



UNIVERSIDADE  
LUSÓFONA

# Aplicação para a gestão de uma Junta de Freguesia

## Trabalho Final de curso

Relatório Final

Nome do Aluno: Filipe Simões

Nome do Orientador: Acácio Carmona

Trabalho Final de Curso | LEI | 27/01/2023

[www.lusofona.pt](http://www.lusofona.pt)

## **Direitos de cópia**

*(Aplicação para a gestão de torneios realizados pela Junta de Freguesia),*  
Copyright de (Filipe Simões), ULHT.

A Escola de Comunicação, Arquitectura, Artes e Tecnologias da Informação (ECATI) e a Universidade Lusófona de Humanidades e Tecnologias (ULHT) têm o direito, perpétuo e sem limites geográficos, de arquivar e publicar esta dissertação através de exemplares impressos reproduzidos em papel ou de forma digital, ou por qualquer outro meio conhecido ou que venha a ser inventado, e de a divulgar através de repositórios científicos e de admitir a sua cópia e distribuição com objectivos educacionais ou de investigação, não comerciais, desde que seja dado crédito ao autor e editor.

---

## Resumo

As Juntas de Freguesia desempenham um papel de relevo e de proximidade com os seus residentes.

Foi definido como objetivo deste trabalho implementar uma ou mais aplicações a definir pela Junta de Freguesia, melhorando a forma de fazer pedidos e comunicar situações e ocorrências facilitando assim a otimização de recursos.

Em reunião realizada com a Junta de Freguesia, esta indicou que a aplicação que lhes seria mais útil era uma aplicação para otimizar a gestão de um projeto criado pela Junta de Freguesia de Almargem do Bispo em 2006, que foi a realização de um torneio de Futsal (Taça Barnabé), mantendo-se a sua realização até ao momento.

Desta forma pretende-se com este trabalho criar uma aplicação que permita agilizar a organização do torneio atualmente extensiva às coletividades da União de Freguesias de Almargem do Bispo, Pêro Pinheiro e Montelavar.

## **Abstract**

The Parish Councils play a significant and close role with their residents. The objective of this project is to implement one or more applications as defined by the Parish Council, improving the process of making requests and communicating situations and incidents, thus facilitating resource optimization.

In a meeting with the Parish Council, they indicated that the most useful application for them would be one to optimize the management of a project created by the Almargem do Bispo Parish Council in 2006, which was the organization of a Futsal tournament (Barnabé Cup), which has been ongoing until now.

In this way, this project aims to create an application that streamlines the organization of the tournament, which currently involves the collectives of the Union of Parishes of Almargem do Bispo, Pêro Pinheiro, and Montelavar.

---

# Índice

Resumo .....	iii
Abstract.....	iii
Índice .....	v
Lista de Figuras.....	vi
1 Identificação do Problema .....	1
2 Viabilidade e Pertinência.....	2
3 Benchmarking .....	3
4 Engenharia .....	5
5 Solução Proposta .....	9
6 Plano de testes e validação .....	12
7 Método e Planeamento .....	13
Bibliografia .....	14
Anexo 1 .....	15
Anexo 2 .....	16
Anexo 3 .....	17
Glossário .....	18

## Lista de Figuras

Figura 1,2 e 3 – Diagramas de Actividade.	8 e 9
Figura 4 – Diagrama Entidade - Modelo.	10
Figura 5 – Diagrama Entidade - Relação.	11
Figura 6 – Model da Tabela CadernoEleitoral	15
Figura 7 - Model da Tabela UsersType	15
Figura 8 – Router da Tabela UsersType	16
Figura 9– Route da Tabela Users	18
Figura 10 e 11 – Código que permite navegar pelas diferentes páginas	18 e 19
Figura 12 – Login	20
Figura 13 – Login Clube bem sucedido	20
Figura 14 – LocalStorage com informação guardada.	21
Figura 15 – Inscrição Jogadores	22
Figura 16 – Inscrição Equipa Técnica	25
Figura 17 – Inscrição Equipas	25
Figura 18 – Informação Equipas Clubes	26
Figura 19 e 20 – Informação Jogadores e Técnicos Clubes	27
Figura 21 – Informação Equipas Admin	28
Figura 22 e 23 – Informação Jogadores e Técnicos Admin	29
Figura 24 – Fichas Jogo e Equipa	30
Figura 25 – Ficha Jogo	31
Figura 26 e 27 – Ficha Jogo downloads	32
Figura 28 – Ficha Equipa	33
Figura 29 – Ficha Equipa 1 PDF	33
Figura 30 – Ficha Equipa 2 PDF	34
Figura 31 – Criação Torneios	34
Figura 32 – Criação Jogos	35
Figura 33 – Novo Jogo Criado	36
Figura 34 – Informação Jogos	36
Figura 35 – Informação Jogos	37

---

Figura 36 – Golos Updated	38
Figura 37 – Cartões Updated	38
Figura 38 – Novo Jogo Informação	39
Figura 39 – Torneios	39
Figura 40 – Jogos	40
Figura 41 – Jogo terminado informação	40
Figura 42 – Jogo não terminado informação	41
Figura 43 – CC Dados	41
Figura 44 – Plano de Trabalho.	45
Figura 45 – Ficha Inscrição Jogador	52
Figura 46 – Informação Equipa	53
Figura 47 – Ficha Equipa	54
Figura 48 – Ficha Jogo 1	55
Figura 49 – Ficha Jogo 2	56

## **Lista de Tabelas**

Tabela 1 – Requisitos Funcionais com respetiva prioridade e grau de cumprimento 5 e 6

Tabela 2 – Requisitos Não Funcionais com respetivo grau de prioridade 6

# **1 Identificação do Problema**

Foi definido como objetivo deste trabalho implementar uma ou mais aplicações a definir pela Junta de Freguesia, melhorando a forma de fazer pedidos e comunicar situações e ocorrências facilitando assim a otimização de recursos.

Na reunião realizada com a Junta de Freguesia, foi indicado que a aplicação que lhes seria mais útil era uma aplicação para otimizar a gestão de um projeto criado pela Junta de Freguesia de Almargem do Bispo em 2006, que foi a realização de um torneio de Futsal (Taça Barnabé).

Desta forma, este trabalho tem como objetivo a organização e gestão do torneio de Futsal (Taça Barnabé), que envolve as coletividades da União de Freguesias de Almargem do Bispo, Pêro Pinheiro e Montelavar.

A aplicação desenvolvida oferece uma variedade de funcionalidades, incluindo a inscrição de jogadores, o registo das fichas de jogo, a atualização dos dados das partidas, a disponibilização do calendário de jogos e a verificação de residência. Esta última funcionalidade garante que apenas jogadores que residam ou tenham pelo menos um dos pais residentes em um dos concelhos da União de Freguesias possam participar do torneio.

Além disso, foram acrescentadas funcionalidades não inicialmente propostas para abordar questões adicionais. Estes incluem a criação da ficha de equipa, a inscrição da equipa técnica, a inscrição das equipas em relação aos escalões e a incorporação de uma funcionalidade que permite a visualização de informações relacionadas às equipas.

Todas as informações dos jogadores, jogos e outros dados relevantes foram registrados em um banco de dados SQL. Os resultados obtidos em relação à proposta inicial foram completos, demonstrando a eficácia do projeto na organização e administração do torneio de Futsal.

## **2 Viabilidade e Pertinência**

Este projeto apresenta viabilidade, pois consiste em uma aplicação que facilita a organização e administração do Torneio de Futsal (Taça Barnabé) pela Junta de Freguesia.

Em termos de pertinência e relevância, a aplicação oferece funcionalidades tais como a inscrição de jogadores, equipa técnica e equipas nos seus respetivos escalões, criação de fichas de jogo e fichas de equipa, atualização de dados pós-jogo, calendário de partidas, verificação de residência de jogadores ou pais, e acesso às informações das equipas.

Essas funcionalidades permitem que a Junta de Freguesia gerencie o torneio de forma mais eficiente e simplificada.

É também, importante destacar que este projeto possui potencial para futuros desenvolvimentos, como a criação de uma aplicação móvel complementar, devido à facilidade de reaproveitamento do backend da aplicação já desenvolvida para a sua construção.

### 3 Benchmarking

Na pesquisa que realizei não encontrei nenhuma aplicação que permitisse implementar todas as funcionalidades pretendidas pela Junta de Freguesia nessa mesma só aplicação.

Contudo foi me possível encontrar varias aplicações que permitem realizar certas das funcionalidades, tais como criação de tournament brackets (Ex. [https://challonge.com/tournaments/bracket\\_generator.html](https://challonge.com/tournaments/bracket_generator.html)) e aplicações que permitem guardar e mostrar informação relativamente aos torneios (Ex. <https://leverade.com/pt/manager/barnabe/tournaments>).

Após a realização desta pesquisa conclui não existem aplicações que realizem todas as funcionalidades pretendidas pela Junta, contudo existem aplicações como as acima mencionadas que permitem criar tournament brackets e manejar informação relativamente ao torneio, logo a nível de mercado os concorrentes são muito poucos para a aplicação que esta a ser desenvolvida neste TFC.

Irei utilizar Node/Express uma vez que ao utilizar estas tecnologias irá permitir a viabilidade do projeto a longo termo, uma vez que se no futuro seja pretendido criar uma aplicação móvel só irá ser necessário a criação de front-end, uma vez que o back-end já se encontra em APIs.

No mercado, é possível também realizar esta aplicação através das seguintes tecnologias:

- Laravel – é um framework PHP livre e open-source.
- Django – é uma framework para desenvolvimento rápido para web, escrito em Python que utiliza o padrão model-template-view.
- .NET Framework – é uma iniciativa da empresa Microsoft, que utiliza uma plataforma única para desenvolvimento e execução de sistemas e aplicações.

Neste momento o mercado de trabalho utiliza muito como base de Full Stack Development o conceito de MERN que tem por base as quatro tecnologias que são MongoDB, Express, React e Node tendo este conhecimento foi feita a decisão de usar três destes componentes substituindo o MongoDB pelo MySQL, pois trata-se de um caso em que faz sentido a utilização de um

sistema de base de dados que trabalhe por base de um sistema de entidade - relação com um Schema pré-definido.

Como foi explicado anteriormente, não existe nenhuma aplicação que realize todas as funcionalidades que foram desenvolvidas neste projeto. Portanto, a solução proposta neste TFC atende a uma demanda no mercado ao oferecer um conjunto abrangente de funcionalidades que atualmente não se encontram disponível em uma única aplicação. Se esta aplicação não tivesse sido desenvolvida, os usuários teriam que recorrer a várias aplicações diferentes para obter o mesmo resultado, o que seria inconveniente e ineficiente.

## 4 Engenharia

### 4.1. Levantamento e análise dos Requisitos

A aplicação permite à Junta de Freguesia administrar o torneio de forma mais simples e eficiente.

Requisitos Funcionais			
ID	Requisito	Prioridade	Cumprimento
RF001	Login: Deve ser possível aos clubes fazer login.	Alta	Cumprido
RF002	Login: Deve ser possível ao administrador fazer login.	Alta	Cumprido
RF003	Inscrever Jogadores: Os clubes devem conseguir inscrever os jogadores.	Alta	Cumprido
RF004	Inscrever Técnicos: Os clubes devem conseguir inscrever os técnicos.	Alta	Cumprido
RF005	Consultar Equipas: Os clubes devem conseguir consultar os jogadores e técnicos que já inscreveram.	Alta	Cumprido
RF006	Consultar Equipas: O administrador devem conseguir consultar os jogadores e técnicos que já foram inscritos.	Alta	Cumprido
RF007	Consultar dia do jogo: Qualquer utilizador que utilize a aplicação deve poder consultar quando é que um jogo irá ser realizado.	Alta	Cumprido
RF008	Download Ficha de Jogo: O administrador deve conseguir fazer download da ficha de jogo pretendida.	Alta	Cumprido
RF009	Download Ficha Equipa: O administrador deve conseguir fazer download da ficha de equipa pretendida.	Alta	Cumprido
RF010	Atualizar informação dos jogos: Os administradores devem conseguir editar a informação dos jogos.	Alta	Cumprido
RF011	Inscrever Equipas: O administrador devem conseguir inscrever os clubes nos escalões em que estes forem participar.	Alta	Cumprido
RF012	Download Cartões Cidadão: O administrador deve conseguir fazer download de toda a informação referente aos cartões de cidadão.	Alta	Cumprido

RF013	Consultar Residência Jogadores e Técnicos: O administrador deve conseguir observar e editar o estado de residência dos jogadores e técnicos.	Alta	Cumprido
RF014	Logout: Deve ser possível o utilizador sair da aplicação.	Alta	Cumprido

**Tabela 1 – Requisitos Funcionais com respetiva prioridade e grau de cumprimento**

Requisitos Não Funcionais		
ID	Requisito	Prioridade
RFN01	Servidor disponível integralmente	Alta
RFN02	A aplicação deve permitir uma navegação simples, lógica e intuitiva com um menu que disponibilize as informações necessárias ao utilizador.	Alta
RFN03	A aplicação deve permitir um utilização permanente, em qualquer hora do dia e dia da semana.	Alta
RFN04	A aplicação deve permitir que as sua operações seja efetuadas com rapidez.	Alta
RFN05	Todas as palavras-passe devem estar cifradas.	Alta
RFN06	A base de dados deve estar protegida de acessos não-autorizados	Alta

**Tabela 2 – Requisitos Não Funcionais com respetivo grau de prioridade**

### Funcionalidades da aplicação propostas anteriormente:

- **Inscrição de Jogadores** – Esta funcionalidade permite aos clubes inscreverem os jogadores que irão representar a equipa, de acordo com o escalão em que estão registados.
- **Calendário de Jogos** – Quando um jogo é criado, é também indicada a data em que este será realizado, e essa data é visualizada no recipiente correspondente a esse jogo.
- **Informação de Jogos** – Ao selecionar um jogo, é possível aceder a informações relevantes sobre o mesmo, tais como os jogadores (onde se pode verificar a lista de jogadores relacionados com o jogo e verificar se algum deles está suspenso), a equipa técnica

(informações sobre a equipa técnica envolvida no jogo) e após a conclusão do jogo, é possível consultar os golos (quem marcou o golo e a que momento do jogo) e os cartões (quem recebeu cartões amarelos ou vermelhos).

- **Criação Ficha de Jogo** – Esta funcionalidade permite à Junta de Freguesia criar as fichas de jogos desejadas para cada jogo. Além disso, é possível também verificar se um jogador está suspenso, ou seja, se pode ou não participar no jogo.
- **Verificação de residência** – Esta funcionalidade foi modificada a pedido da Junta de Freguesia. Inicialmente esta funcionalidade iria ser executada automaticamente, contudo, foi alterada de modo a ser realizada manualmente.

### **Funcionalidades da aplicação adicionadas:**

- **Inscrição da Equipa Técnica** – Esta funcionalidade permite aos clubes inscreverem a equipa técnica que irá representar a equipa, de acordo com o escalão em que estão registados.
- **Inscrição das Equipas** – Esta funcionalidade permite à Junta de Freguesia inscrever as clubes nos escalões em que estes pretendem competir.
- **Informação das Equipas** - Esta funcionalidade permite que os clubes e a Junta de Freguesia tenham acesso às informações sobre quais jogadores ou equipa técnica já foram inscritos para um determinado clube e escalão correspondente. Além disso, oferece a capacidade de remover jogadores ou equipa técnica, caso ocorram erros nas inscrições. Os administradores têm também a capacidade de alterar o status de residência, indicando se o jogador ou equipa técnica é ou não residente de uma das Juntas de Freguesia.
- **Obtenção de dados Cartão Cidadão** - Esta funcionalidade permite à Junta obter um documento Exel com todos os dados correspondentes aos cartões de cidadão.

- **Criação Ficha Equipa** – Esta funcionalidade permite à Junta de Freguesia criar as fichas de equipa desejadas para cada jogo. Isso inclui a criação de uma ficha que mostrará os jogadores e a equipa técnica de uma equipa de um determinado escalão.

Este trabalho tem continuidade académica, uma vez que pode ser criada uma aplicação móvel complementar ao projeto desenvolvido.

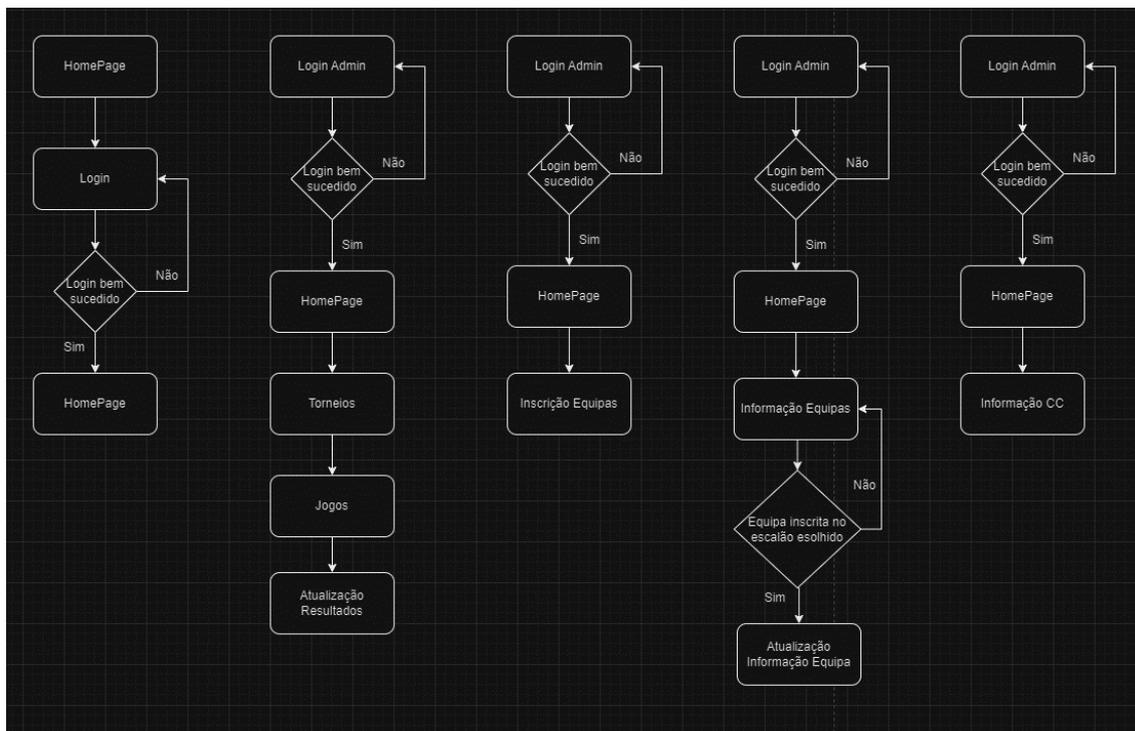
#### 4.2. Diagramas de Casos de Uso ou descrição de cenários de aplicação

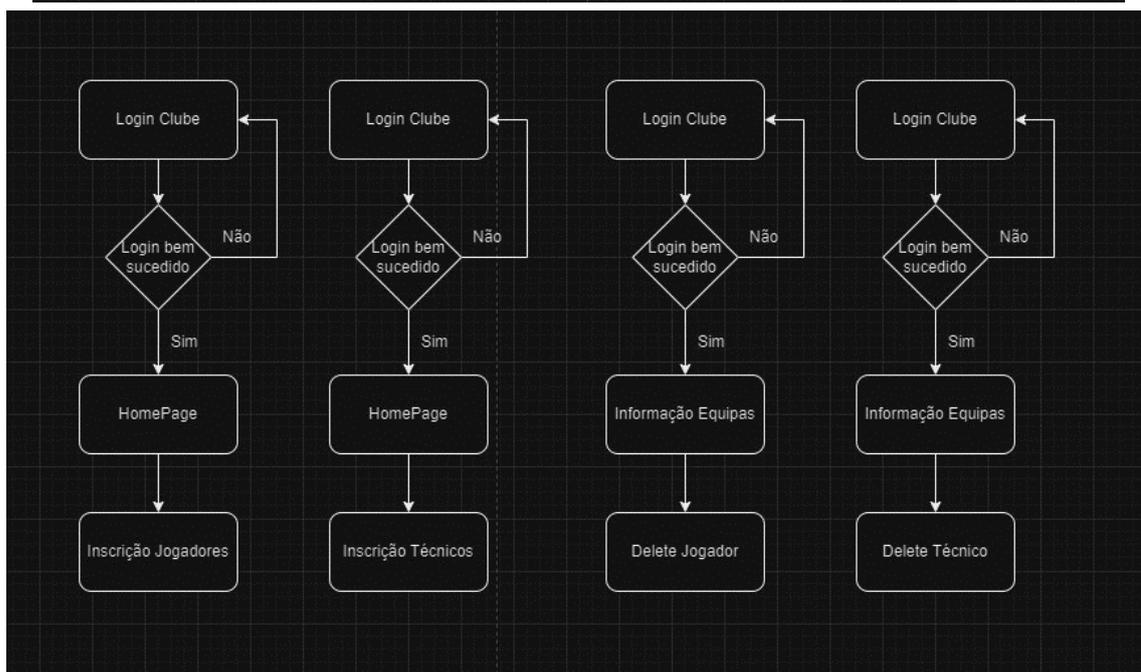
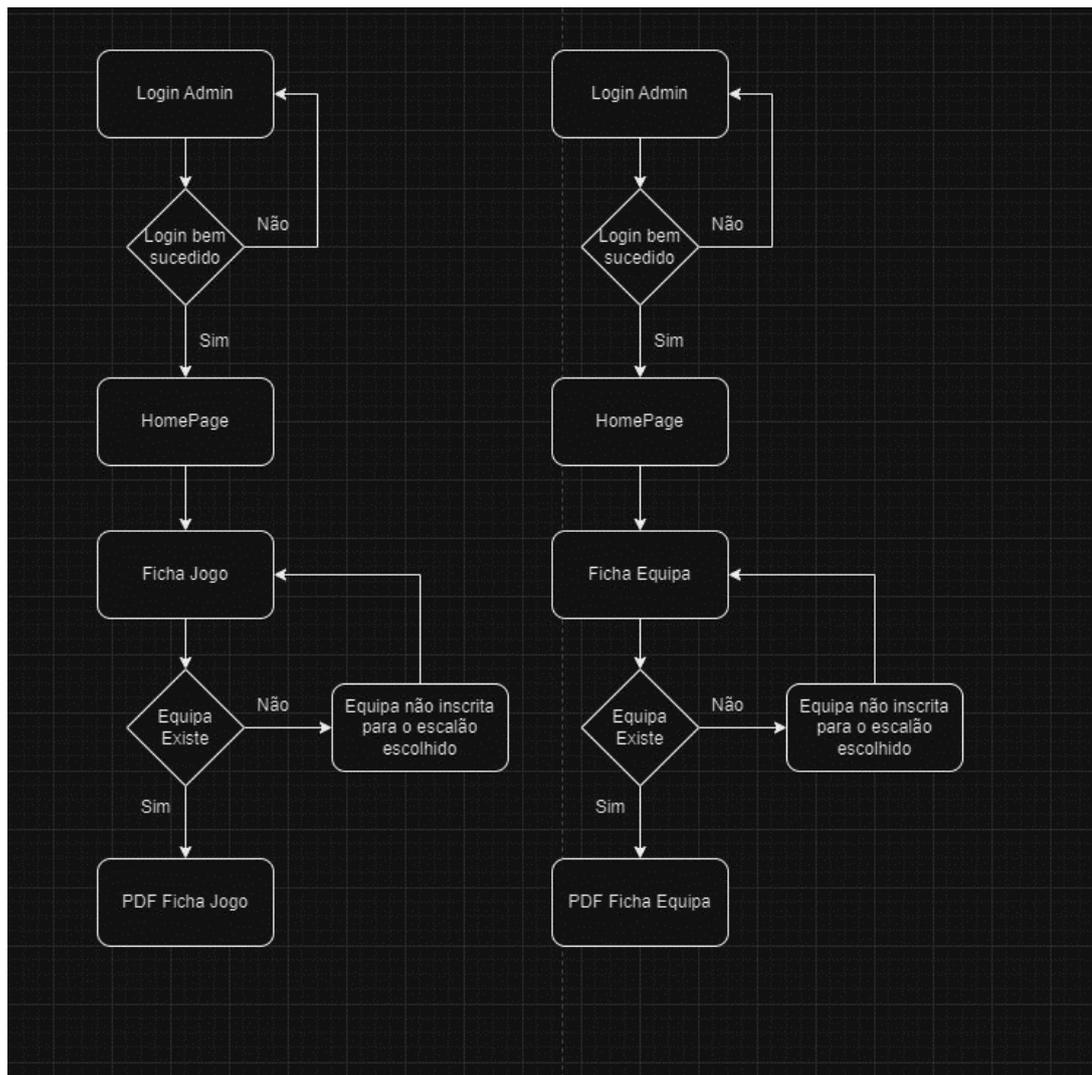
O clube irá inscrever os jogadores e técnicos, logo terá de fazer login na aplicação com os dados correspondentes ao clube e aceder à interface de inscrição.

O Admin. querará atualizar os dados depois dos jogos, logo terá de fazer login na aplicação aceder à página dos torneios escolher o escalão do torneio pretendido e selecionar o jogo que pretende e irá ser redirecionado para uma interface onde poderá atualizar os dados.

Os utilizadores sem estar loged in podem aceder à página inicial, à página dos torneios e à página de documentos.

#### 4.3. Diagramas de Actividades ou equivalente (e.g. BPMN)





**Figuras 1, 2 e 3 – Diagramas de Actividades.**

#### 4.4. Modelos relevantes

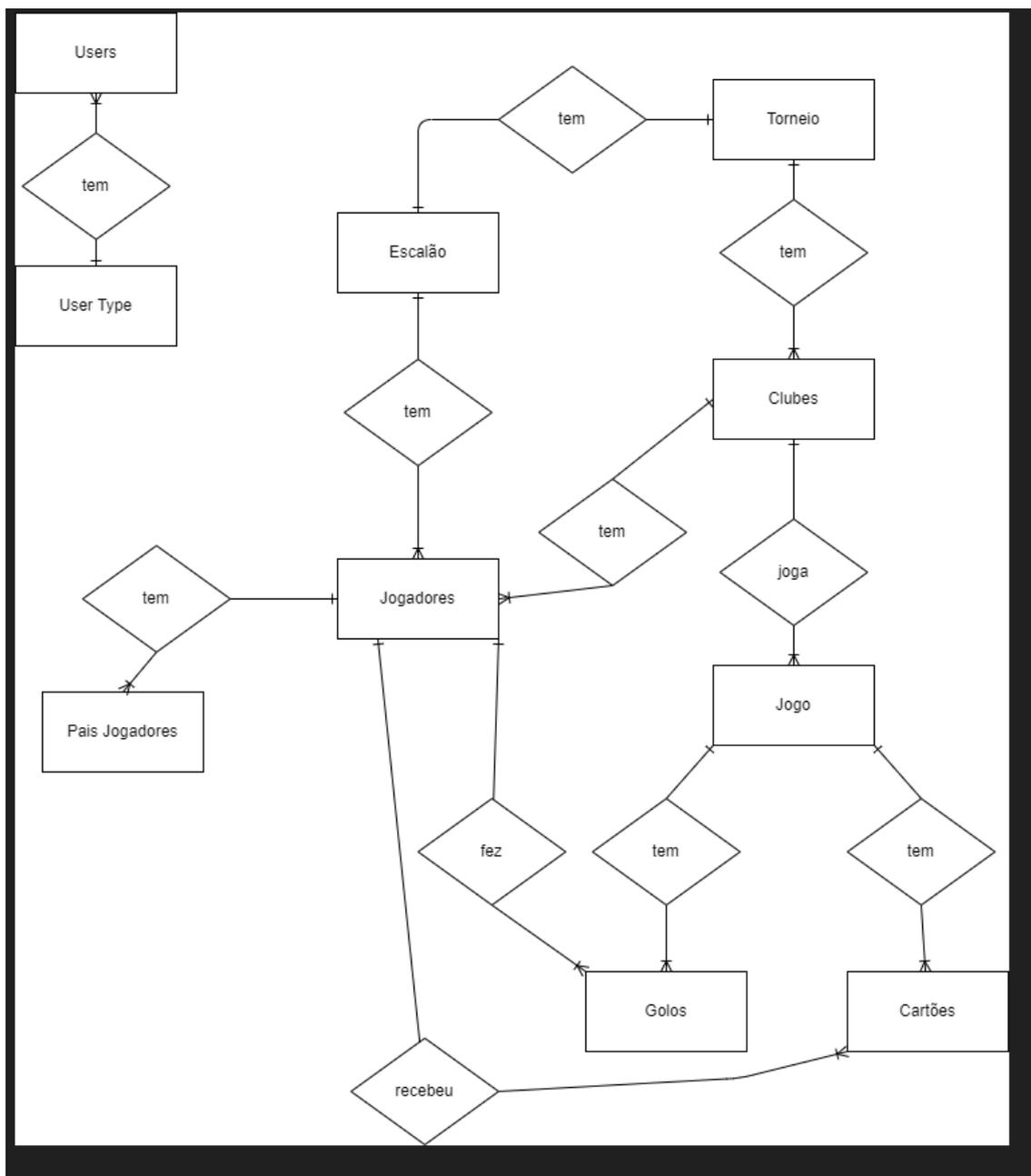


Figura 4 – Diagramas Entidade - Modelo.

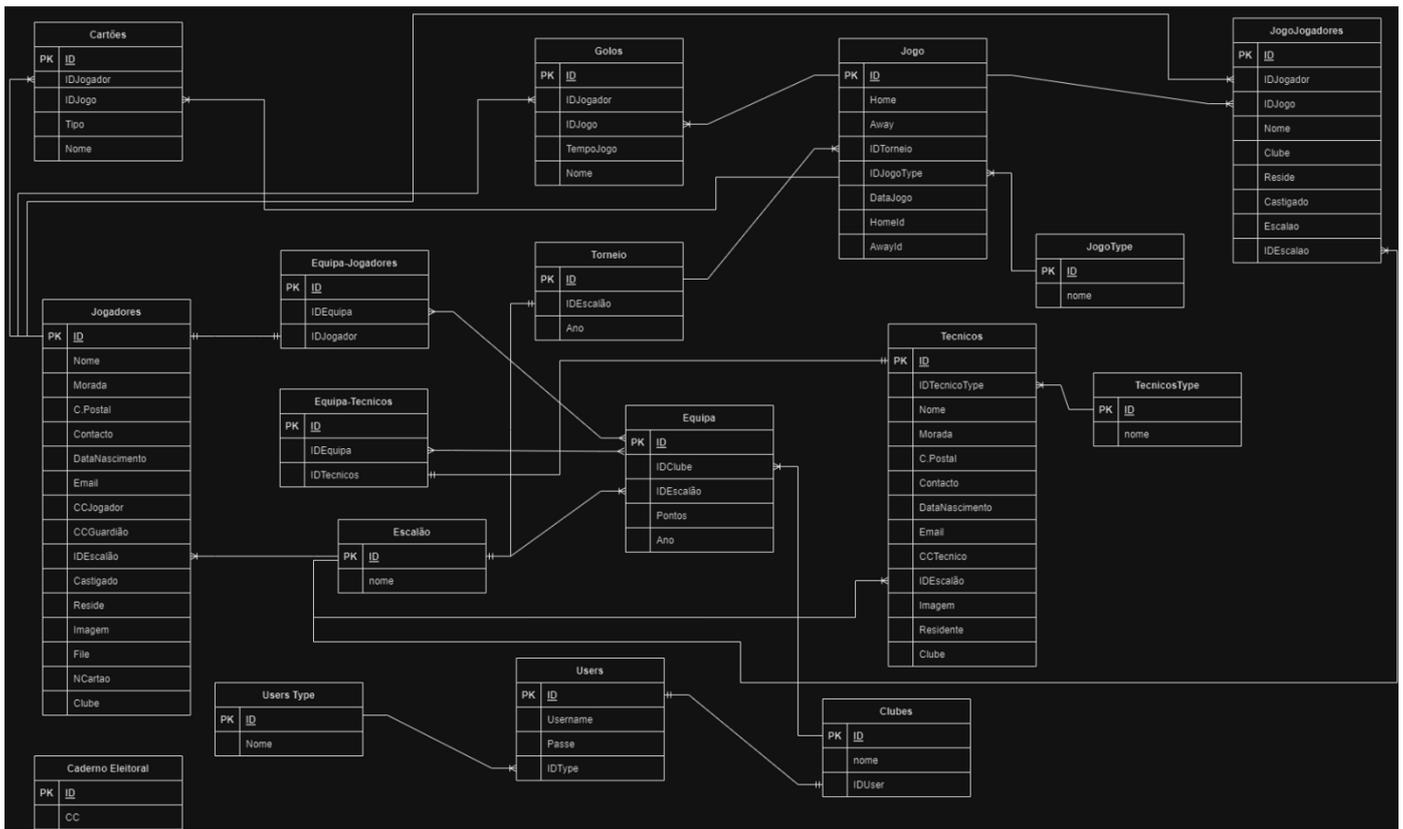


Figura 5 – Diagrama Entidade - Relação.

## 5 Solução Desenvolvida

### 5.1. Introdução

Nos capítulos anteriores já foram identificados os problemas, e como tal, neste capítulo será então apresentada como foi desenvolvida a solução para resolver os problemas em questão.

Links:

- Github – [https://github.com/filipe-21906899/TFC\\_Project](https://github.com/filipe-21906899/TFC_Project)
- Vídeo Demonstrativo:  
[TFC-274-Aplicao-para-a-Gestao-de-uma-Junta-de-Freguesia - YouTube](#)

### 5.2. Arquitectura

A arquitetura do projeto esta dividida em duas partes: temos o lado do Client, que foi desenvolvido usando React e o lado do Server que foi desenvolvido em Express/Node. Isto permite a viabilidade do projeto a longo termo, pois, caso seja desejada uma aplicação móvel no futuro, só será necessário criar o front-end, uma vez que o back-end já está disponível por meio de APIs.

Devido à estrutura do ReactJS cada página/interface corresponde à sua própria componente. Foi criada uma estrutura de pastas permitindo o agrupamento de conteúdo relacionado.

### 5.3. Tecnologias e Ferramentas Utilizadas

**Visual Studio Code** – editor de código simplificado com suporte para operações de desenvolvimento, como debugging, execução de tarefas e controlo de versão.

**NodeJS/ExpressJS (back-end da aplicação)** – utilizado para a programação do lado do servidor e implantado principalmente para servidores sem bloqueio e orientados a eventos, como sites tradicionais e serviços de API de back-end.

**ReactJS (front-end da aplicação)** – é usado para criar interfaces de usuário interativas e aplicativos da Web de maneira rápida e eficiente com muito menos código do que se faria com o JavaScript básico. No React, desenvolvem-se os aplicativos criando componentes reutilizáveis.

**Sass** – é uma extensão de CSS.

**MySQL (base de dados da aplicação)** – é um sistema de gerenciamento de banco de dados e pode ser qualquer coisa, desde uma simples lista de compras até uma galeria de fotos ou até uma vasta quantidade de informação numa rede corporativa.

**SequelizeJS** - Object Relational Mapper baseado em NodeJS que facilita o trabalho com Bases de Dados MySQL entre outras. Um Object Relational Mapper executa funções como manipulação de registos de Bases de Dados, representando os dados como objetos.

**MySQL Workbench** - MySQL Workbench é uma ferramenta de design de banco de dados visual que integra desenvolvimento SQL, administração, design de banco de dados, criação e manutenção em um único ambiente de desenvolvimento integrado para o sistema de banco de dados MySQL.

**Insomnia REST** - Aplicativo desktop de código aberto que facilita a interação e o design, a depuração e o teste de APIs. O Insomnia combina uma interface fácil de usar com funcionalidades avançadas, como auxiliares de autenticação, geração de código e variáveis de ambiente.

#### **Packages instalados:**

**Package server** -

**npm init** - cria no server package.json

**npm install express** - framework para criar o REST API

**npm install cors** - conectar o client e o server

**npm install mysql2**

**npm install nodemon** - reinicia o server sempre que alterações forem guardadas

**npm install sequelize sequelize-cli**

**sequelize init** - irá criar no server os seguintes folders(config, migrations, models, seeders), mas só precisaremos de (models e config)

**npm start** -irá inicializar server e cliente (para inicializar o client e o server precisamos de 2 terminais: primeiro terminal escrever cd server para aceder

ao server e depois o comando para inicial o server e no segundo terminal cd client e depois o comando para inicializar o client)

**npm install bcrypt** - faz com que as passwords fiquem encriptadas

**npm install jsonwebtoken** - gera e valida tokens

**Package client -**

**npm create-react-app .** - irá criar uma aplicação react

**npm install axios** - ajuda a fazer pedidos API mais facilmente

**npm install react-router-dom** - implementa roteamento dinâmico em aplicações web

**npm install formik** - criação mais simplificada de formulários

**npm install yup** - construtor de esquema JavaScript para análise e validação de valor

**npm install xlsx** – usado para que seja possível criar um ficheiro excel

**npm install react-to-print** – permite dar aos users a possibilidade de fazer download de conteúdos

**npm install date-fns** – fornece funções abrangentes para formatação e manipulação de datas

## 5.4. Implementação

**BackEnd:**

Base de Dados :

As tabelas são geradas automaticamente no backend através da criação de models, neste caso foram criados 16, chamados (CadernoEleitoral, Cartoes, Clubes, Equipa, EquipaJogadores, EquipaTecnica, Escalao, Golos, Jogadores, Jogo, JogoType, Tecnicos, TecnicosType, Torneio, Users, UsersType). Será também nestes models que serão especificadas as várias relações entre estas tabelas.

```
JS CadernoEleitoral.js X
server > models > JS CadernoEleitoral.js > <unknown> > exports
1  module.exports = (sequelize, DataTypes) =>{
2
3      const CadernoEleitoral = sequelize.define("CadernoEleitoral",{
4
5          CC :{
6              type: DataTypes.INTEGER,
7              allowNull:false
8          }
9      });
10     return CadernoEleitoral
11 }
```

**Figura 6 – Model da Tabela CadernoEleitoral**

Algumas tabelas estabelecem ligações entre elas, logo para que isso aconteça nos models, é necessário adicionar o seguinte código.

```
JS UsersType.js U X
server > models > JS UsersType.js > <unknown> > exports > UsersType
1  module.exports = (sequelize, DataTypes) =>{
2
3      const UsersType = sequelize.define("UsersType",{
4
5          Nome :{
6              type: DataTypes.STRING,
7              allowNull:false
8          }
9      });
10
11     UsersType.associate = (models) => {
12         UsersType.hasMany(models.Users, {
13             onDelete: "cascade",
14         });
15     };
16     return UsersType
17 }
```

**Figura 7 - Model da Tabela UsersType**

Este pedaço de código indica que o modelo UsersType está associado a outros modelos, especificamente ao modelo Users, e ao usarmos “hasmany”, indicamos que um UserType pode estar relacionado com muitos Users.

Após a criação dos models, foi criado para cada model um respetivo router, e é nesse router que iremos introduzir várias queries que nos permitem aceder à tabela pretendida e adicionar, alterar, eliminar ou obter dados armazenados na base de dados.

```
JS UsersType.js U X
server > routes > JS UsersType.js > router.post("/") callback
1  const express = require('express')
2  const router = express.Router()
3  const { UserType } = require('../models')
4
5  router.get("/", async (req, res) =>{
6    const listOfPost = await UserType.findAll();
7    return res.json(listOfPost);
8  });
9
10 router.post("/", async (req, res) =>{
11   const cc = req.body;
12   await UserType.create(cc);
13   return res.json(cc);
14 })
15
16 module.exports = router
```

**Figura 8 – Router da Tabela UserType**

Temos também um ficheiro chamado index.js onde teremos de adicionar o seguinte código sempre que um novo router for adicionado:

```
const UserTypeRouter = require('./routes/UsersType');
app.use("/users_type", UserTypeRouter);
```

Este código é responsável por configurar uma rota específica para lidar com solicitações relacionadas aos tipos de usuários na nossa aplicação. Quando um cliente faz uma solicitação para o caminho '/users\_type' no nosso servidor, o Express.js redirecionará essa solicitação para as rotas definidas no módulo UserTypeRouter. Isso permite-nos gerenciar as funcionalidades relacionadas aos tipos de usuários de forma separada e organizada, isolando

essa parte da aplicação das outras. Além disso, quando esse router é chamado no frontend, o navegador do cliente, em vez de fazer uma nova solicitação HTTP ao servidor, utilizará o router do lado do cliente para gerenciar a transição de página. Isso significa que o cliente verá uma transição suave entre as páginas da aplicação, com o conteúdo da página sendo renderizado dinamicamente no navegador, sem a necessidade de recarregar a página inteira.

É também através deste router no servidor que acedemos à base de dados quando necessário, por exemplo quando um user solicita informações relacionadas aos tipos de usuários, o router no servidor pode realizar consultas à base de dados, de modo a obter os dados relevantes.

Isso permite que a nossa aplicação forneça informações atualizadas e dinâmicas aos users. Esse procedimento é útil porque permite-nos modularizar o código, manter uma estrutura organizada à medida que adicionamos mais funcionalidades à aplicação e interagir eficientemente com a base de dados sempre que necessário.

Sempre que adicionarmos um novo router, basta seguir esse padrão para garantir que as solicitações sejam tratadas corretamente.

Na implementação do backend um dos routers mais importantes foi o router Users, uma vez que ele será usado para a criação dos dados de login, ou seja, os usernames e passwords. Para manter as Passwords seguras, utilizei a função de hash criptográfico bcrypt, que é amplamente reconhecida pela sua segurança.

```

JS Users.js  X
server > routes > JS Users.js > router.post("/") callback > then() callback
1  const express = require('express');
2  const router = express.Router();
3  const { Users } = require('../models');
4  const bcrypt = require("bcrypt");
5  const {sign} = require('jsonwebtoken');
6
7  router.get("/:usersTypeId", async (req, res) =>{
8      const usersTypeId = req.params.usersTypeId;
9      const users = await Users.findAll({ where: { UsersTypeId: usersTypeId}});
10     return res.json(users);
11 });
12
13 router.post("/", async (req, res) =>{
14     const {Username, Password, UsersTypeId} = req.body;
15     bcrypt.hash(Password, 10).then((hash) => {
16         Users.create({
17             Username: Username,
18             Password: hash,
19             UsersTypeId: UsersTypeId
20         });
21     });
22     return res.json("SUCCESS");
23 });
24

```

Figura 9– Route da Tabela Users

## FrontEnd:

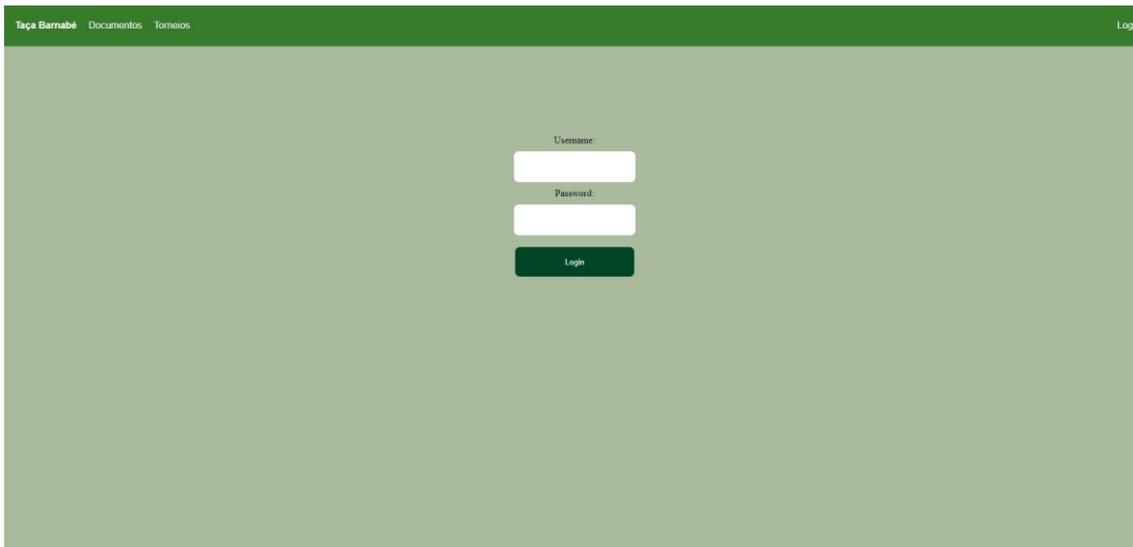
A navegação entre as várias páginas da aplicação é realizado pelo seguinte código.

```

return (
  <div className="App">
    <Router>
      <div className="navbar">
        <div className='Home'>
          <Link to="/">Taça Barnabé</Link>
        </div>
        <Link to="/documentos">Documentos</Link>
        <Link to="/torneio">Torneios</Link>
        {localStorage.getItem('accessToken') && (
          <>
            <Link to="/info">Informação Equipas</Link>
            <Link to="/inscrição">Inscrição Jogadores</Link>
            <Link to="/inscrição_tecnico">Inscrição Equipa Tecnica</Link>
            {isAdmin && <Link to="/ficha_jogo">Ficha de Jogo</Link>}
            {isAdmin && <Link to="/ccdata">CC Dados</Link>}
            {isAdmin && <Link to="/equipa_inscricao">Inscrição Equipas</Link>}
          </>
        )}
      <div className="login-right">
        {!localStorage.getItem('accessToken') && <Link to="/login">Login</Link>}
        <h1>{localStorage.getItem('username')}</h1>
        {localStorage.getItem('accessToken') && <button onClick={logout}> Logout</button>}
      </div>
    </div>
  </div>
)

```

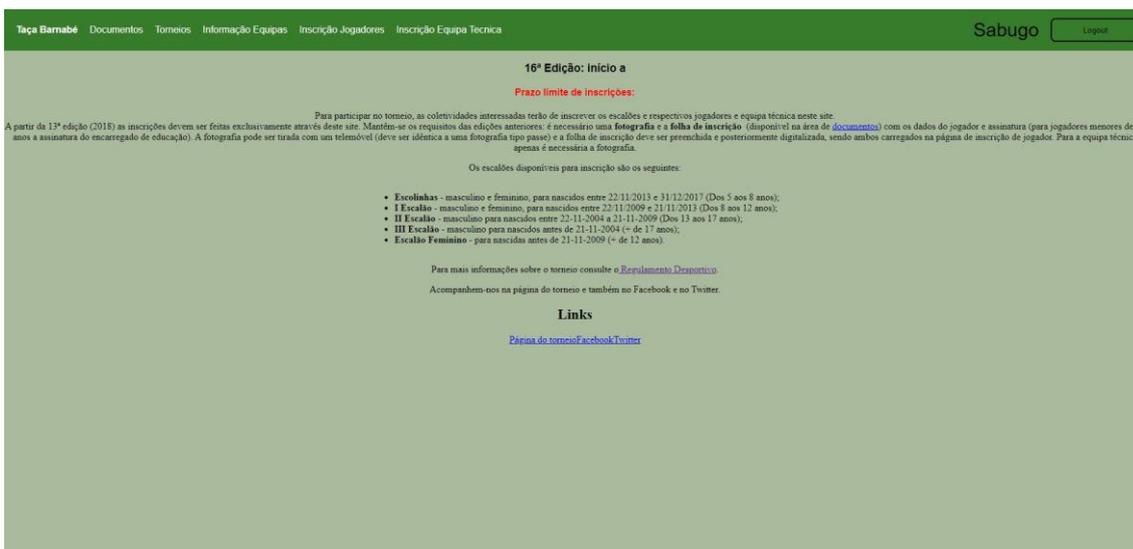




**Figura 12 – Login**

Ao introduzir os dados de login e ao clicar no botão de login, os valores do username e password são enviados para o servidor, e é aí que faremos a verificação. Isto significa que iremos percorrer a tabelas Users na base de dados, e se encontrarmos um utilizador cujo username e password correspondem aos valores recebidos, o login será bem sucedido.

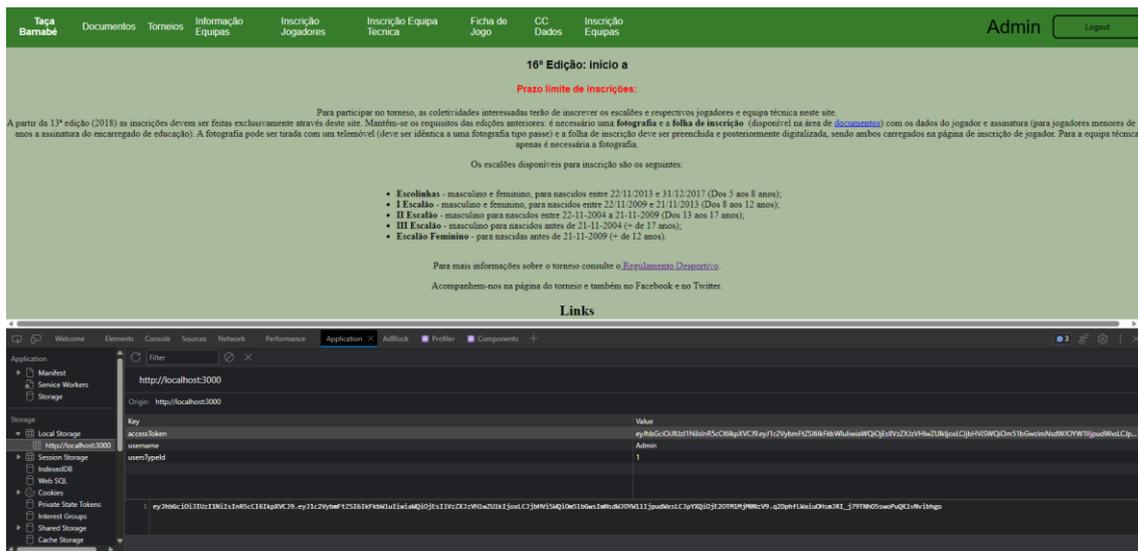
Se o username recebido não existir, a mensagem “Utilizador não existe” será apresentada. No entanto se o username estiver correto mas a password estiver errada será apresentada a mensagem “Palavra-passe errada” Esta é uma prática comum para garantir a segurança do processo de login e para fornecer feedback claro aos utilizadores sobre o motivo pelo qual o login não foi bem-sucedido.



**Figura 13 – Login Clube bem sucedido**

Na figura acima podemos observar que o login foi bem sucedido, o que resultou na apresentação da HomePage. No canto superior direito, podemos ver o nome do usuário que fez login e o botão de logout.

Quando o login é bem-sucedido, o utilizador, seja ele o administrador(Admin) ou um clube, pode armazenar informações relevantes no localStorage'. Isso é uma prática comum para manter informações de sessão ou perfil do utilizador. Na figura abaixo podemos observar como essas informações são armazenadas no 'localStorage'.



**Figura 14 – LocalStorage com informação guardada.**

Em determinadas funcionalidades da aplicação, utilizamos as informações armazenadas no 'localStorage', como por exemplo mostrar o nome do user que fez login.

## Funcionalidades Implementação:

- **Inscrição Jogadores –**

**Figura 15 – Inscrição Jogadores**

Os clubes só podem inscrever jogadores nos escalões em que estiverem inscrito. Portanto, se um clube não estiver inscrito no escalão feminino, essa opção não estará disponível para seleção.

Isso é obtido através do seguinte pedaço de código:

```
useEffect(() => {
  const fetchOptions = async () => {
    try {
      const escalaoResponse = await fetch('http://localhost:3001/escalao');
      const escalaoData = await escalaoResponse.json();

      // Retrieve the 'clubeId' from local storage
      const loggedInClubeId = localStorage.getItem('clubeId');

      // Fetch all the 'equipas' with the 'ClubeId' of the logged-in team
      const equipasResponse = await fetch(
        `http://localhost:3001/equipa?ClubeId=${loggedInClubeId}`
      );
      const equipasData = await equipasResponse.json();

      // Get the current year
      const currentYear = new Date().getFullYear();

      // Extract 'EscalaoId' values from the fetched 'equipas' data
```

```

const escalaoOptionsFiltered = escalaoData.filter((option) =>
  equipasData.some((equipa) => equipa.EscalaoId === option.id &&
equipa.Ano === currentYear)
);

setEscalaoOptions(escalaoOptionsFiltered);

} catch (error) {
  console.error('Error fetching options:', error);
}
};

fetchOptions();
}, []);

```

A inscrição dos jogadores é fácil. Os clubes só precisam de preencher o formulário que se pode ser visto na figura acima. No entanto, dependendo do campo que preencherem, podem existir regras a serem seguidas. Por exemplo, o campo contacto só aceita 9 números, os campos CC e CC Encarregado tem de ter as letras em maiúsculas, e em relação à data de nascimento, os jogadores devem ter uma certa idade, dependendo do escalão, para poderem ser inscrito nele.

A imagem e o ficheiro PDF são automaticamente convertidos para base64 usando o código:

```

const convertToBase64 = (file) => {
  return new Promise((resolve, reject) => {
    const fileReader = new FileReader();
    fileReader.readAsDataURL(file);
    fileReader.onload = () => {
      resolve(fileReader.result);
    };
    fileReader.onerror = (error) => {
      reject(error);
    };
  });
};

const handleProfile = async (e, setFieldValue, name) => {
  const file = e.target.files[0];
  if (file?.size / 1024 / 1024 < 2) {

```

```
const base64 = await convertToBase64(file);
setFieldValue(name, base64);
console.log(base64)
}
else {
  console.log('Image size must be of 2MB or less');
};
}
```

Estes dados serão armazenados na base de dados no formato base64. Após preencher o formulário, ao clicar no botão, o jogador será adicionado à base de dados utilizando o seguinte trecho de código:

```
const response = await fetch('http://localhost:3001/jogadores', {
  method: 'POST',
  headers: {
    'Content-Type': 'application/json',
  },
  body: JSON.stringify(values),
});
```

Após o jogador ser armazenado, iremos procurar na base de dados qual o ID da equipa correspondente a esse jogador, utilizando o seguinte código:

```
const equipaResponse = await
fetch(`http://localhost:3001/equipa/equipaId?EscalaoId=${values.EscalaoId}
&ClubeId=${localStorage.getItem('clubeId')}&Ano=${new
Date().getFullYear()}`);
```

Neste pedido ao servidor, estamos também a enviar o valor do EscalaoId, ClubeId que se encontra guardado no localStorage e o ano atual, de modo a podermos obter o ID da equipa cujo EscalaoId, ClubeId e ano sejam iguais aos valores enviados para o servidor.

Após recebermos o ID da equipa, iremos armazenar esse ID juntamente com o ID do jogador criado na tabela “equipaJogadores”, que é responsável por guardar a associação entre os jogadores e as suas equipas.

- **Inscrição Equipa Técnica –**

**Figura 16 – Inscrição Equipa Técnica**

Assim como para os jogadores, os clube só podem inscrever técnicos nos escalões em que o clube tiver inscrito.

A funcionalidade para inscrever a equipa técnica é semelhante à funcionalidade para inscrever os jogadores, a única diferença esta nas tabelas onde os valores são armazenados e nas informações que serão guardadas. Por exemplo, os técnicos não precisam do ficheiro PDF, ao contrário dos jogadores..

Armazenamos os dados dos técnicos na tabela ‘Tecnicos’ e o tecnicoId e equipaId na tabela ‘EquipaTecnica’.

- **Inscrição Equipas –**

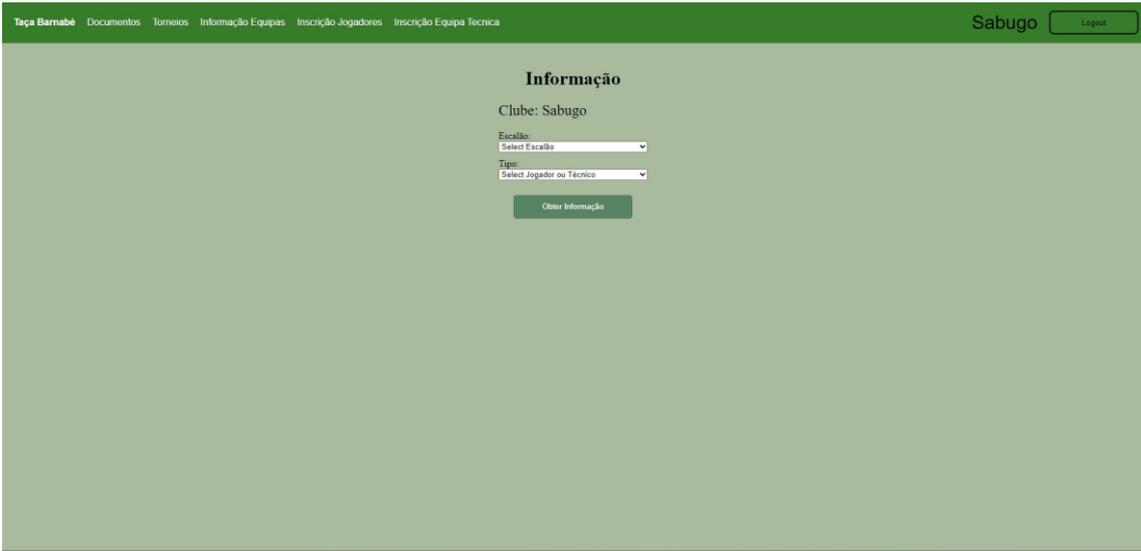
**Figura 17 – Inscrição Equipas**

Esta funcionalidade só esta acessível aos administradores e é através dela que as equipas serão inscritas.

Quando o botão for pressionado, iremos adicionar à tabela ‘Equipa’ o ID do clube e o ID do escalão escolhido, e também será adicionado o ano atual. Isso é feito para garantir que apenas uma inscrição por clube e escalão seja permitida por ano.

No lado do servidor, será realizada uma verificação para determinar se a equipa que queremos inscrever já existe na base de dados. Se não existir, será adicionada à base de dados, mas se já existir uma mensagem de aviso será exibida para informar que o clube já está inscrito no escalão escolhido.

- **Informação Equipas –**



**Figura 18 – Informação Equipas Clubes**

Com esta funcionalidade, será possível observar e, se necessário, eliminar jogadores ou membros da equipa técnica de uma equipa.

Neste caso, quando um clube aceder a esta funcionalidade e selecionar um escalão, só terá acesso aos escalões para os quais está inscrito.

Após selecionar todas as opções do formulário, se o Tipo que foi escolhido é jogador, será apresentada uma tabela com todos os jogadores inscritos na equipa. Se o Tipo for Técnico será nos mostrada uma tabela com a equipa técnica inscritos na equipa. Ao lado da tabela

haverá um formulário que permite ao clube eliminar um jogador ou membro da equipa técnica, por exemplo, se foram inscritos com informações incorretas.

The screenshot shows the 'Informação' page for the Sabugo club. At the top, there is a navigation bar with links for 'Taça Barnabé', 'Documentos', 'Torneios', 'Informação Equipas', 'Inscrição Jogadores', and 'Inscrição Equipa Técnica'. The club name 'Sabugo' and a 'Logout' button are on the right. The main heading is 'Informação' with the club name 'Clube: Sabugo' below it. There are two dropdown menus: 'Escalão' (set to 'III Escalão') and 'Tipo' (set to 'Jogador'). An 'Obter Informação' button is below these. A table lists three players with columns for ID, Nome, Clube, Morada, Código-Postal, Contacto, Data Nascimento, Email, CC Jogador, CC Guardião, Foto, and PDF. To the right is a 'Remover Jogador' form with a dropdown for 'ID Jogador/Técnico' and a 'Resistente' button.

ID	Nome	Clube	Morada	Código-Postal	Contacto	Data Nascimento	Email	CC Jogador	CC Guardião	Foto	PDF
2	Filipe Simões Maria	Sabugo	Rua Mario Pinto	2745-833	999999999	1999-01-16	filipe00727@gmail.com	999999999VE1	999999999VE3	Foto	PDF
7	Fernando Galo	Sabugo	Rua Mario Pinto	2745-833	999999999	1999-01-16	filipe00727@gmail.com	999999999VE2	999999999VT4	Foto	PDF
8	Mario Pinto	Sabugo	Rua Mario Pinto	2745-833	999999999	1999-01-16	filipe00727@gmail.com	999999999VE2	999999999VE2	Foto	PDF

The screenshot shows the 'Informação' page for the Sabugo club, specifically for technical staff. The layout is similar to the previous screenshot, but the 'Tipo' dropdown is set to 'Técnico'. The table lists three technical staff members with columns for ID, Nome, Clube, Morada, Código-Postal, Contacto, Data Nascimento, Email, CC Técnico, Tipo Técnico, and Foto. To the right is a 'Remover Técnico' form with a dropdown for 'ID Jogador/Técnico' and a 'Delete' button.

ID	Nome	Clube	Morada	Código-Postal	Contacto	Data Nascimento	Email	CC Técnico	Tipo Técnico	Foto
9	Filipe Simões Maria	Sabugo	Rua Mario Pinto	2745-833	999999999	1999-01-16	filipe00727@gmail.com	999999999PE2	Treinador	Foto
10	Filipe Simões	Sabugo	Rua Mario Pinto	2745-833	999999999	1999-01-16	filipe00727@gmail.com	999999999PE2	Treinador-Adjunto	Foto
11	tiago reis	Sabugo	Rua Mario Pinto	2745-833	999999999	1999-01-16	filipe00727@gmail.com	999999999PE2	Delegado	Foto

Figura 19 e 20 – Informação Jogadores e Técnicos Clubes

Se o acesso à informação das equipas for realizado pelo administrador(Admin), a funcionalidade será apresentada com um modelo diferente.

The screenshot shows a web application interface with a green header. The header contains a navigation menu with items: 'Taça Barnabé', 'Documentos', 'Torneios', 'Informação Equipas', 'Inscrição Jogadores', 'Inscrição Equipa Técnica', 'Ficha de Jogo', 'CC Dados', 'Inscrição Equipas', 'Admin', and 'Logout'. The main content area is titled 'Informação' and contains three dropdown menus: 'Clube' (with 'Select Clube' as the selected option), 'Escalação' (with 'Select Escalão' as the selected option), and 'Tipo' (with 'Select Jogador ou Técnico' as the selected option). Below these dropdowns is a green button labeled 'Obter Informação'.

**Figura 21 – Informação Equipas Admin**

Conforme ilustrado na figura acima, quando o usuário logged in for o administrador(Admin), será necessário escolher o clube desejado. Se, ao pressionar o botão, for selecionada uma combinação de clube e escalação que não corresponda a uma equipa inscrita, será exibida uma mensagem pop-up informando que o clube não está inscrito nesse escalação.

Se a combinação for válida, o sistema obterá o ID da equipa correspondente aos valores selecionados. Após obter o ID da equipa, o sistema verificará qual o é o Tipo. Se o Tipo for jogador, serão procurados na tabela 'EquipaJogadores' os IDs de todos os jogadores que tenham o valor do campo 'EquipaId' igual ao ID da equipa obtido anteriormente.

Esses IDs serão armazenados em um array, que será percorrido para obter todas as informações necessárias dos jogadores, ou seja, se o array contiver IDs de jogadores, serão obtidos da tabela 'Jogadores' todos os detalhes correspondentes aos IDs armazenados no array.

No caso de o Tipo ser Técnico, o mesmo processo ocorrerá, mas as tabelas utilizadas serão 'EquipaTecnica' e 'Técnicos'. Os detalhes dos técnicos serão obtidos da tabela 'Técnicos'.

**Informação**

Clube: Sabugo  
 Escala: III Escalão  
 Tipo: Jogador

Obter Informação

ID	Nome	Clube	Morada	Código-Postal	Contacto	Data Nascimento	Email	CC Jogador	CC Guardião	Residente	Foto	PDF
2	Filipe Simões Maria	Sabugo	Rua Mario Pinto	2745-833	999999999	1999-01-16	filipe00727@gmail.com	999999999VE1	999999999VE3	Não	Foto	EDE
7	Fernando Galo	Sabugo	Rua Mario Pinto	2745-833	999999999	1999-01-16	filipe00727@gmail.com	999999999VE2	999999999VE4	Sim	Foto	EDE
8	Mario Pinto	Sabugo	Rua Mario Pinto	2745-833	999999999	1999-01-16	filipe00727@gmail.com	999999999VE2	999999999VE2	Sim	Foto	EDE

**Remover Jogador**      **Residência**

ID Jogador Técnico: Select id      ID Jogador Técnico: Select id

Residente      Residente

**Informação**

Clube: Sabugo  
 Escala: III Escalão  
 Tipo: Técnico

Obter Informação

ID	Nome	Clube	Morada	Código-Postal	Contacto	Data Nascimento	Email	CC Técnico	Residente	Tipo Técnico	Foto
9	Filipe Simões Maria	Sabugo	Rua Mario Pinto	2745-833	999999999	1999-01-16	filipe00727@gmail.com	999999999PE2	Sim	Treinador	Foto
10	Filipe Simões	Sabugo	Rua Mario Pinto	2745-833	999999999	1999-01-16	filipe00727@gmail.com	999999999PE2	Sim	Treinador-Adjunto	Foto
11	tiago reis	Sabugo	Rua Mario Pinto	2745-833	999999999	1999-01-16	filipe00727@gmail.com	999999999PE2	Sim	Delegado	Foto

**Remover Técnico**      **Residência**

ID Jogador Técnico: Select id      ID Jogador Técnico: Select id

Delete      Residente

**Figura 22 e 23 – Informação Jogadores e Técnicos Admin**

Como podemos ver nas figuras acima, certos detalhes não são visíveis, a menos que o utilizador seja o administrador (Admin). Um desses detalhes é se o jogador ou o técnico é residente. Além disso, podemos observar que se o jogador ou técnico não for residente, esse campo terá um fundo vermelho.

Outra diferença é que os administradores (Admin) têm acesso a um formulário adicional que lhes permite alterar o estado de residência dos jogadores ou técnicos.

Se um clube ou administrador (Admin) precisar de remover um jogador ou membro da equipa técnica, no formulário 'Remover Jogador' ou 'Remover Técnico', eles simplesmente selecionarão o jogador ou técnico que desejam eliminar, e o valor do ID desse jogador

ou técnico será enviado para o servidor. Para eliminar um jogador ou técnico, foi criada nos router 'Jogadores' e 'Técnicos' uma querie com o método 'delete'. Essa querie irá procurar nas tabelas correspondentes, dependendo se deseja eliminar um jogador ou técnico. Se desejar eliminar um jogador, será pesquisado na tabela 'Jogadores' o jogador cujo ID corresponde ao ID recebido na solicitação e, em seguida, será utilizado o comando 'await jogador.destroy()' para realizar a exclusão do jogador. Para os técnicos, o processo é semelhante, mas com o uso do comando 'await tecnicos.destroy()' para eliminar o técnico.

Quando um administrador (Admin) precisar de alterar o estado de residência de um jogador ou técnico, utilizará o formulário 'Residência'. Neste formulário, será possível escolher o jogador ou técnico cuja residência se pretende alterar. Ao pressionar o botão, será enviado para o servidor o ID do jogador ou técnico selecionado.

Nos routers Jogadores e Tecnicos, foi criado um router.put('/:id') que irá procurar nas tabelas Jogadores ou Tecnicos dependendo do que se deseja alterar. O router vai procurar o ID que corresponde ao ID recebido na solicitação e, quando esse ID é encontrado na tabela, o valor do campo residente será definido como falso o que por sua vez fará com que na tabela a informação de residência seja atualizado para não.

- **Fichas Jogo e Equipa –**

The screenshot displays a web application interface with a green header. The header contains navigation links: 'Taça Barnabé', 'Documentos', 'Torneios', 'Informação Equipas', 'Inscrição Jogadores', 'Inscrição Equipa Técnica', 'Ficha de Jogo', 'CC Dados', 'Inscrição Equipas', 'Admin', and a 'Logout' button. The main content area is divided into two columns. The left column is titled 'Ficha Jogo' and contains three dropdown menus: 'Escalaço' (with 'Select Escalaço' below it), 'Equipa 1' (with 'Select an option' below it), and 'Equipa 2' (with 'Select an option' below it). Below these is a green button labeled 'Obter Ficha Jogo'. The right column is titled 'Ficha Equipa' and contains two dropdown menus: 'Clube' (with 'Select Clube' below it) and 'Escalaço' (with 'Select Escalaço' below it). Below these is a green button labeled 'Obter Ficha Equipa'.

**Figura 24 – Fichas Jogo e Equipa**

É com esta funcionalidade que os administradores irão obter as fichas de jogo e as fichas da equipa.

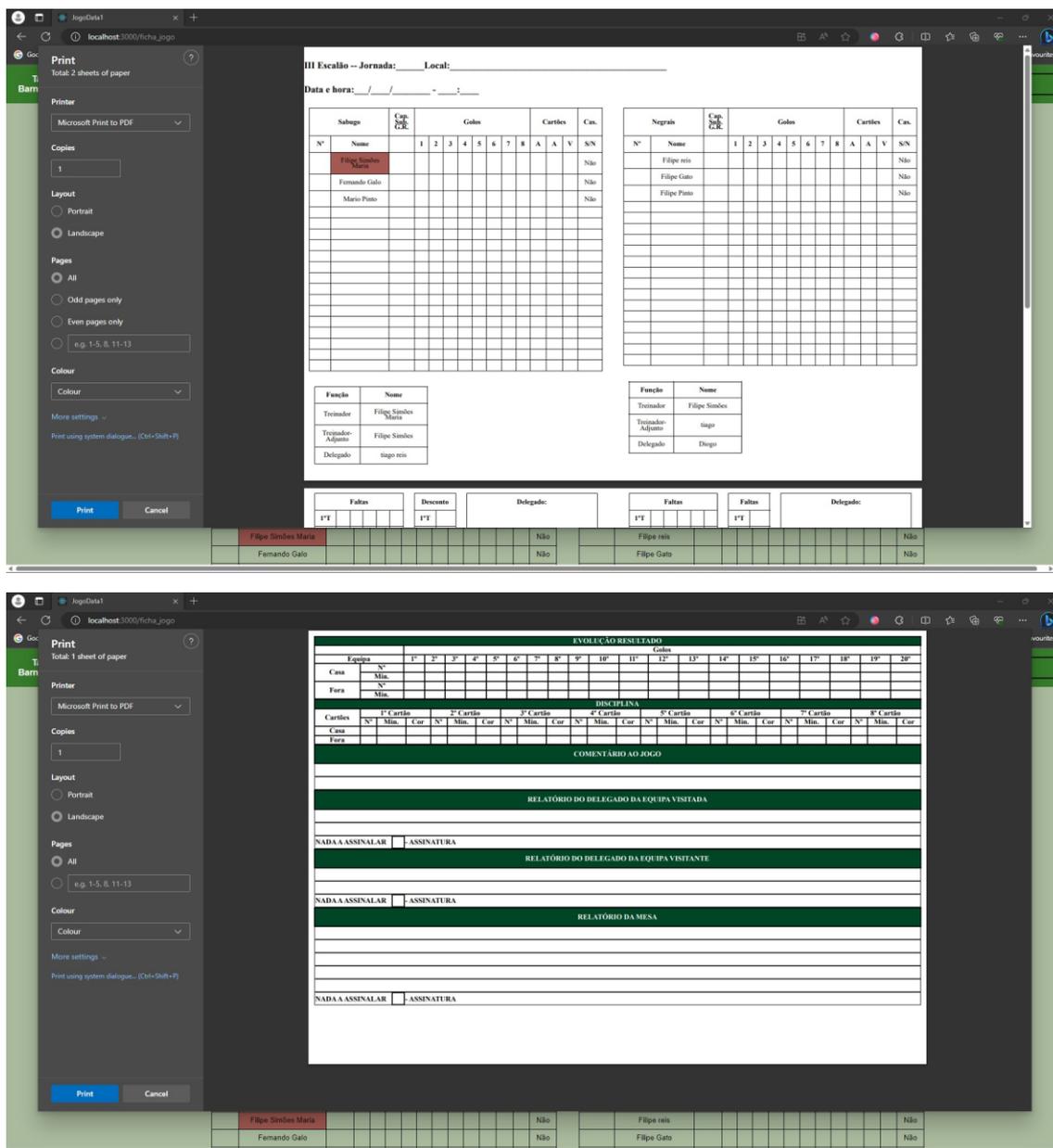
No que diz respeito às fichas de jogo, quando um escalão e as equipas que irão competir uma contra a outra são seleccionados no formulário, a primeira ação é verificar se ambas as equipas seleccionadas estão inscritas nesse escalão. Se uma das equipas não estiver inscrita, será exibida uma mensagem informando qual das equipas não está inscrita no escalão selecionado.

Se ambas as equipas estiverem inscritas no escalão, os dados serão obtidos através das tabelas 'EquipaJogadores' e 'EquipaTecnica' para criar quatro arrays: um com os IDs dos jogadores da 'Equipa1', outro com os IDs dos técnicos da 'Equipa1', um com os IDs dos jogadores da 'Equipa2' e um com os IDs dos técnicos da 'Equipa2'.

Com esses arrays, será possível obter as informações correspondentes aos jogadores e técnicos de ambas as equipas e armazenar os dados dos jogadores e técnicos de cada equipa separadamente. Após obter essa informação, serão criados dois botões que permitirão o download das fichas de jogo.

Para a conversão das tabelas que representam as fichas de jogo em PDF, foi utilizado o 'react-to-print'.

Figura 25 – Ficha Jogo



**Figura 26 e 27 – Ficha Jogo downloads**

As fichas da equipa foram implementadas de forma semelhante às fichas de jogo. Primeiro, verifica-se se a combinação de clube e escalão existe na tabela 'Equipas'. Se não existir, é exibida uma mensagem informando que o clube escolhido não está inscrito nesse escalão. Caso contrário, o sistema obtém o ID da equipa correspondente e, em seguida, recolhe as informações dos jogadores e técnicos que fazem parte dessa equipa.

**Ficha Equipa**

Obter Ficha Equipa

PDF FichaEquipa1 PDF FichaEquipa2

Época: 2023 Equipa: Sabugo Escalação: III Escalão

ID	Nome	Data Nascimento	CC Jogador	Residente
2	Filipe Simões Maria	1999-01-16	99999999	Não
7	Fernando Galo	1999-01-16	99999999	Sim
8	Mário Pinto	1999-01-16	99999999	Sim

Sociedade Recreativa e Desportiva de Sabugo

Escalação: III Escalão Adversário: \_\_\_\_\_

Nome	Jogadores										Cartões						
	Camisola	CAP	SC	GR	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10	A1	A2	V
Filipe Simões Maria																	
Fernando Galo																	
Mário Pinto																	

Figura 28 – Ficha Equipa

Print

total: 1 sheet of paper

Printer: Microsoft Print to PDF

Copies: 1

Layout: Portrait

Pages: All

Colour: Colour

Print

Cancel

Época: 2023 Equipa: Sabugo Escalação: III Escalão

ID	Nome	Data Nascimento	CC Jogador	Residente
2	Filipe Simões Maria	1999-01-16	99999999	Não
7	Fernando Galo	1999-01-16	99999999	Sim
8	Mário Pinto	1999-01-16	99999999	Sim

ID	Nome	Data Nascimento	CC Técnico	Tipo Técnico
9	Filipe Simões Maria	1999-01-16	99999999	Treinador
10	Filipe Simões	1999-01-16	99999999	Técnico-Adjunto
11	Diogo Reis	1999-01-16	99999999	Delegado

Figura 29 – Ficha Equipa 1 PDF

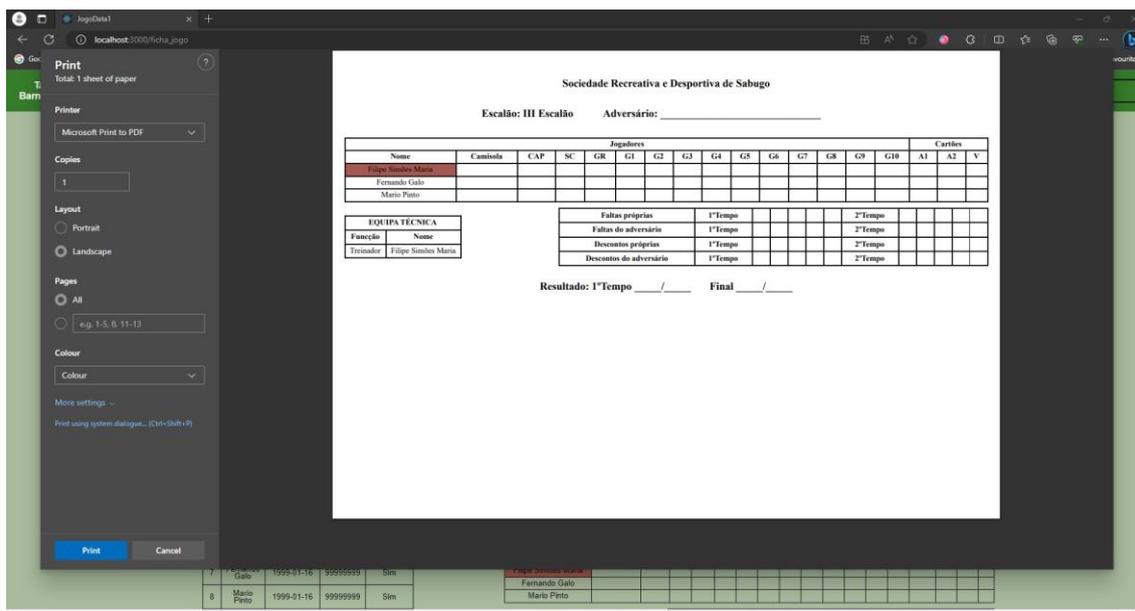


Figura 30 – Ficha Equipa 2 PDF

- Criação Torneio –



Figura 31 – Criação Torneios

Os administradores, nesta funcionalidade, selecionarão o escalão para o torneio que desejam criar. Após selecionar o escalão desejado, ao pressionar o botão, o sistema irá primeiro verificar na tabela 'Torneios' se um torneio com o ano atual e o ID do escalão escolhido já foi criado. Se já existir, será exibida uma mensagem pop-up informando que esse torneio já foi criado. Caso contrário, os valores serão adicionados à tabela, criando assim um novo torneio.

- Criação Jogo –

Back

**Jogos Escalão: III Escalão**

Dia do Jogo:

Tipo de Jogo:

Equipa 1:

Equipa 2:

Submit

1999-01-16	1999-01-16	1999-01-15
Sabugo	Sabugo	Sabugo
Negrais	Negrais	Negrais
GrupoA	GrupoB	GrupoC

**Figura 32 – Criação Jogos**

Nesta funcionalidade, iremos criar um jogo. O procedimento começa com a indicação da data do jogo, seguida pela seleção do tipo de jogo (por exemplo, Grupo A, B, C, D, Semi-Final ou Final) e das duas equipas que irão jogar uma contra a outra.

Após preencher o formulário e pressionar o botão, primeiro será realizada uma verificação para determinar se ambas as equipas escolhidas estão inscritas no escalão em questão. Após essa confirmação, o jogo será adicionado à tabela 'Jogos'. Em seguida, serão obtidos os IDs de ambas as equipas para que possamos armazenar informações dos jogadores de ambas as equipas em uma nova tabela chamada 'JogoJogadores'. Isso é feito para manter os dados originais, ou seja, mesmo que algum jogador seja suspenso após o jogo, as informações sobre a data deste jogo não serão alteradas.



Figura 33 – Novo Jogo Criado

- **Atualização data Jogo –**

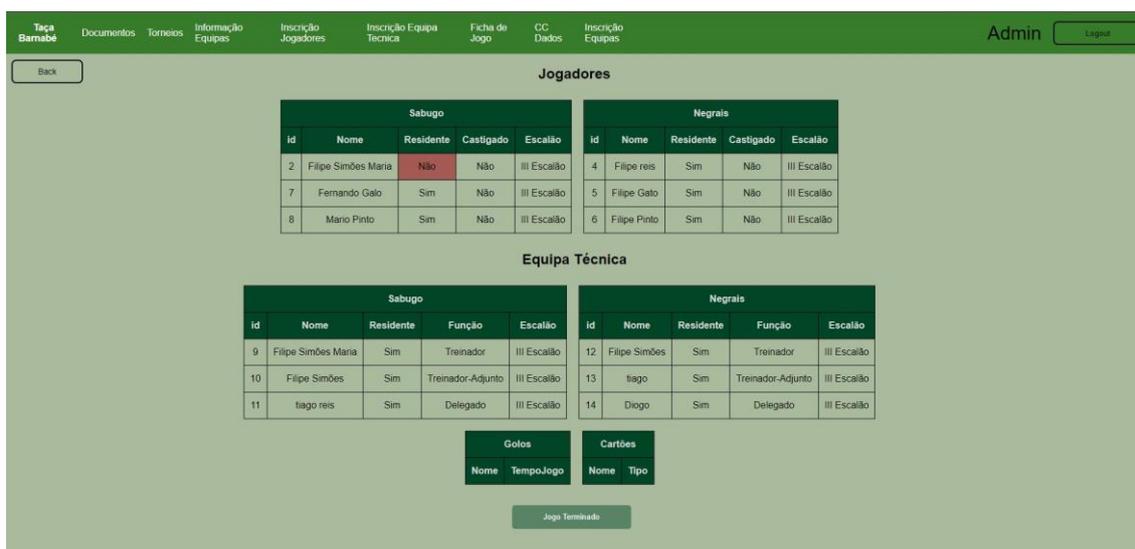


Figura 34 – Informação Jogos

**Equipa Técnica**

Sabugo					Negrais				
Id	Nome	Residente	Função	Escalaço	Id	Nome	Residente	Função	Escalaço
9	Filipe Simões Maria	Sim	Treinador	III Escalaço	12	Filipe Simões	Sim	Treinador	III Escalaço
10	Filipe Simões	Sim	Treinador-Adjunto	III Escalaço	13	tiago	Sim	Treinador-Adjunto	III Escalaço
11	tiago reis	Sim	Delegado	III Escalaço	14	Diogo	Sim	Delegado	III Escalaço

Golos		Cartões	
Nome	TempoJogo	Nome	Tipo
Jogo Terminado			

Golos		Cartões	
Jogador: Select Player	Tempo Jogo: --:--	Jogador: Select Player	Tipo Cartão: Select Tipo Cartão
Introduzir Golos		Introduzir Cartões	

**Figura 35 – Informação Jogos**

As imagens acima representam os dados relacionados aos jogos, e o que ocorre é que quando a página de um jogo é aberta, ela automaticamente vai buscar todas as informações armazenadas na tabela 'JogoJogadores'. Ou seja, enviamos o ID correspondente a este jogo para o servidor e, em seguida, exibimos as informações dos jogadores de ambas as equipas em tabelas separadas, uma para os jogadores da Equipa 1 e outra para os jogadores da Equipa 2. O mesmo é feito para a equipa técnica, conforme pode ser observado nas imagens acima.

Também podemos observar duas tabelas adicionais: a tabela de golos e cartões, além de um botão que indica 'Fim do Jogo'. Nesta funcionalidade, ao pressionar o botão 'Jogo Terminado', se um dos jogadores estiver suspenso, o sistema procurará na tabela 'Jogadores' os jogadores que participaram neste jogo. Se algum jogador estiver com a suspensão (castigo) marcada como 'true', ela será alterada para 'false', e o número de cartões (Ncartao) será definido como 0.

Após pressionar 'Jogo Terminado', os administradores atualizam os golos e cartões. Isso é feito adicionando a informação do jogo no formulário correspondente. Por exemplo, se o jogador Filipe Reis marcou um golo, eles selecionam o jogador Filipe Reis no formulário 'Golos' e indicam o tempo em que ele marcou o golo. Em seguida, na tabela de golos, aparecerá a entrada para Filipe Reis e o tempo em que marcou o golo.

**Jogadores**

Sabugo					Negrais				
id	Nome	Residente	Castigado	Escalão	id	Nome	Residente	Castigado	Escalão
2	Filipe Simões Maria	Não	Não	III Escalão	4	Filipe reis	Sim	Não	III Escalão
7	Fernando Galo	Sim	Não	III Escalão	5	Filipe Gato	Sim	Não	III Escalão
8	Mario Pinto	Sim	Não	III Escalão	6	Filipe Pinto	Sim	Não	III Escalão

**Equipa Técnica**

Sabugo					Negrais				
id	Nome	Residente	Função	Escalão	id	Nome	Residente	Função	Escalão
9	Filipe Simões Maria	Sim	Treinador	III Escalão	12	Filipe Simões	Sim	Treinador	III Escalão
10	Filipe Simões	Sim	Treinador-Adjunto	III Escalão	13	tiago	Sim	Treinador-Adjunto	III Escalão
11	tiago reis	Sim	Delegado	III Escalão	14	Diogo	Sim	Delegado	III Escalão

**Golos**

Nome	TempoJogo
Filipe reis	02:20:00

**Cartões**

Nome	Tipo
Filipe reis	Amarelo
Filipe Pinto	Amarelo
Fernando Galo	Vermelho

Figura 36 – Golos Updated

O mesmo processo será aplicado aos cartões, porém, ao adicionar um cartão, o jogador pode ou não ser suspenso, dependendo do tipo de cartão.

Se o jogador receber um cartão amarelo, o que acontece é que, ao introduzir o cartão, será adicionado 1 à coluna 'Ncartao' na tabela 'Jogadores'. Ou seja, se 'Ncartao' fosse 0, passará a ser 1. Após a adição de 1 ao valor de 'Ncartao', o sistema usará o ID do jogador selecionado e verificará o valor da coluna 'Ncartao'. Se 'castigado' for igual a 'false' na tabela 'Jogadores', o jogador não será suspenso, a menos que o valor de 'Ncartao' seja igual a 2. Nesse caso, o jogador ficará suspenso.

Se o cartão for vermelho, o jogador será automaticamente suspenso.

**Jogadores**

Sabugo					Negrais				
id	Nome	Residente	Castigado	Escalão	id	Nome	Residente	Castigado	Escalão
2	Filipe Simões Maria	Não	Não	III Escalão	4	Filipe reis	Sim	Não	III Escalão
7	Fernando Galo	Sim	Não	III Escalão	5	Filipe Gato	Sim	Não	III Escalão
8	Mario Pinto	Sim	Não	III Escalão	6	Filipe Pinto	Sim	Não	III Escalão

**Equipa Técnica**

Sabugo					Negrais				
id	Nome	Residente	Função	Escalão	id	Nome	Residente	Função	Escalão
9	Filipe Simões Maria	Sim	Treinador	III Escalão	12	Filipe Simões	Sim	Treinador	III Escalão
10	Filipe Simões	Sim	Treinador-Adjunto	III Escalão	13	tiago	Sim	Treinador-Adjunto	III Escalão
11	tiago reis	Sim	Delegado	III Escalão	14	Diogo	Sim	Delegado	III Escalão

**Cartões**

Nome	Tipo
Filipe reis	Amarelo
Filipe Pinto	Amarelo
Fernando Galo	Vermelho

Figura 37 – Cartões Updated

The screenshot displays a web application interface with a green header. The main content area is titled 'Jogadores' and contains two tables: 'Sabugo' and 'Negrais'. Below these is a section for 'Equipa Técnica' with two tables: 'Sabugo' and 'Negrais'. At the bottom, there are buttons for 'Golos' and 'Cartões', and a 'Jogo Terminado' button.

Sabugo					Negrais				
Id	Nome	Residente	Castigado	Escalão	Id	Nome	Residente	Castigado	Escalão
2	Filipe Simões Maria	Não	Não	III Escalão	4	Filipe reis	Sim	Não	III Escalão
7	Fernando Galo	Sim	Sim	III Escalão	5	Filipe Gato	Sim	Não	III Escalão
8	Mario Pinto	Sim	Não	III Escalão	6	Filipe Pinto	Sim	Não	III Escalão

Sabugo					Negrais				
Id	Nome	Residente	Função	Escalão	Id	Nome	Residente	Função	Escalão
9	Filipe Simões Maria	Sim	Treinador	III Escalão	12	Filipe Simões	Sim	Treinador	III Escalão
10	Filipe Simões	Sim	Treinador-Adjunto	III Escalão	13	tiago	Sim	Treinador-Adjunto	III Escalão
11	tiago reis	Sim	Delegado	III Escalão	14	Diogo	Sim	Delegado	III Escalão

**Figura 38 – Novo Jogo Informação**

A imagem acima corresponde a um novo jogo criado. Este jogo foi criado após o jogo mostrado nas imagens anteriores, e podemos observar que o jogador Fernando Galo está suspenso neste jogo devido ao cartão vermelho que recebeu no jogo anterior.

As imagens seguintes representam o que é possível visualizar em relação aos torneios e jogos, seja o utilizador um Clube ou um utilizador que não fez login.

### User Not Logged In e Clubes:

The screenshot displays a web application interface with a green header. The main content area shows two tournament cards for the year 2023. The first card is for 'III Escalão' and the second card is for 'I Escalão'.

**Figura 39 – Torneios**



Figura 40 – Jogos



Figura 41 – Jogo terminado informação

The screenshot shows a web interface for a football tournament. At the top, there is a navigation bar with 'Taça Barnabé', 'Documentos', and 'Torneios'. A 'Login' button is in the top right. Below the navigation bar is a 'Back' button. The main content is divided into three sections: 'Jogadores', 'Equipa Técnica', and 'Golos'/'Cartões'.

**Jogadores**

Sabugo					Negrais				
Id	Nome	Residente	Castigado	Escalação	Id	Nome	Residente	Castigado	Escalação
2	Filipe Simões Maria	Não	Não	III Escalção	4	Filipe reis	Sim	Não	III Escalção
7	Fernando Galo	Sim	Sim	III Escalção	5	Filipe Gato	Sim	Não	III Escalção
8	Mario Pinto	Sim	Não	III Escalção	6	Filipe Pinto	Sim	Não	III Escalção

**Equipa Técnica**

Sabugo					Negrais				
Id	Nome	Residente	Função	Escalação	Id	Nome	Residente	Função	Escalação
9	Filipe Simões Maria	Sim	Treinador	III Escalção	12	Filipe Simões	Sim	Treinador	III Escalção
10	Filipe Simões	Sim	Treinador-Adjunto	III Escalção	13	tiago	Sim	Treinador-Adjunto	III Escalção
11	tiago reis	Sim	Delegado	III Escalção	14	Diogo	Sim	Delegado	III Escalção

**Golos**

Nome	TempoJogo
------	-----------

**Cartões**

Nome	Tipo
------	------

**Figura 42 – Jogo não terminado informação**

- **Obtenção dados CC –**

The screenshot shows a web interface for a football tournament. At the top, there is a navigation bar with 'Taça Barnabé', 'Documentos', 'Torneios', 'Informação Equipas', 'Inscrição Jogadores', 'Inscrição Equipa Técnica', 'Ficha de Jogo', 'CC Dados', and 'Inscrição Equipas'. An 'Admin' button and a 'Logout' button are in the top right. The main content is titled 'CCs Upload' and includes a file selection area with the text 'Select a file: Choose File No file chosen'. Below this is a section titled 'Download CCs para uma folha Excel' with a 'Download' button.

**Figura 43 – CC Dados**

Esta funcionalidade permite obter um ficheiro Excel com todos os números de cartão de cidadão (CC) armazenados nas tabelas 'Jogadores' e 'Técnicos'. Para que esta funcionalidade funcione, primeiro são obtidos todos os valores correspondentes aos CC dos jogadores, técnicos e encarregados de educação que se encontram nas tabelas 'Jogadores' e 'Técnicos'. Esses valores são então armazenados em um array. Com esses valores, um ficheiro Excel é criado e preenchido com as informações dos Cartões de Cidadão, e esse ficheiro é automaticamente downloaded.

O código abaixo demonstra como isso foi possível realizar:

```
const handleDownload = async () => {
  try {
    // Fetch data from the server for Jogadores, Tecnicos, and CCGuardiao
    const jogadoresResponse = await
axios.get('http://localhost:3001/jogadores/cc');
    const tecnicosResponse = await
axios.get('http://localhost:3001/tecnicos/cc');
    const ccGuardiaoResponse = await
axios.get('http://localhost:3001/jogadores/cc/guardiao');

    // Extract data from the responses
    const jogadoresData = jogadoresResponse.data;
    const tecnicosData = tecnicosResponse.data;
    const ccGuardiaoData = ccGuardiaoResponse.data;

    console.log(jogadoresData)
    console.log(tecnicosData)
    console.log(ccGuardiaoData)

    const maxIndex = Math.max(jogadoresData.length,
ccGuardiaoData.length, tecnicosData.length);

    console.log(jogadoresData.length)

    const mappedData = Array.from({ length: maxIndex }, (_, index) => [
      jogadoresData[index] || "",
      ccGuardiaoData[index] || "",
      tecnicosData[index] || ""
    ]);

    console.log(mappedData)

    // Create a new Excel workbook and sheet
    const workbook = XLSX.utils.book_new();
    const sheet = XLSX.utils.aoa_to_sheet([
      ['CCJogador', 'CCEncarregado', 'CCTecnico'],
      ...mappedData
    ]);

    // Add the sheet to the workbook
    XLSX.utils.book_append_sheet(workbook, sheet, 'CC_Data');
```

```
// Set column widths (adjust these values as needed)
const columnWidths = [
  { wch: 16 }, // CCJogador width
  { wch: 16 }, // CCEncarregado width
  { wch: 16 } // CCTecnico width
];
sheet["!cols"] = columnWidths.map(col => ({ width: col.wch }));

// Generate a binary string from the workbook
const excelBuffer = XLSX.write(workbook, { bookType: 'xlsx', type:
'array' });

// Create a Blob from the buffer
const blob = new Blob([excelBuffer], { type:
'application/vnd.openxmlformats-officedocument.spreadsheetml.sheet' });

// Generate a URL for the Blob
const url = URL.createObjectURL(blob);

// Create a temporary anchor element to trigger the download
const anchor = document.createElement('a');
anchor.href = url;
anchor.download = 'CC_Data.xlsx';
anchor.style.display = 'none'; // Hide the anchor element

// Append the anchor to the DOM
document.body.appendChild(anchor);

// Simulate a click on the anchor element to initiate the download
anchor.click();

// Clean up: remove the anchor from the DOM and revoke the Blob
URL
document.body.removeChild(anchor);
URL.revokeObjectURL(url);
} catch (error) {
  console.error('Error downloading data:', error);
}
};
```

Também temos uma funcionalidade chamada "CCs Upload" que está desativada. Esta funcionalidade permite que escolhamos um ficheiro Excel com dados na coluna 1. Em seguida, um botão de upload aparece, que irá ler a coluna 1 do Excel e adicionará os valores à tabela "cadernoEleitoral". Também temos outro botão que serve para remover todos os valores que estavam armazenados na tabela "cadernoEleitoral".

Essa funcionalidade foi implementada para que o aplicativo pudesse automaticamente determinar se um jogador ou técnico era residente quando eram inscritos. No entanto, a pedido da Junta, essa funcionalidade passou a ser manual.

## **5.5. Abrangência**

As disciplinas que foram aplicadas na solução proposta são:

**Base de Dados** – nesta cadeira foi nos ensinado a trabalhar com o MySQL, logo irei usar os conhecimentos adquiridos nesta cadeira de modo a criar a base de dados da aplicação.

**Programação Web** – durante a realização desta disciplina foi realizado um projeto que consistia na realização de uma aplicação web e irei usar estes conhecimentos de modo a criar a aplicação.

**Interação Humano-Máquina** – nesta cadeira aprendemos a criar uma interface tendo em consideração os princípios de utilização e interação entre o Humano e a Máquina.

**Computação Distribuída** – nesta cadeira ensinaram-nos a fazer a distribuição dos meus serviços limitando as conexões entre os componentes isolados, isto leva à nossa arquitetura com uma separação entre cliente e servidor.

## 6 Método e Planeamento

	📌	Name	Duration	Start	Finish	Predecessors
1		📌 Projeto	196 days	27/01/23 08:00	27/10/23 17:00	
2		📌 Planeamento	3 days	18/08/23 08:00	22/08/23 17:00	
3		Elaboração do plano	3 days	18/08/23 08:00	22/08/23 17:00	
4		Aprovação do plano de trabalho	0 days	22/08/23 17:00	22/08/23 17:00	3
5		📌 Início	12 days	23/08/23 08:00	07/09/23 17:00	
6	📌 ❌	Realização do relatório intercalar de 1º semestre	4 days	23/08/23 08:00	28/08/23 17:00	4
7		Reunião com a Junta de Freguesia	1 day	29/08/23 08:00	29/08/23 17:00	6
8		Levantamento da situação existente	7 days	30/08/23 08:00	07/09/23 17:00	7
9		Levantamento de requisitos	3 days	30/08/23 08:00	01/09/23 17:00	7
10		Reunião com a Junta de Freguesia	1 day	04/09/23 08:00	04/09/23 17:00	9
11		Levantamento da situação existente com o professor	1 day	05/09/23 08:00	05/09/23 17:00	10
12		📌 Protótipo	38 days	06/09/23 08:00	27/10/23 17:00	
13		Realização de um protótipo da base de dados	30 days	06/09/23 08:00	17/10/23 17:00	11
14		Apresentação do protótipo ao professor	1 day	18/10/23 08:00	18/10/23 17:00	13
15	❌	Realização relatório intermédio	7 days	19/10/23 08:00	27/10/23 17:00	14
16	📌 ✅	📌 Desenvolvimento	61 days	27/01/23 08:00	21/04/23 17:00	
17	✅	Base de dados	20 days	27/01/23 08:00	23/02/23 17:00	15
18	✅	Testes Base De Dados	15 days	24/02/23 08:00	16/03/23 17:00	17
19	✅	Reunião com a Junta	1 day	17/03/23 08:00	17/03/23 17:00	18
20	✅	Atualização da Base de Dados	1 day	20/03/23 08:00	20/03/23 17:00	19
21	✅	Reunião com a Junta	1 day	21/03/23 08:00	21/03/23 17:00	20
22	✅	Design Básico e Login e Logout	10 days	22/03/23 08:00	04/04/23 17:00	21
23	✅	Testes Login e Logout	6 days	05/04/23 08:00	12/04/23 17:00	22
24	📌 ✅	Realização relatório intercalar de 2º semestre	7 days	13/04/23 08:00	21/04/23 17:00	23
25	✅	📌 Desenvolvimento Final	99 days	24/04/23 08:00	07/09/23 17:00	
26	✅	Inscrição Jogadores e Equipa Técnica	10 days	24/04/23 08:00	05/05/23 17:00	24
27	✅	Testes Inscrição Jogadores e Equipa Técnica	2 days	08/05/23 08:00	09/05/23 17:00	26
28	✅	Reunião com a Junta	1 day	10/05/23 08:00	10/05/23 17:00	27
29	✅	Criação Torneios e Jogos	21 days	11/05/23 08:00	08/06/23 17:00	28
30	✅	Testes Criação Torneios e Jogos	2 days	09/06/23 08:00	12/06/23 17:00	29
31	✅	Reunião com a Junta	1 day	13/06/23 08:00	13/06/23 17:00	30
32	✅	Criação Ficha De jogo	21 days	14/06/23 08:00	12/07/23 17:00	31
33	✅	Testes Criação Ficha de jogo	2 days	13/07/23 08:00	14/07/23 17:00	32
34	✅	Criação funcionalidade com a informação jogadores	21 days	17/07/23 08:00	14/08/23 17:00	33
35	✅	Testes funcionalidade com a informação jogadores	2 days	15/08/23 08:00	16/08/23 17:00	34
36	✅	Apresentação do Projeto a Junta	1 day	17/08/23 08:00	17/08/23 17:00	35
37	📌 ✅	Realização relatório final do TFC	15 days	18/08/23 08:00	07/09/23 17:00	36

**Figura 44 – Plano de Trabalho.**

Este projeto foi realizado e produzido utilizando o plano de trabalho acima mostrado, contudo mesmo seguindo o plano de trabalho existiram atrasos devido a erros e bugs na produção da aplicação, para além disso houveram pequenas alterações quer nas funcionalidades, quer no design da aplicação por parte da Junta de Freguesia.

## **7 Resultados**

Neste capítulo serão apresentados os resultados finais.

Esta aplicação foi desenvolvida de forma responsiva, otimizada e visualmente agradável.

Foi desenvolvida e testada com sucesso, a inscrição das equipas, jogadores e técnicos, criação dos torneios e jogos, atualização dos dados referentes aos jogos, obtenção dos cartões de cidadão, a criação das fichas de jogo e equipa, a obtenção da informação sobre as equipas e a capacidade de alterar valores na base de dados (castigar jogadores, mudar a residência e eliminar jogadores/técnicos). Estes casos foram testados internamente.

Toda a componente foi desenvolvida recorrendo a React e NodeJS/ExpressJS, o que permitiu um desenvolvimento mais rápido e eficaz.

De acordo com o Capítulo 2, podemos constatar o cumprimento de todos os requisitos inicialmente propostos e requisitos adicionais.

A aplicação obteve uma avaliação positiva por parte da Junta de freguesia, garantindo o funcionamento correto e exigido pelos mesmos requisitos.

## **8 Conclusão e trabalhos futuros**

No início do desenvolvimento desta aplicação, não se tem uma ideia clara do que se quer, nem uma noção do tempo e daquilo que se consegue fazer.

A decisão de realizar esta aplicação foi sem dúvida a pensar na minha futura carreira profissional, uma vez que esta é uma das áreas que pretend seguir. Ao realizar este projeto foi-me permitido adquirir imensas competências, conhecimentos e experiência que ainda não tinha em diversas tecnologias.

No decorrer dos semestres, fui confrontado com algumas dificuldades, a maior, foi a criação da base de dados, que foi onde perdi mais tempo para tentar criar uma base de dados que permitisse cumprir todos os requisitos.

Outra dificuldade a referir, foi o tempo perdido a tentar corrigir erros que faziam a aplicação não funcionar e como estava a realizar este trabalho sozinho tornou-se difícil a conciliação e cumprimento dos prazos e metas a atingir.

Graças as dificuldades apresentadas, acabei por adquirir um vasto conhecimento e desenvolvi a capacidade de resolver problemas, o que me ajudou a tornar mais autónomo e independente.

Uma das perspetivas futuras seria continuar a desenvolver a aplicação, isto é criar uma aplicação móvel que a complementa.

Do ponto de vista pessoal, depois de ter tido esta experiência produtiva, e dos conhecimentos adquiridos ao longo da licenciatura, quero aprofundar ainda mais os meus conhecimentos.

## **Bibliografia**

[ULHT21] Universidade Lusófona de Humanidades e Tecnologia,  
[www.ulusofona.pt](http://www.ulusofona.pt), acedido em Out. 2021.

<https://nodejs.org/en/>, acedido a 22/03/2023

<https://expressjs.com/>, acedido a 15/04/2023

<https://reactjs.org/>, acedido a 15/04/2023

<https://www.mysql.com/>, acedido a 25/03/2023

<https://sass-lang.com/>, acedido a 25/03/2023

<https://legacy.reactjs.org/docs/forms.html>, acedido a 15/04/2023

## **Anexo 1 – Planeamento**

**Project**  
*Libre.*

TFC\_GantFinal.pod

## Anexo 2 – Planeamento Anterior

Project  
Libre.

TFC\_Gant.pod

Desde a entrega do relatório anterior até este foram completadas todas as funções em falta.

Houveram alguns imprevistos devidos a erros, o que levou a que o trabalho demorasse mais a ser completado do que era esperado, para além disso também houve algumas alterações no projecto a pedido da Junta o que também levou a que o projecto ficasse um pouco mais atrasado.

## **Anexo 3 – Diagrama Entidade – Relação**



Modelo Entidade-Relação.png

## Anexo 4 – Documentos fornecidos pela Junta



### TAÇA BARNABÉ FICHA DE JOGADOR

**Coletividade:** \_\_\_\_\_

**Nome:** \_\_\_\_\_

**Morada:** \_\_\_\_\_

**C. Postal:** \_\_\_\_\_ - \_\_\_\_\_

**Contacto:** \_\_\_\_\_ **Data de Nascimento:** \_\_\_\_\_ - \_\_\_\_\_ - \_\_\_\_\_

**E-mail:** \_\_\_\_\_

**Nº CC:** \_\_\_\_\_ **Nº CC Enc. Educação:** \_\_\_\_\_

Reside ou é natural da Freguesia de Almargem do Bispo  **Nº Eleitor:** \_\_\_\_\_

Apresentou a declaração médica:

#### Condições inerentes à participação da Taça Barnabé

A Lei de Bases da Actividade Física e do Desporto, aprovada pela Lei nº 5/2007, de 16 de Janeiro, estabelece no nº 2 do artigo 40º que no âmbito das actividades físicas e desportivas desenvolvidas nas instalações desportivas constitui especial obrigação do praticante assegurar-se, previamente, de que não tem quaisquer contraindicações para a sua prática. No seu interesse, assegure-se de que não tem quaisquer contraindicações para a prática das actividades que vai praticar, ou consulte o seu médico em caso de dúvida.

Ao aceitar participar na Taça Barnabé, toma conhecimento das coberturas, capitais, exclusões e franquias que são objecto do Seguro Desportivo efectuado pela colectividade e não poderá exigir qualquer outra contrapartida, a não ser as que estão a coberto do referido seguro. Toma também conhecimento que não poderá exigir qualquer contrapartida financeira à sua colectividade, ou à organização da Taça Barnabé, derivado de algum problema físico ocorrido durante os treinos ou jogos da Taça Barnabé e que possam afectar a sua vida pessoal e profissional.

Os atletas participantes na Taça Barnabé estão sujeitos a análises de controlo anti-doping.

Os atletas autorizam a divulgação do seu nome nos meios de comunicação destinados à Taça Barnabé.

Atendendo ao exposto, a opção pela participação na Taça Barnabé é da sua exclusiva responsabilidade e assume todas as consequências resultantes da prática desta actividade.

**Observações:** \_\_\_\_\_

**Local e Data:** \_\_\_\_\_, \_\_\_\_\_ de \_\_\_\_\_ de \_\_\_\_\_

#### Assinaturas

Do Jogador: \_\_\_\_\_

Do Encarregado de Educação  
(se o jogador(a) for menor de 18 anos): \_\_\_\_\_

O Responsável da Colectividade: \_\_\_\_\_

Figura 45 – Ficha Inscrição Jogador

Tela Início 07/02/22, 11:40

Época: 2022 | Equipa: Negras | Escalão: Escalão

Procurar

### Jogadores

Nome	Data Nascimento	Cartão Cidadão	Estrangeiro	Data inscrição	Última alteração
Afonso Luiz C...	04/12/2008	15863727			
Sebastião Andr...	27/01/2006	30493144			
João Pedro Sá...	20/04/2005	30873075			
David Alexancr...	02/03/2005	51295664			
Duarte Rocha d...	02/04/2005	30436270			
Vicente Rocha ...	25/08/2008	15803555			
André Filipe Po...	27/04/2005	15901069			
Simão Filipe Av...	23/01/2007	15664477			
Martin Malhão ...	19/07/2007	30449021			
João Jorge Rad...	10/01/2007	30715731			
Miguel Marçal ...	16/01/2007	15440821	Sim		
Henrique Lopes...	07/01/2009	30076770			
Pedro Adriano ...	27/11/2008	15861329	Sim		
Santiago André...	28/08/2009	30190122			
Bernardo da S...	29/01/2007	30480393			

Anterior | Página 1 de 1 | 20 linhas | Próxima

Exportar

### Equipa Técnica

Nome	Função	Cartão Cidadão	Data inscrição	Última alteração
Rui Telmo Silva Mar...	Treinador	8553324		
Guilherme de Sous...	Treinador	30239120		
Gustavo Miguel de ...	Treinador	14651048		
Sandra Mata (adm)	Delegado	11896213		

Anterior | Página 1 de 1 | 20 linhas | Próxima

<https://acsbarracoe.herokuapp.com/api/players/search?season=2022&teamid=5&stepid=2> | Página 1 de 1

Figura 46 – Informação Equipa



III Escalão -- Jornada: Local: ALVAREZAS  
 Data e hora: 11/11/2022 - 20h30

Mareira		Camponeses		Cartões	
Nº	Nome	Nº	Nome	1	2
1	Diogo Filipe Duarte		Muno Gomes		
2	Márcio Filipe Duarte		Mauro Sérgio Carvalho		X
3	Rodrigo Marcelino Firmino		Muno Miguel Santos		
4	Ricardo Marcelino Firmino		Ricardo Lopes Gonçalves		
5	Filipe Miguel Jorge		Gonçalo De Piedado		
6	Diogo Miguel Bernardo		Bruno Alexandre Gouveia		
7	Rui Pedro Loures		Tiago Miguel Magalhães		
8	David José Batista		Emanuel Francisco Poixão		
9	Francisco João Mouchreira		Pedro Magalhães		
10	Diogo Carvalho Ferreira		Pedro Miguel Jorge		
11	João Pedro Dias		Pedro Daniel Bernardo		
12	Emanuel Reis Serra		Tiago Alexandre Paulo		
13	Tiago Filipe Francisco		Ruben Miguel Tavares		
14	Diogo Filipe Silva				

Função	Nome	Função	Nome
Delegado	Germanir Manuel Simas	Treineiro	Leonor Augusto
Treineiro	David Emanuel Magina	Sub Delegado	Francisco Aguiar

Função	Nome	Função	Nome
Treineiro	Leonor Augusto	Delegado	Armando Aguiar
Sub Delegado	Francisco Aguiar	Treineiro	Francisco Aguiar

Função	Nome	Função	Nome
Treineiro	Leonor Augusto	Delegado	Armando Aguiar
Sub Delegado	Francisco Aguiar	Treineiro	Francisco Aguiar

Resultado: 1º Tempo 110 Final 211 Mesa: 10260 Escalão: ALVAREZAS MVP: Juágo Francisco

Figura 48 – Ficha Jogo 1



## Anexo 5 – Website Resultado Final

### User not logged in:



HomePage.png



Documents Page.png



Torneios.png



Jogos.png



Dados do Jogo.png

### Clubes:



HomePage.png



Inscrição Jogadores.png



Inscrição Tecnicos.png



Info Equipa.png



Info Jogadores.png



Info Tecnico.png

### Admin:



HomePage.png



Inscrição Equipas.png



Info Equipas.png



Info Jogadores.png



Info Tecnicos.png



Dados CC.png



Ficha Jogo e Equipa.png



FichaJogo.png



FichaJogo PDF1.png



FichaJogo PDF2.png



Ficha Equipa.png



Ficha Equipa PDF1.png



Ficha Equipa PDF2.png



Criação Torneios.png



Criação Jogos.png



Criação Jogos Novo.png



DataJogo.png



DataJogo2.png



GolosAdd.png



NewJogo.png

## **Glossário**

LEI Licenciatura em Engenharia Informática

LIG Licenciatura em Informática de Gestão

TFC Trabalho Final de Curso