

**CENTRO PAULA SOUZA
FACULDADE DE TECNOLOGIA DE FRANCA
“Dr. THOMAZ NOVELINO”**

TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

**LUCAS DE PAULA MELO
BRUNO HENRIQUE**

**ETERNUS COLISEUM:
O físico e o digital em uma só arena**

Trabalho de Graduação apresentado à Faculdade de Tecnologia de Franca - “Dr. Thomaz Novelino”, como parte dos requisitos obrigatórios para obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Orientador: Prof. Me. Alexandre Gomes da Silva

FRANCA/SP

2025

ETERNUS COLISEUM: o físico e o digital em uma só arena

Lucas de Paula Melo¹

Bruno Henrique Messias²

Resumo

Este trabalho apresenta o desenvolvimento de um sistema inovador para a criação e gerenciamento de torneios de jogos eletrônicos (e-sports) e modalidades esportivas tradicionais, com o objetivo de aprimorar a organização, a acessibilidade e a rapidez na realização desses eventos. Diferentemente dos esportes convencionais, que ocorrem em ambientes físicos e exigem esforço corporal direto, os e-sports acontecem em ambiente digital, demandando habilidades cognitivas e estratégicas específicas. A proposta surgiu a partir da identificação de falhas recorrentes na administração de torneios universitários, evidenciando a necessidade de uma solução tecnológica que oferecesse maior controle, transparência e facilidade de uso. O sistema proposto busca beneficiar organizadores, espectadores e participantes, oferecendo ferramentas que simplificam a estruturação de campeonatos, o acompanhamento de resultados e a comunicação entre os envolvidos. A metodologia adotada baseou-se em uma análise de diversos campeonatos amadores e estudantis, aliada a uma pesquisa exploratória e ao desenvolvimento iterativo do sistema, com foco na usabilidade e na melhoria da experiência dos usuários. Assim, o projeto tem como objetivo alcançar campeonatos amadores e estudantis, tornando as competições mais acessíveis, rápidas e eficientes, reduzindo o desgaste organizacional e promovendo melhor gestão de torneios.

Palavras-chave: acessibilidade; esportes; e-sports; sistema de gestão; torneios.

Abstract

This work presents the development of an innovative system for creating and managing electronic game tournaments (e-sports) and traditional sports competitions, aiming to improve the organization, accessibility, and speed of these events. Unlike conventional sports, which take place in physical environments and require direct physical effort, e-sports occur in digital environments, demanding specific cognitive and strategic skills. The proposal arose from practical experience with failures in the administration of a university tournament, highlighting the need for a technological solution that offers greater control, transparency, and ease of use. The proposed system seeks to benefit organizers, spectators, and participants by providing tools that simplify tournament structuring, results tracking, and communication among those involved. The adopted methodology was based on the analysis of various amateur and student tournaments, combined with exploratory research and iterative system development focused on usability and user experience improvement. Thus, the project aims to reach amateur and student tournaments, making competitions more accessible, faster, and more efficient while reducing organizational strain and promoting better tournament management.

Keywords: accessibility; e-sports; management system; sports; tournaments.

¹ Graduando em Análise e Desenvolvimento de Sistemas pela Fatec Dr Thomaz Novelino – Franca/SP. Endereço eletrônico: lucassocialpm@gmail.com

² Graduando em Análise e Desenvolvimento de Sistemas pela Fatec Dr Thomaz Novelino – Franca/SP. Endereço eletrônico: henriquem3ss@hotmail.com

1 Introdução

A organização de torneios competitivos, tanto no âmbito digital quanto presencial, enfrenta obstáculos significativos que limitam seu potencial. A complexidade de gestão, a fragmentação de ferramentas e as barreiras técnicas frequentemente desestimulam organizadores amadores e restringem o crescimento de comunidades competitivas. É neste contexto que o Eternus Coliseum se apresenta como uma solução inovadora - uma plataforma unificada que simplifica radicalmente o processo de criação e administração de torneios.

O aplicativo proposto aborda três desafios principais: a ineficiência nos processos manuais de organização, a dificuldade de engajar participantes casuais e a falta de integração entre espectadores e competidores. Ao automatizar tarefas repetitivas e oferecer uma interface intuitiva, o Eternus Coliseum não apenas economiza tempo valioso, mas também democratiza o acesso à organização de eventos competitivos.

O desenvolvimento do projeto foi conduzido por meio de metodologias ágeis, testes do sistema e uma implementação de tabelas Kanban para visualização do fluxo de trabalho. A relevância desta iniciativa se amplifica quando consideramos o crescimento exponencial do mercado de competições digitais, que movimentou mais de US\$ 1,8 bilhão em 2023, e a crescente demanda por soluções que unam praticidade e capacidade de engajamento.

Mais do que uma ferramenta de gestão, o Eternus Coliseum representa uma plataforma de incentivo à competição saudável, capaz de atender desde pequenos torneios comunitários até eventos de maior escala, sempre com foco na acessibilidade e na experiência do usuário.

1.1 Termo da Abertura do Projeto (TAP)

O Termo de Abertura do Projeto (TAP) é um documento formal que autoriza o início de um projeto e estabelece suas bases, como objetivos, escopo, partes envolvidas e recursos necessários. Ele serve como um contrato entre o gerente do projeto e a organização, garantindo que todos entendam o que será realizado e com quais limites.

Segundo o PMBOK® Guide (7ª ed., PMI, 2021):

“O TAP é essencial porque oficializa o projeto, alinha expectativas entre os stakeholders e reduz riscos de falhas ao deixar claros os requisitos desde o

início. Sem ele, há maior chance de retrabalho, falta de recursos ou desvios nos objetivos.”

1.2 Título do Projeto

Eternus Coliseum: O físico e o digital em uma só arena.

1.3 Objetivo do Projeto

O projeto possui o objetivo de tornar a organização de torneios mais criativa, fácil de acompanhar, rápida e padronizada, para fins de maior organização e praticidade, tornando assim uma opção viável para evitar anotações manuais, perda de dados e modificações sem permissão.

1.4 Justificativa do Projeto

Observa-se que a organização por meios manuais ou simples demais, sempre acabam se tornando caóticas, tornando a experiência de participar um torneio péssima para os organizadores, participantes e até mesmo espectadores, o objetivo é tornar essas competições mais organizadas e transparentes para que os usuários não sejam afetados de forma negativa por experiências ruins.

1.5 Descrição do Projeto

O projeto será feito em um aplicativo, foi realizada uma análise de requisitos para que seja suprida todas as lacunas possíveis do projeto, foi realizada toda a documentação do sistema que seria um aplicativo voltado para sistemas operacionais Android e IOS, foi utilizada a linguagem React Native.

O projeto possui um sistema de cadastro tanto de usuários, organizadores, times, competições e funções específicas dentro de cada cadastro, o projeto conta com uma tela login.

Haverá diferentes níveis hierárquicos de acesso, sendo divididos em administrador, organizador, participante e espectador.

A base das competições será a permissão de criação e gestão das competições, como a seleção de formato de chaves, inclusão de posições, penalidades e demais funções que serão apresentadas no artefato de requisitos.

1.6 Funcionalidades

- Entrar com o login no APP
- Navegar nos menus
- Criar competição
- Gerenciamento de competições
- Acompanhamento de competições
- Cadastro de times
- Cadastro de jogadores
- Escolher um formato de chaveamento
- Registrar pontuação
- Arbitragem
- Plano de assinatura

1.7 Premissas

As premissas do projeto são as entregas de artefatos como:

- Missão, Visão e Valor (Apêndice 1).
- EAP
- BPMN.

A futura implementação com base na integração de parceiros como organizações e marcas registradas.

1.8 Riscos do Projeto

- Atrasos no cronograma devido a problemas técnicos ou falta de recursos.
- Alterações da base do projeto durante o desenvolvimento.
- Troca de linguagem em caso de implementação WEB futura.
- Problemas no design para criação do menu de torneios.

1.9 Equipe do Projeto

A equipe será composta por dois membros, sendo eles:

- Lucas de Paula Melo - Gerente de Projeto.
- Bruno Henrique Messias – Desenvolvedor.

1.10 Orçamento

Será calculado no artefato Métricas, utilizando a planilha de métricas.

1.11 Marcos

O cronograma foi estipulado para as entregas iniciais até a data de entrega da documentação e apresentação do projeto.

- Data de início: novembro de 2024
- Data de entrega até: novembro de 2025

1.12 Estrutura Analítica do Projeto (EAP)

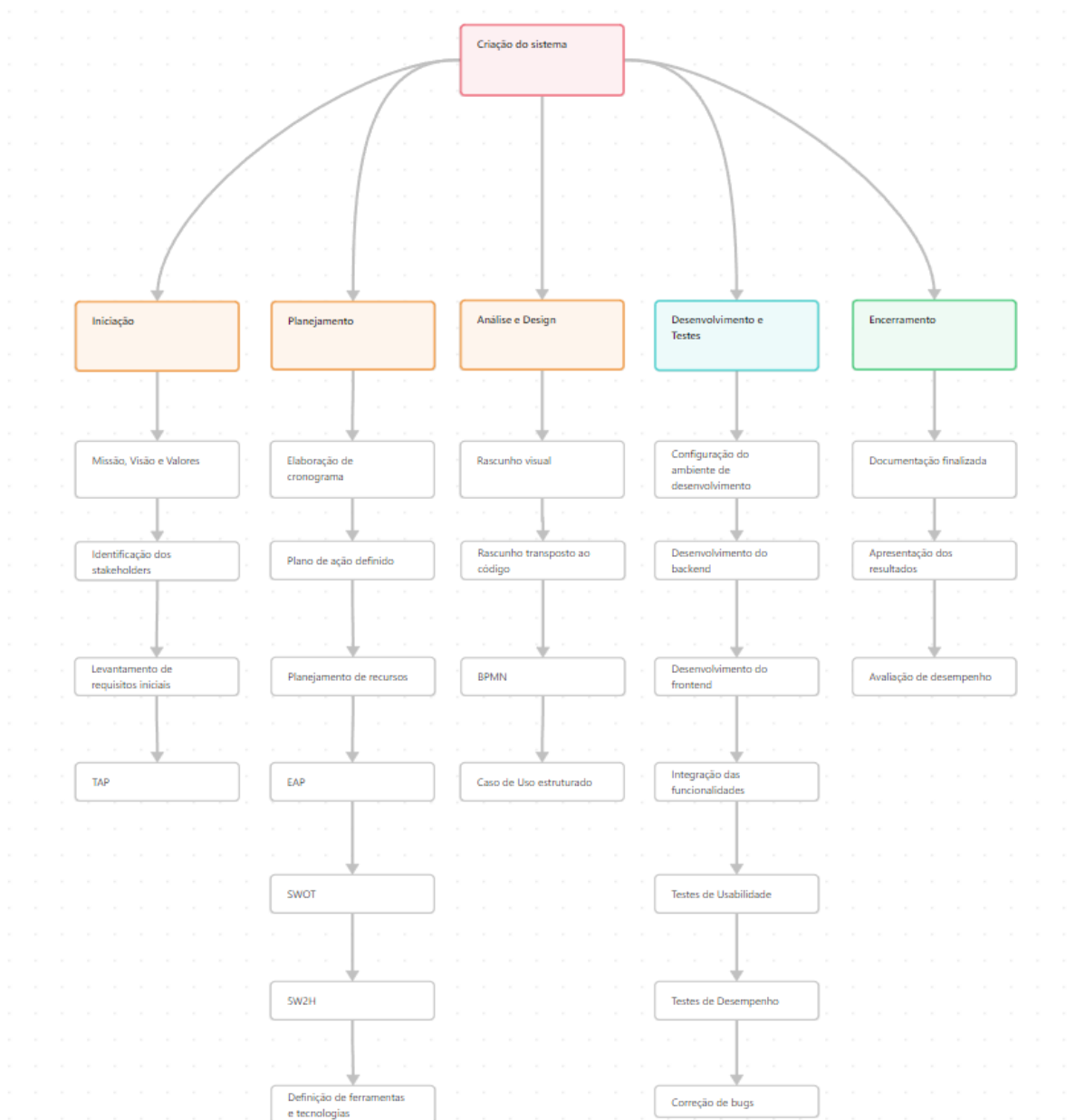
A Estrutura Analítica do Projeto (EAP) é uma ferramenta fundamental no gerenciamento de projetos, pois permite dividir o escopo total do trabalho em partes menores, organizadas de forma hierárquica. Essa decomposição facilita o planejamento, a atribuição de responsabilidades, a estimativa de prazos e custos, além do controle e monitoramento do projeto como um todo. A EAP garante que todas as entregas previstas sejam consideradas, reduzindo riscos de omissões, retrabalho e desorganização.

Segundo Kerzner (2006, apud Maximiano, 2011):

“A EAP é essencial porque “ajuda a garantir que nada importante seja esquecido, pois define claramente todo o trabalho necessário para atingir os objetivos do projeto”.

A estrutura do nosso projeto foi organizada em cinco grandes fases: Iniciação, Planejamento, Análise e Design, Desenvolvimento e Testes, e Encerramento. Começamos pela iniciação, onde definimos a missão, visão e valores do sistema, identificamos stakeholders e levantamos os requisitos iniciais, formalizando tudo no TAP, conforme Figura 1. Em seguida, no planejamento, elaboramos o cronograma, definimos os recursos e ações com apoio de ferramentas como SWOT e 5W2H, além da criação da EAP em si.

Figura 1 - TAP



Fonte: Elaborado pelos autores

Na etapa de Análise e Design, produzimos rascunhos visuais e os convertimos em modelos mais estruturados, incluindo diagramas BPMN e casos de uso. A fase de Desenvolvimento e Testes está dividida entre backend e frontend, com foco na integração das funcionalidades, testes de usabilidade, desempenho e correção de bugs. Por fim, na fase de Encerramento, finalizamos a documentação, apresentamos os resultados e avaliamos o desempenho do projeto.

Optamos por essa estrutura para garantir um fluxo claro e organizado, desde a concepção até a entrega do sistema, facilitando tanto o acompanhamento quanto a justificativa metodológica do projeto.

2. Viabilidade do Projeto

A análise de viabilidade do projeto *Eternus Coliseum* demonstra que sua implementação é tecnicamente possível, economicamente acessível e estrategicamente alinhada ao cenário atual de competições digitais. O uso de ferramentas multiplataforma, como React Native e Firebase, reduz os custos de desenvolvimento e manutenção, permitindo que o produto evolua com menor dependência de infraestrutura robusta. Além disso, a crescente demanda por soluções de gerenciamento de torneios, impulsionada pelo avanço dos e-sports e dos eventos estudantis, reforça o potencial de adoção do aplicativo.

Segundo Osterwalder e Pigneur (2010):

“modelos de negócio bem estruturados geram valor ao conectar necessidades reais do público com soluções claras e escaláveis — exatamente a abordagem aplicada no Canvas do projeto.”

A análise SWOT também contribui para validar a viabilidade do sistema ao demonstrar que os pontos fortes, como acessibilidade e automação de processos, superam as fraquezas identificadas, desde que sejam acompanhados por melhorias contínuas. Dessa forma, o projeto se mostra viável, pois combina baixo custo inicial, alta demanda de mercado e potencial de expansão através de parcerias estratégicas e planos de assinatura.

2.1 Canvas de Negócio

O Business Model Canvas (BMC) é uma ferramenta visual de planejamento estratégico criada por Alexander Osterwalder e Yves Pigneur (2010), que permite descrever, desenhar e analisar modelos de negócios de forma simples e integrada.

Segundo Osterwalder e Pigneur:

“O BMC facilita a compreensão de como uma organização cria, entrega e captura valor. Essa metodologia é amplamente utilizada para estruturar ideias empreendedoras e startups, por ser prática e dinâmica.”

2.2 Análise dos Blocos do BMC

O público-alvo engloba organizadores de torneios, jogadores, equipes e espectadores de e-sports e esportes físicos. Esses segmentos são essenciais para garantir o engajamento e o uso contínuo da plataforma.

A proposta é oferecer uma plataforma prática e automatizada para criação, gerenciamento e acompanhamento de campeonatos. O diferencial está na transparência e engajamento, facilitando a organização e a comunicação entre os participantes.

Os principais canais de contato e entrega de valor incluem o aplicativo mobile (Android/iOS), redes sociais e parcerias com instituições esportivas, permitindo ampla divulgação e acesso facilitado ao serviço.

O relacionamento é mantido por meio de suporte no aplicativo, FAQ, notificações push e atualizações constantes. Isso garante uma comunicação eficiente e mantém o usuário engajado com o produto.

As receitas vêm de planos de assinatura premium, anúncios internos e parcerias comerciais. Esses modelos permitem diversificação financeira e sustentabilidade a longo prazo.

Os recursos principais são a equipe de desenvolvimento, servidores (Firebase), design UI/UX e infraestrutura tecnológica, com expansão futura para armazenamento em nuvem.

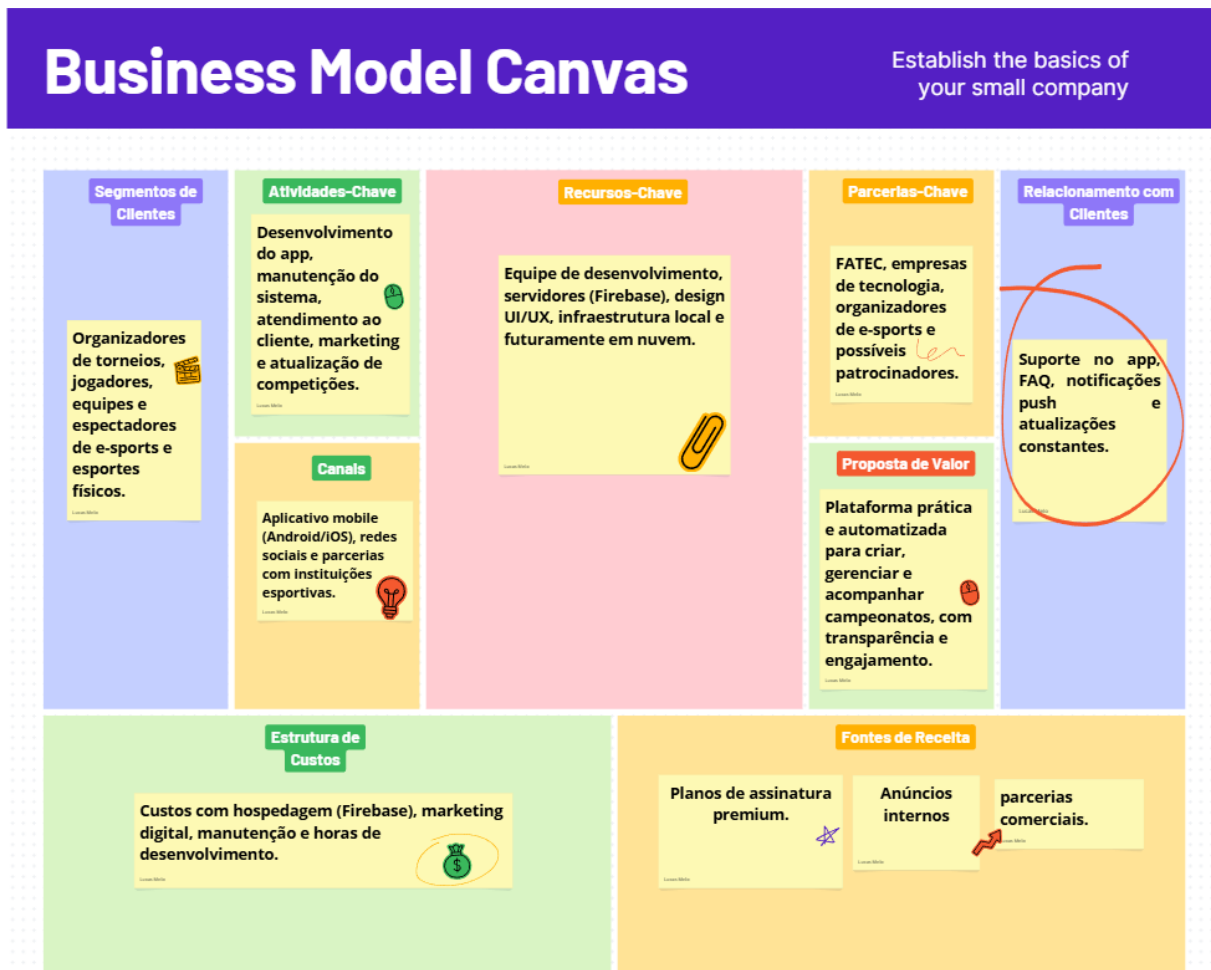
As atividades essenciais incluem desenvolvimento do aplicativo, manutenção do sistema, atendimento ao cliente, marketing digital e atualização de competições para manter o produto relevante e funcional.

As parcerias estratégicas envolvem a FATEC, empresas de tecnologia, organizadores de e-sports e possíveis patrocinadores. Essas alianças são fundamentais para ampliar o alcance e garantir suporte técnico e financeiro.

Os principais custos incluem hospedagem, marketing digital, manutenção e horas de desenvolvimento. Esses investimentos sustentam a operação e a qualidade do serviço.

Podemos ver na Figura 2 o modelo de Canvas elaborado para o desenvolvimento do negócio.

Figura 2 - Canvas

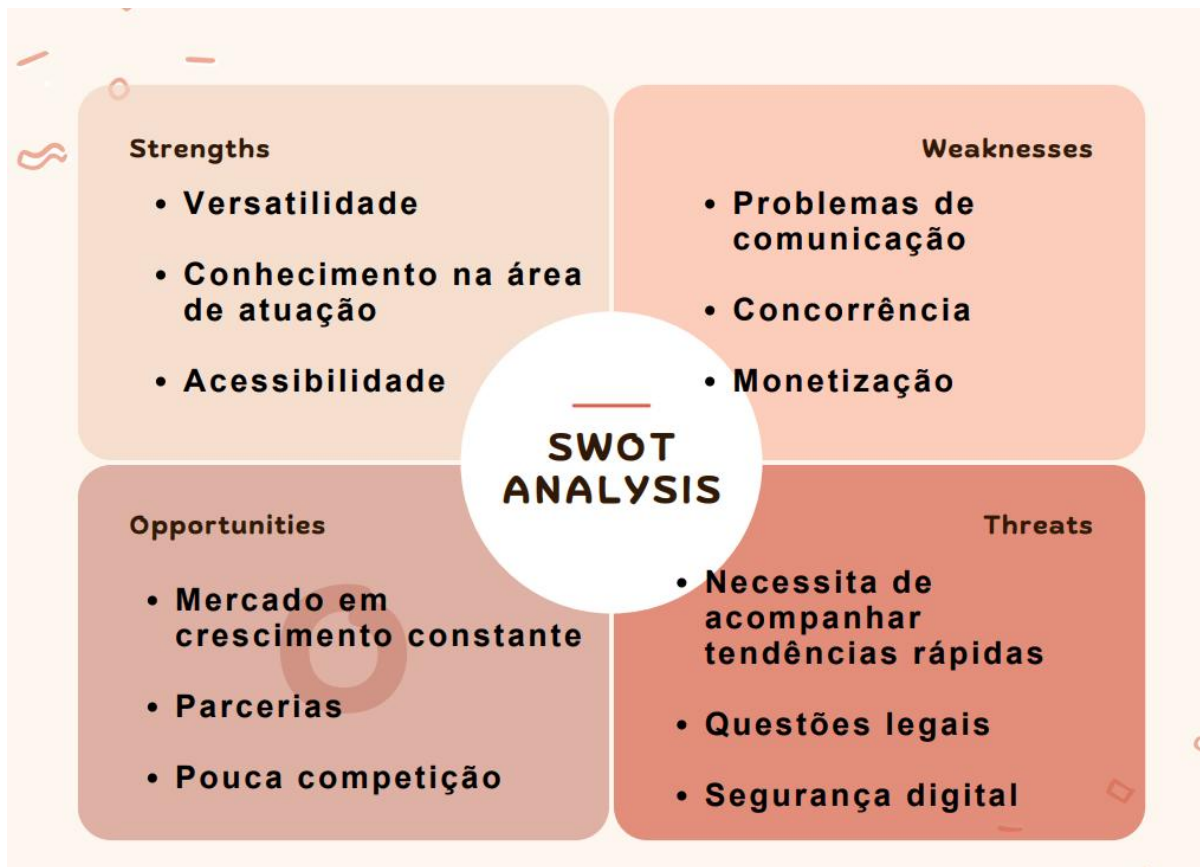


Fonte: Elaborado pelos autores

2.3 Matriz SWOT

A matriz SWOT, também chamada de análise SWOT, é uma ferramenta de planejamento estratégico utilizada para avaliar os pontos fortes (*Strengths*), pontos fracos (*Weaknesses*), oportunidades (*Opportunities*) e ameaças (*Threats*) de uma organização, projeto ou ideia. Essa análise possibilita compreender melhor a situação interna e o ambiente externo, auxiliando na tomada de decisões mais assertivas. Sua importância está em permitir uma visão clara da posição atual do negócio, identificar estratégias para potencializar os pontos fortes e reduzir as fraquezas, antecipar-se a ameaças e aproveitar oportunidades de mercado, servindo assim como base para o planejamento estratégico e para a definição de ações eficazes.

Figura 3 - SWOT



Fonte: Elaborado pelos autores

2.3.1 Fraquezas identificadas

Problemas de comunicação: Falhas na troca de informações entre membros da equipe, o que pode gerar retrabalho, atrasos, muitas vezes devido à falta de tempo.

Concorrência: embora normalmente considerada uma ameaça externa, aqui é tratada como uma fraqueza interna, identificamos que nosso projeto ainda não possui diferenciais fortes o suficiente para competir de forma eficiente no mercado.

Monetização: Dificuldade em transformar o produto ou serviço em fonte de receita sustentável, seja por falta de um modelo de negócio bem definido, estratégias de precificação inadequadas ou baixo alcance do público-alvo.

2.4 Plano de Ação 5W2H do Projeto

O método 5W2H é uma ferramenta de gestão utilizada para planejar, organizar e executar ações de forma clara e objetiva, garantindo que todas as etapas de um projeto sejam bem compreendidas por todos os envolvidos. O nome vem das iniciais

em inglês das perguntas que ele responde: What (o que será feito), Why (por que será feito), Where (onde será feito), When (quando será feito), Who (quem fará), How (como será feito) e How much (quanto custará).

Sua importância está em facilitar o planejamento e o acompanhamento de atividades, evitando falhas de comunicação e promovendo a eficiência na execução das tarefas. Além disso, o 5W2H ajuda na definição de responsabilidades e na mensuração de custos e prazos, tornando-se essencial em processos de melhoria contínua e gestão de projetos.

De acordo com o SEBRAE (2023):

“O 5W2H é uma metodologia prática que contribui para transformar ideias em ações concretas, promovendo a clareza nas etapas de implementação de planos e estratégias.”

Figura 4 – 5W2H

O que?	Por que?	Onde?	Quando?	Quem?	Como?	Quanto?	Status
Planejamento de um aplicativo mobile (Android) para criação e gerenciamento de campeonatos.	Atender à demanda crescente por torneios escolares, universitários e amadores que não têm ferramentas práticas.	Aplicativo será disponibilizado no Google Play e App Store. Mobile	Fase 1 (0–3 meses): Início do projeto.	Alunos responsáveis pelo projeto (Lucas e Bruno)	Planejamento, levantamento de requisitos e documentação inicial.	Calculos a serem feitos dezembro.	Realizado
Funcionalidades: cadastro de times/jogadores, geração de chaves, tabelas e rankings.	Facilitar a organização de campeonatos, reduzindo o trabalho manual e erros.	Banco de dados hospedado remotamente.	Fase 2 (3–6 meses): Fase de desenvolvimento	Alunos responsáveis pelo projeto (Lucas e Bruno)	MVP com criação de campeonatos, cadastro de times e partidas.	Calculos a serem feitos dezembro.	Realizado
Estabelecer parcerias com empresas de games, times de esportes e com a faculdade Fatec.	Criar uma solução intuitiva que pode competir com apps internacionais, mas com foco no público local/regional.	Inicialmente na Fatec Franca.	6–12 meses: Finalização da documentação e fase de testes/implementação.	Alunos responsáveis pelo projeto (Lucas e Bruno)	Por meio de reuniões estratégicas e negociações.	Calculos a serem feitos dezembro.	Em processo

Fonte: Elaborado pelos autores

2.4.1 – Plano de ação

O plano de ação apresentado na tabela 5W2H demonstra o planejamento para o desenvolvimento de nosso aplicativo mobile voltado à criação e gerenciamento de campeonatos. Ele está dividido em três fases principais, cada uma com objetivos claros, responsáveis definidos e status de acompanhamento.

Na primeira etapa, o foco é o planejamento e levantamento de requisitos, um ponto essencial para garantir que o desenvolvimento ocorra com base em informações sólidas e bem definidas. O status “Realizado” mostra que nosso grupo já cumpriu essa etapa inicial com sucesso.

Na segunda fase, voltada para o desenvolvimento do MVP (Produto Mínimo Viável), o plano detalha as funcionalidades principais — como cadastro de times, jogadores e geração de tabelas —, reforçando a preocupação com a usabilidade e funcionalidade básica do aplicativo. Essa fase também está marcada como realizada, o que demonstra nosso avanço consistente.

Por fim, a terceira etapa trata da expansão do projeto por meio de parcerias estratégicas, tanto com empresas do setor quanto com instituições como a Fatec. O status “Em processo” revela que essa fase está em andamento, o que é natural, pois envolve negociações e articulações externas, geralmente mais demoradas.

3 Levantamento de Requisitos

Os requisitos funcionais descrevem as funcionalidades específicas que um sistema, software ou produto deve oferecer para satisfazer as necessidades dos usuários. São declarações formais que definem as operações, serviços ou ações que o sistema deve executar, incluindo a interação entre o usuário e o sistema. Esses requisitos representam as principais tarefas que o sistema deve realizar, com especificações claras sobre entradas, condições de operação, resultados esperados e comportamentos previstos em diferentes situações.

3.1 Elicitação e especificação dos Requisitos

A elicitação de requisitos é uma etapa essencial no processo de desenvolvimento de sistemas, pois é responsável por identificar, compreender e documentar as necessidades e expectativas dos usuários e demais partes interessadas. Sua importância reside no fato de que decisões tomadas com base em requisitos mal compreendidos ou incompletos podem comprometer a qualidade final do produto, gerar retrabalho, aumento de custos e atrasos no cronograma. Além disso, a elicitação contribui para o alinhamento entre a equipe de desenvolvimento e os stakeholders, assegurando que todos compartilhem uma visão comum sobre o que o sistema deve realizar. Ela também permite detectar conflitos, ambiguidades e lacunas nos requisitos desde o início, o que facilita a definição de soluções mais adequadas e evita problemas futuros.

Segundo Sommerville (2011):

“A elicitação é “uma das atividades mais críticas no processo de engenharia de requisitos”, e envolve não apenas a coleta de informações, mas também a análise, negociação e validação dessas informações para garantir que o sistema desenvolvido realmente atenda às necessidades dos usuários.”

O levantamento de requisitos foi conduzido por meio de diálogos com alunos da instituição e de outras instituições, complementado por análises de aplicativos semelhantes e um estudo aprofundado na área de atuação do aplicativo. Essa abordagem permitiu uma compreensão abrangente das necessidades e expectativas dos usuários, culminando na elicitação de requisitos funcionais e não funcionais. Entre os principais requisitos identificados, destacam-se: a necessidade de um acompanhamento dinâmico das tabelas pós-competições, a possibilidade de solicitar a abertura de novas competições, a função de atuar como organizador, bem como demandas por maior segurança e compatibilidade do sistema.

As informações obtidas durante a fase de elicitação foram fundamentadas por meio de diálogos com os usuários, nos quais foram aplicadas perguntas específicas com o objetivo de compreender as necessidades, expectativas e problemas enfrentados em contextos similares ao do sistema proposto. As perguntas formuladas, bem como as respectivas respostas fornecidas pelos participantes, encontram-se organizadas no Apêndice A, ao final deste trabalho, com o intuito de garantir a transparência do processo e oferecer uma base documental para as decisões tomadas ao longo do desenvolvimento.

3.2 BPMN

A Business Process Model and Notation (BPMN) é uma notação gráfica padronizada para modelagem de processos de negócio. Ela fornece uma linguagem comum e intuitiva, compreensível por todos os stakeholders de um projeto, desde os analistas de negócio que desenham os fluxos iniciais, até os desenvolvedores responsáveis pela implementação da tecnologia que irá suportar os processos, e finalmente, pelos gestores que irão monitorar e gerenciar esses processos.

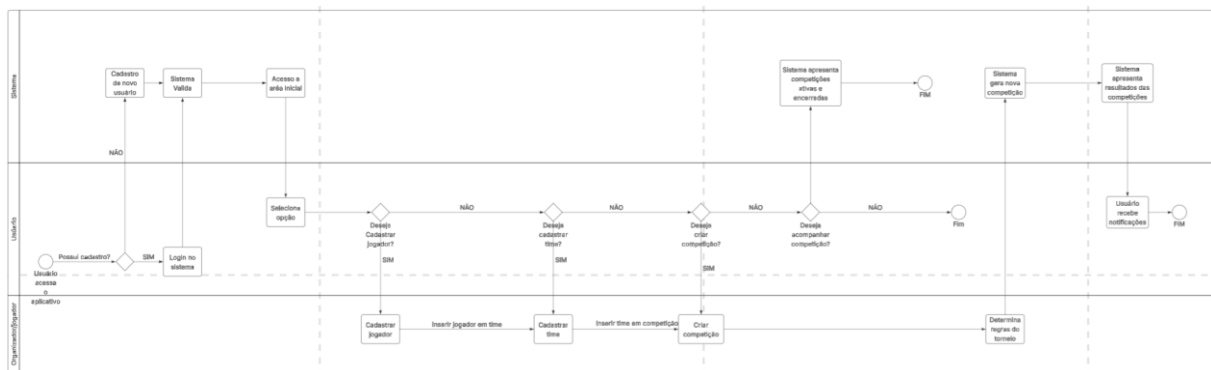
De acordo com Dumas et al. (2018, p. 17):

“O BPMN tornou-se o “padrão 'de facto' para a modelagem de processos”, sendo amplamente adotada globalmente devido à sua capacidade de representar de forma clara a sequência de atividades, eventos, decisões e interações entre diferentes participantes em um processo.”

3.2.1 - Para que serve um BPMN

O principal objetivo da BPMN é funcionar como uma ferramenta visual que facilita a comunicação e o registro de como os trabalhos são realizados em uma organização. Ela ajuda as pessoas a entenderem, analisarem e melhorarem esses processos. De forma prática, ela é útil para: registrar de maneira clara e organizada como as atividades são feitas, criando uma referência para todos; examinar o que está funcionando bem ou não, identificando pontos de lentidão, repetições desnecessárias e onde a automação pode ajudar; e, por fim, para planejar e compartilhar uma versão melhorada do fluxo de trabalho. Além disso, a BPMN vai além de ser apenas um desenho. Ela serve como um guia preciso para orientar a criação de sistemas que darão suporte à execução desses processos, facilitando a automação e garantindo que as áreas de negócio e de tecnologia trabalhem alinhadas em busca do mesmo objetivo.

Figura 5 - BPMN



Fonte: Elaborado pelos autores

Devido ao tamanho do diagrama completo, o BPMN do BelApp pode ser acesso pelo link:

https://lucid.app/lucidchart/b52f7dac-b7df-4d8c-8afd-bda6750994cf/edit?viewport_loc=-1427%2C-1501%2C7754%2C3687%2C18%2C45&invitationId=inv_55a9d342-d42d-47a3-8d75-a7b25b5f9f90

3.2.2 Importância de um BPMN para o projeto

A criação do diagrama de fluxo foi um passo muito importante para o nosso projeto, ajudando a organizar e planejar todo o sistema. Ele foi essencial para dar clareza, permitindo que todos no time desde quem programa até quem vai usar o aplicativo entendessem perfeitamente como o sistema deveria funcionar na prática.

Com esse mapa visual, foi possível enxergar onde poderíamos melhorar e automatizar tarefas, como a criação das chaves dos torneios e o envio de avisos automáticos, tornando tudo mais eficiente e com menos chance de erro. O diagrama também serviu como um guia bem estruturado para o desenvolvimento, garantindo que o que foi planejado fosse corretamente transformado em funcionalidades reais no aplicativo. Por fim, ele mostrou uma conexão direta com as principais necessidades do sistema, representando de forma visual funcionalidades como o controle de participantes, a criação de campeonatos e a atualização instantânea dos resultados, assegurando que o produto atenda exatamente ao que foi imaginado desde o início.

3.3 Requisitos Funcionais

Os requisitos funcionais descrevem as funcionalidades específicas que um sistema deve ser capaz de executar. Eles indicam o que o sistema deve fazer, ou seja, as ações, comportamentos e respostas esperadas diante de determinadas entradas ou interações do usuário. Esses requisitos estão diretamente ligados às necessidades do usuário e aos objetivos do sistema, como permitir login, gerar relatórios ou cadastrar informações.

Quadro 1 – Requisitos Funcionais do Sistema

RF001 – Menu inicial apresentando o projeto	Categoria: ☐ Oculto ☒ Evidente	Prioridade: ☒ Altíssima ☐ Alta ☐ Média ☐ Baixa
Descrição: O sistema deve apresentar uma tela inicial com informações introdutórias sobre o projeto e sua proposta, facilitando a navegação inicial do usuário.		
RF002 – Sistema de login (incluindo login com Google)	Categoria: ☐ Oculto ☒ Evidente	Prioridade: ☒ Altíssima ☐ Alta ☐ Média ☐ Baixa
Descrição: O sistema deve permitir que o usuário acesse sua conta por meio de login tradicional (usuário e senha) e login via conta Google.		
RF003 – Cadastro e autenticação de usuários	Categoria: ☐ Oculto ☒ Evidente	Prioridade: ☒ Altíssima ☐ Alta ☐ Média ☐ Baixa
Descrição: O sistema deve permitir que novos usuários realizem cadastro, e que usuários já registrados se autentiquem para acessar suas funcionalidades.		

RF004 – Exibir mensagens de erro ou sucesso	Categoria: ☐ Oculto ☒ Evidente	Prioridade: ☐ Altíssima ☒ Alta ☐ Média ☐ Baixa
Descrição: O sistema deve apresentar mensagens visuais ao usuário indicando o sucesso ou falha de ações executadas, como login, cadastro, ou criação de eventos.		
RF005 – Gerenciamento de participantes	Categoria: ☐ Oculto ☒ Evidente	Prioridade: ☒ Altíssima ☐ Alta ☐ Média ☐ Baixa
Descrição: O sistema deve permitir o gerenciamento de participantes de campeonatos, incluindo cadastro, edição e exclusão de perfis participantes.		
RF006 – Gerenciamento de equipes	Categoria: ☐ Oculto ☒ Evidente	Prioridade: ☒ Altíssima ☐ Alta ☐ Média ☐ Baixa
Descrição: O sistema deve permitir a criação, edição e gerenciamento de equipes participantes dos campeonatos.		
RF007 – Inclusão de categorias (MOBA, futebol etc.)	Categoria: ☐ Oculto ☒ Evidente	Prioridade: ☒ Altíssima ☐ Alta ☐ Média ☐ Baixa
Descrição: O sistema deve permitir a criação e associação de categorias de campeonatos, como MOBA, futebol, entre outros.		
RF008 – Inclusão de variáveis nas competições	Categoria: ☐ Oculto ☒ Evidente	Prioridade: ☒ Altíssima ☐ Alta ☐ Média ☐ Baixa
Descrição: O sistema deve permitir configurar regras, tempo de partidas, quantidade de jogos e políticas específicas para os campeonatos.		
RF009 – Criação, edição e exclusão de campeonatos	Categoria: ☐ Oculto ☒ Evidente	Prioridade: ☒ Altíssima ☐ Alta ☐ Média ☐ Baixa
Descrição: O sistema deve permitir a criação e o gerenciamento de campeonatos com diversos formatos (mata-mata, suíço, pontos corridos, etc.).		
RF010 – Agendamento de partidas	Categoria: ☐ Oculto ☒ Evidente	Prioridade: ☐ Altíssima ☒ Alta ☐ Média ☐ Baixa
Descrição: O sistema deve permitir o agendamento das partidas com data, hora e local (se aplicável).		

RF011 – Tabela em tempo real	Categoria: ☐ Oculto ☒ Evidente	Prioridade: ☐ Altíssima ☒ Alta ☐ Média ☐ Baixa
Descrição: O sistema deve atualizar e exibir automaticamente a tabela de pontuações conforme os resultados forem inseridos.		
RF012 – Notificações	Categoria: ☐ Oculto ☒ Evidente	Prioridade: ☐ Altíssima ☒ Alta ☐ Média ☐ Baixa
Descrição: O sistema deve enviar notificações aos usuários sobre partidas, resultados, agendamentos ou mensagens administrativas.		
RF013 – Relatórios e estatísticas	Categoria: ☐ Oculto ☒ Evidente	Prioridade: ☐ Altíssima ☒ Alta ☐ Média ☐ Baixa
Descrição: O sistema deve gerar relatórios com estatísticas detalhadas dos campeonatos e desempenho das equipes.		
RF014 – Personalização de perfil	Categoria: ☐ Oculto ☒ Evidente	Prioridade: ☐ Altíssima ☐ Alta ☒ Média ☐ Baixa
Descrição: O sistema deve permitir que os usuários personalizem seus perfis, como avatar, nome de exibição e biografia.		
RF015 – Integração com redes sociais	Categoria: ☐ Oculto ☒ Evidente	Prioridade: ☐ Altíssima ☐ Alta ☒ Média ☐ Baixa
Descrição: O sistema deve oferecer opções de compartilhamento e login via redes sociais como Facebook e Instagram.		
RF016 – Feedback do usuário	Categoria: ☐ Oculto ☒ Evidente	Prioridade: ☐ Altíssima ☐ Alta ☒ Média ☐ Baixa
Descrição: O sistema deve incluir um canal para que os usuários possam enviar comentários, sugestões ou reportar problemas.		
RF017 – Suporte ao usuário	Categoria: ☐ Oculto ☒ Evidente	Prioridade: ☐ Altíssima ☐ Alta ☒ Média ☐ Baixa
Descrição: O sistema deve oferecer uma seção de suporte com perguntas frequentes e canal de contato com administradores.		

RF018 – Sorteio de grupos e chaves	Categoria: ☐ Oculto ☒ Evidente	Prioridade: ☒ Altíssima ☐ Alta ☐ Média ☐ Baixa
Descrição: O sistema deve realizar automaticamente o sorteio de grupos e chaves das competições, baseado nas configurações do campeonato.		
RF019 – Integração com ferramentas de banco de dados.	Categoria: ☐ Oculto ☒ Evidente	Prioridade: ☒ Altíssima ☐ Alta ☐ Média ☐ Baixa
Descrição: O sistema deve armazenar todos os dados em um banco local, podendo ser acessado via ferramentas, garantindo desempenho.		
RF020 – Suporte a streaming	Categoria: ☐ Oculto ☒ Evidente	Prioridade: ☐ Altíssima ☐ Alta ☒ Média ☐ Baixa
Descrição: O sistema deve permitir a vinculação com plataformas de streaming para transmissão das partidas.		
RF021 – Histórico de partidas	Categoria: ☐ Oculto ☒ Evidente	Prioridade: ☐ Altíssima ☐ Alta ☒ Média ☐ Baixa
Descrição: O sistema deve armazenar o histórico de todas as partidas realizadas por usuário, equipe ou campeonato.		
RF022 – Ranking de desempenho	Categoria: ☐ Oculto ☒ Evidente	Prioridade: ☐ Altíssima ☐ Alta ☒ Média ☐ Baixa
Descrição: O sistema deve gerar rankings automáticos com base no desempenho dos jogadores e equipes.		
RF023 – Gerenciamento de arbitragem	Categoria: ☐ Oculto ☒ Evidente	Prioridade: ☐ Altíssima ☐ Alta ☒ Média ☐ Baixa
Descrição: O sistema deve permitir o cadastro e atribuição de árbitros às partidas, com funções específicas durante os jogos.		
RF024 – Sistema de penalidades	Categoria: ☐ Oculto ☒ Evidente	Prioridade: ☐ Altíssima ☒ Alta ☐ Média ☐ Baixa
Descrição: O sistema deve registrar infrações cometidas por jogadores/equipes e aplicar penalidades conforme definido pelo regulamento.		

Fonte: Elaborado pelos autores

O intuito do primeiro requisito é criar uma tela de boas-vindas que apresente o sistema de maneira clara e concisa. A tela inicial deve fornecer aos usuários uma

visão geral do que o sistema oferece, permitindo que eles entendam rapidamente seu propósito e funcionalidades. A ideia é criar uma primeira impressão positiva, destacando os principais recursos e motivando o usuário a continuar navegando.

O segundo requisito visa garantir que os usuários possam acessar o sistema de forma prática e segura, utilizando suas credenciais pessoais ou a autenticação via Google. O objetivo é oferecer uma experiência de login simples e segura, sem que o usuário precise lembrar de mais uma senha. Com o login Google, a integração se torna mais fluida, proporcionando uma forma rápida de acesso sem a necessidade de preenchimento manual de dados.

No terceiro requisito sistema precisa permitir que novos usuários se cadastrem e que os já existentes possam se autenticar facilmente. A intenção deste requisito é garantir que os usuários possam criar uma conta para participar das competições e também acessar suas informações de forma segura. O sistema de autenticação serve para proteger os dados pessoais e garantir que apenas usuários autorizados possam realizar ações dentro da plataforma.

No quarto requisito, o objetivo é manter o usuário informado sobre o resultado de suas ações no sistema. Quando um usuário realiza uma tarefa, como se cadastrar ou enviar um formulário, o sistema precisa confirmar o sucesso da operação ou informar sobre possíveis erros. Isso melhora a interação do usuário com a plataforma, evitando frustrações e esclarecendo qualquer ação que precise ser corrigida.

Neste requisito, o sistema permite que o administrador do sistema possa gerenciar os participantes das competições de maneira organizada. O gerenciamento deve incluir funcionalidades para cadastrar, editar, visualizar e excluir participantes. O intuito aqui é garantir que a base de dados sobre os participantes seja atualizada constantemente e de forma precisa, facilitando a administração do torneio e garantindo que todas as informações estejam corretas.

Já no sexto requisito, o sistema visa permitir a criação e o controle das equipes participantes de forma simples e eficiente. O gerenciamento de equipes inclui a adição de membros, ajustes nas informações e a exclusão de equipes que não estão mais participando. O propósito é facilitar a organização do campeonato, tornando o processo de registro e manutenção das equipes mais ágil e sem erros.

Enquanto no sétimo requisito, este visa garantir que o sistema seja flexível o suficiente para lidar com diferentes tipos de competições. A criação de categorias

como MOBA, futebol, entre outras, permite que os campeonatos sejam organizados de acordo com as especificidades de cada modalidade. O objetivo é oferecer aos organizadores a capacidade de personalizar os torneios, adaptando-os às necessidades de cada tipo de competição.

Neste requisito, o objetivo é permitir que os organizadores configurem regras específicas para cada competição, como o tempo de jogo, número de partidas ou políticas de desempate. A flexibilidade para ajustar essas variáveis assegura que o sistema se adapte às diferentes exigências de cada evento, proporcionando uma experiência personalizada e que atenda às expectativas dos participantes e organizadores.

Já no nono requisito, o sistema visa facilitar a criação e o gerenciamento de campeonatos dentro do sistema. Os organizadores poderão criar campeonatos com diferentes formatos, como mata-mata ou pontos corridos, e fazer ajustes conforme necessário. A edição e exclusão de campeonatos permitem que os eventos sejam ajustados de acordo com o andamento do torneio, garantindo que qualquer erro ou mudança no planejamento possa ser facilmente corrigido.

No décimo requisito, o intuito é permitir que os organizadores do campeonato agendem as partidas com precisão, definindo data, hora e, quando necessário, o local. O agendamento das partidas é fundamental para garantir a organização do torneio, evitando conflitos de horário e proporcionando uma forma eficiente de coordenar a participação das equipes.

Neste requisito, o sistema visa fornecer aos usuários uma tabela de resultados que é atualizada em tempo real, à medida que as partidas são concluídas. O objetivo é que os participantes e espectadores possam acompanhar instantaneamente o progresso da competição, com a tabela refletindo as pontuações e posições das equipes de forma dinâmica.

Já no décimo segundo requisito, esta busca garantir que os usuários sejam mantidos informados sobre eventos importantes, como o início de uma partida ou a atualização dos resultados. As notificações são essenciais para que os participantes e espectadores fiquem sempre atualizados sobre o que está acontecendo, sem a necessidade de acessar o sistema o tempo todo. Elas também ajudam a lembrar de compromissos importantes, como partidas agendadas.

Enquanto no décimo terceiro requisito, o objetivo é fornecer relatórios detalhados e estatísticas sobre as competições e o desempenho dos jogadores ou equipes. Com esses dados, organizadores e participantes poderão analisar o andamento dos campeonatos, identificar áreas de melhoria e visualizar o desempenho das equipes ao longo do torneio. As estatísticas ajudam a tomar decisões informadas e a aprimorar futuros eventos.

Neste requisito, o intuito é permitir que os usuários personalizem suas contas dentro do sistema. Com a possibilidade de adicionar fotos, biografias e outros detalhes, o perfil se torna uma extensão da identidade do usuário, tornando a experiência mais pessoal e envolvente. Essa funcionalidade também ajuda a criar um ambiente mais dinâmico e social, onde os participantes podem se expressar.

Já no quinze, este requisito visa expandir a interação dos usuários com o sistema, permitindo que compartilhem suas conquistas ou eventos nas redes sociais, como Facebook e Instagram. Além disso, a opção de login através dessas plataformas facilita o acesso ao sistema e torna a experiência mais integrada com o cotidiano dos usuários, que frequentemente estão conectados a essas redes.

No décimo sexto requisito, o objetivo é proporcionar um canal de comunicação entre os usuários e os administradores do sistema. Através do feedback, os usuários podem sugerir melhorias, reportar problemas ou compartilhar suas experiências. Isso ajuda os administradores a entender melhor as necessidades dos usuários e a melhorar o sistema continuamente, com base nas opiniões da comunidade.

Enquanto no décimo sétimo requisito, busca oferecer um suporte eficiente aos usuários do sistema. Com uma seção de perguntas frequentes (FAQ) e a possibilidade de contato direto com a equipe de suporte, os usuários terão acesso a soluções rápidas para problemas comuns e poderão resolver questões mais complexas com a ajuda de um atendente. O objetivo é minimizar o impacto de eventuais problemas técnicos, garantindo uma experiência tranquila.

Neste requisito, o objetivo é permitir que o sistema realize sorteios de grupos e chaves de maneira automática e imparcial. Isso elimina a necessidade de intervenção manual, tornando o processo mais rápido e evitando qualquer possibilidade de favorecimento. Os sorteios são fundamentais para organizar as competições de forma justa e garantir que todos os participantes tenham uma chance igual.

Já no dezenove, o requisito visa garantir que os dados do sistema sejam armazenados de maneira segura e eficiente, utilizando um banco local. A ideia é que inicialmente o armazenamento seja local e que futuramente haja a integração com alguma ferramenta.

Enquanto no vigésimo requisito, o objetivo é permitir que as partidas sejam transmitidas ao vivo para os usuários através de plataformas de streaming. Com a integração ao streaming, os participantes e espectadores podem acompanhar os jogos em tempo real, aumentando o engajamento e a visibilidade do evento. Isso também permite que o sistema alcance uma audiência mais ampla e interaja com pessoas fora do ambiente direto da competição.

Neste requisito, o objetivo é permitir que os dados das partidas sejam armazenados para referência futura. O histórico de partidas ajuda os usuários a revisitem resultados passados, analisar o desempenho e manter um registro das competições. Esse recurso é importante tanto para os participantes, que podem verificar seu progresso, quanto para os administradores, que podem acompanhar o andamento do torneio.

Já no vigésimo segundo requisito, o sistema visa criar rankings dinâmicos baseados no desempenho das equipes ou jogadores. Os rankings são atualizados automaticamente conforme os resultados são inseridos, oferecendo uma maneira visual de acompanhar o progresso dos participantes no campeonato. O objetivo é promover a competitividade e transparência, além de incentivar os jogadores a melhorarem seu desempenho ao longo do torneio.

Neste requisito, a finalidade é garantir que cada partida tenha um árbitro designado para supervisioná-la e garantir que as regras sejam seguidas corretamente. O gerenciamento de arbitragem inclui o cadastro de árbitros, a atribuição das partidas e a organização das suas funções durante o evento. A presença de árbitros bem-organizados é essencial para a integridade da competição e para assegurar que todos os participantes sigam as normas estabelecidas.

Já no último requisito, o objetivo é garantir que qualquer infração cometida durante as competições seja registrada e penalizada de acordo com as regras. O sistema de penalidades assegura que o comportamento dos jogadores e equipes esteja em conformidade com o regulamento, proporcionando um ambiente de

competição justo e equilibrado. Ele ajuda a manter a ordem e evita que ações indesejadas afetem o andamento do campeonato.

3.4 Requisitos Não Funcionais

Requisitos não funcionais são especificações que definem os atributos de qualidade de um sistema, descrevendo como ele deve se comportar em vez de quais funcionalidades ele deve executar. Esses requisitos dizem respeito a aspectos como desempenho, segurança, usabilidade, confiabilidade, escalabilidade, portabilidade, entre outros.

Diferentemente dos requisitos funcionais, que tratam das operações diretas do sistema, os não funcionais estabelecem critérios que qualificam o funcionamento dessas operações.

Eles são essenciais para garantir que o sistema atenda às expectativas do usuário não apenas em termos do que faz, mas de como faz, influenciando diretamente a experiência do usuário e a aceitação do produto final. Por exemplo, a exigência de que um sistema responda a comandos em menos de dois segundos é um requisito não funcional relacionado ao desempenho.

Já a necessidade de o sistema estar disponível 99,9% do tempo se refere à confiabilidade. Esses requisitos são fundamentais no processo de desenvolvimento, pois impactam diretamente na arquitetura do software, nas decisões de implementação e na escolha das tecnologias a serem utilizadas.

Segundo Sommerville (2011):

“Os requisitos não funcionais são tão importantes quanto os funcionais, pois “um sistema sem requisitos não funcionais definidos pode não satisfazer as necessidades do usuário, mesmo que todas as funcionalidades estejam corretamente implementadas”.

Quadro 2 – Requisitos não Funcionais do Sistema

RNF001 Desempenho	- O sistema deve ser capaz de suportar uma grande quantidade de usuários simultâneos sem degradação significativa do desempenho.	Qualidade do sistema	() Desejável (X) Obrigatório	(X) Permanente () Transitório
RNF002 - Segurança	O sistema deve ter que todas as informações do usuário sejam armazenadas de forma segura para proteger a privacidade do usuário.	Qualidade do sistema	() Desejável (X) Obrigatório	(X) Permanente () Transitório

RNF003-Disponibilidade	O sistema deve estar disponível 24/7, com tempo de inatividade mínimo para manutenção e atualizações.	Qualidade do sistema	() Desejável (X) Obrigatório	(X) Permanente () Transitório
RNF004-Compatibilidade	O sistema deve ser compatível com várias plataformas (Android, web, desktop) e versões de sistemas operacionais.	Qualidade do sistema	() Desejável (X) Obrigatório	(X) Permanente () Transitório
RNF005-Usabilidade	O sistema deve ser fácil de usar, com uma interface de usuário intuitiva e amigável.	Qualidade do sistema	(X) Desejável () Obrigatório	(X) Permanente () Transitório
RNF006-Manutenção	O sistema deve passar por manutenções regulares, para garantir seu funcionamento contínuo	Qualidade do sistema	() Desejável (X) Obrigatório	(X) Permanente () Transitório
RNF007-Interoperabilidade	O aplicativo deve ser capaz de operar e trocar informações com outros sistemas ou aplicativos. Isso é desejável para permitir uma integração mais suave com outros sistemas que a faculdade possa usar.	Qualidade do sistema	(X) Desejável () Obrigatório	(X) Permanente () Transitório
RNF008-Acessibilidade	O aplicativo deve ser acessível para usuários com deficiências, incluindo aqueles com deficiências visuais, auditivas, motoras ou cognitivas. Isso é desejável para garantir que o aplicativo possa ser usado por todos os usuários.	Qualidade do sistema	(X) Desejável () Obrigatório	(X) Permanente () Transitório

Fonte: Elaborado pelos autores.

3.5 CASOS DE USO

Um caso de uso é uma técnica utilizada na engenharia de software para descrever as interações entre um sistema e seus usuários (atores), com o objetivo de capturar os requisitos funcionais de forma clara e compreensível.

Segundo Pressman (2016, p. 161), “um caso de uso define um conjunto de ações que produzem um resultado observável e de valor para um ator específico do sistema.”

Os casos de uso são ferramentas essenciais na modelagem de requisitos, pois permitem representar o comportamento do sistema sob a perspectiva do usuário, facilitando a comunicação entre analistas, desenvolvedores e clientes.

Em outras palavras, um caso de uso descreve “quem faz o quê com o sistema”, servindo como guia para compreender e documentar as necessidades do usuário.

3.5.1 Importância dos Casos de Uso

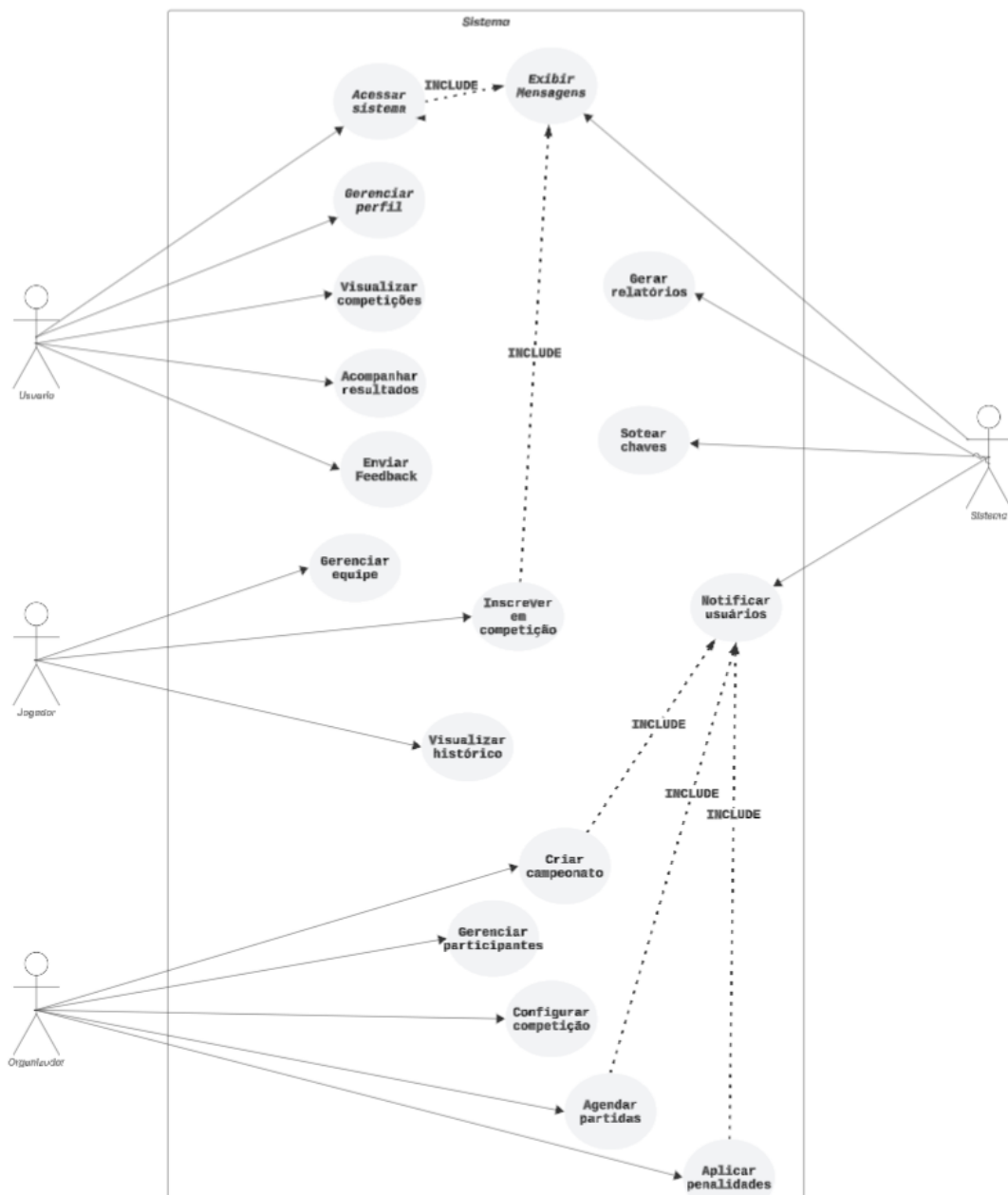
A principal importância dos casos de uso está em facilitar o entendimento e a validação dos requisitos do sistema antes do início do desenvolvimento.

Segundo Jacobson, Booch e Rumbaugh (1999):

“Os casos de uso são uma maneira eficaz de capturar os requisitos funcionais de um sistema, permitindo que analistas e usuários compartilhem uma visão comum sobre o comportamento esperado.”

O uso dessa técnica contribui para uma comunicação eficiente entre as partes envolvidas no projeto, assegurando que o sistema atenda aos objetivos do negócio.

Figura 6 – Casos de Uso



Fonte: Elaborado pelos autores

Quadro 3 – Casos de Uso

Caso de Uso – Acessar Sistema	
ID	UC 001
Descrição	Permite ao usuário fazer login ou cadastrar-se no sistema
Ator Primário	Usuário
Pré-condição	Aplicativo instalado e executando
Cenário Principal	1. Usuário inicia o aplicativo 2. Sistema exibe tela de login/cadastro 3. Usuário informa credenciais ou cria conta 4. Sistema valida dados 5. Sistema concede acesso à área principal
Pós-condição	Usuário autenticado no sistema
Cenário Alternativo	

Caso de Uso – Gerenciar Perfil	
ID	UC 002
Descrição	Permite ao usuário personalizar seu perfil
Ator Primário	Usuário
Pré-condição	Usuário autenticado
Cenário Principal	1. Usuário acessa "Meu Perfil" 2. Sistema exibe dados atuais 3. Usuário edita informações 4. Sistema salva alterações 5. Sistema confirma atualização
Pós-condição	Perfil do usuário atualizado
Cenário Alternativo	

Caso de Uso – Visualizar Competições	
ID	UC 003
Descrição	Permite visualizar competições disponíveis
Ator Primário	Usuário
Pré-condição	Usuário autenticado
Cenário Principal	1. Usuário acessa "Competições" 2. Sistema lista competições ativas 3. Usuário seleciona competição 4. Sistema exibe detalhes 5. Usuário navega entre abas
Pós-condição	Competição visualizada
Cenário Alternativo	

Caso de Uso – Acompanhar Resultados	
ID	UC 004
Descrição	Permite acompanhar resultados em tempo real
Ator Primário	Usuário
Pré-condição	Competição em andamento
Cenário Principal	1. Usuário acessa competição 2. Sistema exibe tabela atualizada 3. Sistema atualiza automaticamente 4. Usuário visualiza resultados 5. Sistema mostra estatísticas

Pós-condição	Resultados acompanhados
Cenário Alternativo	

Caso de Uso – Enviar Feedback	
ID	UC 005
Descrição	Permite enviar feedback sobre o sistema
Ator Primário	Usuário
Pré-condição	Usuário autenticado
Cenário Principal	1. Usuário acessa "Feedback" 2. Sistema exibe formulário 3. Usuário descreve feedback 4. Sistema envia para administração 5. Sistema confirma recebimento
Pós-condição	Feedback registrado
Cenário Alternativo	

Caso de Uso – Gerenciar Equipe	
ID	UC 006
Descrição	Permite gerenciar equipe própria
Ator Primário	Jogador
Pré-condição	Participante autenticado
Cenário Principal	1. Participante acessa "Minha Equipe" 2. Sistema exibe dados da equipe 3. Participante edita informações 4. Participante gerencia membros 5. Sistema salva alterações
Pós-condição	Equipe atualizada
Cenário Alternativo	

Caso de Uso – Inscrever em Competição	
ID	UC 007
Descrição	Permite inscrever equipe em competição
Ator Primário	Jogador
Pré-condição	Equipe criada e competição aberta
Cenário Principal	1. Participante visualiza competição 2. Seleciona "Inscrever Equipe" 3. Sistema valida elegibilidade 4. Participante confirma inscrição 5. Sistema registra participação
Pós-condição	Equipe inscrita na competição
Cenário Alternativo	

Caso de Uso – Visualizar Histórico	
ID	UC 008
Descrição	Permite consultar histórico de participações
Ator Primário	Jogador
Pré-condição	Jogador com histórico
Cenário Principal	1. Participante acessa "Histórico" 2. Sistema lista competições anteriores

	3. Participante aplica filtros 4. Sistema exibe resultados 5. Participante visualiza detalhes
Pós-condição	Histórico consultado
Cenário Alternativo	

Caso de Uso – Criar Campeonato	
ID	UC 009
Descrição	Permite criar novo campeonato
Ator Primário	Organizador
Pré-condição	Organizador autenticado
Cenário Principal	1. Organizador acessa "Criar Campeonato" 2. Sistema exibe formulário 3. Organizador define parâmetros 4. Sistema valida dados 5. Sistema cria campeonato
Pós-condição	Novo campeonato criado
Cenário Alternativo	

Caso de Uso – Gerenciar Participantes	
ID	UC 010
Descrição	Permite gerenciar participantes do campeonato
Ator Primário	Organizador
Pré-condição	Campeonato criado
Cenário Principal	1. Organizador acessa campeonato 2. Seleciona "Participantes" 3. Sistema lista participantes 4. Organizador gerencia inscrições 5. Sistema atualiza lista
Pós-condição	Participantes gerenciados
Cenário Alternativo	

Caso de Uso – Configurar Competição	
ID	UC 011
Descrição	Permite configurar regras e formato da competição
Ator Primário	Organizador
Pré-condição	Campeonato criado
Cenário Principal	1. Organizador acessa "Configurações" 2. Sistema exibe opções 3. Organizador ajusta parâmetros 4. Sistema valida configurações 5. Sistema salva alterações
Pós-condição	Competição configurada
Cenário Alternativo	

Caso de Uso – Agendar Partidas	
ID	UC 012
Descrição	Permite agendar partidas do campeonato
Ator Primário	Organizador

Pré-condição	Chaves definidas
Cenário Principal	1. Organizador acessa "Partidas" 2. Sistema exibe confrontos 3. Organizador define datas/horários 4. Sistema valida agendamento 5. Sistema confirma agendamento
Pós-condição	UC016 - Notificar Usuários
Cenário Alternativo	

Caso de Uso – Aplicar Penalidades	
ID	UC 013
Descrição	Permite aplicar penalidades a participantes
Ator Primário	Organizador
Pré-condição	Infração identificada
Cenário Principal	1. Organizador identifica infração 2. Acessa "Aplicar Penalidade" 3. Sistema exibe tipos de penalidade 4. Organizador seleciona e aplica 5. Sistema registra penalidade
Pós-condição	Penalidade aplicada
Cenário Alternativo	

Caso de Uso – Sortear Chaves	
ID	UC 014
Descrição	Realiza sorteio automático de chaves
Sistema	Sistema
Pré-condição	Inscrições encerradas
Cenário Principal	1. Sistema detecta fim das inscrições 2. Executa algoritmo de sorteio 3. Gera chaveamento 4. Armazena resultados 5. Notifica organizador
Pós-condição	Chaves sorteadas
Cenário Alternativo	

Caso de Uso – Exibir Mensagens	
ID	UC 015
Descrição	Exibe mensagens de feedback ao usuário
Ator Primário	Sistema
Pré-condição	Ação executada pelo usuário
Cenário Principal	1. Sistema detecta ação do usuário 2. Processa resultado 3. Gera mensagem apropriada 4. Exibe mensagem na interface 5. Aguarda interação do usuário
Pós-condição	Mensagem exibida
Cenário Alternativo	

Caso de Uso – Notificar Usuários	
ID	UC 016

Descrição	Envia notificações push aos usuários
Ator Primário	Sistema
Pré-condição	Evento relevante ocorre
Cenário Principal	1. Sistema detecta evento 2. Identifica usuários afetados 3. Gera notificação personalizada 4. Envia notificação push 5. Registra envio
Pós-condição	Usuários notificados
Cenário Alternativo	

Caso de Uso – Gerar Relatórios	
ID	UC 017
Descrição	Gera relatórios estatísticos do sistema
Ator Primário	Sistema
Pré-condição	Dados disponíveis
Cenário Principal	1. Sistema recebe solicitação 2. Coleta dados relevantes 3. Processa estatísticas 4. Gera relatório formatado 5. Disponibiliza relatório
Pós-condição	Relatório gerado
Cenário Alternativo	

Fonte: Elaborado pelos autores

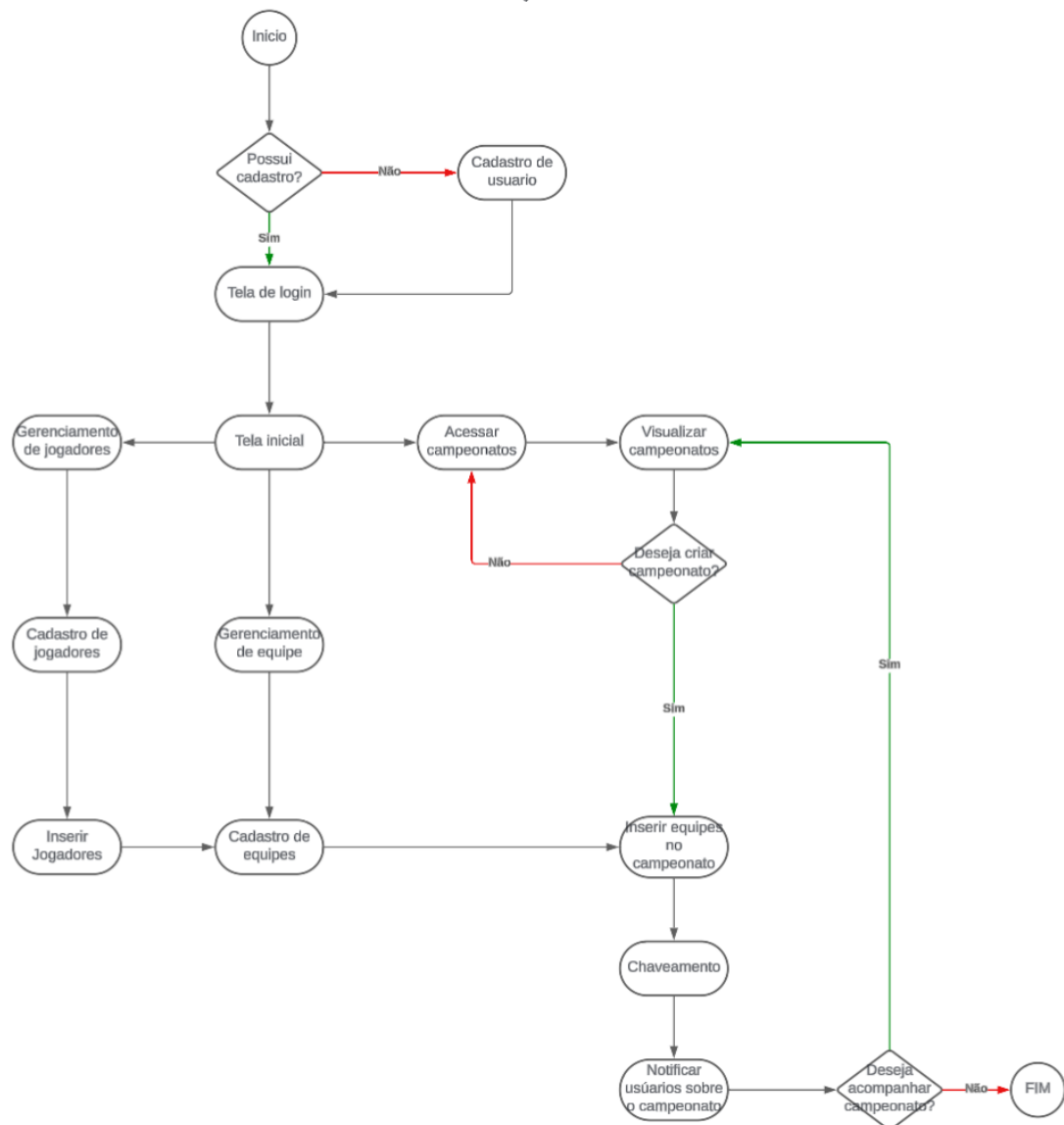
3.6 DIAGRAMA DE ATIVIDADES

O Diagrama de Atividades é um tipo de diagrama da UML (Unified Modeling Language) utilizado para representar o fluxo de atividades dentro de um processo ou sistema. Ele descreve como as ações se sucedem, incluindo decisões, paralelismos e interações entre diferentes partes do sistema.

Em essência, o diagrama mostra o comportamento dinâmico do sistema, permitindo visualizar como tarefas e eventos se encadeiam para atingir um determinado objetivo. É amplamente usado em modelagem de processos de negócios, sistemas de informação e fluxos operacionais, facilitando a comunicação entre analistas, desenvolvedores e usuários.

Segundo Pressman (2016):

“O diagrama de atividades é uma forma de representar graficamente o fluxo de controle entre atividades, destacando o comportamento do sistema durante a execução de processos. Já Sommerville reforça que esse tipo de diagrama ajuda na compreensão dos requisitos e na documentação das funcionalidades de um software.”

Figura 7 – Diagrama de atividades**Fonte:** Elaborado pelos autores

3.6.1 Processo de Cadastro e Autenticação de Usuários

O sistema inicia verificando se o usuário possui cadastro prévio. Caso negativo, o fluxo direciona para o processo de cadastro, onde são coletadas informações básicas como nome, e-mail e senha. O sistema valida a exclusividade do e-mail cadastrado e pode opcionalmente enviar uma confirmação por e-mail. Após o cadastro bem-sucedido, ou caso o usuário já possua registro, o sistema encaminha para a tela de login, onde as credenciais são validadas antes do acesso à área principal.

3.6.2 Navegação na Área Principal e Acesso a Campeonatos

Após a autenticação, o usuário acessa a tela inicial, que oferece duas opções principais: gerenciamento de jogadores ou acesso a campeonatos. Ao selecionar a opção de campeonatos, o sistema apresenta a lista de competições disponíveis para visualização. Neste ponto, ocorre uma verificação importante: o sistema analisa se o campeonato selecionado exige a formação de equipes para participação.

3.6.3 Gerenciamento de Equipes e Inscrição em Campeonatos

Para campeonatos que exigem formação de equipes, o sistema redireciona o usuário para o módulo de gerenciamento de equipes. Este processo envolve duas etapas sequenciais: primeiro, o cadastro da equipe propriamente dita, com informações como nome, sigla e escudo, validando especialmente a exclusividade do nome dentro do campeonato; segundo a inserção de jogadores, onde o usuário pode adicionar atletas já cadastrados no sistema ou realizar novos cadastros, observando os limites mínimo e máximo de jogadores por equipe. Após a confirmação da equipe, o sistema formaliza a inscrição no campeonato.

3.6.4 Chaveamento e Notificações

Quando as inscrições são encerradas, o sistema executa automaticamente o processo de chaveamento, organizando os confrontos com base em critérios pré-definidos como ranking histórico ou sorteio aleatório. Imediatamente após a geração do chaveamento, o sistema dispara notificações para todos os usuários participantes, informando sobre os adversários, datas e horários dos jogos. Este mecanismo de notificação também é acionado sempre que novos resultados são inseridos no sistema.

3.6.5 Acompanhamento e Finalização do Campeonato

Durante a execução do campeonato, os usuários podem acompanhar todos os detalhes através da visualização de tabelas de jogos, resultados atualizados e próximos confrontos. Para administradores, há funcionalidades adicionais como a inserção de resultados e o gerenciamento da classificação. O sistema permanece ativo até a completa finalização do campeonato, quando então encerra seu ciclo operacional.

3.6.6 Tratamento de Situações Especiais

O sistema incorpora mecanismos para lidar com situações excepcionais, como o número ímpar de equipes no chaveamento - onde sugere alternativas como a inclusão de "byes" ou o aguardar por mais inscrições. Também previne duplicidades de jogadores em uma mesma equipe e oferece fluxos de recuperação para casos de esquecimento de senha, garantindo assim a integridade das competições e a experiência do usuário.

3.7 Matriz de Rastreabilidade

A Matriz de Rastreabilidade, também conhecida como *Requirements Traceability Matrix* (RTM) é uma ferramenta fundamental na gestão de requisitos de projeto, principalmente no contexto de Engenharia de Software e Gestão de Projetos. Ela pode ser definida como uma tabela que rastreia a relação entre os requisitos iniciais do negócio ou do usuário e os produtos finais de trabalho do projeto, como documentos de design, componentes de código e casos de teste.

Segundo a norma (ISO/IEC/IEEE, 2018, p. 11)

“Se trata de processos e produtos de engenharia de requisitos, o rastreamento de requisitos é "a capacidade de descrever e acompanhar a vida de um requisito, em ambas as direções (forwards and backwards)"

A Matriz de Rastreabilidade é o artefato prático que materializa essa capacidade.

Sua estrutura básica geralmente associa requisitos de origem (ex.: "O usuário deve poder redefinir sua senha") a outros elementos do projeto, como:

- Requisitos Detalhados: Decomposição do requisito de alto nível.
- Elementos de Design: Componentes do sistema que atendem ao requisito.
- Módulos de Código: Arquivos, classes ou funções que implementam o design.
- Casos de Teste: Cenários específicos para verificar se o requisito foi atendido.

Para PMI (2017), “o rastreamento de requisitos ajuda a garantir que cada requisito agregue valor de negócio, vinculando-o aos objetivos do projeto e do negócio.”

A matriz, portanto, atua como uma "ponte" que conecta a necessidade inicial à solução final, permitindo uma visão completa da linhagem de cada requisito.

3.7.1 A Importância da Matriz de Rastreabilidade

A importância da Matriz de Rastreabilidade transcende a mera organização, impactando diretamente a qualidade, o controle e a eficiência do projeto. Seu uso sistemático é considerado uma das melhores práticas em desenvolvimento de software e gestão de projetos.

Uma de suas principais importâncias reside no **controle de mudanças**. Quando um requisito é alterado ou precisa ser ajustado, a matriz permite identificar rapidamente todos os componentes do projeto que serão impactados, como documentos de design, linhas de código e planos de teste.

Além disso, a matriz é crucial para garantir a **cobertura de testes**. Ela assegura que cada requisito tenha pelo menos um caso de teste associado e que todos os requisitos sejam validados, segundo a (PRESSMAN, 2016) : “Isso evita que funcionalidades críticas sejam esquecidas durante a fase de testes, aumentando significativamente a confiabilidade do produto final.”

Outro benefício fundamental é a verificação da completude. A Matriz de Rastreabilidade permite que a equipe do projeto e os stakeholders verifiquem se todos os requisitos originais foram atendidos pela solução implementada.

Por fim, a matriz facilita a comunicação entre todas as partes interessadas, fornecendo um documento centralizado e claro que demonstra como as necessidades do negócio foram traduzidas em uma solução técnica, promovendo transparência e alinhamento durante todo o ciclo de vida do projeto.

Quadro 4 – Matriz de rastreabilidade

Requisito	Descrição Resumida	Casos de Uso Relacionados	Status
RF001	Exibir menu inicial com informações introdutórias.	UC001 – Acessar Sistema	Atendido
RF002	Sistema de login (usuário/senha e Google).	UC001 – Acessar Sistema	Atendido
RF003	Cadastro e autenticação de usuários.	UC001 – Acessar Sistema	Atendido
RF004	Exibir mensagens de erro ou sucesso.	UC015 – Exibir Mensagens	Atendido
RF005	Gerenciamento de participantes.	UC010 – Gerenciar Participantes	Atendido
RF006	Gerenciamento de equipes.	UC006 – Gerenciar Equipe	Atendido
RF007	Inclusão de categorias de competições.	UC009 – Criar Campeonato / UC011 – Configurar Competição	Atendido
RF008	Definição de regras e variáveis das competições.	UC011 – Configurar Competição	Atendido

RF009	Criação, edição e exclusão de campeonatos.	UC009 – Criar Campeonato / UC011 – Configurar Competição	Atendido
RF010	Agendamento de partidas.	UC012 – Agendar Partidas	Atendido
RF011	Atualização da tabela de resultados em tempo real.	UC004 – Acompanhar Resultados	Atendido
RF012	Envio de notificações sobre partidas e resultados.	UC016 – Notificar Usuários	Atendido
RF013	Geração de relatórios e estatísticas.	UC017 – Gerar Relatórios	Atendido
RF014	Personalização de perfil do usuário.	UC002 – Gerenciar Perfil	Atendido
RF015	Integração com redes sociais.	UC001 – Acessar Sistema	Em desenvolvimento
RF016	Canal de feedback de usuários.	UC005 – Enviar Feedback	Atendido
RF017	Suporte ao usuário e FAQ.	UC016 – Notificar Usuários (parcial)	Em desenvolvimento
RF018	Sorteio automático de grupos e chaves.	UC014 – Sortear Chaves	Atendido
RF019	Funcionamento com banco de dados local.	UC009 – Criar Campeonato / UC010 – Gerenciar Participantes / UC017 – Gerar Relatórios	Atendido
RF020	Suporte à transmissão de partidas (streaming).	UC004 – Acompanhar Resultados	Em desenvolvimento
RF021	Armazenar histórico de partidas.	UC008 – Visualizar Histórico	Atendido
RF022	Gerar ranking de desempenho.	UC004 – Acompanhar Resultados / UC017 – Gerar Relatórios	Atendido
RF023	Gerenciar arbitragem das partidas.	UC010 – Gerenciar Participantes / UC013 – Aplicar Penalidades	Atendido
RF024	Sistema de penalidades para infrações.	UC013 – Aplicar Penalidades	Atendido

Fonte: Elaborado pelos autores

4 Ferramentas Utilizadas

Diversas ferramentas foram empregadas durante as etapas de desenvolvimento do projeto, escolhidas com base em critérios de eficiência, escalabilidade, suporte da comunidade e familiaridade da equipe. Todas possuem documentação abrangente e uma base de conhecimento ampla, o que facilitou significativamente a implementação das soluções.

O Visual Studio (versão mais recente) foi utilizado como o principal ambiente de desenvolvimento. Essa ferramenta, disponibilizada pela Microsoft, oferece suporte robusto a diversas linguagens de programação, integração com plataformas externas e recursos avançados de depuração e controle de versão.

Segundo a Microsoft (2024):

“Visual Studio é “um ambiente de desenvolvimento integrado que permite criar aplicativos modernos e produtivos com alta eficiência”. Sua versão

Community Edition foi escolhida por ser gratuita e oferecer recursos suficientes para atender às necessidades do projeto (MICROSOFT, 2024).”

O Microsoft Office 2016 foi empregado para a elaboração de documentos técnicos, planilhas de análise e apresentações do projeto. Essa suíte de produtividade é amplamente utilizada em ambientes corporativos e acadêmicos, oferecendo ferramentas como Word, Excel e PowerPoint.

Conforme a Microsoft (2016):

“O Office permite “criar, revisar e compartilhar documentos com alta compatibilidade e integração entre plataformas”, o que o tornou essencial para o gerenciamento documental e de relatórios durante o desenvolvimento (MICROSOFT, 2016).”

Para a modelagem visual de processos, a equipe utilizou o Lucidchart (versão gratuita), ferramenta amplamente reconhecida pela sua interface intuitiva e recursos colaborativos em tempo real. O Lucidchart foi essencial na criação de diagramas BPMN e fluxos de sistema, permitindo representar graficamente os processos e interações do projeto. De acordo com a Lucid Software (2024): a ferramenta auxilia equipes a projetar e visualizar sistemas complexos por meio de diagramas claros e colaborativos” (LUCIDCHART, 2024).

O gerenciamento das tarefas e o acompanhamento do progresso do projeto foram realizados com o auxílio do **Trello (versão gratuita)**. A plataforma foi utilizada para organizar o fluxo de trabalho, definir prioridades e acompanhar o andamento das atividades em formato de quadros (boards) e cartões (cards). Segundo a Atlassian (2024), empresa responsável pela ferramenta, o Trello “ajuda equipes a colaborar, gerenciar projetos e atingir metas de maneira visual e flexível” (ATLASSIAN, 2024).

Para o registro de anotações, atas de reuniões e documentação de ideias, foi empregado o Obsidian (versão gratuita). Essa ferramenta se destacou pela capacidade de criar e interligar notas em formato Markdown, permitindo construir uma base de conhecimento integrada e facilmente navegável.

Conforme os desenvolvedores do Obsidian (2024), “o aplicativo “transforma anotações simples em uma rede interconectada de ideias, ideal para o gerenciamento de projetos e estudos.”

4.1 Métodos e Execução do Projeto de Interface

Durante o processo de levantamento de requisitos, foram aplicadas diversas técnicas de análise e planejamento estratégico que auxiliaram na definição do escopo do sistema e na identificação de soluções viáveis. As metodologias empregadas proporcionaram maior clareza, organização e fundamentação nas decisões tomadas ao longo do desenvolvimento.

A **análise SWOT (Strengths, Weaknesses, Opportunities, Threats)** foi utilizada para identificar os pontos fortes e fracos do projeto, bem como as oportunidades e ameaças presentes no ambiente externo.

Segundo Kotler e Keller (2016):

essa ferramenta “permite avaliar fatores internos e externos que podem impactar o desempenho e a viabilidade de um projeto”, sendo amplamente empregada em planejamento estratégico.

A técnica **5W2H (What, Why, Where, When, Who, How, How much)** foi aplicada para estruturar as principais ações e atividades do projeto. Essa metodologia oferece uma visão prática e organizada, facilitando a definição de responsabilidades e prazos.

De acordo com Campos (1992):

“O 5W2H “é um método de planejamento que auxilia na execução sistemática de tarefas, reduzindo ambiguidades e aumentando a eficiência dos processos”.”

Por fim, a notação BPMN foi utilizada para mapear graficamente os processos de negócio e o fluxo de informações entre os usuários e o sistema. Essa metodologia de modelagem padronizada permite representar visualmente os processos de forma compreensível e formal. Segundo o Object Management Group (OMG, 2013), “o BPMN fornece uma notação gráfica padronizada que facilita o entendimento e a comunicação entre analistas, desenvolvedores e gestores”.

5 Desenvolvimento

A etapa de desenvolvimento concentra a implementação prática das funcionalidades planejadas, transformando requisitos e modelos em componentes reais do sistema. Nela são construídos os módulos principais, como autenticação, gerenciamento de campeonatos, telas de navegação e cadastros, sempre seguindo boas práticas de organização, testes e validação contínua. Essa fase garante que o

protótipo evolua de maneira estruturada, funcional e coerente com os objetivos do projeto.

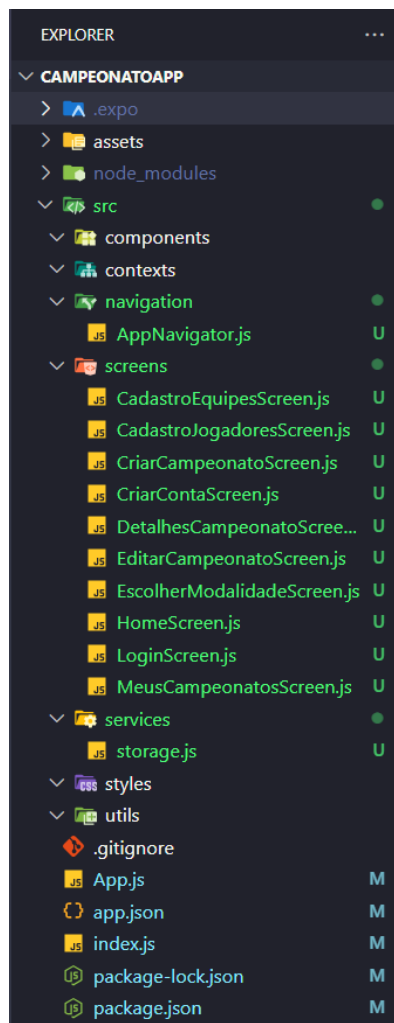
5.1 Arquitetura do Sistema

O protótipo foi desenvolvido utilizando **React Native** com **Expo**, seguindo uma arquitetura baseada em componentes e navegação estruturada. A escolha do React Native se justifica pela capacidade de desenvolvimento multiplataforma (iOS e Android) com uma única base de código, enquanto o Expo facilita o processo de desenvolvimento e testes.

5.1.1 Estrutura de Pastas

A Figura 8 mostra a estrutura de pastas e arquivos, seguindo uma estrutura de projetos organizada.

Figura 8 – Pastas e arquivos Explorer do projeto



Fonte: Elaborado pelos autores

5.2 Camada de Persistência de Dados

A camada de persistência de dados é responsável por armazenar e recuperar informações essenciais do sistema, garantindo segurança, integridade e disponibilidade dos registros. No protótipo, essa persistência é feita por meio de armazenamento local, permitindo salvar usuários, equipes e campeonatos mesmo sem conexão externa. Essa camada dá suporte a todas as funcionalidades do aplicativo, assegurando que os dados sejam mantidos de forma organizada e confiável.

5.2.1 Serviço de Armazenamento Local

O sistema utiliza **AsyncStorage** para persistência de dados local, eliminando a necessidade de conexão com servidor durante a fase de prototipação.

Figura 9 - Código: arquivo storage.js

```
src > services > storage.js > StorageService > removeItem
1  import AsyncStorage from '@react-native-async-storage/async-storage';
2
3  class StorageService {
4    // Salvar dados
5    async setItem(key, value) {
6      try {
7        const jsonValue = JSON.stringify(value);
8        await AsyncStorage.setItem(key, jsonValue);
9        return true;
10     } catch (error) {
11       console.error('Erro ao salvar dados:', error);
12       return false;
13     }
14   }
15
16   // Recuperar dados
17   async getItem(key) {
18     try {
19       const jsonValue = await AsyncStorage.getItem(key);
20       return jsonValue != null ? JSON.parse(jsonValue) : null;
21     } catch (error) {
22       console.error('Erro ao recuperar dados:', error);
23       return null;
24     }
25   }
26 }
```

Fonte: Elaborado pelos autores

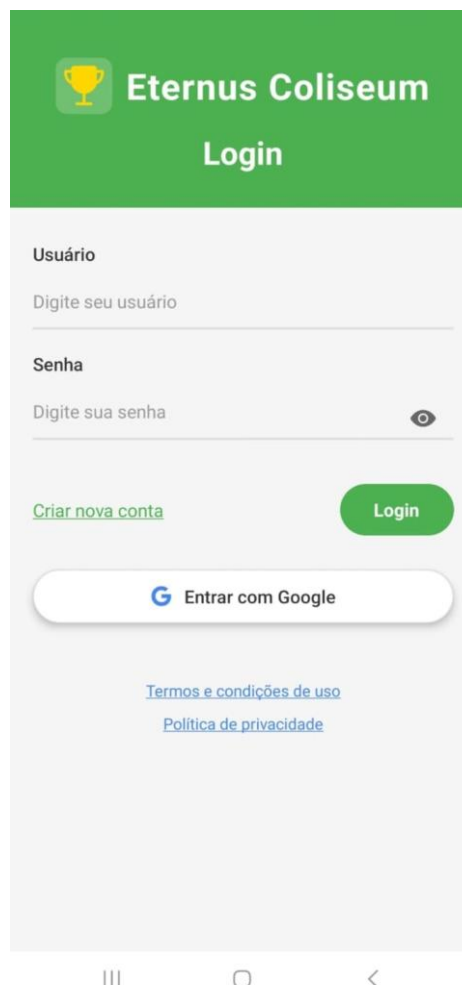
5.3 Sistema de Autenticação

O sistema de autenticação assegura que apenas usuários autorizados possam acessar as funcionalidades do aplicativo. Através de processos de login e cadastro, ele valida credenciais, controla sessões e restringe operações conforme o perfil do usuário. Essa etapa é fundamental para manter a segurança do sistema, proteger informações sensíveis e estruturar níveis de acesso como administrador, organizador e participante.

5.3.1 Tela de Login

A tela de login implementa autenticação local com validação de credenciais e gerenciamento de sessão, conforme podemos ver na Figura 10.

Figura 10 – Captura da tela Login



Fonte: Elaborado pelos autores

5.3.2 Tela de Criação de Conta

Podemos ver na Figura 11, a tela “Criar Conta”, onde permite o cadastro de novos usuários com validações completas.

Figura 11 – Captura da tela Criar Conta

← **Criar Conta**

Nome Completo
Digite seu nome completo

Usuário
Digite seu nome de usuário

Email
Digite seu email

Senha
Digite sua senha (min. 6 caracteres)

Confirmar Senha
Confirme sua senha

Criar Conta

Já tem uma conta? [Fazer Login](#)

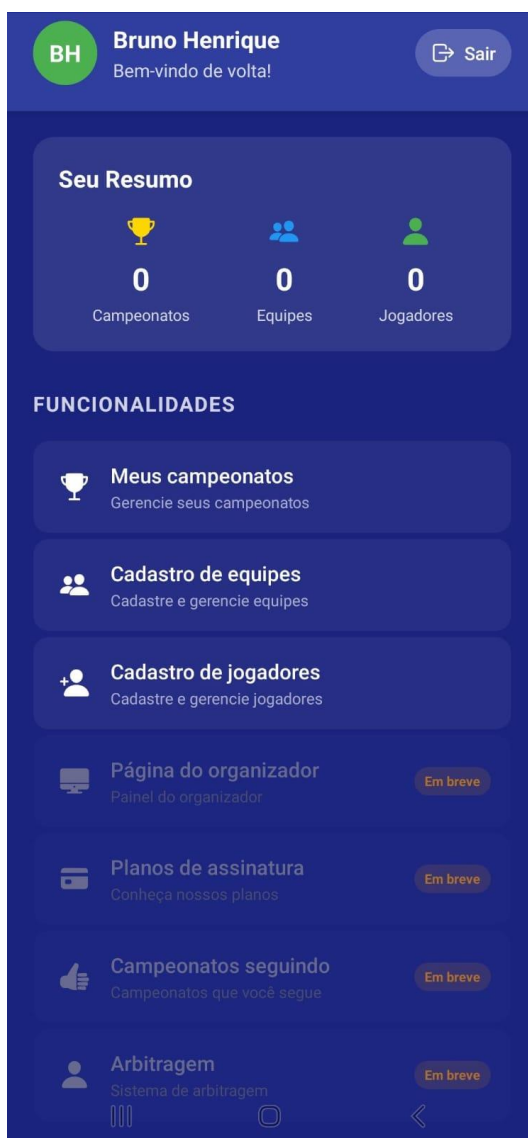
Ao criar uma conta, você concorda com nossos
[Termos e condições de uso](#)
[Política de privacidade](#)

Fonte: Elaborado pelos autores

5.4 Tela Principal (Home)

A HomeScreen serve como hub central do aplicativo, apresentando estatísticas e navegação, conforme podemos ver na Figura 12.

Figura 12 – Captura da tela Home



Fonte: Elaborado pelos autores

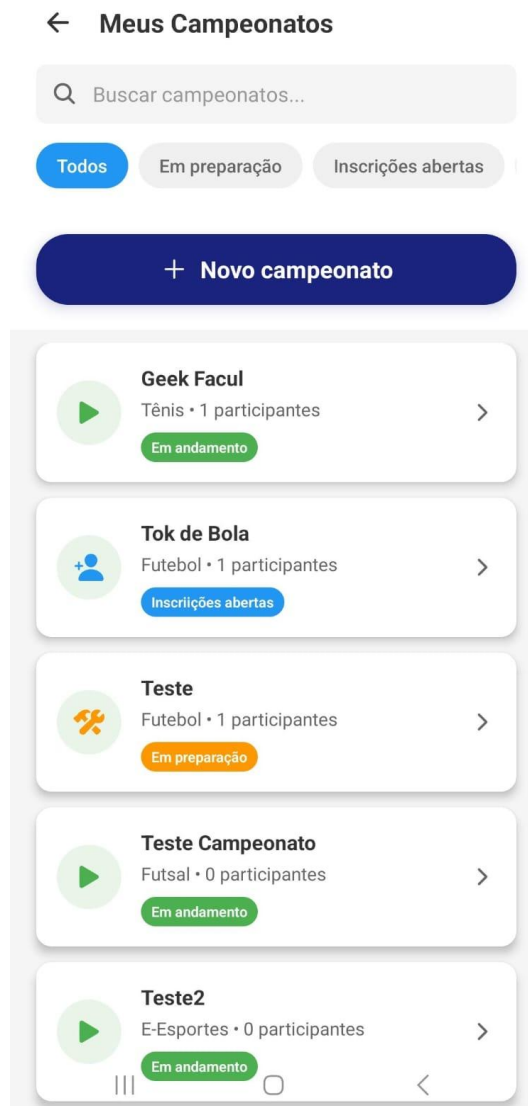
5.5 Gerenciamento de Campeonatos

O gerenciamento de campeonatos reúne todas as funcionalidades relacionadas à criação, edição e acompanhamento das competições. Inclui a definição de modalidades, regras, chaveamento, agendamento de partidas, cadastro de equipes e visualização de resultados. Esse módulo é o núcleo do sistema, oferecendo aos organizadores uma ferramenta prática e dinâmica para conduzir torneios de forma padronizada, rápida e eficiente.

5.5.1 Listagem com Filtros

A tela “Meus Campeonatos”, Figura 13, implementa a busca e filtros múltiplos.

Figura 13 – Captura da tela Meus Campeonatos



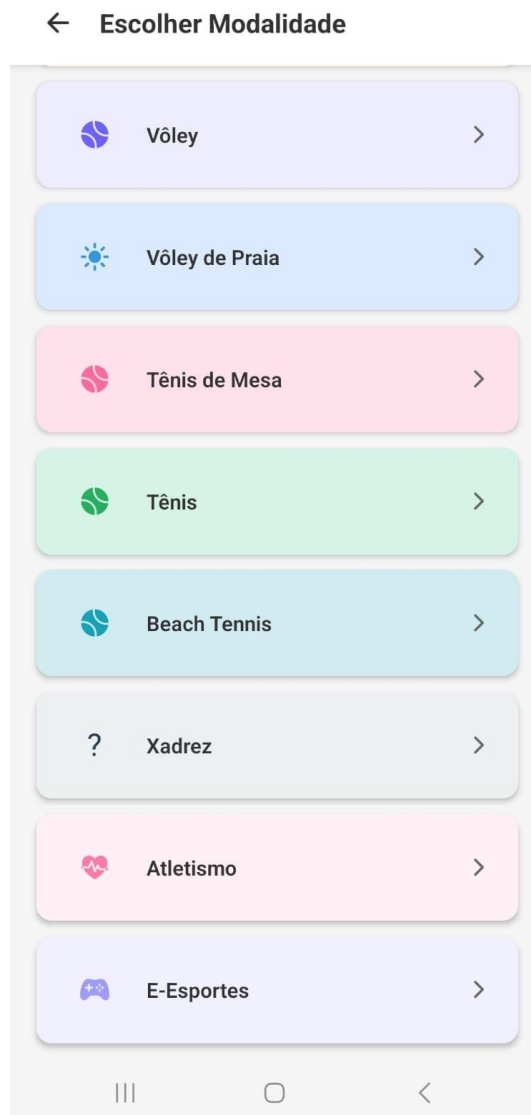
Fonte: Elaborado pelos autores

5.5.2 Criação de Campeonato

Podemos ver na Figura 14 e Figura 15 o fluxo de navegação em duas etapas:

- Escolha de Modalidade (Escolher Modalidade Screen)
- Configuração Detalhada (Criar Campeonato Screen)

Figura 14 – Captura da tela Escolher Modalidade



Fonte: Elaborado pelos autores

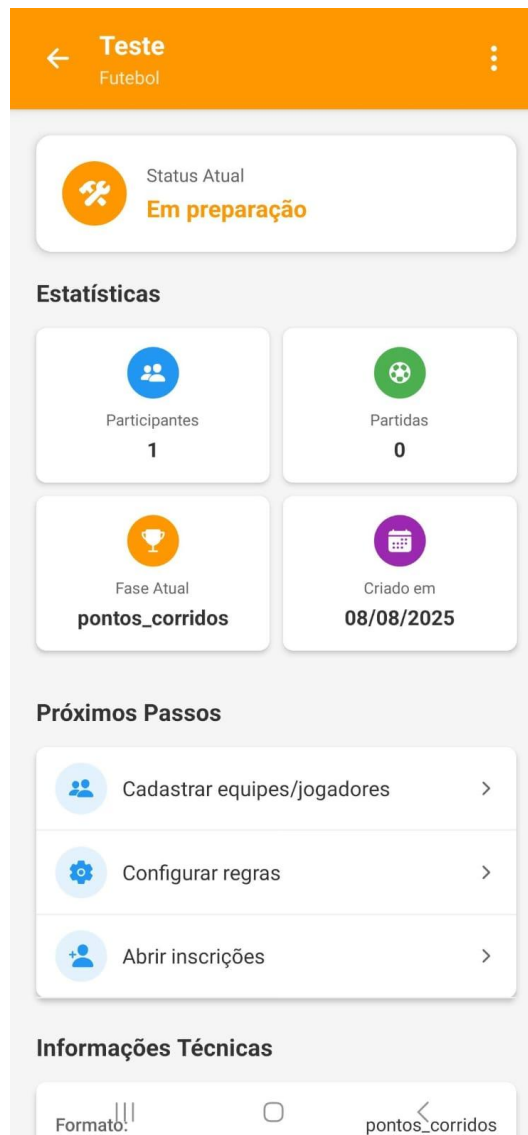
Figura 15 – Captura da tela Criar Campeonato

Fonte: Elaborado pelos autores

5.6 Edição de Campeonatos

A tela de edição utiliza modais bottom-sheet para seleção de opções, conforme a Figura 16.

Padrão de Design: Modal overlay semitransparente com conteúdo deslizante, seguindo guidelines de Material Design.

Figura 16 – Captura da tela Editar Campeonato

Fonte: Elaborado pelos autores

5.7 Cadastro de Equipes

Podemos observar na Figura 17, a tela de “Cadastro de Equipes”, onde o usuário pode registrar uma “Nova Equipe” informando nome, campeonato, categoria, cor do uniforme e dados do responsável. A interface exibe campos de formulário, seleção de cores e opções de categorias, tudo organizado em um modal claro e intuitivo.

Figura 17 – Captura da tela Cadastro de Nova Equipe

Cadastro de Equipes

Buscar equipes...

Todos Geek Facul Tok de Bola Teste

Nova Equipe

Nome da Equipe *

Teste

25 caracteres restantes

Campeonato *

Teste

Categoria

Masculino

Cor do Uniforme

☒ ☐ ☐ ☐ ☐ ☐ ☐ ☐

Responsável pela Equipe

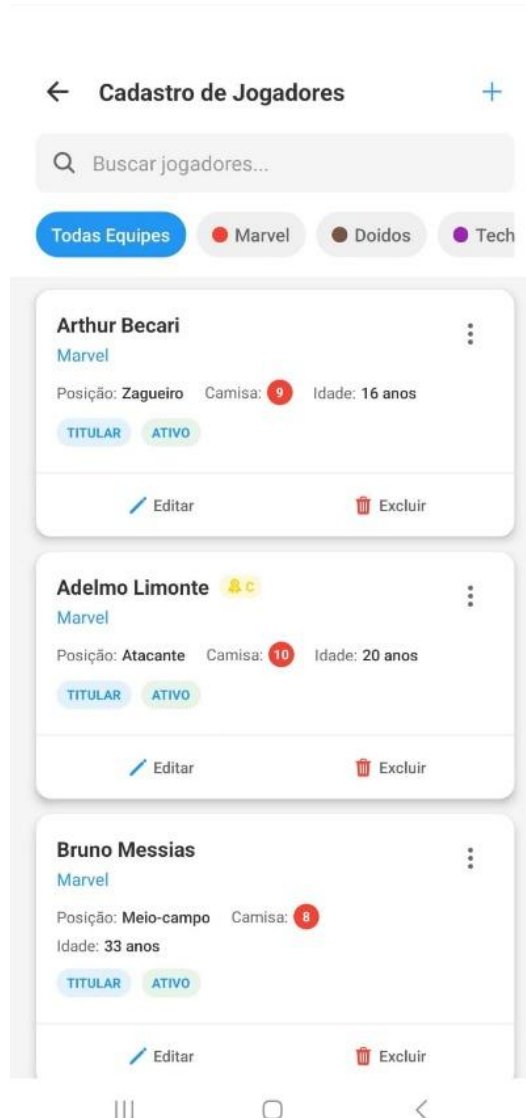
Nome do Responsável *

Nome completo do responsável

Fonte: Elaborado pelos autores

5.8 Cadastro de Jogadores

Na tela de “Cadastro de Jogadores”, podemos observar na Figura 18, onde é possível visualizar, buscar e filtrar jogadores por equipe. Cada cartão exibe informações como posição, número da camisa, idade e status (titular/ativo), além de opções para editar ou excluir o jogador.

Figura 18 – Captura da tela Cadastro de Jogadores

Fonte: Elaborado pelos autores

5.9 Conclusão do Desenvolvimento

O protótipo demonstra uma implementação completa de um sistema de autenticação robusto, com um CRUD completo para campeonatos, equipes e jogadores, onde tivemos os relacionamentos entre as entidades, fazendo filtros e buscas avançadas, com validações em múltiplas camadas e com um soft delete para preservação de dados e uma interface responsiva e intuitiva.

6 Resultados e Discussão

O desenvolvimento do protótipo do Eternus Coliseum, permitiu validar grande parte dos requisitos levantados durante a fase inicial do projeto. As funcionalidades essenciais, como autenticação de usuários, criação e gerenciamento de campeonatos, cadastro de equipes e jogadores, bem como atualização dinâmica das tabelas, foram implementadas com sucesso, demonstrando a viabilidade técnica do sistema. Os testes realizados mostraram que a interface se manteve intuitiva e responsiva, contribuindo para uma experiência de uso positiva. Além disso, o uso de ferramentas como React Native e AsyncStorage mostrou-se adequado para a proposta de um protótipo multiplataforma. No entanto, observou-se que funções mais avançadas, como integração com streaming e notificações push completas, ainda demandam aprimoramentos e maior infraestrutura. De modo geral, os resultados alcançados confirmam que o Eternus Coliseum atende ao objetivo de tornar a organização de torneios mais prática, acessível e eficiente, ao mesmo tempo em que revelam oportunidades de expansão para versões futuras do projeto.

6.1 Formulário de Análise de Portabilidade – Eternus Coliseum

A análise de portabilidade do Eternus Coliseum tem como objetivo verificar a capacidade do sistema em operar adequadamente em diferentes dispositivos móveis e ambientes de desenvolvimento. A portabilidade é um fator essencial para garantir que o aplicativo possa ser acessado de maneira eficiente, responsiva e multiplataforma, independentemente do dispositivo iOS ou Android utilizado.

6.1.1 Configuração Mínima – Usuário Final (Dispositivos Móveis)

Hardware e Software Recomendados:

6.1.1.1 Dispositivos Android:

Sistema Operacional: Android 8.0 (Oreo) ou superior

Processador: Quad-Core de 1.4 GHz ou superior (arquitetura ARM64 ou x86)

Memória RAM: Mínimo de 2 GB (recomendado 3 GB ou mais)

Armazenamento: 100 MB de espaço livre para instalação do app

Resolução de Tela: Mínimo de 720 x 1280 pixels (HD)

Conectividade: Requer conexão com a internet (estável) lembrando que caso a simulação seja em um celular, ambos (computador e celular) precisam estar na mesma rede para se comunicarem.

6.1.1.2 Dispositivos iOS:

Sistema Operacional: iOS 12.0 ou superior

Dispositivos Compatíveis: iPhone 6s ou modelos mais recentes, iPad Air 2 ou superior

Processador: Apple A9 ou superior

Memória RAM: Mínimo de 2 GB

Armazenamento: 100 MB de espaço livre

Resolução de Tela: Suporta todos os tamanhos de tela padrão iOS (desde iPhone SE até iPad Pro)

6.1.2 Ambiente de Desenvolvimento e Testes

O ambiente de desenvolvimento do Eternus Coliseum foi estruturado para garantir estabilidade, reprodutibilidade e facilidade de manutenção durante a construção do protótipo. As implementações foram realizadas utilizando Visual Studio Code e Expo/React Native, permitindo testes rápidos por meio do simulador integrado e dispositivos físicos Android. O uso do Expo facilitou a validação de funcionalidades como navegação, armazenamento local e renderização de componentes, reduzindo o tempo de configuração e depuração. Os testes operacionais foram conduzidos de forma iterativa, verificando fluxos como autenticação, criação de campeonatos, cadastro de equipes e atualização de tabelas. Esse ambiente controlado permitiu identificar erros ainda na fase inicial, garantindo maior consistência ao protótipo antes de migrações para possíveis ambientes de produção. Assim, o ambiente de desenvolvimento e testes adotado mostrou-se adequado à fase atual do projeto, oferecendo suporte eficiente às necessidades de implementação.

6.1.2.1 Hardware Recomendado (Desktop/Notebook):

Processador: Intel Core i5 (8ª geração) ou AMD Ryzen 5 ou superior

Memória RAM: Mínimo 8 GB (recomendado 16 GB)

Armazenamento: 10 GB de espaço livre em SSD

Sistema Operacional: Windows 10/11, macOS 10.14+, ou Linux Ubuntu 20.04+

6.1.2.2 Software Necessário:

Node.js e npm

Versão: Node.js 16.x ou superior

npm: 8.x ou superior

Instalação: <https://nodejs.org/>

Git (para controle de versão)

Versão 2.30 ou superior

Download: <https://git-scm.com/>

6.1.2.3 Editor de Código

Visual Studio Code (recomendado)

Extensões sugeridas:

- ESLint
- Prettier
- React Native Tools
- ES7+ React/Redux/React-Native snippets

6.1.3 Código Único para iOS e Android

100% do código compartilhado entre plataformas

Componentes React Native renderizados nativamente

Sem necessidade de código específico por plataforma

6.1.3.1 Armazenamento Local Portável

AsyncStorage abstrai diferenças entre iOS e Android

Dados migram automaticamente entre dispositivos (com backup)

Estrutura JSON facilita exportação/importação

6.1.3.2 Interface Responsiva

Layout adapta-se a diferentes tamanhos de tela (smartphones e tablets)

Suporta orientações portrait e landscape

Dimensões calculadas dinamicamente via `Dimensions.get('window')`

6.1.3.3 Futuro Web com Expo

Potencial para executar como PWA (Progressive Web App)

Mesmo código pode ser compilado para web com pequenos ajustes

Comando: `npx expo start --web`

6.1.3.4 Limitações Atuais de Portabilidade

Não disponível para Windows Phone (plataforma descontinuada)

Versão web ainda experimental e não totalmente otimizada

Algumas APIs nativas requerem configuração específica por plataforma (futuras implementações como notificações push)

6.1.4 Conectividade e Requisitos de Rede

Durante a fase de prototipação, o projeto foi desenvolvido para operar sem dependência de conexão contínua, utilizando armazenamento local como base de dados provisória. Esse modelo reduziu a necessidade de infraestrutura de rede, permitindo que todas as operações essenciais, como login, cadastro de equipes e gerenciamento de campeonatos fossem executadas offline. No entanto, considerou-se que em um cenário real, a conectividade terá papel fundamental, especialmente em funcionalidades futuras como integração com banco de dados remoto, envio de notificações push, transmissão de partidas e sincronização multiusuário. Para essas operações, será necessária uma conexão estável à internet e protocolos seguros de transferência de dados. Assim, enquanto o protótipo atual exige baixa dependência de rede, as próximas versões deverão incorporar mecanismos robustos de conectividade para garantir desempenho, escalabilidade e consistência das informações.

6.1.4.1 Modo Offline:

Não requer servidor backend (protótipo)

Sincronização futura será opcional

6.1.4.2 Uso de Dados:

Expo Go sincroniza código via Wi-Fi (~5-10 MB por sessão)

Hot reload consome ~1-2 MB por atualização

6.1.4.3 Aplicativo compilado:

Tamanho do APK (Android): ~50-70 MB

Tamanho do IPA (iOS): ~60-80 MB

Nenhum dado consumido após instalação (modo offline)

6.1.4.4 Requisitos para Desenvolvimento:

Wi-Fi: Necessário para Expo Go e hot reload

Velocidade mínima: 10 Mbps (recomendado 20 Mbps)

Latência: <100ms para experiência fluida

6.2 Métricas do Projeto

As métricas aplicadas ao projeto tiveram como objetivo avaliar o andamento do desenvolvimento e monitorar a qualidade das entregas realizadas. Durante a execução, foram analisadas métricas como quantidade de funcionalidades implementadas, tempo gasto em cada tarefa, percentual de requisitos atendidos e progresso do MVP. A matriz de rastreabilidade demonstrou que a maior parte dos requisitos funcionais foi cumprida, validando a coerência entre planejamento e execução. Além disso, o uso de ferramentas como Trello permitiu acompanhar o fluxo de trabalho, identificar atrasos e redistribuir demandas conforme necessário. Mesmo com limitações naturais de um projeto acadêmico, as métricas evidenciaram avanço consistente e desenvolvimento alinhado aos objetivos definidos no início do projeto. Dessa forma, as medições realizadas permitem concluir que o projeto evoluiu de maneira organizada, seguindo boas práticas de monitoramento e garantindo transparência no processo de desenvolvimento.

6.2.1 Análise de Pontos de Função (APF)

As métricas apresentadas na Figura 19 e Figura 20, seguem o método de Análise de Pontos de Função (APF ou FPA - Function Point Analysis), uma técnica consagrada na engenharia de software para medição do tamanho funcional de sistemas. Desenvolvida por Allan Albrecht na IBM em 1979, esta abordagem é reconhecida pela ISO (ISO/IEC 20926) e amplamente utilizada para estimativas de esforço, prazo e custo de projetos de software.

Figura 19 – Métricas (primeira parte)

Função	Nº de ocorrência	Complexidade	Peso	Resultado	Nível de Influência do Sistema (0 a 5)	
entradas	2	Simple	3	6	Comunicação de dados	5
	0	Médio	4	0	Performance	5
	3	complexo	6	18	Volume de transações	5
					Eficiência do usuário final	5
saídas	3	Simple	4	12	Processamento complexo	5
	3	Médio	5	15	Facilidade de implantação	5
	0	complexo	7	0	Múltiplos locais	5
					Processamento distribuído	5
consultas	2	Simple	3	6	Utilização de equipamento	5
	1	Médio	4	4	Entrada de dados on-line	5
	3	complexo	6	18	Atualização on-line	5
					Reutilização de código	5
arquivos	5	Simple	7	35	Facilidade operacional	5
	0	Médio	10	0	Facilidade de mudanças	5
	0	complexo	15	0		
					Total de NI	70
interfaces	6	Simple	5	30		
	0	Médio	7	0		
	0	complexo	10	0		
Total de FP'b				144	FA = multiplicar o NI pela taxa real = 0,65+(0,01*119)	1,35
					FP'r = multiplicar o FP'b pelo FA	194
					Informe o nº de LOC da Linguagem	17,5

Fonte: Elaborado pelos autores

Figura 20 – Métricas (segunda parte)

Estimativas do número médio de LOC por FP		KLOC = Multiplicar o FP'r pelo tipo de linguagem	
Cobol	100	Total de KLOC	3395
Pascal	90		
Linguagens Orientadas a Objeto (C++)	30	Informe o tipo de sistema	3.300
Java / Delphi / Visual Basic / C#	20		
Geradores de Código (SQL + HTML + RUBY + PYTHON + DEMAIS)	15	PRAZO (dividir o KLOC pelo tipo de sistema)	
		(RESULTADO DA DIVISÃO)QTDD DE MESES	1,47
Tipo de Sistema	Produ - Kloc/Loc /mês	QTDD DE DIAS (% x 22)	32,34
Sistema Comercial	2.500	QTDD DE HORAS (% x 6)	194,04
Comércio Eletrônico	3.600	QTDD DE MINUTOS (% x 60)	116,424
Sistema Web	3.300		
		CUSTO - INVESTIMENTO	
		Informe o valor da hora de trabalho	R\$ 40,00
		ISO (NORMA INTERNACIONAL) HORAS/MÊS =	132
		FÓRMULA = 132 * RESULTADO DA DIVISÃO * VALOR DA HORA	
CONFIRMAÇÃO (RESULTADO * 132 * Vr DA HORA)		VALOR TOTAL DO PROJETO =	R\$ 7.761,60

Fonte: Elaborado pelos autores

6.3 Proposta Comercial

O Eternus Coliseum é uma plataforma inovadora desenvolvida para simplificar e automatizar a criação, gestão e acompanhamento de torneios esportivos e de e-sports. A solução une o físico e o digital em uma única arena, oferecendo ferramentas intuitivas para organizadores, participantes e espectadores. Com foco em acessibilidade, transparência e engajamento, a plataforma elimina a complexidade de

processos manuais e centraliza todas as etapas do campeonato em um único ambiente.

6.3.1 Escopo do Projeto

O escopo do Eternus Coliseum foi definido com base nas necessidades identificadas durante o levantamento de requisitos e nas limitações técnicas do período de desenvolvimento. O projeto abrangeu a criação de um protótipo funcional capaz de realizar operações essenciais para a gestão de torneios, como autenticação de usuários, criação e edição de campeonatos, cadastro de equipes e jogadores, geração de chaves e acompanhamento de resultados. Também foram incluídos módulos de visualização, histórico e gerenciamento básico de notificações. Funcionalidades avançadas, como integração com servidores externos, banco de dados em nuvem, transmissão de partidas e automação de processos complexos, foram mantidas fora do escopo desta etapa, ficando previstas para fases futuras de evolução do sistema. Assim, o escopo estabelecido garantiu um protótipo coerente, funcional e alinhado às principais necessidades dos usuários, sem comprometer a viabilidade do desenvolvimento.

6.3.1.1 Funcionalidades Principais Incluídas:

- Cadastro e autenticação de usuários (login tradicional e via Google).

- Criação e gestão de campeonatos (formato mata-mata, pontos corridos, suíço).

- Cadastro de equipes e jogadores.

- Sorteio automático de chaves.

- Agendamento de partidas.

- Acompanhamento de resultados em tempo real.

- Notificações push.

- Relatórios e estatísticas.

- Sistema de penalidades e arbitragem.

- Personalização de perfil.

- Suporte a múltiplas modalidades (MOBA, futebol, etc.).

6.3.1.2 Público-Alvo:

- Organizadores de torneios amadores e estudantis
- Jogadores de e-sports e esportes tradicionais
- Equipes e ligas competitivas
- Instituições de ensino e comunidades gamers

6.3.1.3 Prazo de Desenvolvimento

Com base na metodologia de estimativa por pontos de função e linhas de código (KLOC), o prazo total para o desenvolvimento do projeto é de:

1,47 meses (aproximadamente 1 mês e 14 dias)

Equivalente a 32,34 dias úteis ou 194 horas de desenvolvimento

6.3.2 Investimento Total

O investimento total do projeto foi estimado com base nas horas dedicadas ao desenvolvimento, nos recursos tecnológicos utilizados e nas necessidades operacionais ao longo do processo. Como o projeto foi desenvolvido em ambiente acadêmico, grande parte das ferramentas utilizadas, como Expo, React Native, Visual Studio Code, Trello e Lucidchart, foi acessada em versões gratuitas, reduzindo significativamente os custos diretos. Dessa forma, o principal investimento esteve relacionado à carga horária dedicada pela equipe, bem como aos recursos pessoais utilizados para testes em dispositivos físicos. Mesmo com orçamento reduzido, as entregas previstas foram concluídas de maneira satisfatória, demonstrando que o projeto pode evoluir com investimentos adicionais em infraestrutura de hospedagem, banco de dados remoto, serviços de autenticação e melhorias de desempenho. Assim, o investimento realizado mostrou-se adequado ao escopo e ao objetivo do protótipo apresentado.

6.3.2.1 Custo de Desenvolvimento

Valor total do projeto: R\$ 7.761,60

Valor calculado com base em 194 horas de desenvolvimento, taxa horária de R\$ 40,00 e inclui análise, design, codificação, testes e documentação

6.3.2.2 Opção 1: Assinatura Mensal para Organizadores

Planos a partir de R\$ 99/mês

Inclui até 5 campeonatos simultâneos

Suporte prioritário e atualizações

6.3.2.3 Opção 3: Personalização Sob Demanda

Para clientes corporativos ou com necessidades específicas

Orçamento sob consulta

Inclui integrações com streaming, APIs externas e branding white-label

6.3.2.4 Visão Estratégica

Ser a plataforma unificada de referência para a gestão de competições esportivas e de e-sports, democratizando o acesso à organização de torneios e promovendo uma experiência acessível, transparente e engajante para organizadores, participantes e espectadores.

Buscamos integrar o físico e o digital em uma única arena, eliminando barreiras técnicas e operacionais, e nos tornar o ecossistema preferido para comunidades competitivas amadoras e estudantis, fomentando a competição saudável e o crescimento do cenário competitivo nacional.

Considerações finais

Nosso objetivo inicial foi desenvolver um aplicativo que tornasse a organização de torneios mais dinâmica, acessível e eficiente. A proposta começou de forma simples, focada em oferecer ferramentas básicas de qualidade de vida para os organizadores, como o registro de equipes e o acompanhamento de partidas. Entretanto, ao longo do planejamento e desenvolvimento, o escopo do projeto se expandiu significativamente, incorporando novas funcionalidades e ideias que surgiram durante o processo.

Durante o desenvolvimento, enfrentamos desafios consideráveis, especialmente na etapa de idealização e implementação de certas lógicas de código, o que exigiu um esforço extra de pesquisa e aprendizado. A principal dificuldade esteve relacionada à estruturação do sistema de chaveamento, um componente que demanda

um nível maior de complexidade algorítmica e de integração com o restante do sistema. Apesar disso, conseguimos construir uma base sólida para o funcionamento do aplicativo e validamos diversas de suas funcionalidades essenciais, como o cadastro de usuários, criação de campeonatos e gerenciamento de equipes e jogadores.

O processo de desenvolvimento do Eternus Coliseum nos proporcionou uma experiência de grande valor acadêmico e profissional, permitindo aplicar na prática os conceitos aprendidos ao longo do curso de Análise e Desenvolvimento de Sistemas, além de fortalecer nossas habilidades de trabalho em equipe, resolução de problemas e pesquisa aplicada.

Com os resultados obtidos, acreditamos que o projeto cumpre seu papel como um protótipo funcional e promissor, capaz de evoluir para uma solução completa no futuro. Entre as próximas etapas planejadas estão o aprimoramento do sistema de chaveamento, a implementação de acompanhamento em tempo real por meio de banco de dados online, e a integração com plataformas de streaming, possibilitando uma experiência ainda mais imersiva para jogadores e espectadores.

Em suma, o Eternus Coliseum representa não apenas um projeto de conclusão de curso, mas também o início de uma proposta concreta de inovação no cenário da organização de torneios unindo o físico e o digital em uma única arena, com foco em acessibilidade, praticidade e modernidade.

Referências

ATLASSIAN. **Trello – Organize anything, together.** Disponível em: <https://trello.com/>. Acesso em: 30 out. 2025.

CAMPOS, Vicente Falconi. **TQC – Controle da Qualidade Total (no estilo japonês).** 2. ed. Belo Horizonte: INDG, 1992.

DUMAS, M. et al. **Fundamentals of business process management.** 2. ed. Berlin: Springer, 2018.

ISO/IEC/IEEE. **ISO/IEC/IEEE 29148:2018 - Systems and software engineering — Life cycle processes — Requirements engineering.** Genebra: ISO, 2018.

JACOBSON, I.; BOOCH, G.; RUMBAUGH, J. **The unified software development process.** Reading: Addison-Wesley, 1999.

KOTLER, Philip; KELLER, Kevin Lane. **Administração de Marketing.** 15. ed. São Paulo: Pearson, 2016.

LUCID SOFTWARE. **Lucidchart – Visual Collaboration Suite**. Disponível em: <https://www.lucidchart.com/>. Acesso em: 30 out. 2025.

MAXIMIANO, A. C. A. **Administração de Projetos: Como transformar ideias em resultados**. 5. ed. São Paulo: Atlas, 2011.

MICROSOFT. **Office 2016 – Previous Versions**. Disponível em: <https://www.microsoft.com/pt-br/microsoft-365/previous-versions/microsoft-office-2016>. Acesso em: 30 out. 2025.

MICROSOFT. **Visual Studio – Official Site**. Disponível em: <https://visualstudio.microsoft.com/>. Acesso em: 30 out. 2025.

OBJECT MANAGEMENT GROUP. **Business Process Model and Notation (BPMN) Version 2.0.2**. Needham, MA: OMG, 2013.

OBSIDIAN. **Obsidian – A second brain, for you, forever**. Disponível em: <https://obsidian.md/>. Acesso em: 30 out. 2025.

OSTERWALDER, A.; PIGNEUR, Y. **Business model generation: inovação em modelos de negócios**. Rio de Janeiro: Alta Books, 2010.

PRESSMAN, R. S. **Engenharia de software: uma abordagem profissional**. 8. ed. Porto Alegre: AMGH, 2016.

PROJECT MANAGEMENT INSTITUTE. **A guide to the project management body of knowledge: PMBOK guide**. 7. ed. Pennsylvania: PMI, 2021.

PROJECT MANAGEMENT INSTITUTE. **Um guia do conhecimento em gerenciamento de projetos: guia PMBOK**. 6. ed. Pennsylvania: PMI, 2017.

SEBRAE. **Ferramenta 5W2H: o que é e como aplicar na sua empresa**. 2023. Disponível em: <<https://www.sebrae.com.br/sites/PortalSebrae/artigos/5w2h-o-que-e-e-como-aplicar-na-sua-empresa>>. Acesso em: 5 out. 2024.

SOMMERVILLE, I. **Engenharia de software**. 9. ed. São Paulo: Pearson Prentice Hall, 2011.