com pares chave valor, vindos dos formulários. O GET envia essa informação no URL e o POST envia isso dentro da mensagem. O *Flask* permite responder a pedidos de rotas URL. Na Figura 4.8 podemos ver a comunicação efetuada entre o cliente-servidor que segue o modelo MVC, onde o *controller* faz pedidos e recebe respostas ao *model* e permite a visualização dos dados através do *view*.

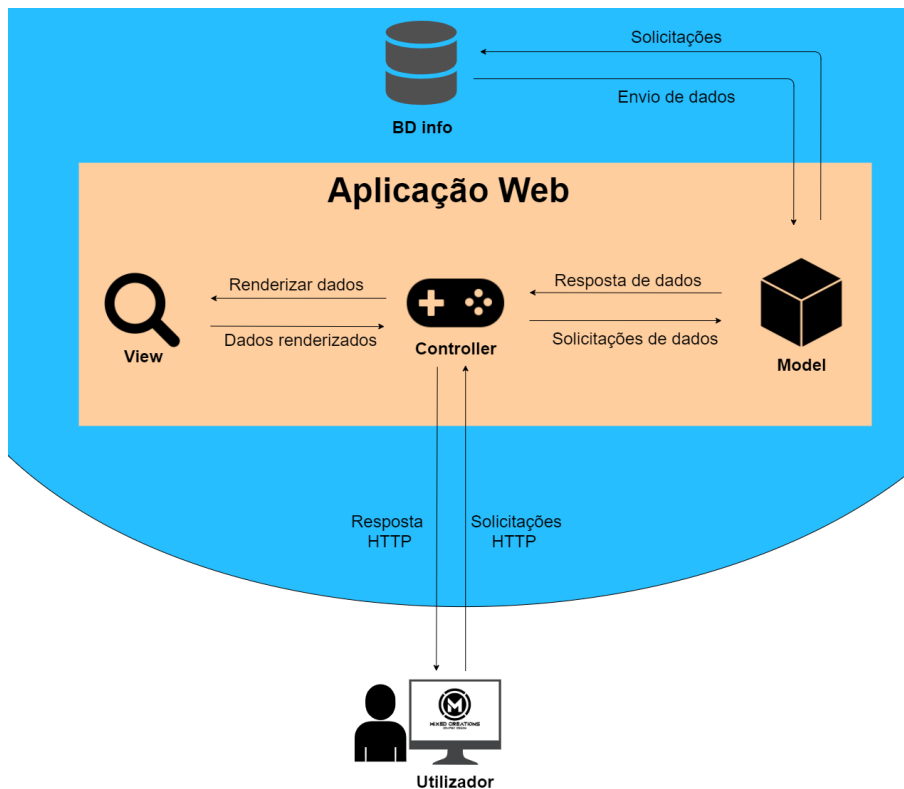


Figura 4.8: Implementação do padrão MVC

As rotas são os URL da aplicação Web. Cada uma é associada, no controlador, a uma função (*controller action, function view*) e ao inserir um URL a aplicação Web irá tentar encontrar uma rota correspondente e caso consiga, chama a função do controlador associado a esta. Na Figura 4.9 está representado um *snippet* de código da rota da página que permite a edição de um jogador.

```
@app.route('/editar_jogador/<string:id>', methods=['GET', 'POST'])
def editar_jogador(id):
    j=Jogador.query.get(id)
    form = EditJogadorForm()
    if request.method=='POST':
        jogador = Jogador.query.get(id)
        jogador.nome = request.form['nome']
        jogador.data_nascimento = request.form['data_nascimento']
        jogador.nacionalidade = request.form['nacionalidade']
        jogador.equipa = request.form['equipa']
        jogador.selecao = request.form['selecao']
        db.session.commit()
        script_inserreequipa(request.form['equipa'])
        return redirect(url_for('ver_jogador'))
    return render_template('editar_jogador.html', title='Editar Jogador', id=id, form=form,
```

Figura 4.9: Rota da página de edição de um jogador

Os métodos *GET* e *POST* estão representados na Figura 4.10, onde está um exemplo da edição dos dados de um jogador e a eliminação de uma equipa da base de dados, o método *GET* é sempre seguido do *POST* com os novos dados a apresentar.

```

127.0.0.1 - - [02/Jun/2020 13:03:42] "POST /editar_jogador/74706 HTTP/1.1" 302 -
127.0.0.1 - - [02/Jun/2020 13:03:42] "GET /ver_jogador HTTP/1.1" 200 -
127.0.0.1 - - [02/Jun/2020 13:06:01] "GET /ver_equipa HTTP/1.1" 200 -
127.0.0.1 - - [02/Jun/2020 13:06:02] "GET /apagar_equipa/11 HTTP/1.1" 302 -
127.0.0.1 - - [02/Jun/2020 13:06:02] "GET /ver_equipa HTTP/1.1" 200 -

```

Figura 4.10: Métodos Get/Post

### 4.4.3 Estrutura das páginas e Funcionalidades

Na Figura 4.26 está esquematizada toda a estrutura da aplicação Web desenvolvida. Existem 7 páginas principais (Página Inicial, Jogadores, Equipas, Jogos, Fotos, Jogos Equipas e Sobre) e dentro de cada uma delas existem ligações com páginas de formulários com determinadas funções para cada caso específico.

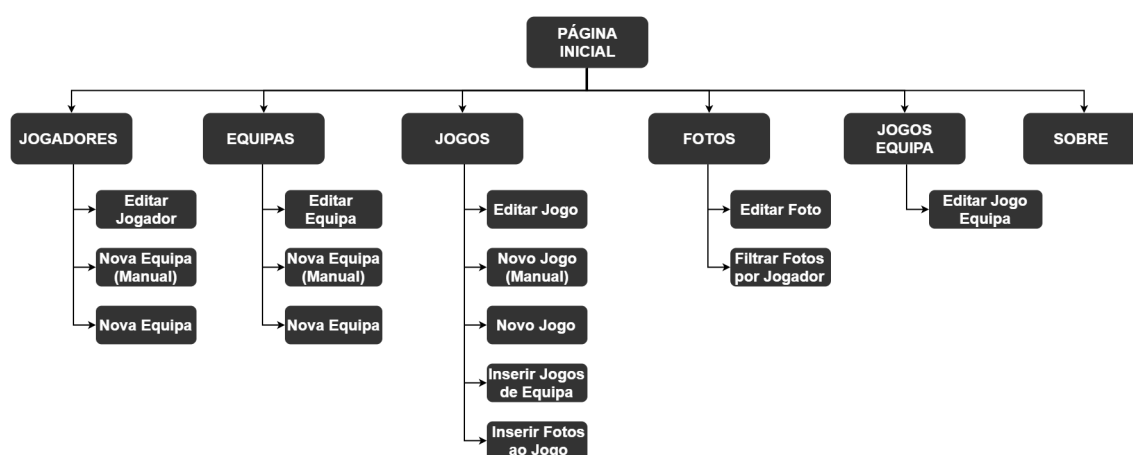


Figura 4.11: Estrutura das Páginas

A implementação de todas as funcionalidades que são realizadas pela aplicação Web foram desenvolvidas de forma a satisfazer todos os requisitos funcionais. Nas próximas sub-seções estão representados várias funcionalidades possíveis de realizar na aplicação Web:

- Adicionar uma equipa descrito na Secção 4.4.3.1
- Adicionar um jogador descrito na Secção 4.4.3.2
- Adicionar jogos descrito na Secção 4.4.3.3
- Adicionar fotos descrito na Secção 4.4.3.4
- Adicionar dados de forma manual descrito na Secção 4.4.3.5
- Editar ou eliminar dados representado descrito na Secção 4.4.3.6

#### 4.4.3.1 Funcionalidade para adicionar uma equipa

O Zerozero tem a sua base de dados organizada por identificadores únicos para cada jogador, equipa e jogo. Para retirarmos as informações todas as informações que necessitamos de um determinado jogador ou equipa a única coisa que precisamos de extrair do

Zerozero é o seu identificador (id), a partir daí o programa vai abrir as páginas que necessita para extrair a informação do jogador, da sua equipa e dos seus respetivos jogos. Os ids de cada equipa estão presentes no URL da página de cada uma, na Figura 4.12 podemos observar o exemplo da página do Sport Lisboa e Benfica onde o seu identificador é o número 4.

The screenshot shows the Zerozero website interface for the Benfica team. The browser address bar displays the URL `zerozero.pt/equipa.php?id=4`. The page header includes the Benfica logo, the team name "Benfica", and location information "Portugal · Lisboa · fundado em 1904". A dropdown menu shows the season "2019/2020". A navigation menu contains links for "MENU", "PÁGINA INICIAL", "ESTATÍSTICAS", "TODOS OS JOGOS", and "NOTÍCIAS". Below the menu, there are three video thumbnails with titles: "Antevisão em 60 segundos - Benfica x CD Tondela", "Estádios da Luz e de Alvalade podem vir a receber Liga dos Campeões", and "João Ferreira: «Já me estreei pela equipa principal, mas agora quero g...". A "VER TODOS VÍDEOS" link is positioned below these thumbnails. The main content area is divided into a "RESUMO (2019/20) " table and a "NOTÍCIAS" section. The table lists performance metrics for various leagues, and the news section features a headline about Julian Weigl's move to Benfica.

	J	V	E	D	DG
Liga dos Campeões	6	2	1	3	10-11
Europa League	2	0	1	1	4-5
Liga Portuguesa	24	19	2	3	52-14
Taça de Portugal	6	5	1	0	15-7
Supertaça	1	1	0	0	5-0
Taça da Liga	3	0	3	0	3-3
	42	27	8	7	89-40

**NOTÍCIAS**

**EX-JOGADOR DO BORUSSIA DORTMUND**  
**Weigl elogia Benfica e explica a mudança: «Eles queriam um jogador sonante para causar sensação na Champions»**  
 Julian Weigl aterrou em Lisboa para reforçar o Benfica na janela de transferências de janeiro do presente ano e fê-lo com grande pompa e circunstância, não fossem os ...

Figura 4.12: Visualização da página da equipa SL Benfica no Zerozero

Para adicionarmos o Benfica à base de dados, a única coisa que precisamos de passar para a aplicação Web desenvolvida é mesmo o identificador e gravar como podemos observar na Figura 4.13.

Figura 4.13: Inserir a equipa na aplicação Web

Após efetuar esta ação, a aplicação Web adicionou a equipa à base de dados com o seu nome, atribui-lhe o ícone do Futebol e registou o estádio do Sport Lisboa e Benfica (Luz) como podemos ver na Figura 4.14 que está pronta para, quando o utilizador quiser, obter os dados dos próximos jogos do clube.

	Nome	Modalidade	Local		
	Benfica		Estádio do Sport Lisboa e Benfica (Luz)		
	Sporting		Estádio José Alvalade		
	SC Covilhã		Estádio Municipal José dos Santos Pinto		
	Atlético CP		Estádio da Tapadinha		
	Elétrico		Estádio Municipal de Ponte de Sor		
	CD Tondela		Estádio João Cardoso		
	Sporting Futsal		Pavilhão João Rocha		
	Aldenovense		Parque Desportivo de Vila Nova de São Bento		
	SC Braga Futsal		Pavilhão Desportivo Universitário de Gualtar		
	AD Fundão Futsal		Pavilhão Municipal do Fundão		

Mostrando 1/10 de 37 resultados

Anterior Próximo

Figura 4.14: Visualização das equipas na aplicação Web

#### 4.4.3.2 Funcionalidade para adicionar um jogador

À semelhança do que acontece com as equipas, os ids de cada jogador estão presentes no URL das suas páginas pessoais como podemos observar na Figura 4.15 um exemplo do Franco Cervi, cliente da Mixed Creations.

The screenshot shows the player profile for Franco Cervi on the Zerozero website. The URL is zerozero.pt/player.php?id=424231. The profile includes a photo, name, age (26), position (Avançado), and current club (Benfica). A navigation menu is visible at the top. Below the profile, there is a section for 'RESUMO 2019/20' with a table of statistics:

	J	M	GM
Liga dos Campeões	5	398	1
Europa League	1	90	0
Liga Portuguesa	16	1125	2
Taça de Portugal	5	337	1
<b>Total</b>	<b>27</b>	<b>1950</b>	<b>4</b>

Below the table, it says 'J=Jogos, M=Minutos, GM=Golos marcados' and 'TODOS OS JOGOS'. To the right, there is a 'DADOS PESSOAIS' section with fields for Name, Birthdate (1994-05-26), Birthplace (Argentina), Nationality (Argentina), and Current Club (Benfica).

Figura 4.15: Visualização da página do jogador Franco Cervi no Zerozero

Para adicionarmos o Cervi à base de dados, a única coisa que precisamos de passar para a aplicação Web desenvolvida é mesmo o identificador e gravar como podemos observar na Figura 4.16.

The screenshot shows a web application interface for adding a player. At the top, there is a logo with the letter 'M' and the word 'EQUIPAS'. Below the logo, there is a red button labeled 'Voltar'. The main heading is 'INSERE O JOGADOR'. Below this, there is a label 'Id' followed by a text input field containing the value '424231'. Below the input field, there is a button labeled 'Gravar'.

Figura 4.16: Inserir o jogador na aplicação Web

Após efetuar esta ação, a aplicação Web adicionou o jogador à base de dados com o seu nome, data de nascimento, nacionalidade e equipa como vemos na Figura 4.17. A aplicação Web tem a capacidade de adicionar a sua equipa à tabela das equipas no caso de esta não existir, como na Secção anterior já tinha adicionado a equipa do Sport Lisboa e Benfica, o processo não teve de ser repetido. A aplicação Web está pronta para quando o utilizador quiser atualizações, adicionar os próximos jogos do clube do Cervi, que neste caso é o SL Benfica.

**JOGADORES**

Novo Jogador Inserir Manualmente Jogador

Pesquisar:

	Nome	Data Nasc.	Nacionalidade	Equipa	Seleção		
	Iker López	1992-11-20	Espanha	Soliss FS Talavera Futsal	-		
	Ted	1999-09-28	Portugal	Pedreles Futsal	-		
	Diogo Basílio	1996-10-10	Portugal	Eléctrico Futsal	-		
	Tiago Nunes	1999-02-02	Portugal	V. Setúbal Futsal	-		
	Rafael Bocum	1991-03-29	Portugal	AD Fundão Futsal	-		
	Carlos Simões	1999-05-24	Portugal	CRI Alhadense Futsal	-		
	Mile Svlilar	1999-08-27	Bélgica	Benfica	-		
	Franco Cervi	1994-05-26	Argentina	Benfica	-		
	Venâncio	1993-07-01	Brasil	Halle-Gooik Futsal	-		
	Gustavo Santos	1996-03-03	Portugal	ADR Retaxo Futsal	-		

Mostrando 21/30 de 58 resultados

[Anterior](#) [Próximo](#)

Figura 4.17: Visualização dos jogadores na aplicação Web

#### 4.4.3.3 Funcionalidade para adicionar jogos

Para adicionar os próximos eventos desportivos disponíveis à aplicação Web existem várias possibilidades. A primeira é ao clicar no botão "Inserir Todos Jogos" que vai adicionar os próximos 1 a 6 jogos de todas as equipas presentes na minha base de dados, caso existam e consoante a sua disponibilidade no Zerozero. O resultado final do clique no botão pode ser observado na Figura 4.18 onde vemos que foram criadas duas tabelas, uma com os jogos que irão decorrer nos próximos 7 dias por ordem crescente de datas e outra com todos os jogos existentes na base de dados.

**JOGOS**

Inserir Todos Jogos Novo Jogo por Equipa Inserir Manualmente Jogo

PRÓXIMOS 7 DIAS

Atualiza Todos Jogos

Pesquisar:

	Equipa Casa	Equipa Fora	Data	Hora	Local	Competição	Fase	JorGrupElim			
↻	Benfica	CD Tondela	2020-06-04	19h15	Estádio do Sport Lisboa e Benfica (Luz)	Liga NOS 2019/20	Campeonato	Jornada 25	📧	📄	🗑️
↻	V. Guimarães	Sporting	2020-06-04	21h15	Estádio D. Afonso Henriques	Liga NOS 2019/20	Campeonato	Jornada 25	📧	📄	🗑️
↻	Portimonense	Benfica	2020-06-10	19h15	Estádio Municipal de Portimão	Liga NOS 2019/20	Campeonato	Jornada 26	📧	📄	🗑️

Mostrando 1/3 de 3 resultados

Anterior Próximo

TODOS OS JOGOS

Pesquisar:

	Equipa Casa	Equipa Fora	Data	Hora	Local	Competição	Fase	JorGrupElim			
↻	Benfica	FC Porto	2020-08-03	17h15	Estádio Nacional do	Taça de Portugal Placard	Final		📧	📄	🗑️

Figura 4.18: Visualização, após a inserção, de todos os próximos jogos de todas as equipas

Uma outra possibilidade, supondo que adicionamos uma nova equipa à base de dados neste preciso momento e não queremos que o *script* verifique de novo a existência de novos jogos ou não para todas as equipas, podemos optar por adicionar os próximos jogos apenas para uma determinada equipa. Imaginemos que acabamos de adicionar o SC Braga ao leque das equipas e queremos adicionar somente os próximos jogos desta equipa, para efetuar este processo selecionamos o SC Braga e clicamos no gravar como podemos ver na Figura 4.19.

**INSERE O JOGO**

Voltar

Choose a team: SC Braga

Gravar

- Benfica
- SC Braga
- Sporting
- SC Covilhã
- Atlético CP
- Eléctrico
- Sporting Futsal

Figura 4.19: Inserção dos próximos jogos do SC Braga

O resultado final é o da adição de 3 jogos, onde o SC Braga irá defrontar as equipas do Famalicão, Boavista e Santa Clara, sendo que apenas o jogo que se vai realizar na Cidade

do Futebol está agendado para os próximos 7 dias como podemos ver na Figura 4.20.

PRÓXIMOS 7 DIAS					
Atualiza Todos Jogos					
	Equipa Casa	Equipa Fora	Data	Hora	Local
	Benfica	CD Tondela	2020-06-04	19h15	Estádio do Sport Lisboa e Benfica (Luz)
	V. Guimarães	Sporting	2020-06-04	21h15	Estádio D. Afonso Henriques
	Santa Clara	SC Braga	2020-06-05	19h00	Cidade do Futebol
	Portimonense	Benfica	2020-06-10	19h15	Estádio Municipal de Portimão

Mostrando 1/4 de 4 resultados

TODOS OS JOGOS					
	Equipa Casa	Equipa Fora	Data	Hora	Local
	FC Famalicão	SC Braga	2020-06-19	21h15	Estádio Municipal de Famalicão
	SC Braga	Boavista	2020-06-13	21h00	Estádio Municipal de Braga
	Santa Clara	SC Braga	2020-06-05	19h00	Cidade do Futebol

Figura 4.20: Visualização, após a inserção dos jogos do SC Braga, de todos os próximos jogos de todas as equipas

A atualização dos dados dos jogos é um fator importantíssimo devido às constantes mudanças de horário, data ou até mesmo de local que podem acontecer. De forma a contornar este problema, foi criada uma função que vai verificar se houve atualizações nos dados de um determinado jogo e no caso de existirem, corrigir essas informações. Este processo é possível de efetuar para um determinado jogo específico. De forma a que não tenha de ser feito um a um, foi criado um botão que verifica se houve atualizações nos jogos que irão decorrer nos próximos 7 dias e corrige-as em caso afirmativo.

#### 4.4.3.4 Funcionalidade para adicionar fotografias

Uma das inovações introduzidas à aplicação Web é a possibilidade de introduzir os *links* que vão conter as edições finais em foto a promover um determinado jogo. Como a Mixed Creations trabalha muito com a *cloud* com os jogadores, o facto de usar os *links* que serão usados para o envio do produto final para os jogadores é bastante eficiente evitando ocupação de memória. Como são fotos que envolvem um certo nível de edição, as mesmas acabam por se tornar um pouco pesadas e através desta forma, o processo torna-se bastante eficiente. As fotos podem ser introduzidas através da escolha do jogo, da inserção do respetivo *link* e da seleção jogador para que a mesma foi realizada. A aplicação Web é "inteligente" o suficiente para detetar os jogadores que vão participar num determinado jogo, pelo que ao selecionarmos as fotos do jogo, que por exemplo coloca frente a frente a equipa do Benfica e do Tondela, a aplicação Web só deixa associar fotos ao jogador Franco

Cervi porque é o único jogador presente na base de dados que pertence a uma dessas duas equipas, tal feito é possível de observar na Figura 4.21.

Figura 4.21: Escolha do jogador na inserção de fotos a um determinado jogo

O processo anterior pode ser repetido para todos os jogos presentes na base de dados, pelo que adicionei também a foto realizada para o jogador Rodrigo Battaglia do Sporting CP no jogo que dita uma deslocação ao terreno do Vitória SC. O resultado final é o da visualização de todas as fotos inseridas para aquele determinado jogo. No caso de querermos ver todas as fotos existentes na base de dados, que neste momento são as duas inseridas para os jogadores Franco Cervi e Rodrigo Battaglia, referentes aos primeiros jogos após o regresso das competições fruto da pandemia de COVID-19, podendo ser vistas na Figura 4.22.

Equipa Casa	Equipa Fora	Competição	Fase	JorGrupElim	
Benfica	CD Tondela	Liga NOS 2019/20	Campeonato	Jornada 25	[Edit] [Delete]
Guimarães	Sporting	Liga NOS 2019/20	Campeonato	Jornada 25	[Edit] [Delete]

Figura 4.22: Visualização de todas as fotos existentes na aplicação Web

No caso de querermos ver todas as fotos realizadas apenas para um jogador específico é possível fazê-lo através da seleção do nome do mesmo como podemos observar na Figura 4.23.



Figura 4.23: Visualização de todas as fotos do Rodrigo Battaglia existentes na aplicação Web

#### 4.4.3.5 Funcionalidade para adicionar dados de forma manual

De forma a permitir que situações excepcionais sejam criadas, outra forma de adicionar dados de um jogador, equipa ou jogo é através da inserção manual dos mesmos. Na Figura 4.24 podemos ver um exemplo em que é exibido um formulário, onde é permitido ao utilizador a inserção manual com o preenchimento obrigatório de todos os campos.

Figura 4.24: Inserção de forma manual de uma equipa

#### 4.4.3.6 Funcionalidade para editar ou eliminar dados

Para além do simples adicionar dados, é importante também ser possível a edição e até mesmo a eliminação dos mesmos quando não sejam necessários. Um exemplo prático e simples é quando um jogador deixa de querer os serviços da Mixed Creations, não faz sentido que este continue registado juntamente com todos os outros. A eliminação é possível clicando no botão do lixo que se encontra presente em cada linha de todas as tabelas com

dados presentes na aplicação Web. A edição também deve ser possível de fazer para todo o tipo de dados pelo que ao seleccionar o lápis a verde, o utilizador é redirecionado para um formulário autopreenchido e editável como podemos observar na Figura 4.24 onde após clicar no gravar os dados serão atualizados.



The screenshot shows a web application interface with a black header containing a logo with the letter 'M' and navigation links for 'EQUIPAS', 'JOGADORES', and 'JOGOS'. Below the header is a form for editing team data. The form includes a red 'Voltar' button with a left-pointing arrow, followed by three text input fields: 'NOME:' with the value 'Benfica', 'MODALIDADE:' with the value 'Futebol', and 'LOCAL:' with the value 'Estádio do Sport Lisboa e Benfica'. At the bottom of the form is a 'Gravar' button.

Figura 4.25: Edição dos dados de uma equipa

## 4.5 Lançamento na nuvem

Após o desenvolvimento de todas as funcionalidades e interligação de todas as tecnologias, o lançamento da aplicação Web na nuvem irá permitir a disponibilização da aplicação Web 24 horas por dia através do acesso à seguinte página <http://mixedcreations.herokuapp.com/> [1]. O *Heroku* foi a tecnologia escolhida pelo facto de suportar várias linguagens e permitir ao programador criar, executar e lançar aplicações na Web. Na Figura 4.26 podemos observar a aplicação Web da Mixed Creations a correr na Internet.



Figura 4.26: Aplicação Web a correr na nuvem (Heroku)

## 4.6 Produto Final

Com a aplicação Web a funcionar, está tudo pronto para a realização de um cartaz para um evento desportivo sem a necessidade de recorrer a *websites* externos que contenham as informações do mesmo. Assim sendo, na Figura 4.27 está o produto final da edição de uma foto para o Franco Cervi que irá participar no jogo do Benfica contra o Portimonense, onde as informações presentes na foto foram todas retiradas automaticamente da aplicação Web desenvolvida.



Figura 4.27: Produto final da promoção de um jogo feita para o jogador Franco Cervi

Após a realização do cartaz promocional, é possível associar este a um determinado jogo para que no futuro fiquem registados os cartazes que foram feitos ao longo do tempo para um determinado jogo e jogador. Na Figura 4.28 está ilustrado a associação do cartaz ao

jogo e na Figura 4.29 está representado o histórico de todas as fotos realizadas onde podemos observar a presença do produto final do cartaz feito para o jogo do Benfica. A aplicação Web foi desenvolvida de forma a permitir a visualização de todos cartazes elaborados por jogo ou por jogador.

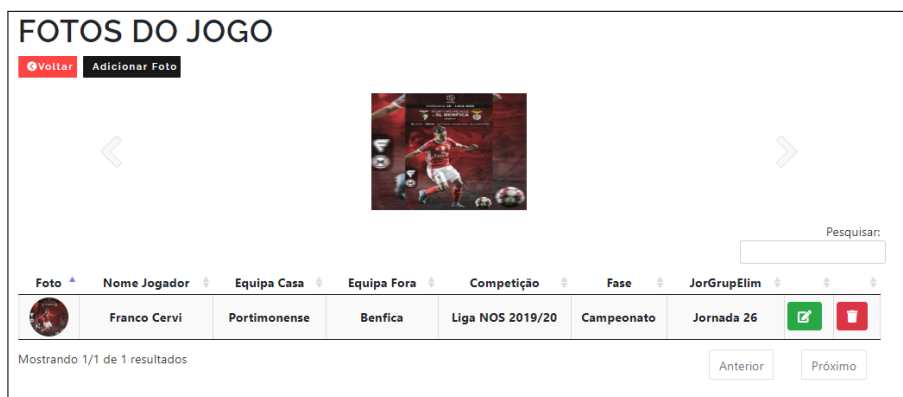


Figura 4.28: Registo do cartaz desenvolvido na aplicação Web

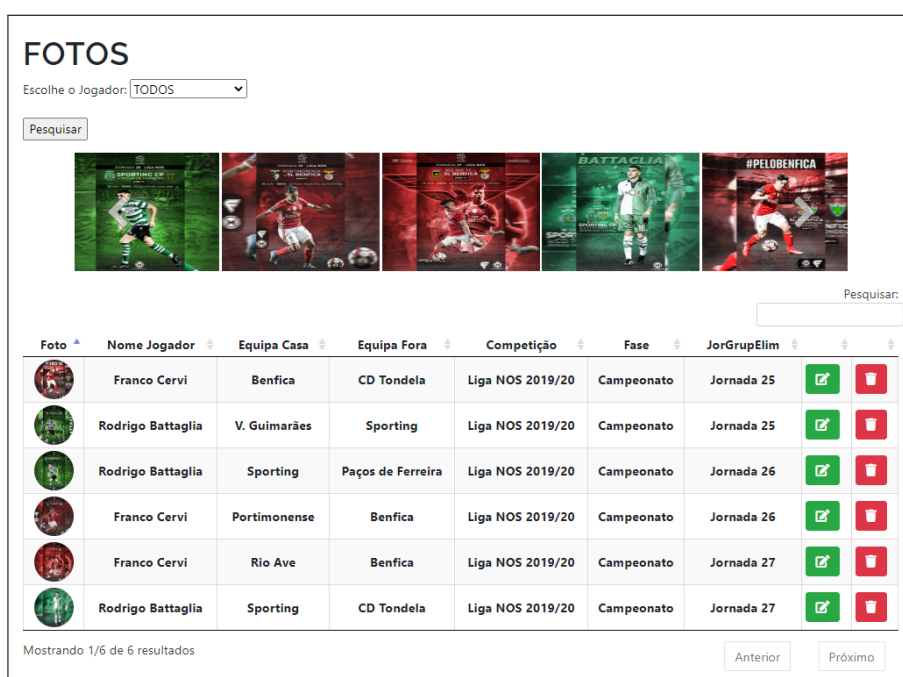


Figura 4.29: Histórico de todos os cartazes realizados

## 4.7 Conclusões

Neste Capítulo a arquitetura e a implementação foram a palavra de ordem. A arquitetura da plataforma foi a primeira coisa a ser desenhada permitindo uma melhor percepção de como é que iriam ser utilizadas as ferramentas e tecnologias escolhidas. A escolha do Zerozero deveu-se ao facto de ser a maior base de dados de futebol e modalidades do mundo, o que permite a obtenção dos dados seguindo o mesmo esquema para os diversos desportos. Se tivesse optado por escolher outras fontes de informação teria de desenvolver

um algoritmo específico para obter os jogos de Futebol, um outro para os jogos de Futsal, entre outros. Este processo iria dispendir muito mais tempo, já para não falar de que corria o risco de obter informações de *websites* onde os dados possam demorar a ser atualizados. Assim sendo, o Zerozero pela sua reputação e qualidade na informação tornou-se na escolha mais fácil e acertada. Poderiam ter sido usadas outras tecnologias como o *Node.js*, o *Laravel* mas devido à facilidade e rapidez de implementação optei pelo *Flask* que me permitiu criar rotas para as minhas várias páginas da aplicação Web de forma bastante simples e intuitiva. A criação da base de dados implementa o *SQLAlchemy* que possibilita a interação uma base de dados *SQLite*. A linguagem utilizada foi o *Python* devido à facilidade de adaptação da mesma e por suportar várias bibliotecas que foram cruciais para o desenvolvimento de todas as funcionalidades propostas. No que diz respeito à mineração de dados, a biblioteca *BeautifulSoup* foi a escolhida para efetuar o *web scraping* e obter todos os dados necessários, enquanto que o *RegEx* teve a função de tratar todos os dados obtidos. Após a aplicação Web estar feita seguiu-se o seu lançamento na *cloud* através do *Heroku*, que dá suporte a todas as ferramentas e tecnologias utilizadas na aplicação Web. No final de contas, a conjugação de todas estas ferramentas permitiu o desenvolvimento de uma ferramenta muito poderosa que implementa um modelo automático e não supervisionado.



# Capítulo 5

## Testes e Análise de Resultados

O quinto Capítulo é o dos testes e análise de resultados, na Secção 5.1 foram feitos os testes para confirmar se estava tudo de acordo com os requisitos funcionais e após a realização destes, é possível comparar os resultados previstos com os alcançados. Na Secção 5.2 está presente a análise de resultados onde podemos ver o tempo dispendido na obtenção da informação antes e depois da *web application* estar a funcionar.

### 5.1 Testes

#### 5.1.1 Testes e Resultados Previstos

De forma a validar a forma de como foram implementadas as funcionalidades, os testes e os resultados previstos são importantíssimos para avaliar se a aplicação Web cumpre todos os requisitos funcionais especificados na Secção 3.1.1 do Capítulo 3. Na Tabela 5.1 podemos observar uma tabela com os testes, os resultados previstos e com a associação de cada um aos requisitos funcionais correspondentes.

Tabela 5.1: Testes e Resultados Previstos

<b>Nº</b>	<b>RF</b>	<b>Funcionalidade</b>	<b>Resultado Previsto</b>
1	RF3.1	Representar os jogadores numa tabela	Mostrar os jogadores na tabela
2	RF2.1	Representar as equipas numa tabela	Mostrar as equipas na tabela
3	RF1.1, RF1.2, RF1.3	Representar os jogos e respetiva informação em duas tabelas (Próximos 7 dias e todos os jogos)	Mostrar os jogos nas duas tabelas
4	RF5.1	Representar os ids das equipas presentes nos jogos numa tabela	Mostrar os ids de todas as equipas presentes em todos os jogos
5	RF4.1	Representar as fotos de um jogador numa tabela e galeria	Mostrar as fotos de um jogador numa tabela e galeria
6	RF6.1	Representar na página inicial uma imagem de apresentação seguida dos atuais jogadores, equipas e modalidades	Mostrar uma imagem de apresentação e uma tabela com os atuais jogadores, equipas e respetivas modalidades
7	RF4.2	Representar as fotos de um determinado jogo numa tabela e galeria	Mostrar as fotos associadas a um determinado jogo numa tabela e galeria
8	RF1.4	Pesquisar dados de todas as tabelas através de texto	Mostrar dados pesquisados
9	RF1.5, RF2.2, RF3.2, RF4.3	Permitir a edição das informações presentes nas diversas tabelas	Mostrar os dados correspondentes à linha selecionada
10	RF1.5, RF2.2, RF3.2, RF4.3	Permitir a eliminação das informações presentes nas diversas tabelas	Mostrar a tabela correspondente à linha eliminada sem a mesma
11	RF1.5, RF2.2, RF3.2	Permitir a atualização das informações presentes nas diversas tabelas	Mostrar a tabela correspondente com as atualizações efetuadas no caso de existirem
12	RF1.4	Ordenar os dados de uma tabela através da seleção de um determinado campo associado (nomes, data, hora, entre outros)	Mostrar os dados da respetiva tabela filtrados com o campo selecionado, ordenando do primeiro para o último ou vice-versa
13	RF1.1, RF2.1, RF3.1, RF4.1, RF5.1, RF6.1	Representar todas as tabelas com um máximo de 10 linhas e adicionar botões para a frente e trás no caso de a tabela conter mais de 10 linhas	Mostrar todas as tabelas com um máximo de 10 linhas e com dois botões para andar para a frente e para trás
14	RF2.3, RF3.3	Permitir a adição de jogadores e equipas através do id do Zerozero do jogador ou da equipa	Mostrar a respetiva tabela já com o novo jogador equipa
15	RF1.6, RF2.3, RF3.3, RF4.4	Permitir a adição de jogadores, jogos, equipas ou fotos de forma manual a partir de formulários	Mostrar a respetiva tabela já com o novo jogador, jogo, equipa ou foto
16	RF1.6	Permitir a adição de todos os jogos disponíveis de todas as equipas existentes na base de dados	Mostrar a respetiva tabela já com os próximos jogos de todas as equipas
17	RF1.6	Permitir a adição de jogos por equipa	Mostrar a respetiva tabela já com os novos jogos da respetiva equipa
18	RF3.4	Permitir que o campo seleção seja editável de forma a poder adicionar os jogadores consoante as convocatórias	Mostrar a respetiva tabela dos jogadores com o respetivo campo seleção atualizado
19	RF6.2	Representar numa imagem publicitária todos os contatos da Mixed Creations	Mostrar na página inicial uma imagem publicitária com os contatos da Mixed Creations
20	RF7.1	Representar um texto que faça um enquadramento acerca do desenvolvimento da aplicação Web	Mostrar uma página sobre onde será enquadrado o desenvolvimento da aplicação Web

### 5.1.2 Resultados Alcançados

Após todos os testes enumerados na Secção anterior efetuados, na Tabela 5.2 podemos observar as funcionalidades alcançadas e rapidamente chegamos à conclusão que todas as funcionalidades foram implementadas com sucesso.

Tabela 5.2: Resultados Alcançados

Nº	Funcionalidade Alcançada
1	Os jogadores são corretamente representados na tabela
2	As equipas são corretamente representadas na tabela
3	Os jogos são corretamente representados nas duas tabelas
4	Os ids das equipas e jogos são corretamente representados na tabela
5	As fotos de um jogador são corretamente representadas na tabela e galeria
6	A imagem de apresentação e os atuais jogadores, equipas e respetivas modalidades ligados à Mixed Creations são corretamente representados na tabela
7	As fotos de um determinado jogo são corretamente representadas na tabela e galeria
8	Os dados pesquisados são corretamente filtrados e representados na tabela
9	Os dados correspondentes à linha selecionada são corretamente preenchidos no formulário
10	A tabela correspondente à linha eliminada é corretamente representada sem a linha eliminada
11	A tabela é corretamente representada após as atualizações
12	A tabela é corretamente apresentada por ordem em função do campo selecionado
13	As tabelas são corretamente apresentadas com um máximo de 10 linhas por tabela e com os dois botões
14	A tabela é corretamente apresentada já com o novo jogador ou equipa inserido automaticamente
15	A tabela é corretamente apresentada já com o novo jogador, equipa, jogo ou foto inserido manualmente
16	A tabela é corretamente apresentada já com os próximos jogos de todas as equipas
17	A tabela é corretamente apresentada já com os novos jogos da respetiva equipa
18	A tabela é corretamente apresentada já o campo seleção atualizado
19	A imagem publicitária com os contatos da Mixed Creations é corretamente apresentada
20	O texto sobre o desenvolvimento da aplicação Web é corretamente apresentado

## 5.2 Análise de Resultados

Os resultados obtidos com a realização desta dissertação são bastante positivos, o facto de ter desenvolvido uma aplicação Web que tem a capacidade de mitigar um dos problemas que surgia semanalmente é motivo para poder afirmar que o objetivo foi alcançado com sucesso.

Antes do desenvolvimento da aplicação Web para aceder às informações de um determinado evento desportivo era necessário aceder a um *browser*, navegar pelo Zerozero ou pelo *website* da Federação Portuguesa de Futebol (FPF) que contém ligações para os outros *websites* de cada associação distrital de futebol espalhada pelo país. Para obter as informações de um determinado jogo é necessário aceder à respetiva competição e assim podem ser vistos os dados desse evento desportivo. Todo este processo multiplicado

por todas as equipas dos jogadores pertencentes à Mixed Creations resulta num processo demorado que no final de cada mês ainda soma algumas horas que podem ser bastante reduzidas com a introdução da automação. De forma a poder avaliar as diferenças entre o uso e o não uso da aplicação Web, decidi fazer três testes que darão a informação do tempo despendido na obtenção da informação dos eventos desportivos. Para efetuar os testes serão usadas duas fontes de informação. Uma delas foi o Zerozero por se tratar da maior base de dados do mundo e por conter registos de todas as competições dos jogadores registados na aplicação Web. A outra foi o *website* da FPF devido a ser a principal fonte de informação oficial e por a esmagadora maioria dos clientes da Mixed Creations pertencer a equipas que competem em competições da FPF ou das respetivas associações de futebol locais. Em último, será testada a aplicação Web que implementa um modelo automático e não supervisionado.

Posto isto, foi feita uma simulação na procura da informação do próximo jogo de cada equipa dos jogadores com ligação à Mixed Creations de forma manual nos *websites* do Zerozero e da FPF, sendo registado o tempo final despendido por semana na obtenção dessa informação. A informação de cada evento desportivo foi efetuada antes da realização de cada cartaz, no início de cada semana, e no final da semana o processo era repetido para confirmar que não haviam ocorrido alterações na data ou local do evento. Usando o Zerozero a soma do tempo despendido ao final de uma semana normal de trabalho registou uma média de 127 minutos na procura da informação de 40 eventos desportivos no início da semana e no final da semana na confirmação dos dados do mesmo, o que equivale a uma média 2 horas e 7 minutos. Se multiplicarmos por 4 que é o número de semanas que tem um mês tem estaremos a falar de uma média 8 horas e 28 minutos, o que corresponde em média a um dia de trabalho. Usando o *website* da FPF e as respetivas associações o tempo ainda foi superior, registou uma média de 150 minutos na procura e confirmação da informação de um mesmo número de eventos desportivos, o que equivale a cerca de 10 horas semanais. Este aumento de tempo em comparação com o Zerozero pode ser explicado por algumas falhas a nível informático nos centro de resultados das diversas associações de futebol. Apesar de se tratar de uma fonte de informação oficial, aconteceu várias vezes o sistema não reconhecer a temporada atual (2019/20) e apresentar as competições de épocas anteriores. Isso implica uma constante seleção da época desportiva 2019/20 no topo da página Web e um conseqüente, maior tempo na obtenção da informação. Por fim foram efetuados os testes à aplicação Web desenvolvida e se tivermos em conta que o processo é feito de forma automática e não supervisionado podemos logo à partida esperar excelentes resultados. Com a aplicação aberta e após um clique no botão "Insere todos jogos" e após outro clique num outro botão "Atualiza Jogos" que confirma em tempo real se houve alguma alteração nos dados de algum jogo foi registado uma média de 2 minutos semanais despendidos na obtenção da mesma informação, estamos a falar de uma poupança enorme de tempo e o processo é feito de forma automática pelos *scripts* desenvolvidos para o efeito.

Na Figura 5.1 temos um gráfico que mostra em minutos o tempo despendido na obtenção da informação dos eventos desportivos em função da fonte de informação escolhida para

a obtenção da informação.

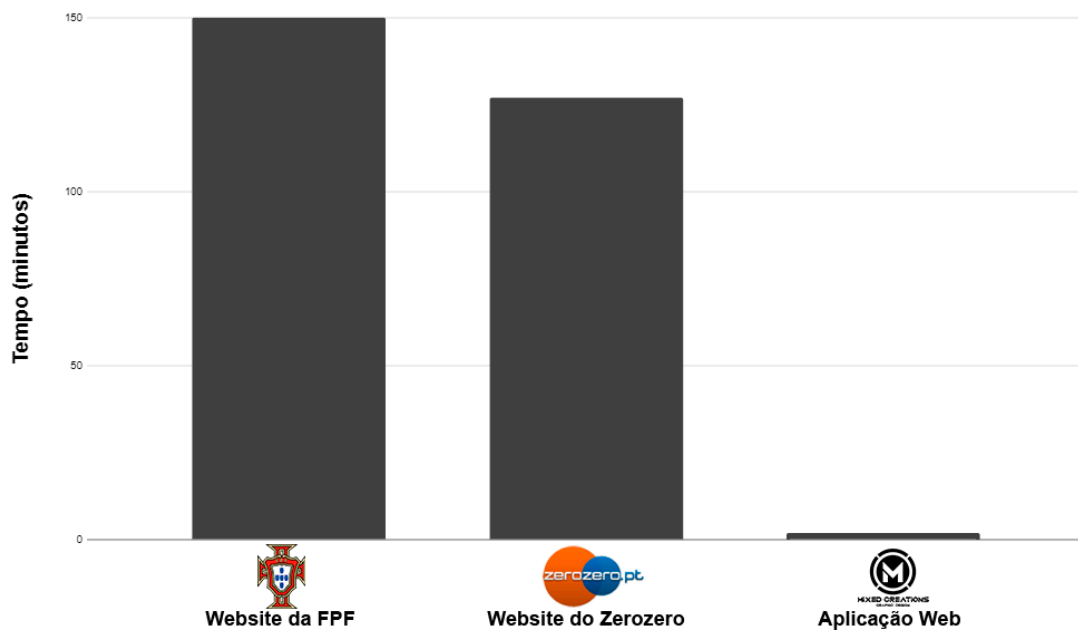


Figura 5.1: Tempo médio dispendido na obtenção da informação de eventos desportivos

Após analisar os resultados obtidos podemos afirmar que com a criação da aplicação Web o único tempo dispendido na procura da informação é o do clique nos botões de inserção ou atualização dos dados, o processo de procura e obtenção dos dados passa a ser feito de forma automática. O utilizador pode muito bem efetuar outras tarefas enquanto o programa se encarrega de adicionar novas informações dos próximos eventos desportivos disponíveis. Assim sendo, podemos afirmar que o desenvolvimento da aplicação Web é um grande contributo para a Mixed Creations nos seus tempos de resposta, evitando no final do mês perdas de tempo superiores às 8 horas.



# Capítulo 6

## Conclusões e Trabalho Futuro

### 6.1 Conclusões Principais

Ao começar a dissertação foram definidas metas que foram tidas em conta ao longo da realização da mesma. O principal objetivo foi o de criar uma ferramenta inovadora, que não existe no mercado, para dar resposta às necessidades da Mixed Creations. À medida que fui especificando os requisitos e criando o protótipo foram-me nascendo novas ideias de inovação que acabaram por ser implementadas na aplicação Web, como por exemplo a possibilidade de ver os trabalhos finais desenvolvidos para um evento onde participou um determinado jogador.

Numa fase inicial foi feito o enquadramento e a motivação da dissertação, falando da Mixed Creations e com ênfase na cada vez maior importância do *marketing* desportivo na vida dos clubes, bem como dos atletas. Os objetivos definidos antes da realização da aplicação Web foram atingidos com sucesso, sendo criado um sistema automático e não supervisionado que será uma mais valia para o que vai ser o trabalho da Mixed Creations na próxima época desportiva. Também poderá ser vista como algo bastante útil para outras pessoas que procuram ter os registos de todos os jogos dos seus jogadores, nomeadamente agências ou agentes desportivos.

Após ser feita uma pesquisa das soluções que existem no mercado, a conclusão é que apesar de existirem ferramentas que permitam a obtenção dos dados das equipas não há nenhuma que nos permita ter a nossa lista de jogadores e de equipas, tendo como objetivo a organização da forma como visualizamos os eventos desportivos. Além disso, outro ponto a favor é o facto de poderem ser associadas fotos a jogadores que participaram num determinado jogo, isso é claramente um ponto a favor em termos da flexibilidade da aplicação Web. As ferramentas e tecnologias escolhidas revelaram-se as opções certas, não constituindo uma entrave ao desenvolvimento da aplicação Web.

A definição dos requisitos foi um processo que implicou uma estruturação muito forte para que o sistema pudesse dar resposta a todas as funcionalidades descritas. Uma boa definição foi meio caminho andado para encontrar as melhores soluções que dessem resposta ao problema de forma simples e direta. A definição de layouts muito cedo também foi importantíssima e permitiu uma visão muito melhor de como seria a aplicação Web no final de todo o processo.

Durante a implementação foram surgindo vários problemas mas graças às tecnologias escolhidas foram sempre encontradas as melhores soluções para os solucionar. Toda a implementação foi efetuada com sucesso, destacando-se o automatismo e a flexibilidade da mesma, sendo introduzidas novas ideias à medida do desenvolvimento do projeto. O facto de ter lidado com novas plataformas permitiu-me desenvolver novos conheci-

tos e mais competências. Tanto o *Flask* como o *Heroku* são duas ferramentas de trabalho muito poderosas que auxiliam de forma bastante intuitiva e simples o trabalho do programador.

Após todo o trabalho desenvolvido, posso afirmar que este superou todas as minhas expectativas. A nível pessoal consegui melhorar as minhas capacidades de organização e de comunicação, o facto de ter tido sempre uma estrutura bem definida ao longo de todo o projeto foi claramente uma mais valia, tendo sempre como objetivo a longo prazo um possível melhoramento e um acréscimo de novas funcionalidades. Posso afirmar com toda a certeza que foi uma experiência bastante positiva e muito aliciante ter podido fazer uma ponte entre a Engenharia Informática, o desporto e o meu projeto pessoal, a *Mixed Creations*.

## 6.2 Trabalho Futuro

Para trabalho futuro há vários caminhos que podem ser tidos em conta. O primeiro poderá passar pelo aumento da informação que é obtida, poderiam ser adicionadas informações pós jogo como a disponibilização dos resultados finais, marcadores e toda a estatística associada ao evento desportivo e ao rendimento na época do atleta, registo de golos, assistências, cartões, entre outros. Outra intenção futura é a implementação de técnicas de PLN, tal como a categorização de texto, isto é, a obtenção das informações de forma automática no texto. O objetivo passa a ser o de detetarmos que um determinado conjunto específico de caracteres corresponde, por exemplo, à altura de um jogador. Isto irá permitir que a aplicação Web seja inteligente o suficiente para que esta não esteja dependente só de uma determinada fonte de informação. Ao escrevermos "Franco Cervi Benfica" na aplicação Web, esta deverá ser o suficientemente "inteligente" para ir descobrir todas as informações pessoais daquele jogador. A criação de uma aplicação móvel para *android* e *iOS* que implemente todas as funcionalidades que já estão implementadas na aplicação Web e as que foram descritas anteriormente, seriam uma ferramenta muito poderosa pois permitiriam em qualquer sítio e em qualquer lugar o acesso à informação sem a dependência de uma fonte principal de informação. Por outro lado, e tendo em conta de que a aplicação Web criada já é uma ferramenta bastante inovadora no que diz respeito à organização do calendário desportivo, um outro dos caminhos que pode ser seguido é o da disponibilização de um serviço com as informações que a aplicação Web dá ao utilizador segundo alguns níveis de utilizador. Podíamos ter um utilizador *free* onde apenas podia registar 5 jogadores na aplicação Web e outro *premium* que pagava para utilizar a aplicação Web de forma ilimitada acedendo às informações dos jogadores e equipas que associasse à sua conta.

# Bibliografia

- [1] Mixed Creations. Aplicação da mixed creations. Acedido em junho de 2020. Available from: <http://mixedcreations.herokuapp.com/>. v, vii, 3, 10, 59
- [2] Fábio Mota. Mixed creations. Repositorium UBI, 2018. 1
- [3] Mixed Creations. Aplicação da mixed creations desenvolvida no projeto final de licenciatura. Acedido em junho de 2020. Available from: <https://mixedcreations2017.wixsite.com/mixedcreations>. 1
- [4] Bruno André Paulino Costa. O impacto do social media marketing na qualidade dos relacionamentos dos fans desportivos com os clubes de futebol. 2014. <http://repositorium.sdum.uminho.pt/handle/1822/29352>. 1, 18
- [5] zerozero. Zerozero. Acedido em junho de 2020. Available from: <https://www.zerozero.pt/>. 8
- [6] sportradar. Sportradar. Acedido em junho de 2020. Available from: <https://www.sportradar.com/>. 8
- [7] Wyscout. Wyscout. Acedido em junho de 2020. Available from: <https://wyscout.com/>. 9
- [8] Flashscore. Flashscore. Acedido em junho de 2020. Available from: <https://www.flashscore.pt/>. 10
- [9] Sofascore. Sofascore. Acedido em junho de 2020. Available from: <https://www.sofascore.com/pt/>. 10
- [10] Livescore. Livescore. Acedido em junho de 2020. Available from: <https://www.livescore.com/>. 10
- [11] Besoccer. Besoccer. Acedido em junho de 2020. Available from: <https://pt.besoccer.com/>. 10
- [12] Mixed Creations. Facebook da mixed creations. Acedido em junho de 2020. Available from: <https://www.facebook.com/mixedcr/>. 11
- [13] Mixed Creations. Instagram da mixed creations. Acedido em junho de 2020. Available from: <https://www.instagram.com/mixed.creations/>. 11
- [14] Angura Guruge. *Web Services: Theory and Practice*. Elsevier, 1 edition, 2004. 14
- [15] P. Wang and A. Kuzmanovic. “what’s the score?”: A first look at sports live data feed services. In *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pages 220–225, 2018. <https://ieeexplore.ieee.org/document/8406822>. 18

- [16] D. M. Thomas and S. Mathur. Data analysis by web scraping using python. In *2019 3rd International conference on Electronics, Communication and Aerospace Technology (ICECA)*, pages 450–454, 2019. <https://ieeexplore.ieee.org/document/8822022>. 18
- [17] D. K. Mahto and L. Singh. A dive into web scraper world. In *2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom)*, pages 689–693, 2016. <https://ieeexplore.ieee.org/document/7724353>. 18
- [18] K. Sundaramoorthy, R. Durga, and S. Nagadarshini. Newsone – an aggregation system for news using web scraping method. In *2017 International Conference on Technical Advancements in Computers and Communications (ICTACC)*, pages 136–140, 2017. <https://ieeexplore.ieee.org/document/8067594>. 19
- [19] A. Maududie, W. E. Y. Retnani, and M. A. Rohim. An approach of web scraping on news website based on regular expression. In *2018 2nd East Indonesia Conference on Computer and Information Technology (EIConCIT)*, pages 203–207, 2018. <https://ieeexplore.ieee.org/abstract/document/8878550>. 19
- [20] R. Diouf, E. N. Sarr, O. Sall, B. Birregah, M. Bousso, and S. N. Mbaye. Web scraping: State-of-the-art and areas of application. In *2019 IEEE International Conference on Big Data (Big Data)*, pages 6040–6042, 2019. <https://ieeexplore.ieee.org/abstract/document/9005594>. 19
- [21] Rosario Girardi Leandro Balby Marinho. Mineração na web. UFMA. Acedido em junho de 2020. Available from: <https://paginas.fe.up.pt/~mgi03006/ARI/MineracaoNaWeb.pdf>. 19
- [22] Minewiskan. Modelos de mineração (analysis services-mineração de dados). Acedido em junho de 2020. Available from: [https://docs.microsoft.com/pt-br/analysis-services/data-mining/mining-models-analysis-services-data-mining?view=asallproducts-allversions#bkmk\\_md1Arch](https://docs.microsoft.com/pt-br/analysis-services/data-mining/mining-models-analysis-services-data-mining?view=asallproducts-allversions#bkmk_md1Arch). 19
- [23] D. Fernandes, D. Clemente, G. Soares, P. Sebastião, F. Cercas, R. Dinis, and L. S. Ferreira. Cloud-based implementation of an automatic coverage estimation methodology for self-organising network. *IEEE Access*, 8:66456–66474, 2020. <https://ieeexplore.ieee.org/document/9060956>. 20
- [24] Python. Python. Acedido em junho de 2020. Available from: <https://www.python.org/doc/>. 41
- [25] BeautifulSoup. BeautifulSoup. Acedido em junho de 2020. Available from: <https://pypi.org/project/beautifulsoup4/>. 41
- [26] W3Schools. RegEx. Acedido em junho de 2020. Available from: [https://www.w3schools.com/python/python\\_regex.asp/](https://www.w3schools.com/python/python_regex.asp/). 41

- [27] SQLAlchemy. SQLAlchemy. Acedido em junho de 2020. Available from: <https://www.sqlalchemy.org/>. 41
- [28] Flask. Flask. Acedido em junho de 2020. Available from: <https://www.palletsprojects.com/p/flask/>. 41
- [29] Heroku. Heroku. Acedido em junho de 2020. Available from: <https://www.heroku.com/>. 42
- [30] Visual Studio Code. Visual Studio Code. Acedido em junho de 2020. Available from: <https://code.visualstudio.com/>. 42
- [31] Adobe Photoshop. Adobe photoshop. Acedido em junho de 2020. Available from: <https://www.adobe.com/pt/products/photoshop.html>. 42