

UNIVERSIDADE FEDERAL DE OURO PRETO
INSTITUTO DE CIENCIAS EXATAS E APLICADAS
DEPARTAMENTO DE COMPUTAÇÃO E SISTEMAS

JUNIOR REIS DOS SANTOS

**MÓDULO *WEB* PARA GERENCIAMENTO DE CAMPEONATOS AMADORES DE
FUTEBOL NO MUNICÍPIO DE JOÃO MONLEVADE E REGIÃO**

João Monlevade

2023

JUNIOR REIS DOS SANTOS

MÓDULO *WEB* PARA GERENCIAMENTO DE CAMPEONATOS AMADORES DE FUTEBOL NO MUNICÍPIO DE JOÃO MONLEVADA E REGIÃO

Monografia apresentada ao curso Sistemas de Informação do Instituto de Ciências Exatas e Aplicadas, da Universidade Federal de Ouro Preto, como requisito parcial para aprovação na Disciplina “Trabalho de Conclusão de Curso II”.

Orientador: Theo Silva Lins

João Monlevade

2023



FOLHA DE APROVAÇÃO

Júnior Reis dos Santos

Módulo Web para Gerenciamento de Campeonatos Amadores de Futebol no Município de João Monlevade e Região

Monografia apresentada ao Curso de Sistemas de Informação da Universidade Federal de Ouro Preto como requisito parcial para obtenção do título de Bacharelado em Sistemas de Informação

Aprovada em 29 de março de 2023

Membros da banca

Doutor - Theo Silva Lins - Orientador(a) Universidade Federal de Ouro Preto
Doutor - Fernando Bernardes de Oliveira - Universidade Federal de Ouro Preto
Mestre - Tiago França Melo de Lima - Universidade Federal de Ouro Preto

Theo Silva Lins, orientador do trabalho, aprovou a versão final e autorizou seu depósito na Biblioteca Digital de Trabalhos de Conclusão de Curso da UFOP em 11/04/2023



Documento assinado eletronicamente por **Theo Silva Lins, PROFESSOR DE MAGISTERIO SUPERIOR**, em 11/04/2023, às 15:39, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



A autenticidade deste documento pode ser conferida no site http://sei.ufop.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **0507037** e o código CRC **4B575265**.

AGRADECIMENTOS

Primeiramente, gostaria de agradecer a Deus por me dar forças e coragem para concluir este trabalho de conclusão de curso.

Agradeço à minha família pelo amor, apoio e incentivo ao longo de toda a minha jornada acadêmica. Agradeço aos meus pais, que sempre me encorajaram a estudar e me deram todo o suporte necessário para que eu pudesse chegar até aqui. Agradeço também aos meus irmãos, que sempre me apoiaram e me motivaram a nunca desistir.

Aos meus amigos e colegas de curso, em especial Jean, Rosabel, Verônica, Luana, Christian e Wandrovski, agradeço pela amizade, companheirismo, pelas horas de estudo compartilhadas, apoio e suporte.

Agradeço aos professores e orientadores, que dedicaram seu tempo e conhecimento para nos ensinar e nos guiar nessa jornada. Sem a orientação e o conhecimento deles, este trabalho não teria sido possível.

Por fim, agradeço a todos aqueles que, de alguma forma, contribuíram para a realização deste trabalho e para a minha formação acadêmica. Espero poder retribuir todo o apoio e suporte recebidos ao longo do caminho.

“It's just a spark, but it's enough to keep me
going.”

(Paramore)

RESUMO

A realização de um campeonato de futebol nem sempre é uma tarefa simples e exige muita organização de planejamento. Muitas vezes os simpatizantes do futebol amador que desejam realizar seus próprios campeonatos podem perder tempo em planilhas eletrônicas, folhas de papel e em diversos sistemas disponíveis na *web* que não atende à todas as suas necessidades.

Considerando tal problema, o presente trabalho, propõe o desenvolvimento de um módulo *web* para a criação de gerenciamentos de campeonatos de futebol amador. O sistema busca otimizar o processo de organização e administração de campeonatos através de ferramentas para controle e inserção de dados que simulam a estrutura de um campeonato de futebol, gerar súmulas para serem preenchidas com os dados das partidas e permitir seu *upload* no sistema. Além disso, o sistema permite que os jogadores e demais torcedores possam acompanhar resultados, tabelas e demais informações dos campeonatos disputados.

Palavras- chave: sistema *web*, gerenciador de campeonatos, campeonatos de futebol, futebol amador, engenharia de software.

ABSTRACT

Holding of a football championship is not always a simple task and requires a lot of organization and planning. Amateur football supporters who wish to hold their own championships can often waste time on electronic spreadsheets, sheets of paper and on various systems available on the web that do not meet all their needs.

Considering this problem, the present work proposes the development of a web module for the creation of management of amateur soccer championships. The system seeks to optimize the process of organizing and administering championships through tools for controlling and inserting data that simulate the structure of a football championship, generating scoresheets to be filled in with data from the matches and allowing them to be uploaded to the system. In addition, the system allows players and other fans to follow results, tables and other information about the disputed championships.

Keywords: web system, championship manager, soccer championships, amateur soccer, software engineering.

Lista de Ilustrações

Figura 1 – Relação entre as camadas <i>Model</i> , <i>View</i> e <i>Controller</i>	18
Figura 2 – Diagrama de Caso de Uso.....	28
Figura 3 – Mapa do Site – Administrador Campeonato	29
Figura 4 – Mapa do Site – Super Administrador	29
Figura 5 – Mapa do Site – Usuário Comum	30
Figura 6 – Modelo Entidade Relacionamento para a ACL.....	31
Figura 7 – Modelo Entidade Relacionamento.....	32
Figura 8 – Tela Inicial Campeonatos.....	35
Figura 9 – Tela Seleciona Jogadores Participantes	36
Figura 10 – Tela Informações Campeonatos.....	37
Figura 11 – Tela Partidas.....	37
Figura 12 – Tela Gerencia Papeis	38
Figura 13 – Tela Inicial Campeonatos.....	43
Figura 14 – Tela Inicial Campeonatos vista em um smartphone.....	43
Figura 15 – Tela que lista os Campeonatos – visão do administrador de campeonatos	44
Figura 16 – Tela que lista os Campeonatos – visão de usuário comum.....	44
Figura 17 – Tela que lista os Campeonatos – visão de usuário comum em um smartphone.....	45
Figura 18 – Tela que mostra as Informações do campeonato.....	45
Figura 19 – Tabela de classificação de um campeonato.....	46
Figura 20 – Tabela de classificação de um campeonato vista em um smartphone	46
Figura 21 – Tabela de classificação de um campeonato com grupos.....	47
Figura 22 – Tabela de classificação de um campeonato com grupos vista em um smartphone.....	47
Figura 23 – Tela que lista as partidas do campeonato – visão do administrador de campeonato.....	48
Figura 24 – Tela que lista as partidas do campeonato – visão do usuário comum.....	48
Figura 25 – Tela que lista as partidas do campeonato vista em um smartphone.....	49
Figura 26 – Tela que confirma os jogadores que irão disputar o campeonato pelo time	49
Figura 27 – Tela que confirma os jogadores que irão disputar o campeonato pelo time vista em um smartphone	50
Figura 28 – Tela que lista os usuários e seus papéis.....	50
Figura 29 – Súmula gerada pelo sistema.....	51

Lista de Quadros

Quadro 1 – Comparativos entre as funcionalidades.....	22
Quadro 2 – Requisitos do Sistema.....	24
Quadro 3 – Erros reportados nos testes.....	39

LISTA DE ABREVIATURAS

CRUD – *Create, Read, Update e Delete*

CSS – *Cascading Style Sheets*

HTML – *HyperText Markup Language*

HTTP – *Hypertext Transfer Protocol*

JSON – *JavaScript Object Notation*

LMF – *Liga Monlevadense de Futebol*

MVC – *Model, View, Controller*

PDF – *Portable Document Format*

PHP – *Hypertext Preprocessor*

SGBD – *Sistema de Gerenciamento de Banco de Dados*

SQL – *Structured Query Language*

PWA – *Progressive Web Application*

URL – *Uniform Resource Locator*

Sumário

1 INTRODUÇÃO	13
1.1 Problema	14
1.1.1 Solução Proposta	14
1.2 Objetivos	14
1.2.1 Objetivos Específicos	14
1.3 Metodologia	15
1.4 Organização do Trabalho	16
2. REFERENCIAL TEÓRICO	17
2.1 Frameworks	17
2.1.1 Laravel	17
2.1.2 O Padrão MVC	17
2.2 Bootstrap	18
2.3 Banco de Dados	19
2.3.1 MariaDB	19
2.4 Servidores Web	20
2.4.1 Apache	20
2.5 Hospedagem e versionamento do código-fonte	20
2.6 Trabalhos Relacionados	21
2.6.1 Copa Fácil	21
2.6.2 Challenge.place	21
2.6.3 Comparações com a aplicação proposta	21
3 DESENVOLVIMENTO	23
3.1 Ambiente de Desenvolvimento	23
3.2 Stakeholders e Usuários do Sistema	23
3.2.1 Usuários do Sistema	23
3.3 Funcionalidades do Sistema	24
3.4 Casos de Uso	27
3.5 Mapa do Site	27
3.5.1 Administrador de Campeonato	28
3.5.2 Super Administrador	29
3.5.3 Usuário Comum	30
3.6 Banco de Dados	30
3.7 Desenvolvimento do Sistema	32

3.7.1 Autenticação e Controle Acesso.....	32
3.7.2 Geração da Súmula em PDF.....	33
3.7.3 PWA Laravel	33
3.7.4 Responsividade	34
4.1 Telas do Sistema	35
4.1.1 Campeonatos	35
4.1.2 Seleciona Jogadores Participantes.....	35
4.1.3 Informações de um Campeonato.....	36
4.1.4 Partidas	37
4.1.5 Gerenciamentos dos Papeis dos Usuários	38
4.2 Testes	39
5 CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS.....	40
REFERÊNCIAS	42
APÊNDICE A – Telas do Sistema.....	43
APÊNDICE B – Súmula Gerada pelo Sistema.....	51

1 INTRODUÇÃO

O futebol é o esporte coletivo mais praticado no mundo e, no Brasil essa característica se torna ainda mais significativa, haja vista algumas pesquisas afirmaram que quase metade da população brasileira acima de 16 anos envolve-se com o futebol de alguma forma (JÚNIOR et al., 2017).

A prática do futebol pode-se dar tanto de forma profissional quanto amadora. Sobre as principais diferenças entre as duas práticas, Silva (2011, apud. JÚNIOR et al., 2017) diz que o profissional tem como principal foco a perspectiva do trabalho, a busca de resultados e da vitória, além de se tornar um meio de sobrevivência. Já para GOERG (2010, apud. JÚNIOR et al., 2017) o futebol amador é uma opção de lazer para os indivíduos que têm uma carga de trabalho durante a semana, e que buscam relaxar no final de semana, através desta prática amadorística.

Sobre a prática amadora de futebol Pimenta (2009) relata:

“Futebol amador é um termo bastante utilizado nacionalmente, tanto no meio urbano, quanto no meio rural. Embora seja uma prática esportiva amadora, procura manter uma estrutura espelhada no futebol profissional. Os times, em geral, contam com uma diretoria, presidência, diretoria técnica – alguns deles, inclusive, com registro em cartório; muitos possuem sede, mesmo que esta seja na casa do presidente; os diretores procuram os melhores jogadores, alguns em bairros distantes, e estes, em geral, recebem dinheiro para atuar – para os dirigentes é importante montar um time competitivo, contando para isso com a contribuição financeira de sócios e doações de torcedores e comerciantes do bairro. Alguns times possuem torcida organizada, com charangas, gritos de guerra, hinos e uniformes padronizados; disputam torneios e campeonatos organizados por ligas amadoras e pelo poder executivo - algumas destas competições são regidas pelas regras do Football Association as mesmas do futebol profissional”. (PIMENTA, 2009, p. 16)

No município de João Monlevade, a Liga Monlevadense de Futebol (LMF) é uma instituição que organiza campeonatos amadores no município, além de promover a cultura esportiva, da atividade física e do lazer, proporcionando a prática do futebol de modo não profissional.

Para a realização de campeonatos amadores de futebol, assim como qualquer outro evento esportivo, é preciso de organização e planejamento para que sua realização ocorra da melhor forma possível, sem que ocorra erros que possam comprometer seu resultado.

Diante do contexto apresentado, nota-se a necessidade de se desenvolver um módulo para o gerenciamento de campeonatos realizados pela LMF, que além de

proporcionar aos organizadores maior facilidade para o gerenciamento dos campeonatos facilitará o compartilhamento dos resultados com os atletas e torcedores.

1.1 Problema

A LMF é uma instituição que realiza campeonatos amadores de futebol na cidade de João Monlevade e municípios vizinhos. Para tal, a LMF se vale de diversos recursos para estruturar e organizar os campeonatos, como o uso de planilhas, diversos *softwares* de terceiros e papel e caneta. A grande diversidade de ferramentas traz a dificuldade em gerenciar os campeonatos, uma vez que as informações estão dispersas em diversos locais.

Este módulo irá compor um sistema *web* que além do gerenciamento de campeonatos também permitirá o gerenciamento dos próprios times.

1.1.1 Solução Proposta

Para ajudar no gerenciamento dos campeonatos foi proposto o desenvolvimento de um módulo para o controle dos campeonatos realizados pela LMF onde será possível criar campeonatos e as partidas que o compõem, bem como ferramentas de visualização da classificação dos times nos campeonatos que poderá ser acessada pelos torcedores.

1.2 Objetivos

Este trabalho tem como principal objetivo o desenvolvimento de um módulo *web* para o gerenciamento de campeonatos realizados pela LMF.

1.2.1 Objetivos Específicos

Os objetivos específicos do trabalho são:

- Modelar e implementar a aplicação *web* para gerenciamento dos campeonatos amadores de futebol;
- Validar a aplicação por meio de testes com os usuários do software;

- Hospedar a aplicação em um servidor para a utilização do software para a sua utilização.

1.3 Metodologia

Num primeiro momento realizou-se a revisão da literatura, onde buscou-se estudar e identificar os processos relacionados a criação de campeonatos amadores de futebol, estudar trabalhos correlatos e identificar pontos de melhoria para o sistema proposto. Nesta etapa também foi realizada a identificação das tecnologias que seriam utilizadas para a construção do sistema. Para tal, foi feita uma busca *on-line* sobre as linguagens de programação, *Frameworks*, e demais ferramentas disponíveis no mercado atualmente.

O procedimento adotado para identificar trabalhos correlatos foi procurar pelos sistemas de gerenciamento de campeonatos de futebol outrora já usados pela LMF e fazer uma análise de seu uso em comparação com a aplicação proposta neste trabalho.

A próxima etapa foi responsável pelo o levantamento de requisitos e documentação do modelo proposto. Para a identificação dos requisitos foram realizadas conversas no modelo de entrevistas não estruturada com membros da LMF. Este mesmo modelo de entrevista foi utilizado no decorrer do desenvolvimento para que fosse tirada dúvidas em relação ao desenvolvimento do projeto.

Uma vez de posse dos requisitos necessários foram desenvolvidos os diagramas de caso de uso e esboço do mapa do sistema.

A próxima etapa foi a implementação do *software*. Para o desenvolvimento foi escolhida uma abordagem de divisão em módulos para que sua manutenção e reutilização futura seja viável. Nesta etapa também foi pensada em que ordem se daria o desenvolvimento do sistema a se começar pelo CRUD dos campeonatos e finalizando a visualização dos resultados das partidas.

A etapa seguinte foi a realização de testes. Para os testes foram realizados com usuários com o objetivo de avaliar a performance da aplicação e a experiência dos usuários.

Por último, foi analisado e discutido os resultados obtidos através dos testes, além de identificar possíveis melhorias, correções e considerações gerais sobre a aplicação.

1.4 Organização do Trabalho

O presente trabalho é organizado em quatro capítulos, sendo o primeiro responsável por realizar a introdução do tema do projeto desenvolvido. O capítulo 2 discorre sobre a fundamentação teórica e trabalhos correlatos na mesma área. Em seguida, o capítulo 3, detalha a metodologia usada no desenvolvimento do projeto. Por fim, no capítulo 4 são apresentadas as considerações finais.

2. REFERENCIAL TEÓRICO

Este capítulo apresenta toda a fundamentação teórica do projeto. É discutido acerca das ferramentas e tecnologias utilizadas para o desenvolvimento do projeto, bem como, sobre sistemas correlatos.

2.1 Frameworks

Um dos principais objetivos da Engenharia de Software é o reuso. Através da reutilização de software obtém-se o aumento da qualidade e redução do esforço de desenvolvimento (Gimenes & Huzita, 2005, apud JÚNIOR, 2006). Um *Framework* é capaz de definir uma arquitetura e oferecer construtores básicos para a criação de sistemas. Para Buschmann et al. (1996), Pree (1995) e Pinto (2000) um *Framework* é definido como um software parcialmente completo projetado para ser instanciado (apud JÚNIOR, 2006).

2.1.1 Laravel

O *Framework* utilizado no desenvolvimento deste trabalho é o Laravel, cuja primeira versão foi lançada em 2011. Nos últimos anos, o *Framework* se desenvolveu rapidamente e é bastante popular entre os desenvolvedores PHP. Como a maioria dos Frameworks de aplicações *web*, o Laravel segue o padrão *Model-View-Controller*. Um dos grandes pontos fortes do Laravel é a criação de *migrations*, que são basicamente o controle de versão do banco de dados da sua aplicação. Dessa forma, torna-se muito simples modificar a estrutura do banco de dados e compartilhá-la entre os desenvolvedores.

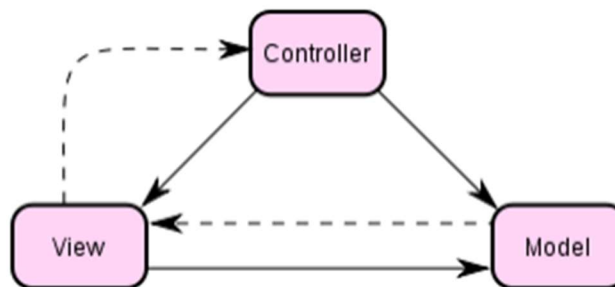
2.1.2 O Padrão MVC

O padrão MVC foi proposto em 1980 como uma abordagem para design de interfaces gráficas de usuários (GUI), permitindo múltiplas representações de um objeto e que interações sejam feitas com estas representações (SOMMERVILLE, 2011 apud ASSUNÇÃO, 2017).

A organização em camadas é a chave para a independência entre os componentes e esta independência é que vai atingir os objetivos de eficiência, escalabilidade, reutilização e facilidade de manutenção padrão de projeto *Model-View-Controller* (MVC) (LUCIANO e ALVES, 2011). Por esse motivo o padrão MVC é bastante utilizado para a criação de sistemas *web*. A Figura 1 mostra o relacionamento entre as camadas *Model*, *View* e *Controller*. Os papéis das camadas podem ser definidos da seguinte forma:

- *Model*: responsável por gerenciar os dados do sistema, esta camada recebe as requisições do *Controller* e então as processa repassando as requisições para a camada de *View*;
- *View*: define como os dados serão apresentados pelo sistema, e nesta camada que o usuário interage com o sistema seja pela visualização e/ou entrada de dados;
- *Controller*: – traduz as interações do usuário com a camada de *View*, mapeando-as e definindo quais dados serão utilizados, quais tarefas serão executadas e quais *Views* serão utilizadas para a visualização.

Figura 1 – Relação entre as camadas *Model*, *View* e *Controller*



Fonte: LUCIANO e ALVES (2011)

2.2 Bootstrap

O Bootstrap foi inicialmente desenvolvido por integrantes do Twitter em 2010 trata-se de um *Framework* para desenvolvimento *front-end*, que permite criar páginas *web* com um mínimo de códigos e possui uma gama de componentes personalizados, baseados em CSS, HTML e Java Script. Seu uso facilita a geração de componentes na interface, evitando o trabalho de personalização de cada componente.

O Bootstrap conta um grid flexível baseado em doze colunas junto com cinco diferentes breakpoints, além de possuir uma variedade de componentes responsivos. Este é um aspecto muito importante do projeto, uma vez que se espera que o sistema desenvolvido aqui possa ser também acessado por meio de dispositivos móveis.

2.3 Banco de Dados

Um banco de dados é um sistema computadorizado cuja finalidade geral é armazenar informações e permitir que os usuários busquem e atualizem essas informações quando as solicitar, Date (2004). O principal objetivo de um banco de dados é armazenar registros que podem ser consultados quando necessário.

Existem diversos tipos de bancos de dados no mercado. Aqui destaca-se os bancos de dados relacionais. Sobre os bancos de dados relacionais Macário e Baldo (2005) dizem:

Proposto inicialmente em 1970, revolucionou o mercado desta área e, apesar das novas tendências de software como orientação a objetos, continua sendo o modelo dominante no mercado de banco de dados. (...) No modelo relacional a principal construção para representação dos dados é a relação, uma tabela com linhas não ordenadas e colunas. Uma relação consiste de um esquema e de uma instância.

Com o objetivo de gerenciar os bancos de dados existem os Sistemas de Gestão de Banco de Dados (SGBD), sobre eles Elmasri e Navathe (2018) definem:

Um sistema gerenciador de banco de dados (SGBD) é uma coleção de programas que permite aos usuários criar e manter um banco de dados. O SGBD é, portanto, um sistema de software de propósito geral que facilita os processos de definição, construção, manipulação e compartilhamento de bancos de dados entre vários usuários e aplicações. A definição de um banco de dados implica especificar os tipos de dados, as estruturas e as restrições para os dados a serem armazenados em um banco de dados.

Para a execução deste trabalho, foi utilizado o modelo relacional de banco de dados e como SGBDs o MariaDB.

2.3.1 MariaDB

O MariaDB é um sistema de código aberto, de acordo com a própria empresa mantenedora, o MariaDB é uma versão melhorada do *MySQL*, possuindo mecanismos e funcionalidades disponíveis apenas na versão privada do *MySQL*.

2.4 Servidores *Web*

Os servidores da *Web* permitem que as pessoas acessem sites na Internet. Eles processam as solicitações do usuário por meio do *Hypertext Transfer Protocol* (HTTP), o protocolo usado para transmitir informações na Internet. O termo *Web Server* pode se referir a duas tecnologias: computador (*hardware*) ou programa (*software*).

2.4.1 Apache

O Apache é um servidor HTTP que surgiu no mercado em 1995, foi desenvolvido de forma colaborativa para implementar um servidor que tem como objetivo fornecer um servidor HTTP seguro, eficiente e escalável. O servidor *web* Apache é poderoso, robusto, seguro e de código aberto, além de ser gratuito e é capaz de executar código em PHP.

Atualmente, o mesmo se encontra na versão 2.5, a mesma utilizada durante o desenvolvimento deste trabalho. O servidor está hospedado em um ambiente que utiliza o sistema operacional Debian 11.

2.5 Hospedagem e versionamento do código-fonte

Para realizar a hospedagem e versionamento do código-fonte do sistema foi utilizado o GitHub. o GitHub é um serviço baseado em nuvem que hospeda o Git – sistema de controle de versão. O GitHub permite a colaboração em projetos compartilhados enquanto se mantêm um registro detalhado do progresso.

Como mencionado, o GitHub incorpora os recursos de controle de versão do Git, o que significa que o usuário pode acompanhar e gerenciar as mudanças feitas no código-fonte em tempo real, enquanto têm acesso a todos os outros recursos do Git disponíveis. Além disso, o GitHub possui uma interface considerada mais amigável para o usuário o que torna seu uso mais acessível para pessoas com menos conhecimentos técnicos.

2.6 Trabalhos Relacionados

Nesta seção serão apresentados alguns sistemas relacionados ao sistema desenvolvido.

2.6.1 Copa Fácil

O Copa Fácil é um sistema que permite a criação de campeonatos esportivos, como futebol, vôlei, basquete, entre outros. O sistema, no plano gratuito, permite que se cadastre até 300 jogadores por campeonato, sendo esta uma limitação muito restritiva, mesmo para campeonatos amadores de futebol. O Copa Fácil também permite a geração de alguns relatórios, como o de jogadores, além de permitir gerar uma carteirinha para o jogador. Entretanto, a opção de gerar uma súmula apenas é disponibilizada mediante uma assinatura.

2.6.2 *Challenge.place*

O *Challenge.place* é um site/aplicativo que permite gerenciar campeonatos de diversos tipos diferentes de esportes e e-esportes. O *Challenge.place* além de gerenciar os campeonatos, permite também gerar estatísticas sobre as partidas disputadas e disponibilizar os dados do campeonato on-line para que o público possa acompanhar o resultado dos jogos.

O Sistema do *Challenge.place* é bem completo no quesito de permitir gerenciar campeonatos de futebol. Ficando os pontos negativos em que não é possível gerenciar os times, com exceção do elenco, além disso, o *Challenge.place* não permite a geração de uma súmula das partidas, nem a integração com o gerenciamento do time. Sendo estas duas funções disponíveis no sistema desenvolvido para a LMF.

2.6.3 Comparações com a aplicação proposta

Com base nos sistemas apresentados, este trabalho tem como objetivo desenvolver um sistema próprio para a LMF, de forma a atender suas necessidades e com a possibilidade de expansão num futuro. Buscando atender os profissionais de forma mais precisa.

A Tabela 1 representa as principais funcionalidades em comum dos softwares e quais destas são contempladas em cada um. Os requisitos aplicados ao software proposto por este trabalho. Nas demais colunas, há a relação das funcionalidades dos sistemas “Copa Fácil” e “*Challenge.place*”.

Quadro 1 – Comparativos entre as funcionalidades

Funcionalidade	Sistema desenvolvido para LMF	Copa Fácil	<i>Challenge.place</i>
Geração de Súmula	Sim	Sim ¹	Não
Diferentes tipos de usuário dentro do próprio sistema	Sim	Não	Não
Atualização em tempo real	Não	Sim	Sim
Galeria de fotos	Não	Sim ¹	Sim ¹
Cadastrar jogadores	Sim	Sim ¹	Sim ¹
Estatísticas	Não	Não	Sim ¹
Compartilhar resultados com o público	Sim	Sim	Sim

Fonte: Elaboração Própria

¹ Recurso pago;

3 DESENVOLVIMENTO

Neste capítulo as etapas realizadas durante o desenvolvimento do trabalho são detalhadas. São apresentadas as atividades referentes ao levantamento de requisitos, arquitetura do banco de dados, desenvolvimento do sistema, testes realizados e a sua implantação.

3.1 Ambiente de Desenvolvimento

O ambiente de desenvolvimento utilizado foi configurado com as seguintes tecnologias:

- Linguagens de programação: PHP 8.2.0, HTML, JavaScript, CSS e JQuery;
- *Framework*: Laravel 9.47.0 e Bootstrap 5.2;
- Gerenciador de dependências do PHP: Composer 2.3.10;
- Servidor: Apache 2.5;
- Banco de Dados: MariaDB 10.4.27;
- Hospedagem do código-fonte: GitHub;
- Editor de texto: Visual Studio Code;
- Sistema Operacional: Windows.

3.2 Stakeholders e Usuários do Sistema

Sommerville (2011) diz que os *stakeholders* são aqueles que são beneficiados pelo sistema tanto direta quanto indiretamente. Os principais *stakeholders* do Sistema do sistema proposto são: os organizadores dos campeonatos, os responsáveis pelos times, os jogadores e os torcedores.

3.2.1 Usuários do Sistema

Após identificar os stakeholders, foi identificado que seria necessário a existência de diferentes tipos de usuários para o sistema em questão, sendo:

- Super Administrador: responsável por controlar permissões de acesso e recursos necessários para que os demais usuários possam utilizar o sistema;
- Administrador de Campeonatos: este usuário possui as permissões para realizar todo o controle dos campeonatos que vai desde a criação do próprio até o controle das partidas realizadas e seus resultados;
- Administrador de Times: o administrador do time é responsável por controlar as atividades dos times, como o controle dos jogadores que compõem seu elenco, por exemplo;
- Usuário (torcedor): este usuário pode apenas visualizar os dados dos campeonatos e partidas.

3.3 Funcionalidades do Sistema

Nesta seção são descritas as funcionalidades do sistema levantadas em conversas com membros da LMF com o intuito de desenvolver uma aplicação que atenda às necessidades dos diferentes atores. O Quadro 1 exibe os requisitos levantados para o sistema.

Quadro 2 – Requisitos do Sistema

Requisito	Usuário	Descrição
RF001 – Cadastrar Campeonato	Administrador de Campeonatos e Super Administrador	Criar um campeonato
RF002 – Excluir Campeonato	Administrador de Campeonatos e Super Administrador	Excluir um campeonato
RF003 – Alterar Campeonato	Administrador de Campeonatos e Super Administrador	Alterar os dados de um campeonato

Requisito	Usuário	Descrição
RF004 – Incluir um time como participante do Campeonato	Administrador de Campeonatos e Super Administrador	Inserir um time a um campeonato
RF005 – Excluir um time de um Campeonato	Administrador de Campeonatos e Super Administrador	Excluir um time participante de um campeonato
RF006 – Selecionar jogadores que irão disputar o campeonato pelo time	Administrador de Campeonatos e Super Administrador	Selecionar quais jogadores irão disputar o campeonato pelo time
RF007 – Excluir jogadores participantes do time do campeonato	Administrador de Campeonatos e Super Administrador	Excluir os jogadores participantes de um campeonato
RF008 – Cadastrar partida	Administrador de Campeonatos e Super Administrador	Cadastrar novas partidas ao campeonato
RF009 – Excluir Partida	Administrador de Campeonatos e Super Administrador	Excluir partidas do campeonato
RF010 – Alterar partida	Administrador de Campeonatos e Super Administrador	Editar os dados de uma partida
RF011 – Visualizar classificação do campeonato	Administrador de Campeonatos e Super Administrador	Visualizar a classificação de um campeonato

Requisito	Usuário	Descrição
RF012 – Visualizar partidas realizadas	Administrador de Campeonatos e Super Administrador	Visualizar as partidas realizadas bem como seus resultados
RF013 – Visualizar próximas partidas	Administrador de Campeonatos e Super Administrador	Visualizar as próximas partidas do campeonato
RF014 – Ver súmula da partida	Administrador de Campeonatos e Super Administrador	Visualizar a súmula das partidas
RF015 – Fazer upload da súmula preenchida	Administrador de Campeonatos e Super Administrador	Após a realização da partida e com a súmula devidamente preenchida o administrador do campeonato deve conseguir enviar uma cópia para o sistema
RF016 – Remover arquivo da súmula enviada do sistema	Administrador de Campeonatos e Super Administrador	Caso seja necessário, o administrador do sistema deve conseguir remover o arquivo enviado para o sistema
RF016 – Criar grupos	Administrador de Campeonatos e Super Administrador	Criar novos grupos em um campeonato
RF017 – Excluir grupos	Administrador de Campeonatos e Super Administrador	Excluir grupos em um campeonato

Requisito	Usuário	Descrição
RF018 – Editar grupos	Administrador de Campeonatos e Super Administrador	Editar grupos em um campeonato
RF019 – Selecionar times participantes de um grupo	Administrador de Campeonatos e Super Administrador	Selecionar os times que participaram de um grupo em um campeonato
RF020 – Criar Conta	Qualquer Usuário	O usuário pode criar conta no sistema
RF021 – Alterar dados	Qualquer Usuário	O usuário altera seus dados
RF023 – Alterar Senha	Qualquer Usuário	O usuário pode alterar sua senha
RF024 – Controlar Papéis Usuário	Super Administrador	Controlar os papéis dos outros usuários

Fonte: Elaboração Própria

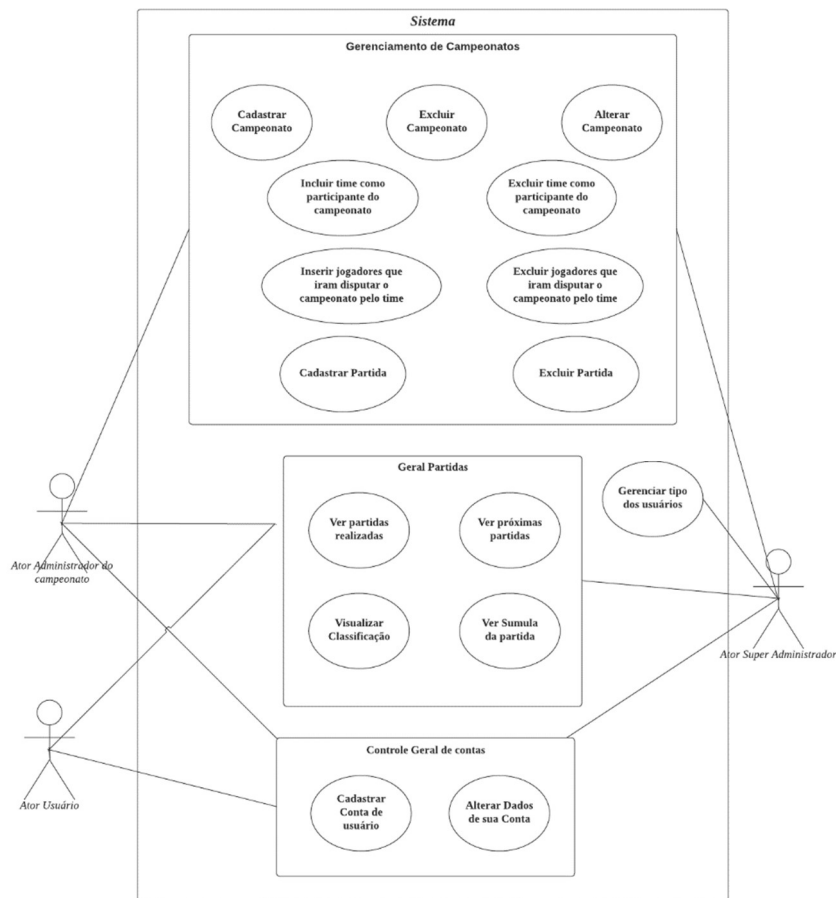
3.4 Casos de Uso

Após identificados diferentes tipos de usuários, seus papéis e definidas as principais funcionalidades que eles poderiam executar dentro do sistema, como pode ser visto na Figura 2, um diagrama de caso de uso envolve a modelagem das interações entre os objetos do sistema. Cada utilizador do sistema tem um nível de acesso específico, podendo assim interagir com mais funcionalidades associadas ao seu nível.

3.5 Mapa do Site

Aqui serão exibidos os diagramas de mapa de *site*, que demonstram as funções que podem ser acessadas dentro do módulo de campeonato e o caminho percorrido até chegar em cada opção. Foram criados três mapas de site, correspondendo à cada *homepage* dos usuários. Os seguintes usuários: Administrador de Campeonato, Super Administrador e Usuário Comum.

Figura 2 – Diagrama de Caso de Uso

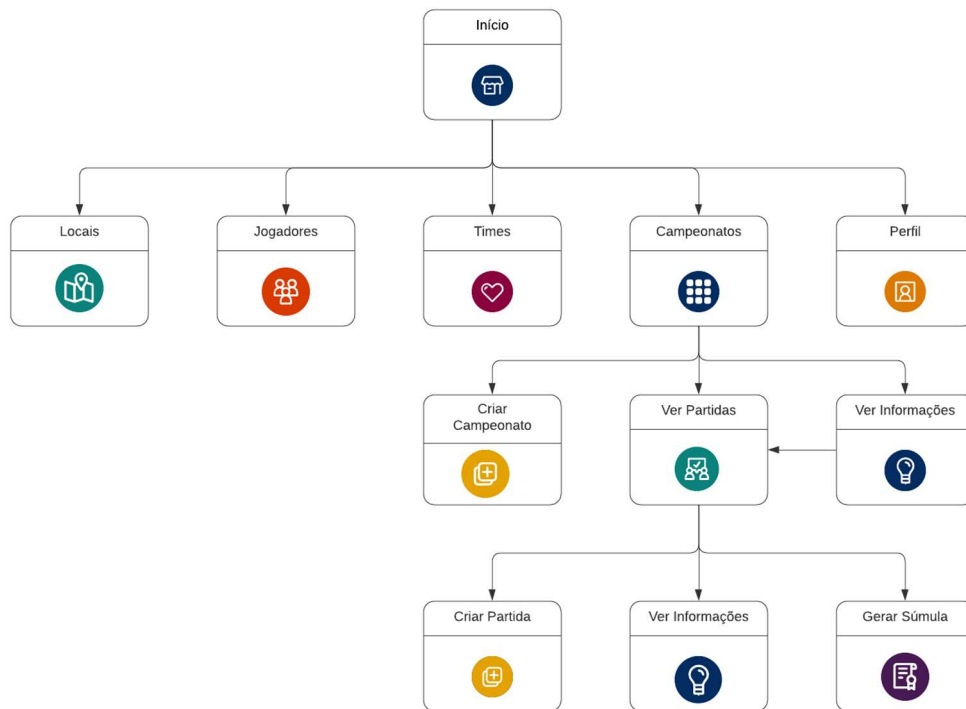


Fonte: Elaborado pelo autor

3.5.1 Administrador de Campeonato

Este mapa demonstra todas as opções oferecidas para o usuário Administrador de Campeonatos, vale ressaltar que para realizar suas ações o usuário deve estar logado no sistema. O mapa de site descrito pode ser visto na Figura 3.

Figura 3 – Mapa do Site – Administrador Campeonato

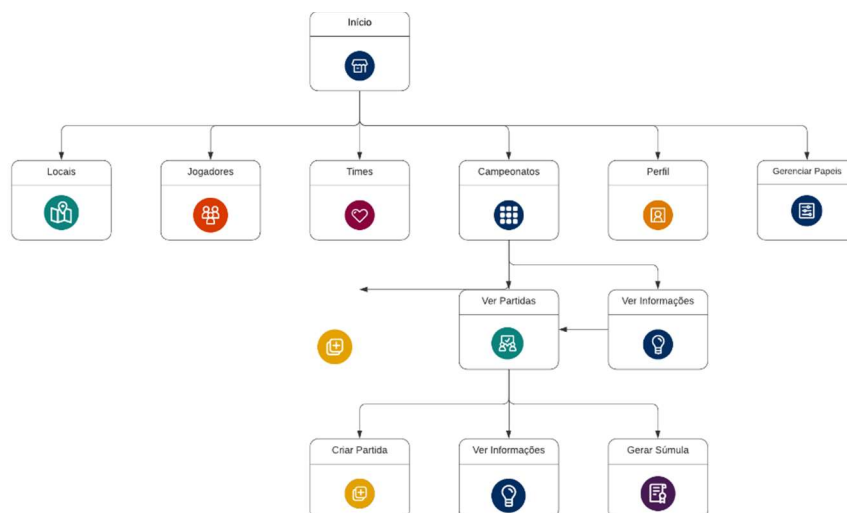


Fonte: Elaboração própria

3.5.2 Super Administrador

A Figura 4 por sua vez representa o mapa do *site* que é acessado pelos usuários do tipo Administrador de Campeonato.

Figura 4 – Mapa do Site – Super Administrador

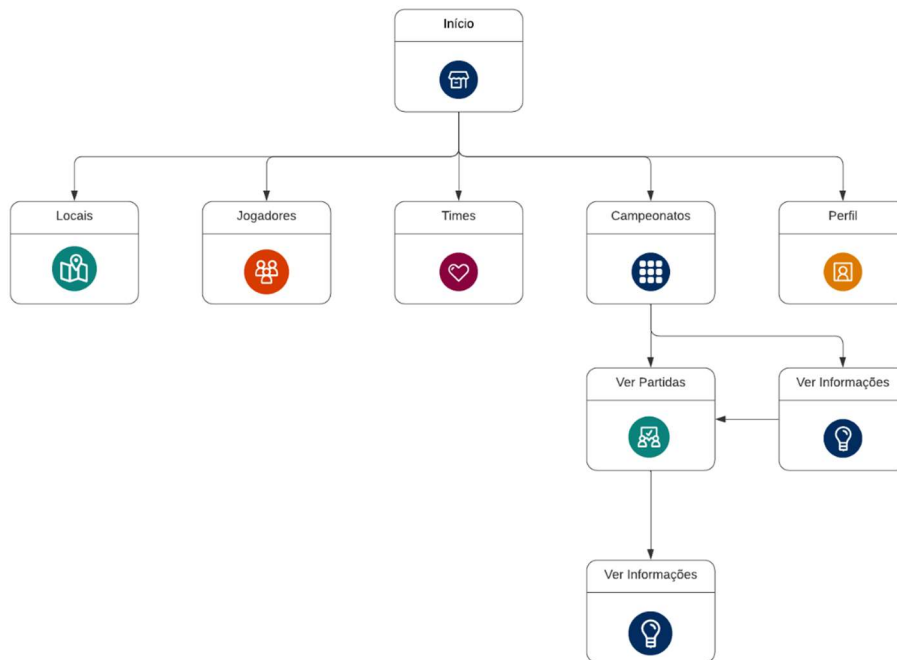


Fonte: Elaboração própria

3.5.3 Usuário Comum

A Figura 5 representa o mapa do site que é acessado pelos Usuários Comuns do sistema. Este tipo de usuário apenas faz a leitura de dados.

Figura 5 – Mapa do Site – Usuário Comum

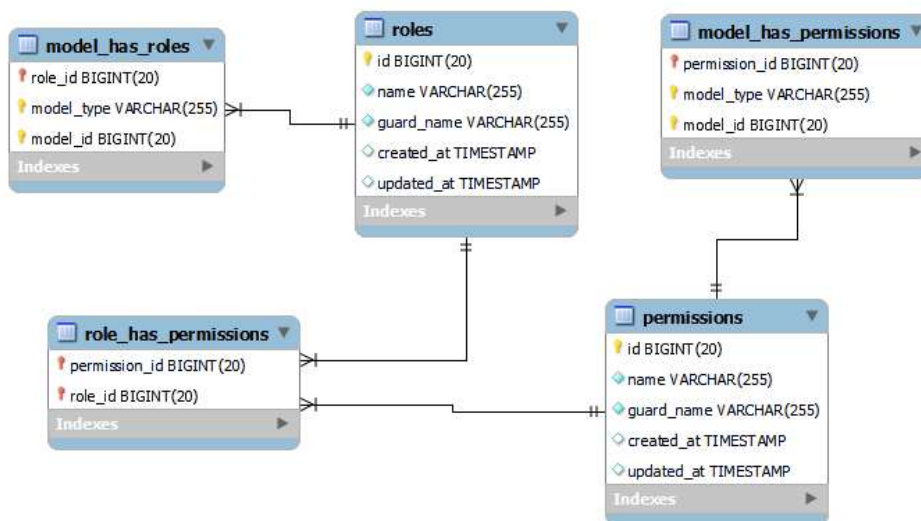


Fonte: Elaboração própria

3.6 Banco de Dados

A seguir é apresentado o Modelo Entidade Relacionamento do banco de dados, o qual foi subdividido em duas partes, a Figura 6 representa o esquema relacional do sistema e a Figura 7, as tabelas referentes à Lista de controle de acesso também conhecida por sua sigla em inglês ACL que será discutido na seção 3.7.2.

Figura 7 – Modelo Entidade Relacionamento



Fonte: Elaboração própria

3.7 Desenvolvimento do Sistema

O desenvolvimento do sistema foi dividido em cinco etapas. Primeiramente foi realizado o desenvolvimento das funcionalidades de Create, Read, Update e Delete (CRUD) para os campeonatos, seguido do CRUD para as partidas. Após finalizados os CRUD foi realizado o desenvolvimento de autenticação dos usuários e o controle de acesso ao sistema, esta é uma tarefa importante, uma vez que o sistema terá diferentes tipos de usuários e existe diferença entre as atividades que cada um pode realizar. Em seguida foram desenvolvidas ferramentas de visualização dos resultados dos campeonatos. Por fim foi desenvolvida a funcionalidade que permite gerar uma súmula em formato PDF e realizar seu *upload* no sistema.

3.7.1 Autenticação e Controle Acesso

Como o sistema possui diferentes tipos de usuários, realizar a implementação de uma ferramenta que possibilite gerenciar o papel que cada usuário pode assumir foi essencial. Para a criação deste módulo foi utilizado o pacote **Laravel-permission** desenvolvido pela empresa belga Spatie.

Esse pacote permite a criação e gerenciamento de papéis e permissões. No desenvolvimento do sistema foram criadas apenas papéis, uma vez que o **Laravel-permission** permite que se defina no *Controller* a quais métodos os papéis podem ter acesso. Sendo assim, um usuário ao utilizar sistema e tentar acessar um método que

não foi definido como sendo de acesso liberado a um dos papéis que possui é automaticamente barrado de realizar a ação.

No *Laravel-permission* é possível que um usuário possua mais de um papel e um papel pode ser dado a n-usuários, sendo este um relacionamento do tipo muitos-para-muitos.

Ao realizar o cadastramento de uma conta no sistema, todo usuário é concedido automaticamente o papel de “usuário comum”, sendo este usuário impossibilitado de fazer qualquer inserção de dados. O controle do papel que cada usuário possui é realizado pelo Super Administrador do Sistema que pode conceder ou remover os papéis dos usuários.

Durante o desenvolvimento do sistema também houve o cuidado de não se exibir ao usuário botões aqueles que seu papel não possui acesso. Desta forma, a um usuário comum, por exemplo, não é exibido os botões que permitem cadastrar e editar um campeonato.

3.7.2 Geração da Súmula em PDF

Para a geração da Súmula em PDF foi utilizada a biblioteca **barryvdh/laravel-dompdf**. Esta é uma variação da biblioteca **dompdf** que foi adaptada para o Laravel. O Laravel utiliza a *Blade*, um mecanismo de modelagem que não restringe o uso de simples códigos PHP nos *templates* das telas.

Assim como a biblioteca original, o **barryvdh/laravel-dompdf** consegue converter as páginas HTML para um documento no formato PDF que pode ser salvo no dispositivo e impresso caso seja necessário. Um versão da súmula pode ser visualizado no Apêndice B.

3.7.3 PWA Laravel

Afim de se criar uma aplicação móvel para o sistema, foi escolhido trabalhar com o PWA Laravel que possui o poder de transformar uma aplicação *web* para se comportar como um aplicativo.

Progressive Web Applications (PWA) são softwares fornecidos pela *web*; codificados usando linguagens de programação comuns como JavaScript, HTML ou CSS; e que se parecem, se comportam e funcionam como um aplicativo móvel.

Esta é a maneira mais rápida de transformar uma aplicação *web* em um aplicativo.

Como o sistema desenvolvido é necessário que se crie um arquivo chamado *manifest.json*. Este arquivo contém informações sobre o aplicativo da *web*, incluindo algumas informações básicas incluem o nome do aplicativo, URL inicial, cor do tema e ícones. Todas as informações necessárias são especificadas no formato JSON.

Também é necessário configurar um *Service Worker*. Um *Service Worker* é um tipo especial de *worker* baseado em eventos, o qual é registrado para um determinado *path* e origem. Na prática, ele é um arquivo JavaScript que pode controlar as páginas do site ao qual ele está associado, interceptando e modificando requisições e a navegação em si.

Esta tecnologia estava sendo estudada para ser implementada no sistema de forma a se possibilitar que os usuários pudessem acessá-lo através de um aplicativo para dispositivos móveis, porém, como o servidor não fornece suporte ao PWA.

3.7.4 Responsividade

Como se deseja que os usuários do sistema consigam acessá-lo a partir de dispositivos móveis, tais como uma *smartphone*, a responsividade do sistema foi um ponto importante. SILVA e MOREIRA (2019) definem o *Web Design Responsivo* como a busca em proporcionar, de maneira automática, a adaptação da interface e seus elementos nos diferentes tamanhos de tela dos dispositivos que acessam determinada página *web*.

Para atingir tal objetivo, a utilização do Bootstrap como *Framework* do *front-end* foi de extrema importância, uma vez ele oferece um conjunto de componentes HTML e CSS, tais como botões, formulários, tabelas, etc. que podem ser alterados por meio do uso de classes nestes elementos HTML.

4 Resultados

Este capítulo apresenta os principais resultados obtidos durante o desenvolvimento do software. Os resultados consistem nas telas e descrição da aplicação e os testes de usabilidade realizados.

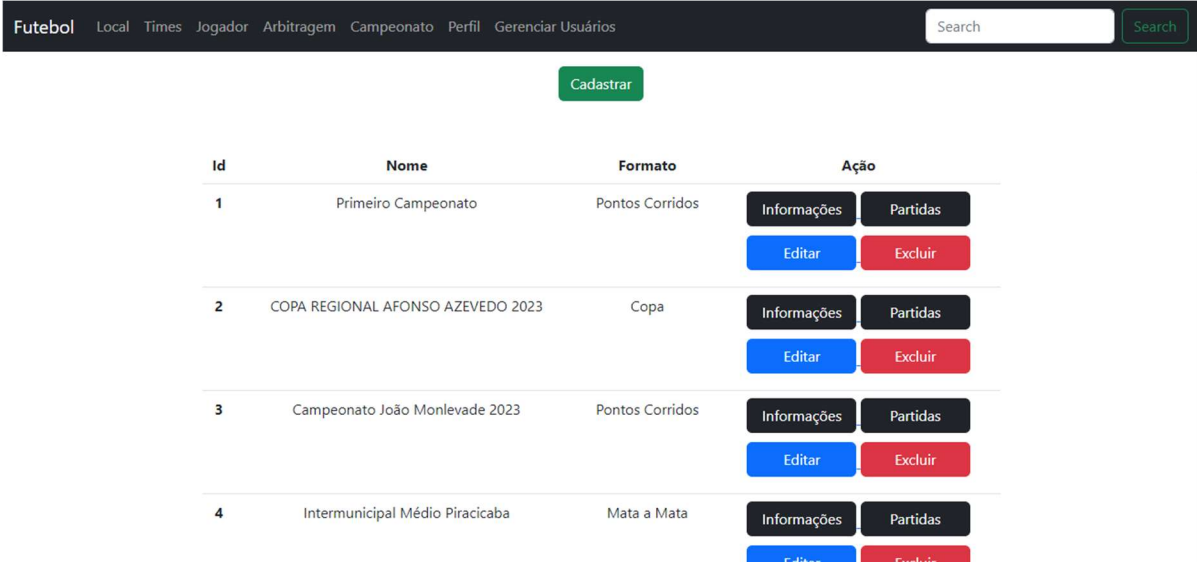
4.1 Telas do Sistema

A seguir são descritas as principais funcionalidades do sistema. São apresentadas as principais telas e as suas respectivas características e uma breve descrição do seu funcionamento. As demais telas estão disponíveis no Apêndice A.

4.1.1 Campeonatos

A tela inicial do gerenciamento dos campeonatos apresenta os campeonatos cadastrados no sistema. Um campeonato pode ser cadastrado, visualizado editado e excluído. Como pode ser visto na Figura 8, esta tela também permite que se busque por campeonatos pelo seu nome, formato, período em que ocorre e time participante.

Figura 8 – Tela Inicial Campeonatos



Id	Nome	Formato	Ação	
1	Primeiro Campeonato	Pontos Corridos	Informações	Partidas
			Editar	Excluir
2	COPA REGIONAL AFONSO AZEVEDO 2023	Copa	Informações	Partidas
			Editar	Excluir
3	Campeonato João Monlevade 2023	Pontos Corridos	Informações	Partidas
			Editar	Excluir
4	Intermunicipal Médio Piracicaba	Mata a Mata	Informações	Partidas
			Editar	Excluir

Fonte: Elaboração própria

4.1.2 Seleciona Jogadores Participantes

Uma vez criado os campeonatos, é possível adicionar os times. Uma regra de negócio diz que um jogador por participar apenas por um time em um determinado campeonato. Sendo assim, não é possível que um jogador dispute um campeonato por dois times. Para ajudar na tarefa de selecionar os jogadores que disputaram a partida, foi utilizado do CSS para estilizar de vermelho os nomes dos jogadores que já estão disputando o campeonato. Estes mesmos jogadores vem com o *selectbox* desmarcado enquanto os demais vem marcados, a fim de facilitar o processo de seleção. Essa tela pode ser vista na Figura 9.

Figura 9 – Tela Seleciona Jogadores Participantes

Fonte: Elaboração própria

Além da estilização utilizando o CSS também foi criada uma regra de validação para impedir que jogador dispute o campeonato por mais de um time.

4.1.3 Informações de um Campeonato

Para exibir as informações e os resultados dos campeonatos, foi construída uma tela que além das informações básicas dos campeonatos mostra também uma tabela com a classificação. A classificação segue as regras onde para cada vitória um time ganha 3 pontos, 1 ponto para os empates e 0 em casos de derrota.

Por se tratar de campeonatos amadores nem sempre as informações de cartões são registradas e por isso estes fatores não foram considerados nas tabelas.

Como é mostrado na Figura 10, nesta tela é possível visualizar também as partidas recentes e as próximas partidas a serem disputadas.

Figura 10 – Tela Informações Campeonatos

Últimas Partidas		Próximas Partidas	
América Mineiro	0 X 1 Ipatinga	Atlético Mineiro	X Democrata
Vila Nova	2 X 2 Caldense	Ver Mais	
Cruzeiro	2 X 3 Patrocinense		
Atlético Mineiro	0 X 0 Democrata		
Ver Mais			

Time	Pontos	Partidas	Vitorias	Empates	Derrotas
Cruzeiro	6	2	2	0	0
Democrata	4	2	1	1	0
Ver Mais					

Fonte: Elaboração própria

4.1.4 Partidas

Para a listagem das partidas dos campeonatos são dadas as opções de se editar os dados das partidas. Gerar a súmula pré-preenchida da partida. Editar o resultado e enviar a súmula já preenchida para ser armazenada no servidor. Como

Início Local Times Jogador Arbitragem Campeonato Perfil Gerenciar Usuários

Figura 11 – Tela Partidas

Time Casa	Time Visitante	Data	Local	Status	Resultado	Opções
Caldense	Ipatinga	10/03/2023 20:03	Campo Quebra Pé	Encerrada	2 X 2	Editar Remover Súmula Editar Resultado Baixar Súmula
Cruzeiro	Atlético Mineiro	10/03/2023 20:03	Campo Quebra Pé	Encerrada	5 x 0	Editar Gerar PDF Editar Resultado Enviar Sumula

Fonte: Elaboração própria

pode ser visto na Figura 11, caso a súmula já tenha sido enviada para o servidor e dada a opção de remover a súmula do servidor e também a opção de se realizar o seu *download*.

4.1.5 Gerenciamentos dos Papeis dos Usuários

Como o sistema comporta diferentes tipos de usuário é necessário que se faça seu controle. Para isso a biblioteca *Laravel-permission* foi utilizada. Esta biblioteca permite a criação de papeis que são atribuídos aos usuários.

Cada usuário pode possuir mais de um papel e este controle é realizado pelo Super Administrador do sistema. Cabe a este usuário conceder ou remover os papeis dos demais usuários do sistema. Para tal controle o sistema possui uma tela onde é possível pesquisar os usuários por seu nome ou papel. Esta tela pode ser visualizada na Figura 12.

Figura 12 – Tela Gerencia Papeis

Usuario:

Selecione...
▼

Papel:

AdminTime
▼

Pesquisar

Atribuir Papel

Novo Usuário

Usuario	Papel	Ação
junior	AdminTime	Excluir
Jean	AdminTime	Excluir
Janio	AdminTime	Excluir
Veronica	AdminTime	Excluir

Fonte: Elaboração própria

4.2 Testes

Segundo (PRESSMAN, 2005), o principal objetivo do teste de sistema é localizar bugs, falhas, defeitos e verificar a funcionalidade do sistema que está sendo desenvolvido. Ainda segundo (PRESSMAN, 2005), o teste funcional busca apontar, entre várias coisas, que os requisitos funcionais do sistema são atendidos, que as entradas são aceitas corretamente, que as saídas esperadas são produzidas e que a integridade das informações externas é verificada, mantido, não sendo necessário se preocupar com a estrutura lógica interna do sistema. Sendo assim, sempre que uma nova funcionalidade era implementada ela era testada para verificar se seu funcionamento está conforme o esperado. As demais funcionalidades já implementadas também eram verificadas para confirmar se a alteração do código fonte não gerou erros. Desta forma, o sistema é testado pouco a pouco e todas as funções enviadas anteriormente serão testadas novamente, sendo este um esforço realizado para garantir que não haja falhas nos processos.

Para a realização dos testes, também se contou com a participação de bolsistas de um projeto do orientador deste trabalho. Estes testes foram realizados de forma informal e tinham com função simular o uso da aplicação a fim de se encontrar erros e possíveis melhorias do projeto. Com feedback destes testes realizados foi construída o Quadro 3.

Quadro 3 – Erros reportados nos testes

Erros
Informações de Complemento e Estado para os Locais não está sendo salva.
Necessidade de inserir os dados novamente caso um dado obrigatório não fosse preenchido ao criar novas partidas.
Possibilidade de criar grupos com número negativo de times.
Erro ao adicionar um novo jogador
<i>Back to top</i> não funcionando.

Fonte: Elaboração própria

5 CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS

O desenvolvimento deste trabalho demonstrou ser um processo muito mais complexo do que o esperado inicialmente. Garantir que o sistema atenda às necessidades da LMF não é algo simples. Um dos principais pontos a ser notado é que, ao longo do projeto, parte dos requisitos mudaram, devido a um amadurecimento por parte de todos os envolvidos e das ideias propostas para o sistema.

A utilização dos Frameworks Laravel e Bootstrap foram dois fatores primordiais que contribuíram para a diminuição da complexidade de adaptação do sistema aos novos requisitos, auxiliando muito no processo como um todo, pois este permitiu que o sistema seja modular e o mesmo possa ser melhorado no futuro, além de simplificar o processo de criação de telas responsivas para o sistema.

Entre as principais dificuldades encontradas, desenvolver uma forma de exibir os chaveamentos dos campeonatos com formato Mata a mata e Copa, se demonstrou uma tarefa muito complexa, onde a exibição de forma que possa ser considerada correta, clara e objetiva não foi atingida e por isso a mesma não foi entregue. Esperava-se também entregar um aplicativo que utilizasse a tecnologia PWA, afim de se poder adaptar o sistema *web* desenvolvido para um aplicativo; infelizmente, o servidor utilizado não atende as configurações do *Progressive Web Application* (PWA). A sua implantação estava sendo estudada para que os usuários pudessem utilizar o software, por meio dos recursos do Laravel, realizando a instalação de um aplicativo em dispositivos móveis; e embora o uso através de um aplicativo não seja viável, ainda é possível acessar o sistema usando o navegador do *smartphone*. Isso é possível, pois o desenvolvimento de todo o sistema foi pensado considerando para aproveitar os aspectos de responsividade que o Bootstrap é capaz de oferecer.

Como mencionado o sistema, assim como qualquer outro, possui capacidade de crescimento e pontos de melhoria. visando complementar ainda mais o sistema, algumas ideias de trabalhos futuros foram sugeridas:

- Desenvolver uma ferramenta para a visualização dos chaveamentos dos campeonatos com formato Mata a mata e Copa;
- Criar a aplicação para *smartphone* utilizando PWA do *Framework* Laravel;

- Criar dashboards para acompanhamento campeonatos e seus resultados;
- Disponibilizar uma opção de atualização dos resultados partidas em tempo real;
- Permitir que se extraia dados e estatísticas, tais como os times com melhor artilharia, defesa, etc.

Pode-se considerar que, embora nem todos os objetivos propostos para o trabalho foram completamente atingidos, o mesmo teve um desempenho considerado satisfatório; sendo o sistema capaz de gerenciar os campeonatos organizados pela LMF, sendo este o principal objetivo do trabalho, além de permitir o gerenciamento das sumulas das partidas. Espera-se que seu uso venha realmente a melhorar a realização de campeonatos por parte da LMF.

Por fim, o trabalho contém muitas sugestões de melhorias para serem desenvolvidos futuramente. De modo geral, a aplicação permanecerá disponível para uso e testes dos usuários

REFERÊNCIAS

ASSUNÇÃO, Oto Braz. **Desenvolvimento de um sistema web para avaliação de classes da Universidade Federal de Ouro Preto**. 2017. 74 f. Monografia (Graduação em Sistemas de Informação) – Instituto de Ciências Exatas e Aplicadas, Universidade Federal de Ouro Preto, João Monlevade, 2017.

DATE, C. J. **Introdução a Sistemas de Banco de Dados**. Elsevier Editora, 2004.

ELMASRI, Ramez e NAVATHE, Shamkant B. **Sistemas de Banco de Dados**. Pearson Universidades. 1a Edição, 2019

JÚNIOR, J. C. C. L. et al. Futebol amador: lazer e saúde. **Anais III JOIN / Edição Brasil**. Campina Grande: Realize Editora, 2017. Disponível em: <<https://editorarealize.com.br/artigo/visualizar/49857>>. Acesso em: 20/01/2023.

JÚNIOR, Celso Gomes Barreto. **Agregando Frameworks de Infra- Estrutura em uma Arquitetura Baseada em Componentes: Um Estudo de Caso no Ambiente AulaNet**. 2006. Tese (Mestrado). Programa de Pós-Graduação em Informática, Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, 2006.

LUCIANO, Josue; ALVES, Wallison. PADRÃO DE ARQUITETURA MVC: MODEL-VIEW-CONTROLLER. Bebedouro - SP, 2011. **Revista EPeQ Fafibe**, 3ª. Ed., Vol. 01 (2011). Centro Universitário UNIFAFIBEI, Bebedouro - SP, 2011.

MACÁRIO, Carla Geovana do N.; BALDO, Stefano Monteiro. O Modelo Relacional. jan. 2005. Disponível em: < <https://www.ic.unicamp.br/~geovane/mo410-091/Ch03-RM-Resumo.pdf>>. Acesso em: 20/01/2023.

PIMENTA, Rosângela; SERGIO, José. **Desvendando o jogo: futebol amador e pelada na cidade e no sertão**. 2009. Tese (Doutorado). Programa de Pós-Graduação em Sociologia, Universidade Federal de Pernambuco, Recife, 2009.

SILVA, Rebeca Pinto Cruz e; MOREIRA, Leonardo Oliveira. **Um estudo de frameworks de design responsivo e avaliação na perspectiva da acessibilidade na web**. Artigo apresentado ao Curso de Sistemas e Mídias Digitais da Universidade Federal do Ceará, em 2019, como requisito para obtenção do título de bacharel em Sistemas e Mídias Digitais.

APÊNDICE A – Telas do Sistema

Figura 13 – Tela Inicial Campeonatos

Fonte: Elaboração própria

Figura 14 – Tela Inicial Campeonatos vista em um smartphone

Fonte: Elaboração própria

Figura 15 – Tela que lista os Campeonatos – visão do administrador de campeonatos

Id	Nome	Formato	Ação
1	Campeonato João Monlevade 2023	Pontos Corridos	Informações Partidas Editar Excluir
2	Copa JM	Copa	Informações Partidas Editar Excluir
4	Copa Regional Afonso Azevedo 2023	Copa	Informações Partidas Editar Excluir
5	Intermunicipal Médio Piracicaba	Mata a Mata	Informações Partidas Editar Excluir

Fonte: Elaboração Própria

Figura 16 – Tela que lista os Campeonatos – visão de usuário comum

Id	Nome	Formato	Ação
1	Campeonato João Monlevade 2023	Pontos Corridos	Informações Partidas
2	Copa JM	Copa	Informações Partidas
4	Copa Regional Afonso Azevedo 2023	Copa	Informações Partidas
5	Intermunicipal Médio Piracicaba	Mata a Mata	Informações Partidas

© 2023 UFOP, Inc. · [Privacy](#) · [Terms](#) [Back to top](#)

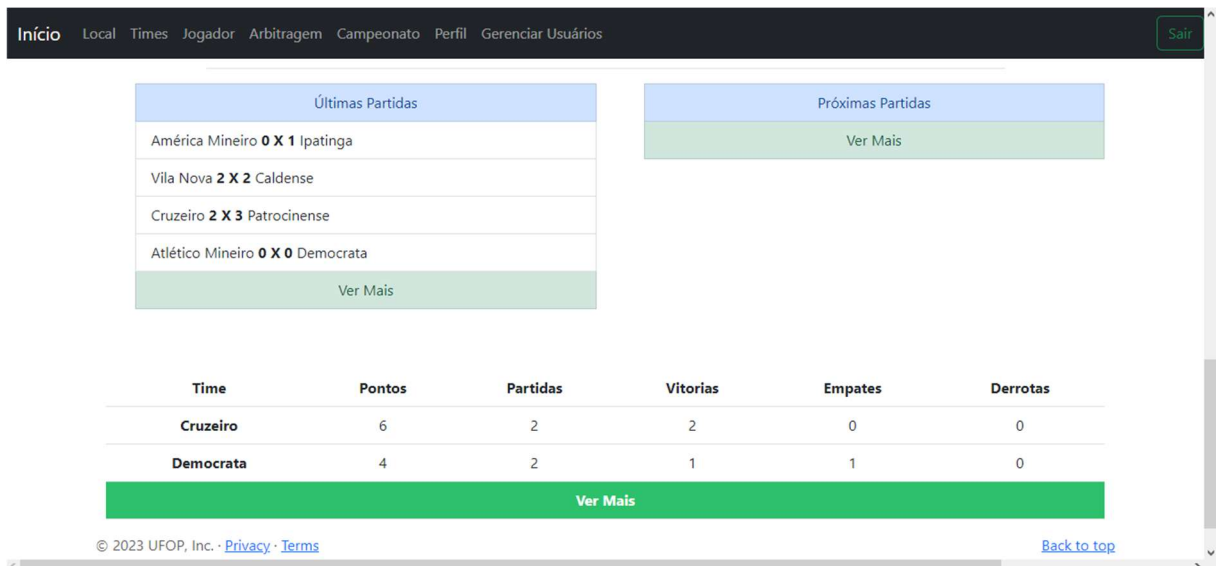
Fonte: Elaboração Própria

Figura 17 – Tela que lista os Campeonatos – visão de usuário comum em um smartphone



Fonte: Elaboração própria

Figura 18 – Tela que mostra as Informações do campeonato



Fonte: Elaboração Própria

Figura 19 – Tabela de classificação de um campeonato

Time	Pontos	Partidas	Vitórias	Empates	Derrotas
Cruzeiro	6	2	2	0	0
Democrata	4	2	1	1	0
Vila Nova	3	2	1	0	1
Caldense	3	2	1	0	1
Atlético Mineiro	1	2	0	1	1
Ipatinga	1	2	0	1	1
Patrocinense	0	2	0	0	2

Fonte: Elaboração Própria

Figura 20 – Tabela de classificação de um campeonato vista em um smartphone

Time	Pontos	Partidas	Vitor
Cruzeiro	6	2	2
Democrata	4	2	1
Vila Nova	3	2	1
Caldense	3	2	1
Atlético Mineiro	1	2	0
Ipatinga	1	2	0
Patrocinense	0	2	0

Fonte: Elaboração Própria

Figura 21 – Tabela de classificação de um campeonato com grupos

Grupo A					
Time	Pontos	Partidas	Vitorias	Empates	Derrotas
América Mineiro	3	1	1	0	0
Cruzeiro	1	1	0	1	0
Caldense	1	1	0	1	0
Atlético Mineiro	0	1	0	0	1

Grupo B					
Time	Pontos	Partidas	Vitorias	Empates	Derrotas
Vila Nova	0	0	0	0	0
Ipatinga	0	0	0	0	0
Democrata	0	0	0	0	0

Fonte: Elaboração Própria

Figura 22 – Tabela de classificação de um campeonato com grupos vista em um smartphone

Grupo A			
Time	Pontos	Partidas	Vitorias
América Mineiro	3	1	1
Cruzeiro	1	1	0
Caldense	1	1	0
Atlético Mineiro	0	1	0

Grupo B			
Time	Pontos	Partidas	Vitor
Vila Nova	0	0	0
Ipatinga	0	0	0

192.168.3.7

Fonte: Elaboração Própria

Figura 23 – Tela que lista as partidas do campeonato – visão do administrador de campeonato

Partidas

Adicionar Partida

Time Casa	Time Visitante	Data	Local	Status	Resultado	Opções
América Mineiro	Vila Nova	10/03/2023 20:02	Campo Quebra Pé	Encerrada	1 x 0	Editar Remover Súmula Editar Resultado Baixar Súmula
Caldense	Ipatinga	10/03/2023 20:03	Campo Quebra Pé	Encerrada	3 x 2	Editar Gerar PDF Editar Resultado Enviar Súmula

Fonte: Elaboração Própria

Figura 24 – Tela que lista as partidas do campeonato – visão do usuário comum

Partidas

Time Casa	Time Visitante	Data	Local	Status	Resultado	Opções
América Mineiro	Vila Nova	10/03/2023 20:02	Campo Quebra Pé	Encerrada	1 x 0	Baixar Súmula
Caldense	Ipatinga	10/03/2023 20:03	Campo Quebra Pé	Encerrada	3 x 2	Baixar Súmula

Fonte: Elaboração Própria

Figura 25 – Tela que lista as partidas do campeonato vista em um smartphone

Time Casa	Time Visitante	Data
América Mineiro	Vila Nova	10/03/2023 20:02
Caldense	Ipatinga	10/03/2023 20:03

192.168.3.7

Fonte: Elaboração Própria

Figura 26 – Tela que confirma os jogadores que irão disputar o campeonato pelo time

Selecionar Jogadores

Selecione os jogadores que iram disputar o campeonato **Copa Regional Afonso Azevedo 2023** pelo time **Cruzeiro***

- ANTONIO
- BRUNO
- CARLOS
- DANIEL
- EDUARDO
- FELIPE
- FRANCISCO
- GABRIEL
- JOAO
- JOSE
- Junior
- LUCAS
- LUIS
- LUIZ
- MARCELO
- MARCOS

Fonte: Elaboração Própria

Figura 27 – Tela que confirma os jogadores que irão disputar o campeonato pelo time vista em um smartphone



Fonte: Elaboração Própria

Figura 28 – Tela que lista os usuários e seus papéis

Pesquisar

Usuario:

Papel:

Usuario	Papel	Ação
Junior	AdminCampeonato	<input type="button" value="Excluir"/>
Jean C	AdminCampeonato	<input type="button" value="Excluir"/>

Fonte: Elaboração Própria

