

CENTRO ESTADUAL DE EDUCAÇÃO TECNOLÓGICA PAULA SOUZA  
FACULDADE DE TECNOLOGIA DA PRAIA GRANDE  
CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE  
E DESENVOLVIMENTO DE SISTEMAS

André Luiz Nery de Oliveira

Kauã da Silva Duarte

Thalita Bispo Soares

RELATÓRIO TÉCNICO DE CONCLUSÃO DE CURSO  
KLUBINHOS: REDE SOCIAL PARA CLUBES DE LEITURA

PRAIA GRANDE  
JUNHO/2024

André Luiz Nery de Oliveira

Kauã da Silva Duarte

Thalita Bispo Soares

## RELATÓRIO TÉCNICO DE CONCLUSÃO DE CURSO

KLUBINHOS: REDE SOCIAL PARA CLUBES DE LEITURA

Relatório técnico apresentado como exigência parcial para aprovação no Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas da Faculdade de Tecnologia da Praia Grande.

Orientador: Prof.<sup>a</sup> Me. Simone Maria Viana Romano

PRAIA GRANDE  
JUNHO/2024

OLIVEIRA, André Luiz Nery de

KLUBINHOS: REDE SOCIAL PARA CLUBES DE LEITURA. / André Luiz Nery de Oliveira, Kauã da Silva Duarte, Thalita Bispo Soares – Praia Grande, 2024.

90f.

Relatório técnico (Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas) - - Faculdade de Tecnologia de Praia Grande – Centro Estadual de Educação Tecnológica Paula Souza

Orientadora: Profa. Ms. Simone Maria Viana Romano

1. Desenvolvimento de software 2. Livro 3. Redes virtuais.

I. OLIVEIRA, André Luiz Nery de , II. DUARTE, Kauã da Silva, III. SOARES, Thalita Bispo IV. ROMANO, Simone Maria Viana V. Centro Estadual de Educação Tecnológica Paula Souza – Faculdade de Tecnologia de Praia Grande

## **AGRADECIMENTOS**

Gostaríamos de agradecer aos nossos colegas da Fatec-PG por todas as trocas de conhecimento, risadas, ajudas e apoio nos momentos difíceis. Aos nossos familiares por sempre nos incentivarem e por todo o suporte imprescindível para a conclusão deste curso. Aos nossos professores por todos os ensinamentos, em especial à nossa orientadora, Simone, que sempre nos motivou, incentivou e cobrou para garantir que sempre estivéssemos dando o nosso melhor. Por fim, gostaríamos de agradecer a Fatec de Praia Grande e a todos aqueles que acreditam, incentivam e financiam o ensino público, que possibilita que nós e tantas outras pessoas consigam uma formação de qualidade.

## RESUMO

Este relatório técnico detalha o desenvolvimento de uma plataforma interativa e intuitiva para a gestão de clubes de leitura. O objetivo principal é criar um ambiente virtual dedicado que facilite a interação entre os membros, promovendo a cultura da leitura e proporcionando uma experiência colaborativa e envolvente. O projeto aborda os desafios enfrentados pelos clubes de leitura dispersos em múltiplas redes sociais e propõe uma solução integrada para unificar atividades, agendar reuniões, votar em próximas leituras e avaliar livros já lidos. A plataforma visa aumentar a coesão e a qualidade das interações dentro das comunidades literárias, aproveitando as vantagens da conectividade digital sem perder a essência das interações literárias. Para isso, optou-se pela criação de uma aplicação web utilizando de linguagem PHP com Laravel no *back-end*, MySQL como banco de dados e as ferramentas React e Vite para o *front-end*. A aplicação foi desenvolvida conforme o planejado, com os requisitos devidamente desenvolvidos e em pleno funcionamento, embora alguns testes ainda sejam necessários antes da ampla divulgação. Espera-se que em breve a aplicação esteja disponível para o público geral, permitindo que as pessoas possam gerenciar seus clubes de leitura com facilidade e eficiência.

Palavras-chave: livro, gerenciamento de clube, gestão de leitura

## **ABSTRACT**

This technical report details the development of an interactive and intuitive platform for managing book clubs. The main objective is to create a dedicated virtual environment that facilitates interaction among members, promotes the culture of reading, and provides a collaborative and engaging experience. The project addresses the challenges faced by book clubs spread across multiple social networks and proposes an integrated solution to unify activities, schedule meetings, vote on future readings, and evaluate books already read. The platform aims to increase cohesion and the quality of interactions within literary communities, leveraging the advantages of digital connectivity without losing the essence of literary interactions. To achieve this, a web application was developed using PHP with Laravel for the back-end, MySQL as the database, and React and Vite for the front-end. The application was developed as planned, with the requirements properly implemented and fully functional, although some tests are still necessary before wide release. It is expected that the application will soon be available to the general public, allowing people to manage their book clubs easily and efficiently.

Keywords: book, club management, reading management

## LISTA DE ILUSTRAÇÕES

Figura 1 – Método Kanban utilizado no aplicativo Notion .....	12
Figura 2 – Distribuição da faixa etária dos entrevistados.....	14
Figura 3 – Frequência de leitura dos entrevistados.....	15
Figura 4 – Representação do <i>model</i> no <i>back-end</i> Klubinhos.....	25
Figura 5 – Representação do <i>controller</i> no <i>back-end</i> Klubinhos.....	27
Figura 6 - Exemplo de <i>request</i> utilizando o RapidAPI.....	30
Figura 7 – Exemplo de componentização da página de calendário .....	34
Figura 8 – Requisição GET no <i>back-end</i> realizada pelo <i>front-end</i> .....	37
Figura 9 – Painel do protótipo Klubinhos no Figma.....	39
Figura 10 – Caso de uso da rede social Klubinhos .....	42
Figura 11 – Diagrama de classes do Klubinhos .....	43
Figura 12 – Modelo Conceitual do Banco de Dados .....	46
Figura 13 – Modelo Lógico do Banco de Dados.....	47
Figura 14 – Tela de login .....	50
Figura 15 – Tela de cadastro de usuário .....	51
Figura 16 – Tela para criar/participar de um clube .....	51
Figura 17 – Tela para criação de clube .....	52
Figura 18 – Tela encontrar um clube de leitura .....	53
Figura 19 – Tela para entrar em um clube .....	53
Figura 20 – Tela de Dashboard .....	54
Figura 21 – Tela de Dashboard com comentário .....	55
Figura 22 – Tela de Lista de Reuniões .....	56
Figura 23 – Tela de Criação de Reunião .....	56
Figura 24 – Tela de Edição de Reunião .....	57
Figura 25 – Tela de Lista de Eventos e Reuniões .....	58
Figura 26 – Tela de Criação de evento .....	58
Figura 27 – Tela de Edição de evento .....	59
Figura 28 – Tela do perfil do usuário .....	60
Figura 29 - Tela de edição do perfil .....	60
Figura 30 – Respostas da questão 4 .....	71

Figura 31 – Respostas da questão 5 .....	71
Figura 32 – Respostas da questão 6 .....	72
Figura 33 – Respostas da questão 7 .....	72
Figura 34 – Respostas da questão 8 .....	73
Figura 35 – Respostas da questão 9 .....	73
Figura 36 – Respostas da questão 10 .....	74
Figura 37 – Respostas da questão 11 .....	74
Figura 38 – Respostas da questão 12 .....	75
Figura 39 – Respostas da questão 13 .....	75
Figura 40 – Respostas da questão 14 .....	76
Figura 41 – Respostas da questão 15 .....	76
Figura 42 – Respostas da questão 16 .....	77
Figura 43 – Respostas da questão 17 .....	77
Figura 44 – Respostas da questão 18 .....	78
Figura 45 – Respostas da questão 19 .....	78
Figura 46 – Respostas da questão 20 .....	79
Figura 47 – Respostas da questão 21 .....	79
Figura 48 – Respostas da questão 22 .....	80
Figura 49 – Respostas da questão 23 .....	80

## LISTA DE ABREVIATURAS E SIGLAS

ABNT	- Associação Brasileira de Normas Técnicas
API	- <i>Application Programming Interface</i>
CRUD	- <i>Create, Read, Update, Delete</i>
CSS	- <i>Cascading Style Sheets</i>
E2E	- <i>End-to-end</i>
FATEC PG	- Faculdade de Tecnologia da Praia Grande
HTTP	- <i>Hypertext Transfer Protocol</i>
JSON	- <i>JavaScript Object Notation</i>
MVC	- <i>Model, View, Controller</i>
PHP	- <i>Hypertext Preprocessor</i>
REQ	- Requisito
RESTful	- <i>Representational State Transfer</i>
SQL	- <i>Structured Query Language</i>
UML	- <i>Unified Modeling Language</i>
VS Code	- <i>Visual Studio Code</i>
XAMPP	- Apache, MySQL, PHP e Pearl

## SUMÁRIO

1 INTRODUÇÃO .....	7
1.1 Objetivo Geral .....	8
1.1.1 Objetivos Específicos .....	9
2. PLANEJAMENTO DO PROJETO .....	11
3 DESENVOLVIMENTO DO PROJETO .....	13
3.1 Etapas De Desenvolvimento .....	13
3.1.1 Planejamento .....	13
3.1.2 Construção .....	16
3.1.3 Implantação .....	18
3.1.4 Avaliação e Manutenção .....	19
3.2 Critérios De Inovação .....	19
3.3 Divulgação Do Produto .....	19
4 levantamento de requisitos.....	21
4.1 Análise e Classificação dos Requisitos .....	21
4.1.1 Requisitos Funcionais .....	21
4.1.2 Requisitos Não Funcionais.....	22
5 DESENVOLVIMENTO SISTEMA/SOFTWARE/APLICATIVO.....	23
5.1 Back-end: Linguagem e Software e Framework.....	23
5.1.1 PHP E Laravel.....	23
5.1.2 Arquitetura MVC.....	25
5.1.3 Ferramentas auxiliares no desenvolvimento .....	28
5.2 Front-end: Linguagem de Software e Framework .....	31
5.2.1 React e Vite.js .....	31
5.2.2 Arquitetura de componentização.....	32
5.2.3 Comunicação com o back-end .....	35
5.2.4 Estilização e Responsividade das telas ao usuário.....	37
5.2.5 Ferramentas auxiliares no desenvolvimento .....	38
5.3 Ferramentas Utilizadas em Conjunto com o Back-end e Front-end .....	39
6 PROCESSOS.....	41

6.1 Casos de Uso.....	41
6.2 Diagrama de Classes .....	43
7 Projeto de Banco de dados .....	45
7.1 Modelo Conceitual.....	45
7.2 Modelo Lógico .....	46
7.3 Modelo Físico .....	47
8 PROJETO Técnico.....	48
8.1 Implementação.....	48
8.2 Testes.....	48
8.3 Descrição das Interfaces.....	49
8.3.1 Tela de Login.....	50
8.3.2 Tela de Cadastro de Usuário.....	50
8.3.3 Tela para criar/participar de um clube .....	51
8.3.4 Tela para criação de clube .....	52
8.3.5 Tela para encontrar um clube de leitura .....	52
8.3.6 Tela para entrar em um clube .....	53
8.3.7 Telas de Dashboard .....	54
8.3.8 Telas de Reunião .....	55
8.3.9 Telas do Calendário .....	57
8.3.10 Telas de Perfil .....	59
9 CONSIDERAÇÕES FINAIS, RECOMENDAÇÕES E LIMITAÇÕES .....	61
REFERÊNCIAS.....	62
APÊNDICE A – Questionário .....	65
APÊNDICE B – RESULTADO Do questionário.....	71
APÊNDICE C – Requisitos Funcionais .....	83
APÊNDICE D – Requisitos Não Funcionais.....	85
APÊNDICE E – MODELO FÍSICO .....	87

## 1 INTRODUÇÃO

A leitura tem um papel fundamental na história e no desenvolvimento humano, sendo uma fonte inestimável de conhecimento, proporcionando o desenvolvimento do intelecto, da imaginação e a formação de valores. O hábito de leitura transcende fronteiras e gerações, proporcionando conexões que podem ser potencializadas com clubes de leitura. Os clubes de leitura oferecem a oportunidade de explorar as obras literárias, ao mesmo tempo que se constrói comunidades que compartilham experiências e interesses e confrontam opiniões e ideias.

A ascensão rápida das tecnologias digitais nos últimos anos mudou radicalmente a maneira como as pessoas consomem e compartilham conteúdo, e a literatura também foi muito impactada com essa transformação. A popularização dos dispositivos móveis, plataformas de streaming, e-books e redes sociais redefiniu não apenas o acesso à informação, mas também a interação com textos literários. A leitura, antes restrita às páginas físicas dos livros, agora se expande para diversos formatos digitais, proporcionando flexibilidade e acessibilidade.

As redes sociais, por sua vez, se transformaram em espaços virtuais de compartilhamento de recomendações, resenhas e discussões sobre obras literárias, conectando leitores de todo o mundo. A comunidade literária cresceu e se fortaleceu tanto nestes ambientes, que até termos específicos foram cunhados para designar os usuários e produtores de conteúdo focados em livros de cada rede social. Um exemplo disso é o caso do termo “*Booktok*”, que junta a palavra “*book*” (que significa livro em inglês), e o nome da rede social “*TikTok*”, e que só no Brasil já contava com mais de 15 bilhões de visualizações até setembro de 2023 (ZANARDI; MESQUITA; SCAVACIN, 2023).

Com essas movimentações nas redes sociais, o mercado literário nacional tem se reinventado. Livrarias físicas e virtuais têm reservado espaços em suas prateleiras para seções específicas de recomendações populares nas redes sociais. O mercado literário também vem crescendo e, segundo pesquisa do Sindicato Nacional dos Editores de Livros (2023), livrarias exclusivamente virtuais passaram a representar a

maior participação no faturamento das editoras no ano de 2022. Esse crescimento tem sido sentido também nos eventos de literatura, com a tradicional Bienal Internacional do Livro do Rio de Janeiro de 2023, que contou com espaços e palestras dedicados aos criadores de conteúdo literário digital, batendo recordes de participação e vendas (CBN, 2023).

Essas tendências digitais não apenas refletem uma mudança nos hábitos de consumo, mas também apontam a necessidade de inovação no âmbito dos clubes de leitura. Os clubes de leitura tradicionais enfrentam obstáculos significativos, desde a dificuldade na organização de reuniões presenciais até a limitação na troca efetiva de ideias entre os membros. Nesse cenário, a tecnologia se apresenta como aliada para superar essas barreiras, proporcionando uma solução inovadora para revitalizar a experiência dos clubes de leitura. Muitos grupos de leitura têm surgido destas novas interações sociais. Entretanto, eles normalmente se encontram dispersos por múltiplas redes sociais e, com isso, enfrentam desafios consideráveis ao tentar proporcionar uma experiência coesa e enriquecedora para os seus membros.

Plataformas generalistas, embora úteis para diversos propósitos, não são, por natureza, projetadas para atender às dinâmicas específicas e necessidades exclusivas de um clube de leitura. A dispersão de discussões em diferentes espaços online dificulta a organização, a continuidade das interações e a gestão eficiente do conteúdo relacionado à leitura.

O desafio é, portanto, encontrar uma maneira de aproveitar os benefícios da conectividade digital sem perder a essência e a intimidade que caracterizam essas comunidades literárias. Diante desse cenário, a concepção de um aplicativo exclusivo para clubes de leitura surge como uma resposta a esses desafios, visando unificar as atividades, oferecer funcionalidades específicas para a leitura e, assim, aprimorar a coesão e a qualidade das interações dentro dessas comunidades virtuais.

## **1.1 Objetivo Geral**

Este relatório técnico tem como objetivo desenvolver uma plataforma web interativa e intuitiva para a gestão de clubes de leitura, funcionando como uma rede

social dedicada, promovendo o engajamento nas discussões de leituras e proporcionando uma comunidade dinâmica através de uma ferramenta simples e eficiente.

### **1.1.1 Objetivos Específicos**

Este trabalho tem como objetivos específicos os seguintes pontos:

- **Facilitar a Interação entre Membros:** Desenvolver funcionalidades que permitam aos usuários criarem perfis, enviarem mensagens, participarem de discussões e compartilharem informações de leitura, promovendo um ambiente colaborativo e de troca de ideias.
- **Organização de Reuniões:** Implementar um sistema de agendamento que permita aos usuários marcarem e organizarem reuniões do clube de leitura, com lembretes para garantir a participação ativa dos membros e mantendo os registros de reuniões passadas.
- **Votação de Próximas Leituras:** Desenvolver um mecanismo de votação democrático onde os membros do clube possam sugerir e votar em novos livros para leitura, garantindo que as escolhas reflitam o interesse coletivo do grupo.
- **Avaliação de Leituras Passadas:** Criar um sistema de avaliação e resenha onde os membros possam dar notas e escrever críticas sobre os livros lidos, permitindo a construção de um banco de dados com avaliações detalhadas e úteis para futuras referências.
- **Gestão de Clubes:** Fornecer ferramentas administrativas para os líderes de clubes gerenciarem os membros e atividades do clube, facilitando a coordenação e crescimento da comunidade.
- **Interface Amigável e Acessível:** Garantir que a plataforma seja intuitiva, responsiva e acessível, proporcionando uma experiência de usuário agradável em diferentes dispositivos, incluindo desktops, tablets e smartphones.

- **Segurança e Privacidade:** Implementar medidas robustas de segurança para proteger os dados dos usuários e garantir a privacidade das interações e informações compartilhadas na plataforma.

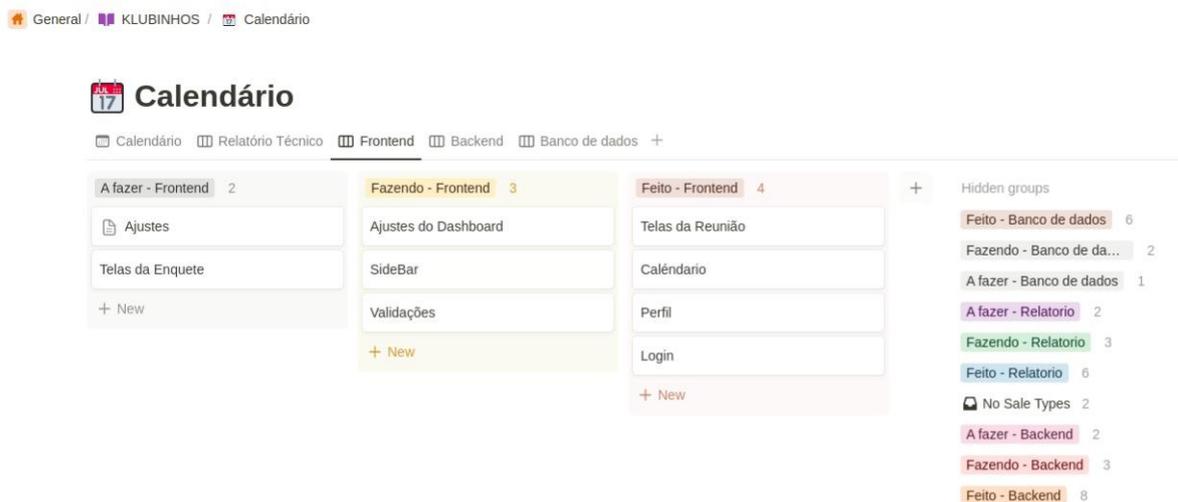
## 2. PLANEJAMENTO DO PROJETO

Para a organização do grupo e planejamento do projeto, foi escolhido utilizar uma adaptação do método Kanban. O método Kanban é um método de gerenciamento de trabalho visual desenvolvido pela Toyota na década de 1940 para melhorar a eficiência na produção (ANDERSON, 2011). Ele utiliza um quadro dividido em colunas que representam diferentes estágios de um processo. Cada tarefa é representada por um cartão que se move entre as colunas conforme o trabalho avança, ajudando a identificar e eliminar gargalos. Esse método já se mostrou eficiente em diferentes áreas, podendo ser implementado com o auxílio de diferentes ferramentas e, de modo geral, sua implementação proporciona uma melhor visualização das demandas, permitindo uma melhor organização e fluxo de trabalho (ELY, 2024). Para sua utilização neste projeto, a equipe utilizou um quadro Kanban no aplicativo Notion.

Notion é um aplicativo que combina várias funcionalidades em uma única plataforma, permitindo a criação e organização de notas, tarefas, calendários e bancos de dados em uma interface personalizável, onde os usuários podem construir suas próprias páginas e sistemas usando blocos modulares que suportam diferentes tipos de conteúdo, como texto, imagens, tabelas e mais (NOTION, 2024). Além disso, o Notion se destaca pela sua capacidade de colaboração, permitindo que várias pessoas editem e compartilhem documentos em tempo real.

No quadro Kanban utilizado, as tarefas foram divididas em colunas representando os diferentes estágios de desenvolvimento, como "A Fazer", "Em Progresso" e "Concluído" (Figura 1). Cada tarefa foi representada por um cartão, contendo detalhes específicos, prazos e responsáveis, promovendo transparência e colaboração contínua entre os membros da equipe, resultando em um desenvolvimento mais ágil e coordenado.

**Figura 1.** Método Kanban utilizado no aplicativo Notion.



Fonte: Notion, 2024.

Para uma melhor organização, o projeto foi dividido nas cinco fases abaixo, desenvolvidas em ordem:

1. Planejamento e análise: Definição dos objetivos, requisitos e recursos necessários para o projeto.
2. Design da interface e experiência do usuário: Criação de protótipos e design visual para garantir uma interface intuitiva e agradável.
3. Desenvolvimento *back-end* e *front-end*: Construção do sistema e funcionalidades, tanto no lado do servidor quanto no lado do cliente.
4. Testes e garantia de qualidade: Verificação e validação do sistema para assegurar que ele funcione corretamente e atenda aos requisitos especificados.
5. Implementação e lançamento: Implantação da aplicação em ambiente de produção e disponibilização para os usuários finais.

### 3 DESENVOLVIMENTO DO PROJETO

Com base nas necessidades de uma rede social para gestão de clube de leitura que possa atingir um amplo público de forma eficiente e na experiência prévia do grupo de trabalho, optou-se pela criação da rede social em aplicação *web*.

#### 3.1 Etapas De Desenvolvimento

As etapas de desenvolvimento seguiram as fases estabelecidas no capítulo anterior, e foram realizadas como registrado a seguir.

##### 3.1.1 Planejamento

###### a) Identificação do cliente

A aplicação é voltada para pessoas envolvidas ou com interesse em participar de clubes de leitura, organizando reuniões e leituras. Para o início do planejamento e desenvolvimento do projeto, foi utilizada a experiência dos próprios desenvolvedores com gestão de clubes de leitura online e as suas vivências em ambientes virtuais de discussão de leitura. Foram feitas também conversas informais com participantes de clubes de leitura a fim de estabelecer as ideias iniciais.

Para melhor identificação do cliente e definição dos requisitos, foi elaborado um questionário no aplicativo de gerenciamento de pesquisa *Google Forms*. O questionário contou com vinte e quatro questões, sendo vinte e duas de múltipla escolha e duas de resposta aberta e opcionais. As questões do formulário foram divididas em cinco seções, e a seleção de exibição das seções era feita com base nas respostas de algumas questões, como a existência ou não de experiência prévia em grupos de leitura e o interesse ou não em fazer parte de um. A identificação do entrevistado era opcional.

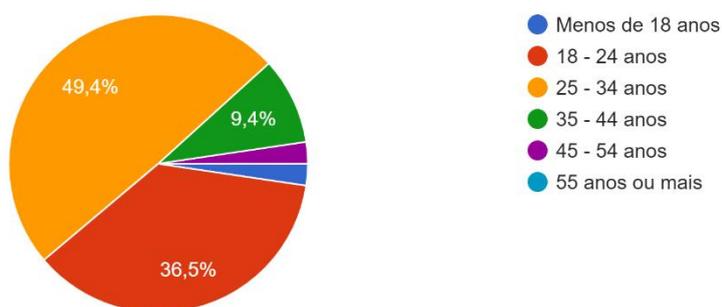
O questionário foi compartilhado em grupos pessoais dos desenvolvedores no aplicativo *Whatsapp*, além de compartilhado em redes sociais como o *Twitter*. Ao todo,

o questionário recebeu oitenta e cinco respostas, sendo a maioria na faixa etária entre 25 e 34 anos (Figura 2).

**Figura 2.** Distribuição da faixa etária dos entrevistados.

Qual a sua faixa etária?

85 respostas



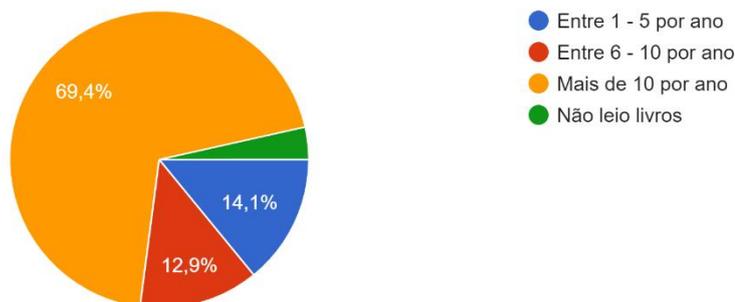
Fonte: Google Forms, 2024.

O objetivo do questionário era coletar respostas de diferentes grupos de usuários, leitores e não leitores, para ter a opinião de uma amostra heterogênea. Entretanto, é importante ressaltar que, por o questionário ter sido compartilhado nas redes sociais por grandes canais de leitura, espera-se um viés de opinião das pessoas já consumidoras de literatura que, presume-se, sejam os principais usuários da aplicação. Esse viés foi confirmado, com os entrevistados apresentando em sua maioria uma média de leitura de mais de 10 livros por ano (Figura 3), número bem maior do que a média dos leitores brasileiros, de 2,6 livro por ano (INSTITUTO PRÓ-LIVRO, 2021).

**Figura 3.** Frequência de leitura dos entrevistados.

Com que frequência você costuma ler livros?

85 respostas



Fonte: Google Forms, 2024.

O formulário na íntegra está apresentado no Apêndice A. Os demais resultados estão disponíveis no Apêndice B.

#### b) Identificação de requisitos

Para a elencar os requisitos, foram utilizadas as experiências prévias dos desenvolvedores e as respostas do questionário divulgado. Neste processo, buscou-se entender as necessidades e problemas enfrentados no gerenciamento de um clube de leitura, e elencar as possíveis soluções para estes problemas e novos que podem surgir com o uso da aplicação.

Dentre as perguntas do questionário utilizadas no levantamento dos requisitos, destacam-se as que buscam identificar o que as pessoas buscam em um clube de leitura, entender como as reuniões e escolhas de clubes de leituras ocorrem e o tipo de recursos que elas gostariam de encontrar em uma aplicação deste tipo.

#### c) Levantamento de recursos e custos

Para a implementação da aplicação *web*, foram despendidos recursos para a compra do domínio ([www.klubinhos.com](http://www.klubinhos.com)) da aplicação no site *GoDaddy*. O domínio foi reservado pela duração de um ano. Já a hospedagem da aplicação será realizada

na plataforma da AWS, também pelo período de um ano. Tanto a hospedagem quanto o domínio foram financiados pelos desenvolvedores.

Para o desenvolvimento do projeto foram utilizados os equipamentos pessoais dos desenvolvedores, além de equipamentos da FATEC de Praia Grande.

#### d) Avaliar a viabilidade

Para avaliar a viabilidade, foi realizada uma extensa busca a fim de confirmar a ausência de produtos equivalentes no mercado. Os produtos similares encontrados se apresentaram restritos à clubes específicos, não permitindo que outros clubes de fazerem uso de suas plataformas.

O formulário de sondagem de requisitos também revelou o interesse de grande parte dos entrevistados em um produto do tipo, incluindo ainda uma aceitação pela possibilidade de cobrança pelo serviço, o que permitiria uma viabilidade financeira a longo prazo.

Entretando, para garantir o desenvolvimento da primeira versão do sistema dentro do cronograma estabelecido, alguns recursos de interesse, mas julgados não essenciais, foram suprimidos com o intuito de serem adicionados em atualizações futuras após o apropriado tempo de desenvolvimento, garantindo a qualidade da aplicação.

### **3.1.2 Construção**

A construção do aplicativo foi realizada seguindo os seguintes pontos:

#### a) Segurança

Visando proteger os dados pessoais dos usuários e suas interações, algumas medidas de segurança foram implementadas na aplicação. Entre elas, o sistema conta com login para autenticação dos usuários e administradores a partir do e-mail pessoal e uma senha escolhida por eles, que é criptografada para armazenamento no banco de dados. Para garantir a segurança durante a navegação, a aplicação conta com a

criptografia, restrição e garantia de integridade dos cookies, que também expiram após um tempo, reduzindo o risco de ataques ou manipulações de terceiros.

#### b) Versionamento do código fonte

Para o versionamento do código fonte, foi utilizado o sistema de controle Git e a plataforma de hospedagem de código-fonte e arquivos com controle de versão Github. Git é um sistema de controle de versão distribuído que permite aos desenvolvedores rastreamento de alterações no código-fonte e colaborar eficientemente em projetos de software. Já o GitHub é uma plataforma de hospedagem de código que utiliza Git e oferece ferramentas adicionais para gerenciamento de projetos, revisão de código e colaboração. Juntos, Git e GitHub facilitam o desenvolvimento colaborativo e a manutenção de históricos detalhados de alterações, e ainda permitem a divulgação destes códigos com uma comunidade online (CHACON E STRAUB, 2014).

#### c) Construção da aplicação

A construção da aplicação web foi feita utilizando o Visual Studio Code (VS Code). O VS Code é um editor de código-fonte desenvolvido pela Microsoft, amplamente utilizado por desenvolvedores para escrever e editar código de maneira eficiente. É conhecido por sua leveza, rapidez e extensibilidade, oferecendo suporte para várias linguagens de programação e sistemas operacionais. O VS Code inclui funcionalidades como *debugging*, controle de versão integrado com Git, realce de sintaxe, autocompletar inteligente, além de possibilitar a inclusão de extensões para personalizar e ampliar suas capacidades. Este editor se destaca por sua interface amigável e ferramentas poderosas que ajudam a aumentar a produtividade dos desenvolvedores, o que fez com que se tornasse tão popular que 74,09% dos desenvolvedores ao redor do mundo o utilizam (STACK OVERFLOW, 2023).

#### d) Criação do teste

Os testes realizados foram de usabilidade, portabilidade, segurança e conformidade. Todos os testes foram separados nas seguintes fases: planejamento, preparação, execução e análise.

O teste de usabilidade visa garantir que a aplicação seja intuitiva e fácil de usar. Já o teste de portabilidade tem o objetivo de verificar se a aplicação se adapta bem a diferentes ambientes. Por se tratar de uma aplicação web, o acesso pode ocorrer de telas de diferentes tamanhos, incluindo dispositivos portáteis, e deve ser capaz de se adaptar e ajustar a cada uma delas. O teste de segurança, por sua vez, tem como objetivo garantir que os usuários tenham acesso somente aos recursos que seus privilégios garantem, além de garantir que as informações pessoais fiquem protegidas e, no caso de senhas e cookies, criptografadas. Por fim, nos testes de conformidade são verificados se a aplicação atende aos requisitos propostos.

A criação e execução desses testes foram realizadas repetidamente ao longo de todo o ciclo de desenvolvimento, garantindo a estabilidade e correto funcionamento da aplicação.

### **3.1.3 Implantação**

A implantação é uma etapa extremamente importante pois pode garantir a fidelização e, até mesmo, promoção da aplicação por parte dos usuários. Sendo assim, a implantação desta aplicação deve acontecer em duas etapas.

Inicialmente, a rede social será compartilhada com apenas alguns clubes de leitura selecionados, a fim de permitir os ajustes necessários e garantir o bom funcionamento da aplicação. Após este período e a conclusão das correções necessárias, o aplicativo será amplamente divulgado para acesso do público geral.

Como a aplicação foi desenvolvida com o objetivo de ser usada por qualquer pessoa e apresenta uma interface simples e intuitiva, não existe necessidade de treinamento para seu uso.

### **3.1.4 Avaliação e Manutenção**

A avaliação e manutenção da qualidade do produto serão realizadas através do contínuo monitoramento e execução de melhorias. Para garantir o perfeito funcionamento da plataforma, implementaremos práticas de testes de software regularmente para identificar e corrigir quaisquer falhas ou *bugs*. *Feedback* dos usuários será coletado constantemente por meio de pesquisas, relatórios de problemas e testes de grupos controlados, permitindo ajustes e aprimoramentos baseados nas necessidades reais dos usuários. A manutenção da plataforma incluirá atualizações regulares de segurança, otimizações de desempenho e a adição de novas funcionalidades, garantindo que a aplicação permaneça segura, eficiente e alinhada com as expectativas dos usuários.

### **3.2 Critérios De Inovação**

Os critérios de inovação da aplicação incluem a integração de funcionalidades específicas para clubes de leitura, como agendamento de reuniões, votação de próximas leituras e avaliações colaborativas de livros, que não são comumente encontradas em plataformas convencionais de redes sociais. Diferente de outras soluções no mercado, a plataforma oferece uma interface intuitiva e amigável, desenhada para facilitar a interação entre os membros dos clubes de leitura, de forma gratuita e não restrita a somente clubes específicos, permitindo que qualquer usuário possa gerenciar seu próprio grupo.

### **3.3 Divulgação Do Produto**

Para garantir uma maior visibilidade da aplicação, foi pensada uma estratégia de *marketing* ampla. Após o fim dos períodos de testes, serão utilizados uma combinação de canais de divulgação, incluindo o site oficial com informações detalhadas sobre as funcionalidades e benefícios da plataforma, campanhas nas redes sociais (Facebook, Instagram, Twitter) para alcançar um público mais amplo.

Os canais nas redes sociais foram criados todos com o mesmo nome e endereço de acesso, facilitando a identificação e busca. Será feito também colaborações com influenciadores literários, como o que foi feito para a divulgação do questionário, e blogs especializados em leitura. Além disso, também existe a possibilidade de realização de webinars e eventos online para demonstrar o uso da plataforma e engajar potenciais usuários. Esta abordagem tem o objetivo de maximizar o alcance e atrair uma comunidade ativa de amantes de livros.

## **4 LEVANTAMENTO DE REQUISITOS**

Os requisitos são fundamentais para o desenvolvimento do projeto, uma vez que auxiliam na coleta, análise e documentação das necessidades, expectativas e restrições das partes interessadas envolvidas no projeto.

### **4.1 Análise e Classificação dos Requisitos**

A análise e classificação dos requisitos são importantes para o desenvolvimento do software, uma vez que auxiliam a entender o que realmente é necessário, ajudam a evitar problemas e estabelecer a procedência do projeto, para que assim os recursos possam ser utilizados de maneira eficaz para que o projeto possa atender as necessidades dos usuários.

Os requisitos funcionais e não funcionais foram classificados e suas descrições podem ser encontradas no Apêndice C e D do projeto.

#### **4.1.1 Requisitos Funcionais**

**REQ-001 Cadastro de usuário**

**REQ-002 Criação e participação do clube do livro**

**REQ-003 Gerenciar postagens**

**REQ-004 Gerenciar reunião**

**REQ-005 Gerenciar agenda de eventos**

**REQ-006 Gerenciar enquete**

**REQ-007 Votos**

**REQ-008 Avaliações de livros**

**REQ-009 Gerar relatórios**

**REQ-010 Gerenciar integrantes**

**REQ-011 Consultar integrantes**

**REQ-012 Gerenciar administrador**

#### **4.1.2 Requisitos Não Funcionais**

- REQ-001 Segurança**
- REQ-002 Portabilidade**
- REQ-003 Implementação**
- REQ-004 Implementação**
- REQ-005 Usabilidade**
- REQ-006 Confiabilidade**
- REQ-007 Manutentabilidade**
- REQ-008 Entrega**
- REQ-009 Segurança**
- REQ-010 Usabilidade**
- REQ-011 Escalabilidade**

## 5 DESENVOLVIMENTO SISTEMA/SOFTWARE/APLICATIVO

Tendo em vista que a rede Klubinhos trata-se de uma rede social, a aplicação pode receber muitos acessos simultâneos, além de contar com diversas funcionalidades em sua rede, tornando-se necessário um desenvolvimento técnico que suporte uma grande quantidade de acessos a aplicação, a fim de garantir uma excelente experiência ao usuário, incentivo à leitura e a conexão com novas pessoas, de acordo com seu interesse.

O sistema foi desenvolvido utilizando uma combinação de tecnologias de ponta para garantir uma experiência de usuário otimizada e uma implementação eficiente dos recursos desejados.

Para isso, foram utilizadas ferramentas compatíveis com a necessidade do projeto e de grande reconhecimento pelo mercado de tecnologia, sendo elas percorridas nos tópicos a seguir.

### 5.1 *Back-end*: Linguagem e Software e Framework

Quanto a linguagem de código e sua arquitetura de desenvolvimento do *back-end* para garantir a aplicabilidade plena do Klubinhos, o desenvolvimento deste projeto envolve a integração de várias tecnologias, sendo as principais PHP com Laravel e, para banco de dados, MySQL.

#### 5.1.1 PHP E Laravel

O desenvolvimento do *back-end* do Klubinhos foi realizado com o uso integrado de diversas tecnologias-chave, sendo as mais notáveis o PHP como linguagem de programação e o Laravel como framework. A seguir, serão exploradas em detalhes essas ferramentas e como elas contribuíram para a construção de uma arquitetura de *back-end* robusta e eficiente.

O PHP é uma linguagem de programação amplamente utilizada para o desenvolvimento no lado do servidor. Escolhida para o desenvolvimento do back-end deste projeto, o PHP oferece uma série de vantagens.

Primeiro, o PHP é simples de usar, mas poderoso, com uma sintaxe que é fácil de entender e seguir. Segundo o PHP é uma linguagem de programação muito eficiente, capaz de lidar com uma grande quantidade de solicitações e operações simultaneamente. Terceiro, o PHP é amplamente adotado na indústria, o que significa que existe uma grande quantidade de documentação, tutoriais e recursos disponíveis para ajudar os desenvolvedores a resolverem problemas e implementar funcionalidades (LOCAWEB, 2023).

Para complementar o PHP, foi utilizado o Laravel, um framework PHP que proporciona uma estrutura robusta para o desenvolvimento de aplicativos web. O Laravel é conhecido por sua sintaxe expressiva, que torna o código mais legível e fácil de manter.

O Laravel segue o padrão de "convenção sobre configuração", o que significa que ele fornece soluções sensatas para a maioria das tarefas comuns de desenvolvimento web, sem a necessidade de configurações extensas. Isso facilita muito a implementação de diferentes funcionalidades do sistema (ALURA, 2015).

Uma das facilidades que o Laravel trouxe para o desenvolvimento do projeto foi a biblioteca já disponível para autenticação de usuários, manipulação de dados e a criação de *Applications Programming Interfaces* (APIs).

A autenticação, fornecida pelo Laravel na aplicação web, facilitou com seus recursos para criação de sistemas de login e registro, redefinição de senhas, verificação de e-mails e proteção contra-ataques de força bruta, proporcionando uma experiência de usuário segura e confiável.

O Laravel também ofereceu uma interface intuitiva para manipular dados de banco de dados. Isso permitiu que ao longo do desenvolvimento fossem realizadas as operações de CRUD (*Create, Read, Update, Delete*) em seus dados de forma simples e eficiente, sem ter que escrever consultas *Structured Query Language* (SQL) complexas (GUIMARÃES, 2019).

Além das facilidades que o PHP possa trazer em conjunto com o Laravel, foi utilizado também o XAMPP em conjunto com o Mysql, para integração com o banco de dados e o fácil manuseio para a estrutura de dados ao longo do desenvolvimento.

### 5.1.2 Arquitetura MVC

A arquitetura *Model-View-Controller* (MVC) é um padrão de design de software de grande utilização no meio da programação, sendo um dos principais componentes em aplicações e projetos estruturados em PHP que utilizam o framework Laravel (COODESH, 2024). Esta arquitetura, é pensada para separar a regra de negócios (*Model*), aos dados que são apresentados (*View*) e o controle do fluxo de entrada (*Controller*), no que facilita a manutenção e na promoção da escalabilidade do código.

O model é o componente que representa os dados e a lógica de negócios. Dentro do contexto do Laravel, os *Models* são os elementos que interagem diretamente com o banco de dados, realizando as operações fundamentais de CRUD e estabelecendo as regras de validação para os dados, garantindo assim a sua integridade.

**Figura 4:** Representação do *model* no *back-end* Klubinhos

```
10 references | 0 implementations
class Club extends Model
{
    use HasFactory;

    0 references
    protected $fillable = [
        'name',
        'nick_club',
        'description',
        'banner',
        'user_id',
        'banner_imagem'
    ];

    0 references
    public static $rules = [
        'nick_club' => 'unique:clubs,nick_club',
    ];
}
```

Fonte: Autoral, 2024

O *model* apresentado na Figura 4 refere-se à entidade do banco de dados do *Club*, tabela responsável pelas informações do clube de leitura na aplicação. Ela se estende a classe *Model*, permitindo interagir com a tabela *clubs* no banco de dados. É utilizado o *trait HasFactory* para facilitar a criação de instâncias e define os atributos permitidos para preenchimento em massa (*\$fillable*), como os atributos *name*, *nick\_club*, *description*, *banner*, *user\_id*, e *banner\_imagem*. Além disso, especifica uma regra de validação (*\$rules*) que garante que o campo *nick\_club* seja único na tabela *clubs*.

Este modelo serve como uma ponte entre a aplicação Laravel e a tabela *clubs*, proporcionando operações seguras e eficientes de criação, leitura, atualização e exclusão de registros. As medidas de segurança incluem a proteção contra atribuição em massa e a garantia de unicidade de certos campos.

Já o *view* é responsável pela apresentação dos dados ao usuário. No caso do Laravel, as *Views* são compostas por arquivos *Blade*, porém em nosso escopo de projeto optamos por utilizar como *view* uma estrutura diferente do que é composta no php com laravel, estamos utilizando como front-end uma aplicação a parte para garantir uma melhor manutenção de código, do qual será explicado nos tópicos abaixo.

Por fim temos o *controller*, sendo este responsável como um intermediário entre o *Model* e a *View*, processando as solicitações do usuário, interagindo com o *Model* para recuperar ou modificar dados conforme necessário, e passando esses dados para a *View* para que sejam exibidos ao usuário.

**Figura 5:** Representação do *controller* no back-end Klubinhos

```
6 references | 0 implementations
class ClubController extends Controller
{
    1 reference | 0 overrides
    public function registerClub(Request $request)
    {
        $validator = Validator::make($request->all(), [
            'nick_club' => 'required|unique:clubs,nick_club',
        ]);

        if ($validator->fails()) {
            return response()->json([
                'message' => 'Nick Club already exists'
            ], 401);
        }

        $club = new Club;
        $club->name = $request->name;
        $club->nick_club = $request->nick_club;
        $club->description = $request->description;
        $club->banner = $request->banner;
        $club->user_id = $request->user_id;
        $club->save();

        $clubIntegrantesController = new ClubIntegrantesController();
        $integranteRequest = new Request([
            'user_id' => $club->user_id,
            'club_id' => $club->id,
            'role' => 'admin',
        ]);

        $clubIntegrantesController->create($integranteRequest);

        return response()->json([
            'club' => $club,
            "message" => "Club record created"
        ], 201);
    }
}
```

Fonte: Autoral, 2024

Na Figura 5, foi utilizado o *controller* que é chamado de ClubController e tem um método chamado registerClub. Este método é responsável por registrar um novo clube em nossa base de dados. No início do método, um objeto validator é criado para validar os dados recebidos na requisição. A regra de validação especificada aqui é que o campo nick\_club é obrigatório e deve ser único na tabela, se a validação falhar, uma resposta *JavaScript Object Notation* (JSON) (uma formatação para transmitir dados de um sistema para outro, de forma organizada) é retornada com uma mensagem de erro e um código de status *Hypertext Transfer Protocol* (HTTP) 401. Se

a validação for bem-sucedida, um novo objeto do model de Club é criado e seus atributos são preenchidos com os dados da requisição. O objeto Club é então salvo no banco de dados.

Em seguida, um novo objeto ClubIntegrantesController é criado e o método create é chamado nele. Este método é responsável por criar um registro na tabela de integrantes do clube. Para isso, uma nova requisição é criada com o user\_id do clube e o tipo do usuário no clube, que é definido como 'admin'.

Finalmente, uma resposta JSON é retornada com o objeto Club recém-criado e uma mensagem de sucesso. O código de status HTTP 201 é retornado, indicando que um novo recurso foi criado com sucesso.

Por fim, o PHP, tendo como sua vantagem a incorporação da arquitetura MVC em sua estrutura, facilita o desenvolvimento de aplicações web robustas e escaláveis. Além disso, ele oferece uma série de conveniências que otimizam o processo de desenvolvimento. Todos estes elementos contribuem para uma experiência de desenvolvimento rápida, eficiente e organizada.

### **5.1.3 Ferramentas auxiliares no desenvolvimento**

Além das facilidades que o PHP, junto com o laravel podem auxiliar na etapa de toda arquitetura do nosso back-end, foram utilizadas como ferramentas auxiliares que contribuíram para verificar a integridade com os dados que estamos gerindo e a escalabilidade do sistema, sendo estes:

#### **a) XAMPP**

O XAMPP é um pacote de servidor web de código aberto que inclui Apache, MySQL, PHP e Perl, usado principalmente para testar aplicações web em um servidor local. Ele oferece facilidade de instalação, permite executar ambientes de desenvolvimento locais, é compatível com várias plataformas e suporta diversas tecnologias. Isso facilita o desenvolvimento rápido e eficiente de aplicações PHP. Ele permite que o desenvolvimento do projeto seja realizado rapidamente por um servidor

web local e já iniciando suas aplicações PHP sem a necessidade de configurações complexas.

A integração do XAMPP com PHP traz vários benefícios. Entre eles, destaca-se a possibilidade de testar sites e aplicações antes de publicá-los em um servidor web real. Isso ajuda os desenvolvedores a identificarem e corrigirem erros mais rapidamente, melhorando a qualidade final do projeto.

Ademais, o XAMPP suporta outras tecnologias importantes como MySQL para gerenciamento de banco de dados e Apache como servidor web. Isso fornece aos desenvolvedores uma plataforma completa para desenvolver, testar e construir aplicações web. Assim, o XAMPP é uma ferramenta valiosa no arsenal de um desenvolvedor web.

#### **b) phpMyAdmin**

O phpMyAdmin é uma ferramenta de código aberto para administração de servidores MySQL e MariaDB, que oferece uma interface gráfica do usuário para simplificar operações complexas. Ele suporta a criação, cópia, exclusão e renomeação de bancos de dados, tabelas, visualizações, campos e índices, além de permitir a execução de consultas SQL. A ferramenta também facilita a importação e exportação de dados, suporta a administração de múltiplos servidores, permite a criação de consultas complexas, gerencia permissões de usuário e transforma dados armazenados. É amplamente utilizado em hospedagens baseadas em cPanel e pode ser instalado em servidores próprios.

Esta ferramenta foi de grande auxílio para verificar a integridade com o banco de dados, bem como a aplicação está se comportando ao receber requisições realizadas pelo usuário.

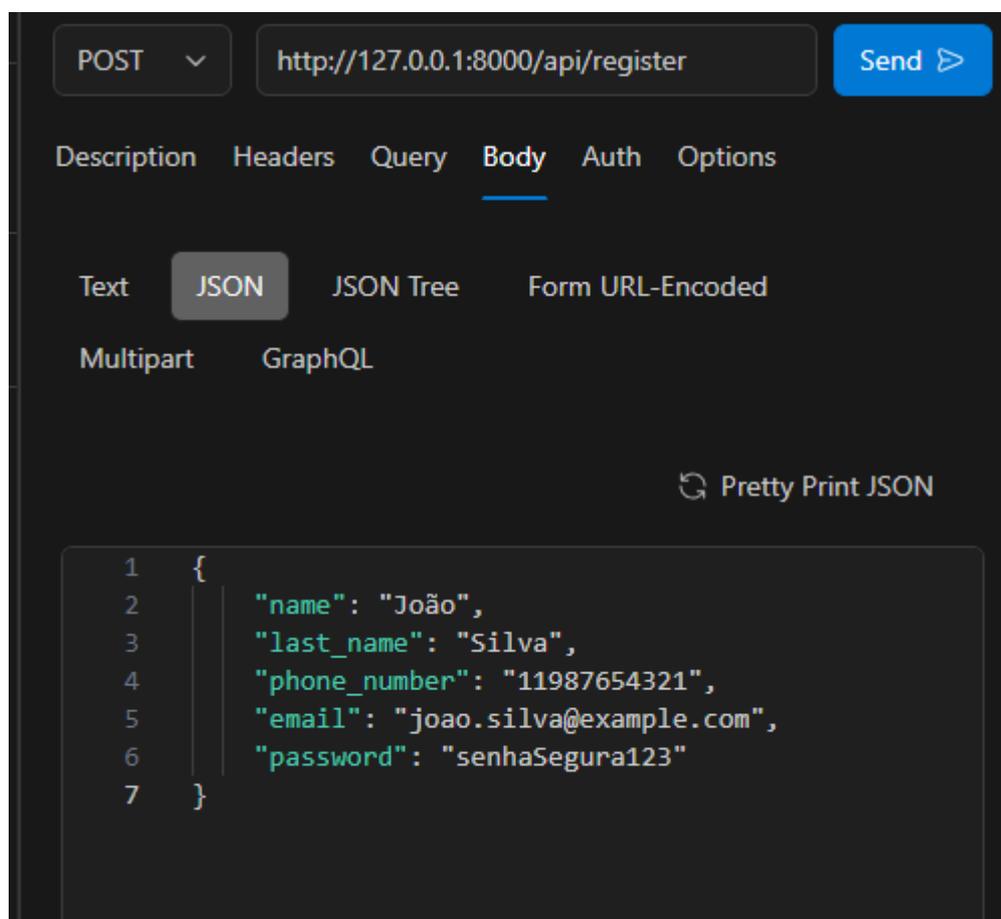
#### **c) RapidAPI**

O RapidAPI é uma extensão para o Visual Studio Code (VSCode), do qual é uma ferramenta essencial para realizar os testes de APIs integrado, geração automática de código, suporte para parâmetros e variáveis, sincronização de projetos com a nuvem e um manipulador de links de terminal. Ela aumenta a produtividade,

oferece flexibilidade e é fácil de usar, tornando o teste e a integração de APIs mais eficientes e acessíveis.

Ao longo do desenvolvimento do projeto foi realizado diversos testes de integração das rotas de API criadas, a fim de garantir a persistência dos dados em nosso banco de dados. Na Figura 6 temos um exemplo de como foi utilizada a extensão para o auxílio da codificação.

**Figura 6:** Exemplo de *request* utilizando o RapidAPI



Fonte: Autoral, 2024

No código apresentado na imagem, está sendo batido no *endpoint* de registrar o usuário da aplicação. Para isso, no corpo da requisição são enviados os campos obrigatórios para registro, como o nome, sobrenome, telefone, e-mail e a senha que o usuário for criar.

## 5.2 *Front-end*: Linguagem de Software e Framework

Neste tópico serão abordados os elementos de desenvolvimento da *view* da aplicação, o qual é responsável pela apresentação dos dados ao usuário. No caso do Laravel, as *views* são compostas por arquivos *Blade*, um sistema de *templates* que permite a renderização de conteúdo dinâmico de forma simples e intuitiva, melhorando a experiência do usuário. Contudo, com o objetivo de otimizar o desenvolvimento do projeto e garantir uma melhor manutenibilidade, foi optado por seguir um modelo de *view* apartado, utilizando uma linguagem de programação mais consolidada no mercado e com maiores componentes para agilizar o processo de entregar uma interface plena e de fácil construção por parte do desenvolvedor.

A linguagem de programação optada foi o Javascript através da biblioteca React, em conjunto com o *framework* do Vite.js, além de combinar outras ferramentas como o Bootstrap e TypeScript.

### 5.2.1 React e Vite.js

O desenvolvimento do front-end Klubinhos, foi implementado com o uso integrado de diversas tecnologias-chave, sendo as mais notáveis o React como linguagem de programação e o Vite.js como framework. A seguir será explorado em detalhes essas ferramentas e como elas contribuíram para a construção de uma arquitetura do *front-end* leve e eficiente para a experiência do cliente.

React é uma biblioteca JavaScript de código aberto, muito utilizada para criar interfaces de usuário interativas e reativas. Com o React, é possível dividir a interface do usuário em componentes reutilizáveis, o que facilita a construção e manutenção de grandes aplicações web.

Vite.js é uma ferramenta leve e extremamente rápida para construção de aplicações web, conhecida pelo seu rápido início e compilação instantânea, graças ao uso de carregamento de módulos em tempo real durante o desenvolvimento. Vite suporta React, Vue.js e outros frameworks populares (DIGITALPRODUCTSDEV, 2021).

A combinação de React com Vite oferece diversas vantagens:

- **Desenvolvimento Rápido:** A união de React e Vite cria um ambiente de desenvolvimento rápido, possibilitando a visualização imediata de mudanças no navegador, eliminando o tempo de espera entre fazer uma alteração de código e ver o resultado.
- **Melhor Desempenho:** Vite otimiza a compilação e execução do código, resultando em uma melhor experiência para o usuário final, devido a tempos de carregamento mais rápidos e menor uso de recursos.
- **Suporte ao TypeScript:** Tanto React quanto Vite têm suporte nativo ao TypeScript, benefício para quem prefere usar TypeScript para adicionar tipagem estática ao código, ajudando a detectar erros mais cedo e melhorar a manutenção do código.
- **Ecosistema Robusto:** React possui um amplo ecossistema de bibliotecas e ferramentas que podem ser facilmente integradas ao projeto. Vite, por outro lado, é uma ferramenta nova em rápido desenvolvimento que ganha popularidade rapidamente devido à sua abordagem inovadora para o desenvolvimento web.

Além disso, Vite.js oferece um processo de desenvolvimento rápido e eficaz, permitindo organizar o código de maneira mais organizada e sustentável através da abordagem de desenvolvimento modular. O Vite também otimiza o desempenho, alcançando tamanhos de arquivo menores e carregamentos de aplicativos mais rápidos.

### 5.2.2 Arquitetura de componentização

No front-end do Klubinhos, foi explorada a arquitetura de componentes do React com o objetivo de criar uma interface de usuário modular e fácil de manter. Esta arquitetura se baseia na ideia de dividir a interface do usuário em partes independentes e reutilizáveis, chamadas de componentes.

Cada funcionalidade criada no desenvolvimento do Klubinhos foi pensada em componentização, desde a criação de perfis, passando pela gestão e criação de

clubes, até a interação no fórum, foi implementada como um componente separado e autônomo. Esta abordagem traz inúmeros benefícios para a manutenção e atualização do sistema.

Na Figura 7, pode-se observar na classe Calendário a forma como foi utilizada a componentização e como são tratados dentro do serviço de front-end. A estrutura da classe está organizada de forma a separar a lógica de recuperação e exibição de eventos em um componente específico (GOMES, 2020).

Um desses benefícios é a facilidade de trabalhar com cada componente de forma independente. Isso significa que, se uma funcionalidade precisa ser alterada ou atualizada, os desenvolvedores podem se concentrar apenas naquele componente específico, sem afetar o restante do sistema. Este nível de isolamento ajuda a evitar bugs e torna o processo de atualização muito mais eficiente.

Além disso, a arquitetura de componentes pode levar a um código mais limpo e organizado. Esta organização é possível porque durante o desenvolvimento pode encapsular a lógica de negócios e a interface do usuário de cada funcionalidade em um único componente. Isso torna o código mais fácil de entender e depurar e, também, pode facilitar a reutilização de código em diferentes partes do sistema (APPMASER, 2023).

Portanto, a componentização no React foi uma forma eficaz encontrada ao longo do desenvolvimento do projeto de trazer uma aplicação eficaz de construir interfaces de usuário complexas, mas de uma maneira que seja fácil de entender, manter e atualizar.

Figura 7: Exemplo de componentização da página de calendário

```

1 import React, { useState, useEffect } from 'react';
2 import axios from 'axios';
3 import Cookies from 'js-cookie';
4 import ptBR from 'date-fns/locale/pt-BR';
5 import { format, addDays } from 'date-fns';
6 import { Link } from 'react-router-dom';
7 import styles from './Calendario.module.css';
8
9
10 export function Calendario() {
11   const [events, setEvents] = useState([]);
12   const [loading, setLoading] = useState(true);
13   const club_id = Cookies.get('club_id');
14
15   useEffect(() => {
16     async function fetchEvents() {
17       try {
18         const responseReuniao = await axios.get('http://127.0.0.1:8000/ap
19         const responseCalendar = await axios.get('http://127.0.0.1:8000/a
20
21         const reuniaoEvents = responseReuniao.data.map(reuniao => ({
22           ...reuniao,
23           data_evento: reuniao.data_reuniao, // Convertendo o campo data_
24           tipo: 'reuniao' // Adicionando o tipo de evento
25         }));
26
27         const allEvents = [...responseCalendar.data, ...reuniaoEvents];
28         // Ordenar eventos por data crescente
29         allEvents.sort((a, b) => new Date(a.data_evento) - new Date(b.dat
30         setEvents(allEvents);
31         setLoading(false);
32       } catch (error) {
33         console.error(error);
34         setLoading(false);
35       }
36     }
37
38     fetchEvents();
39   }, [club_id]);
40
41   // Organize os eventos por data
42   const eventsByDate = {};
43   events.forEach(event => {
44     const eventDate = format(addDays(new Date(event.data_evento), 1), "dd
45     if (!eventsByDate[eventDate]) {
46       eventsByDate[eventDate] = [];
47     }
48     eventsByDate[eventDate].push(event);
49   });
50
51   // Função para formatar a descrição com quebra de linha a cada 60 caract
52   const formatarDescricao = (descricao: string) => {
53     if (descricao.length > 60) {
54       return descricao.match(/.{1,60}/g).join('\n');
55     }
56     return descricao;
57   };
58
59   return (
60     <div className="container mb-4">
61       <div className="form-group">
62         <div className="row">
63           <div className="col">
64             <b>Calendário</b>
65           </div>
66           <div className="col-md-4">
67             <Link to="/createcalendario">
68               <button>Criar evento</button>
69             </Link>
70           </div>
71         </div>
72       </div>
73
74       {loading ? (
75         <div>Carregando...</div>
76       ) : (
77         Object.keys(eventsByDate).map(date => (
78           <div className="row mt-4">
79             <div className="col">
80               <span className="mt-4">{date}</span>
81             </div>
82             <div>
83               <ul className="list-group">
84                 {eventsByDate[date].map(event => (
85                   <li key={event.id} className="list-group-item mt-4 ${sty
86                     (event.tipo === 'reuniao') ? (
87                       <span className="material-symbols-outlined" style={{
88                         color: '#5b6b77' }}>
89                       <span className="material-symbols-outlined" style={{
90                         color: '#5b6b77' }}>
91                       <div className="d-flex">
92                         <div className="mt-1">
93                           <span className="d-block">{event.titulo}</span>
94                           <span style={{ color: '#5b6b77' }}>{formatarDescric
95                         </div>
96                       </div>
97                     </li>
98                   </li>
99                 </ul>
100               </div>
101             </div>
102           </div>
103         </div>
104       )
105     </div>
106   );
107 }
108

```

Fonte: Autoral, 2024

### 5.2.3 Comunicação com o back-end

A comunicação entre o front-end (a parte da aplicação web com a qual o usuário interage diretamente, como a interface gráfica e os elementos visuais) e o back-end (a parte da aplicação que processa as solicitações do usuário, acessa o banco de dados e realiza as operações necessárias para fornecer as respostas) da aplicação web Klubinhos é realizada através das APIs RESTful (uma abordagem para a comunicação entre sistemas que utiliza o protocolo HTTP para transferir dados e é baseada nos princípios do REST) que são fornecidas pelas rotas disponibilizadas no Laravel do back-end (DEV MEDIA, 2016). Este último é responsável por administrar as requisições HTTP (protocolo de comunicação utilizado na navegação web) de forma assíncrona.

Esta funcionalidade de requisições assíncronas significa que o sistema não é impedido ou bloqueado enquanto aguarda pela resposta do servidor. Esta abordagem assegura uma experiência de usuário altamente suave e extremamente responsiva. O motivo deste comportamento é que o usuário não fica em espera que o servidor responda antes de poder interagir com a aplicação. Em vez disso, o usuário pode continuar a interagir com a aplicação, enquanto o sistema aguarda pela resposta do servidor, aumentando assim a eficiência geral e a satisfação do usuário. Hugo Costa, em seu artigo, Introdução REST (MEDIUM, 2019), explica da seguinte forma como é realizada a troca de informações entre o back-end e o front-end:

"No REST a implementação do cliente (front-end) e do servidor (back-end) são feitas separadamente sem nenhuma interferência de um ou outro. Isso significa que os dois podem ser desenvolvidos e refatorados separadamente. Mas uma hora ou outra o cliente irá precisar se comunicar com a aplicação REST para poder usufruir dos dados que a mesma disponibiliza para consumo."

A API RESTful é uma interface eficiente e flexível para a comunicação entre sistemas, baseada nos princípios do REST. Ela apresenta separação clara entre cliente (o front-end) e servidor (o back-end), independência de estados armazenados no servidor (cada solicitação do cliente contém todas as informações necessárias para

o servidor processá-la, sem depender de informações armazenadas anteriormente no servidor), possibilidade de cache (permitindo que os dados sejam armazenados temporariamente para melhorar o desempenho), uso consistente de métodos HTTP (como GET para recuperar dados, POST para criar novos recursos, PUT para atualizar recursos existentes, DELETE para excluir recursos e PATCH para atualizar parcialmente recursos) e arquitetura em camadas, que baseia-se em dividir o sistema em componentes independentes que se comunicam entre si através de interfaces bem definidas. As operações HTTP incluem GET, POST, PUT, DELETE e PATCH. A API RESTful é escalável, capaz de lidar com um grande número de solicitações simultâneas, flexível e simples de usar e manter (Figura 8).

**Figura 8:** Requisição *GET* no *back-end* realizada pelo *front-end*

```

export function Calendario() {

  const [events, setEvents] = useState([]);
  const [loading, setLoading] = useState(true);
  const club_id = Cookies.get('club_id');

  useEffect(() => {
    async function fetchEvents() {
      try {
        const responseReuniao = await axios.get(`http://127.0.0.1:8000/api/reuniao/getAllReuniaoByClub/${club_id}`);
        const responseCalendar = await axios.get(`http://127.0.0.1:8000/api/calendar/getAllEventsByClub/${club_id}`);

        const reuniaoEvents = responseReuniao.data.map(reuniao => ({
          ...reuniao,
          data_evento: reuniao.data_reuniao, // Convertendo o campo data_reuniao para o mesmo formato que os eventos do calendário
          tipo: 'reuniao' // Adicionando o tipo de evento
        }));

        const allEvents = [...responseCalendar.data, ...reuniaoEvents];
        // Ordenar eventos por data crescente
        allEvents.sort((a, b) => new Date(a.data_evento) - new Date(b.data_evento));
        setEvents(allEvents);
        setLoading(false);
      } catch (error) {
        console.error(error);
        setLoading(false);
      }
    }
  });
}

```

Fonte: Autoral, 2024

Para assegurar uma experiência de usuário fluida e interativa, a arquitetura de comunicação do sistema é cuidadosamente projetada para permitir uma interação eficiente entre o *front-end* e o *back-end*.

As requisições HTTP, que é a entrada principal de comunicação entre o *front-end* e o *back-end*. Em resumo, a comunicação eficiente entre o *front-end* e o *back-end*, possibilitada pelas APIs RESTful do Laravel e o tratamento assíncrono das requisições HTTP, contribui para uma experiência de usuário superior, caracterizada por interações fluidas e respostas rápidas.

#### 5.2.4 Estilização e Responsividade das telas ao usuário

O design do Klubinhos foi pensado e criado com a responsividade desde o início de seu desenvolvimento. Essa escolha foi feita para garantir uma experiência de usuário consistente e fluida em uma variedade de diferentes dispositivos e telas das mais diversas resoluções, independentemente do tamanho da tela ou do sistema operacional.

Para estilizar a interface do usuário de maneira eficaz e visualmente atraente, foi utilizada a linguagem de estilo *Cascading Style Sheets* (CSS). Essa linguagem permite uma grande quantidade de controle sobre a aparência da aplicação, desde a cor e o tamanho do texto até o layout das páginas (TECNOBLOG, 2021). Através do uso de CSS, ao longo do desenvolvimento foi capaz de criar um design que não só agrada de uma forma estática, mas que proporciona uma experiência consistente e intuitiva para o usuário em toda a aplicação e na jornada da nossa rede de leitura.

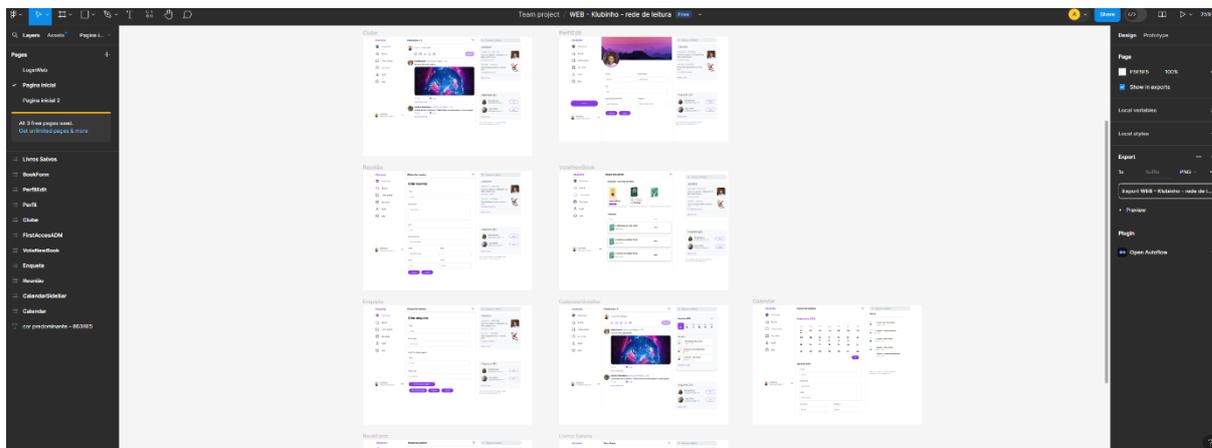
Além disso, para auxiliar na estilização e acelerar o processo de desenvolvimento, foi utilizado o Bootstrap. O Bootstrap é um dos frameworks CSS mais populares e conhecidos na comunidade de tecnologia, conhecido por sua flexibilidade e eficácia. Ele oferece uma variedade de componentes pré-estilizados que podem ser facilmente integrados ao sistema. O uso do Bootstrap facilita o processo de desenvolvimento e, também, garante um design consistente e atraente, já que os componentes seguem as mesmas regras de design.

### **5.2.5 Ferramentas auxiliares no desenvolvimento**

No desenvolvimento do *front-end* do Klubinhos, foram utilizadas ferramentas para garantir a eficiência e eficácia do processo. Duas das ferramentas mais notáveis que foram utilizadas para auxiliar no desenvolvimento do *front-end* foram o Figma e o Servidor do Vite.js:

#### **a) Figma:**

O Figma é uma ferramenta de design de interface do usuário baseada em navegador que é amplamente utilizada na indústria do design. Ele permitiu que fosse realizado no início de planejamento do projeto uma visão preliminar e mais aprofundada do protótipo, e colaborou em designs de interface do usuário em tempo real, tudo dentro de uma única plataforma. Os seus recursos utilizados foram o desenho vetorial, prototipagem interativa e compartilhamento do board entre a equipe. No projeto Klubinhos, o Figma foi fundamental para o processo de criação de telas, proporcionando um ambiente de design intuitivo e colaborativo (Figura 9).

**Figura 9:** Painel do protótipo Klubinhos no Figma

Fonte: Autoral, 2024

### **b) Servidor Vite.js**

O Servidor do Vite.js, auxiliou na capacidade de visualização e teste das telas criadas. O Vite.js é uma ferramenta que oferece um servidor de desenvolvimento extremamente rápido. Com esta funcionalidade, foi possível fazer alterações no código e ver os resultados instantaneamente, sem a necessidade de recarregar toda a página. Isso torna o desenvolvimento e a depuração de telas um processo muito mais rápido e eficiente. Além disso, o Vite.js também otimiza a compilação e execução do código, resultando em tempos de carregamento mais rápidos e menor uso de recursos.

Essas duas ferramentas, quando usadas em conjunto, oferecem uma experiência de desenvolvimento de *front-end* rápido e eficiente.

### **5.3 Ferramentas Utilizadas em Conjunto com o *Back-end* e *Front-end***

Além das ferramentas mencionadas em cada tópico específico do *back-end* e *front-end*, foram utilizadas outras que foram manuseadas em conjunto tanto com o *back-end*, quanto ao *front-end*. Sendo estas descritas a seguir:

**a) GitHub**

O GitHub foi utilizado como a plataforma principal para o gerenciamento de repositórios do projeto. Ele facilitou o controle de versões e permitiu a colaboração eficaz entre os membros da equipe. O GitHub também ajudou a rastrear e gerenciar mudanças no código do projeto, tornando o processo de desenvolvimento mais organizado.

**b) Draw.io**

Draw.io foi usado para criar diagramas UML e casos de uso. Esta ferramenta de diagramação online permitiu a visualização clara da estrutura do projeto e do fluxo de processos, facilitando a compreensão e o planejamento do sistema.

**c) GoDaddy:**

GoDaddy foi o site escolhido para a hospedagem do projeto. A plataforma oferece uma variedade de serviços, incluindo registrador de domínios, hospedagem web e e-mail profissional, garantindo que o site esteja acessível para os usuários finais.

## 6 PROCESSOS

Para garantir um desenvolvimento de sistemas robusto e complexo como o Klubinhos é necessário criar um processo de atividades padronizadas e bem definidas que orientam o desenvolvimento e a implantação. Esses processos são vitais para assegurar que o trabalho seja realizado com um alto nível de qualidade e excelência.

A *Unified Modeling Language* (UML) foi utilizada para a construção dos processos para a rede social Klubinhos. A UML é uma linguagem de modelagem que auxilia na visualização, especificação, construção e documentação dos artefatos de um sistema de software. Ela fornece uma forma padronizada de representar os processos e facilita a comunicação entre os membros da equipe (BOOCH; RUMBAUGH; JACOBSON, 2006).

### 6.1 Casos de Uso

Um caso de uso é uma descrição detalhada das ações que um ator deve realizar para alcançar um objetivo específico. Apresenta-se, então, um "ator" que pode ser um usuário humano, um sistema externo ou um componente de software (PROGRAMMER, 2015). Cada caso de uso descreve uma sequência de eventos que ocorrem durante a interação entre o ator e o sistema, geralmente representada em um diagrama de caso de uso.

Os casos de uso são ferramentas valiosas para especificar requisitos funcionais e ajudar as equipes de desenvolvimento a entenderem e projetar sistemas de acordo com as necessidades dos usuários.

Os casos de uso, utilizados para garantir uma plena experiência e usabilidade do sistema, estão apresentados na Figura 10.

**Figura 10:** Caso de uso da rede social Klubinhos



Fonte: Autoral, 2024

O sistema Klubinhos permite aos administradores e integrantes realizarem várias funções. Os administradores podem adicionar e remover membros, aprovar novos integrantes, promover administradores, gerar relatórios, criar e editar clubes do livro e reuniões. Os integrantes podem criar contas, entrar em clubes do livro, editar

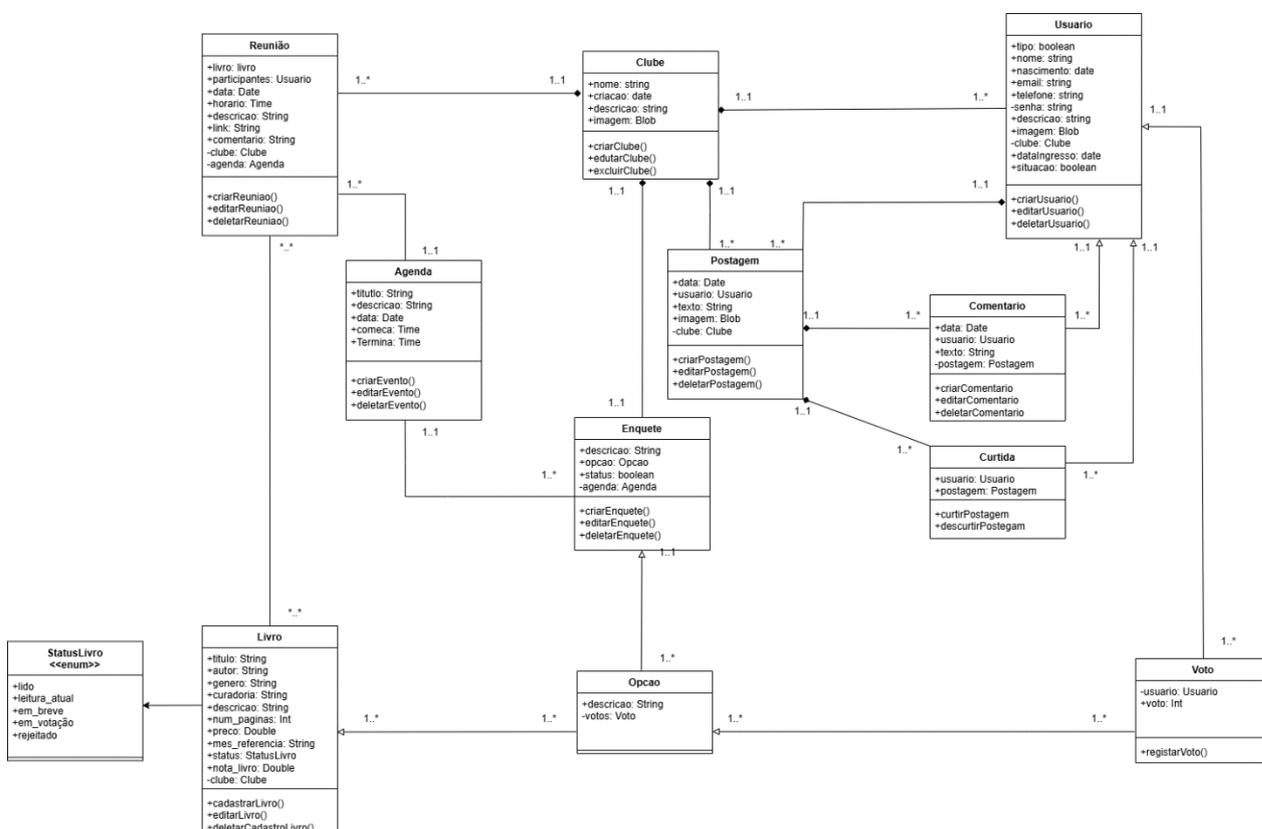
perfis, criar e editar postagens e enquetes, votar em enquetes, sugerir reuniões e acessar e publicar em páginas de reuniões.

## 6.2 Diagrama de Classes

O diagrama de classes é um tipo de diagrama estrutural no UML que descreve a estrutura de um sistema mostrando as classes do sistema, seus atributos, operações ou métodos e as relações entre os objetos (LUCIDCHART, 2024).

No projeto Klubinhos, o diagrama de classes foi utilizado para representar a estrutura do sistema, de forma clara e organizada (Figura 11). Auxiliando no processo de desenvolvimento e a entender as diferentes entidades envolvidas no sistema, bem como seus relacionamentos, responsabilidades e colaborações. Este diagrama foi fundamental para visualizar a arquitetura do sistema e facilitar a comunicação entre os membros da equipe.

Figura 11: Diagrama de classes do Klubinhos



Fonte: Autoral, 2024

O Klubinhos é um sistema de gerenciamento de clubes de leitura que utiliza um diagrama de classes para representar suas funcionalidades. Ele é composto por várias classes, cada uma com seus próprios atributos, métodos e relacionamentos. Por exemplo, a classe Usuário representa os usuários do sistema, com atributos como nome e e-mail, e permite a criação, edição e exclusão de usuários. Essa classe está relacionada com outras, como Clube, Postagem e Comentário.

Outra classe importante é a Clube, que representa os próprios clubes de leitura, com atributos como nome e descrição, e permite a criação, edição e exclusão de clubes. Ela se relaciona com usuários, reuniões, postagens e livros. O sistema também possui classes para gerenciar reuniões, agendas, postagens, comentários, curtidas, enquetes, votos e livros, proporcionando uma gestão completa e envolvente de clubes de leitura. Os usuários podem se cadastrar, criar clubes, participar de reuniões, interagir por meio de postagens e comentários, realizar enquetes, registrar votos e gerenciar livros dentro dos clubes.

## **7 PROJETO DE BANCO DE DADOS**

O banco de dados é o sistema que gerencia, organiza, armazena e recupera informações de maneira eficiente, visando a segurança dos dados ou conferências futuras.

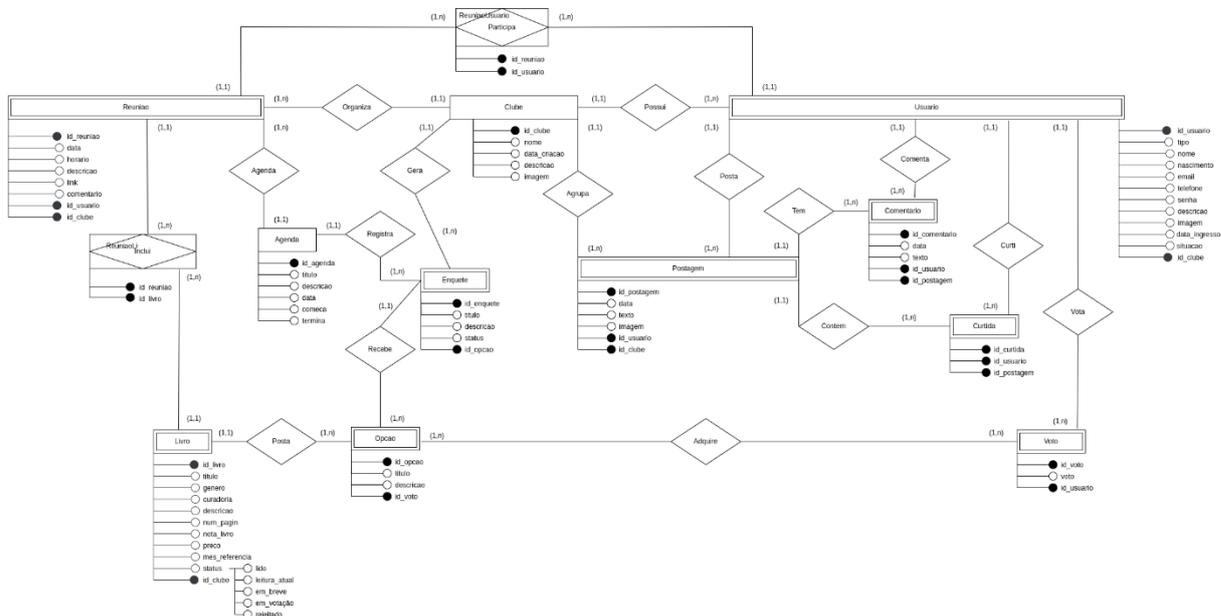
O banco de dados do projeto foi pensado e desenvolvido para atender as necessidades da aplicação, prezando a garantia de que os dados seriam armazenados de forma organizada e segura, permitindo que sejam facilmente acessados e manipulados conforme o necessário.

### **7.1 Modelo Conceitual**

Um modelo conceitual de banco de dados é uma representação abstrata que ilustra as principais entidades, conceitos e seus relacionamentos dentro de um domínio específico.

Como parte do processo de construção do projeto, o diagrama (Figura 12) foi desenvolvido para permitir uma visualização clara das entidades, assim como seus dados e relacionamentos. Ele estabelece uma base sólida e fornece orientação para o desenvolvimento da rede social para clubes de leitura, permitindo uma compreensão abrangente das interações entre as diferentes partes do sistema.

**Figura 12 – Modelo Conceitual do Banco de Dados**

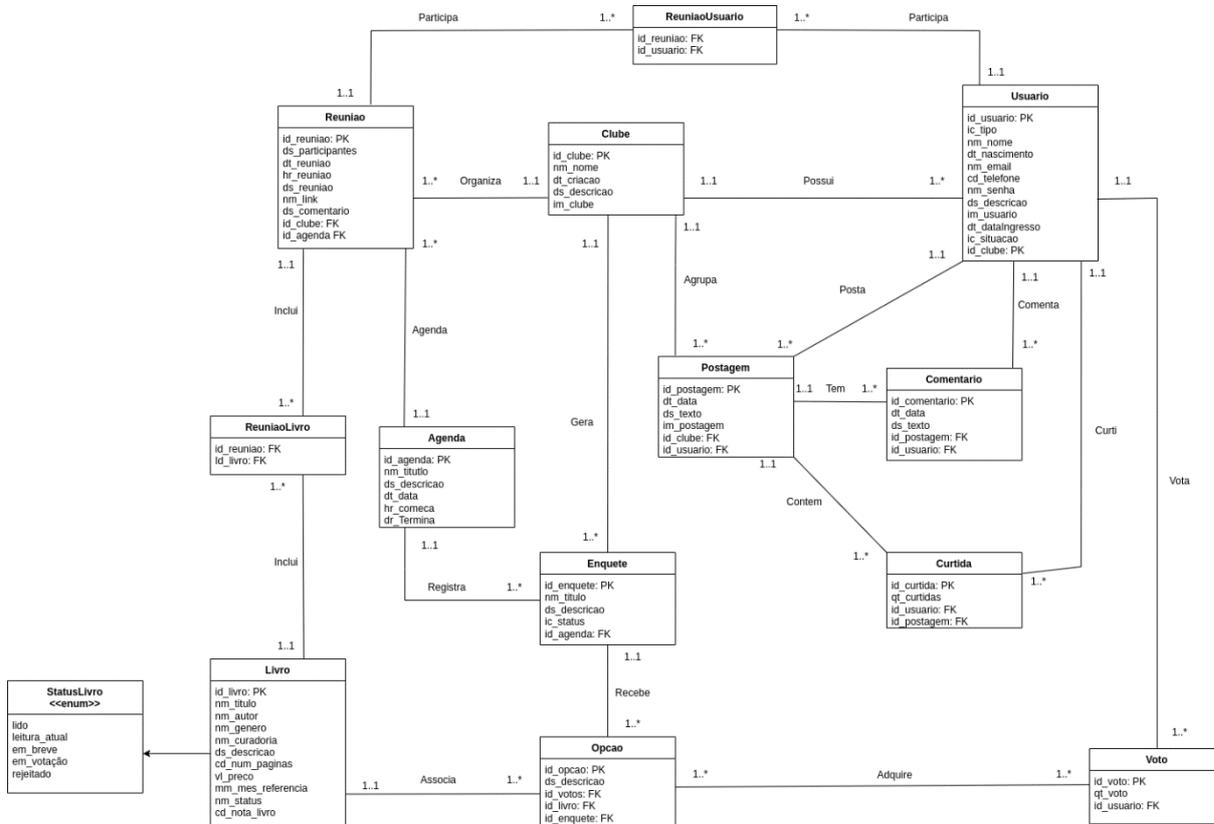


Fonte: Autoral, 2024

## 7.2 Modelo Lógico

Um modelo lógico de banco de dados (Figura 13) é uma representação mais detalhada e específica do modelo conceitual, trazendo com mais detalhes a maneira na qual as entidades serão representadas e relacionadas dentro do banco de dados (AMAZON WEB SERVICES, 2024). O modelo inclui a definição exatas das tabelas, atributos, chaves primárias e estrangeiras que serão utilizadas para organizar os dados.

**Figura 13 – Modelo Lógico do Banco de Dados**



Fonte: Autoral, 2024

### 7.3 Modelo Físico

O modelo físico do banco de dados é a representação detalhada do modelo lógico, apresentando como os dados serão armazenados no sistema de gerenciamento do banco de dados (SGBD), incluindo a estrutura das tabelas, chaves primárias, chaves estrangeiras e tipos de dados.

O script SQL usado para criação das tabelas e inserção dos dados pode ser encontrado no Apêndice E – Modelo Físico.

## 8 PROJETO TÉCNICO

Finalizada a parte de desenvolvimento da aplicação web, será dado início a última etapa de validações e testagens em massa de produto, realizando testes de ponta a ponta, também conhecido como *End-to-end* (E2E), coletando dados de desempenho e usabilidade do serviço.

### 8.1 Implementação

A implementação do serviço será feita após a conclusão de todo o desenvolvimento do projeto, como também realizado o teste de ponta-a-ponta nos servidores locais.

Desde modo, a implementação ainda está em fase de homologação, tendo em vista que o projeto está em sendo validado localmente e correspondendo as expectativas que se esperam nos requisitos funcionais e não funcionais, conforme mencionados anteriormente neste relatório técnico.

### 8.2 Testes

Os testes de software é uma etapa que acompanha desde a fase inicial do sistema, sendo já realizadas em suas primeiras linhas de código. As fases dos testes que foi realizado no projeto seguiu um processo estruturado, que vai desde a preparação até a avaliação final do sistema. Este processo é fundamental para assegurar a qualidade e a funcionalidade correta do software antes de sua implantação

Este ciclo de vida dos testes é essencial para minimizar erros e garantir que o software atende aos padrões de qualidade e funcionalidade esperados. Em cada fase, o sistema passa por rigorosos processos de validação e correção, assegurando uma entrega de produto de alta qualidade ao cliente (OBJECTIVE, 2022).

Deste modo, conforme explicado anteriormente, foi realizado o teste de ponta a ponta da aplicação ao todo, para garantir uma completa funcionalidade e

redundância do software. Após estas etapas, serão realizados os testes em produção, porém de modo controlado e com um número reduzido de usuários para colher *feedbacks* e analisar as informações extraídas por estes seletos usuários.

Após realizado o teste de produção com o seletto grupo, o projeto será encaminhado para as fases finais da entrega do produto.

O plano de implantação do Klubinhos seguirá as seguintes etapas, para garantir sua plena funcionalidade:

**Preparação:** Certificar de que todos os requisitos do sistema estão instalados e funcionando corretamente, incluindo servidores, banco de dados e dependências de software;

**Instalação e Configuração:** Será feita a instalação e configuração do sistema Klubinhos no ambiente de produção e disponibilização em seu domínio ([www.klubinhos.com](http://www.klubinhos.com)). Incluindo a configuração do banco de dados, a instalação de todas as dependências e a configuração de parâmetros específicos do sistema.

**Testes:** Realizar novamente testes rigorosos de ponta a ponta para garantir que o sistema esteja funcionando corretamente no ambiente de produção.

**Lançamento:** Será feita a disponibilidade do sistema no domínio reservado e será feito o monitoramento de perto para resolver quaisquer problemas que possam surgir.

**Manutenção:** Realizar periodicamente o monitoramento e a manutenção contínua do sistema para garantir seu desempenho e estabilidade.

Como a aplicação visa ser aberta ao público geral, intuitiva e de fácil utilização, não será necessário treinamentos nem capacitações para seu uso.

### 8.3 Descrição das Interfaces

As figuras abaixo representam as telas da aplicação com o design e identidade visual anteriormente projetadas.

### 8.3.1 Tela de Login

A tela de login (Figura 14) é a primeira tela de interação direta do usuário com a aplicação, nesta tela o usuário pode optar por seguir em cadastrar-se ou entrar com seus dados, se já cadastrados.

**Figura 14:** Tela de login

A imagem mostra a interface de login do sistema Klubinhos. No canto superior esquerdo, há o texto "KLUBINHOS" em uma barra azul. O formulário de login é centralizado e contém o seguinte conteúdo:

- Um cabeçalho de boas-vindas: "Bem-vindo(a) ao Klubinho" e um link "Sem conta? Cadastre-se" no canto superior direito.
- O título "Login" em negrito.
- Dois campos de entrada: "e-mail" e "senha", cada um com o rótulo "Entre com o seu e-mail" e "Entre com a sua senha" respectivamente.
- Um botão azul "Entrar" centralizado abaixo dos campos.
- Um link "Esqueceu a senha?" no canto inferior direito do formulário.

Fonte: Autoral, 2024

### 8.3.2 Tela de Cadastro de Usuário

A tela de cadastro de usuário (Figura 15) é o ponto de partida para novos membros, sendo projetada para coletar informações importantes para a criação do perfil dos usuários, incluindo nome, sobrenome, e-mail, telefone e senha.

**Figura 15:** Tela de cadastro de usuárioClique'."/>

Bem-vindo(a) ao Klubinho [Tem uma conta? Clique](#)

### Cadastro

Nome  Sobrenome

Informe seu email

E-mail

Telefone

Nº de celular

Informe sua senha

Senha

[Cadastrar](#)

Fonte: Autoral, 2024

### 8.3.3 Tela para criar/participar de um clube

A tela para criar ou participar de um clube (Figura 16) permite com que o usuário possa escolher se prefere criar um clube ou participar de um clube já existente.

**Figura 16:** Tela para criar/participar de um clube

Seja bem-vindo(a) ao Klubinho

[Criar Clube](#)

[Participar de Clube](#)

Fonte: Autoral, 2024

### 8.3.4 Tela para criação de clube

A tela para criação de um clube (Figura 17) é a tela na qual o usuário (administrador) pode criar o nome e endereço do clube.

**Figura 17:** Tela para criação de clube

A imagem mostra a interface de usuário para a criação de um clube. No canto superior esquerdo, há o texto "KLUBINHOS" em uma barra lateral de cor roxa. O conteúdo principal é um formulário branco centralizado com o título "Crie um Clube". O formulário contém:

- Uma mensagem de boas-vindas: "Bem-vindo(a) usuário".
- O título "Crie um Clube".
- Dois campos de entrada de texto rotulados "Nome do Clube" e "Endereço do Clube".
- Um campo de entrada de texto rotulado "@klubinho" com um ícone de lupa à direita.
- Um botão roxo "Criar Clube" na base do formulário.
- Um link azul "Buscar Clube Existente" na base do formulário.

Fonte: Autoral, 2024

### 8.3.5 Tela para encontrar um clube de leitura

Na tela para encontrar um clube de leitura (Figura 18), o usuário pode digitar o endereço (*nickname*) para fazer parte de um clube de leitura.

**Figura 18:** Tela encontrar um clube de leitura

KLUBINHOS

Bem-vindo(a)

**Faça parte de um clube!**

Digite o apelido do clube

klubinho

[Clique Aqui para Criar Clube](#)

Pesquisar

Fonte: Autoral, 2024

### 8.3.6 Tela para entrar em um clube

Na tela para entrar em um clube (Figura 19), o usuário pode clicar em “Entrar” para fazer parte do clube ou “Voltar” para retornar a tela para encontrar um clube de leitura.

**Figura 19:** Tela para entrar em um clube

KLUBINHOS

Bem-vindo(a)

**Deseja entrar em Bookaholics?**

Entrar

Voltar

Fonte: Autoral, 2024

### 8.3.7 Telas de Dashboard

Na tela de Dashboard (Figuras 20 e 21) o usuário pode interagir com os outros membros do clube, criar postagens e comentários, visualizar os integrantes do clube e o calendário com os eventos e reuniões da semana.

**Figura 20:** Tela de Dashboard

The screenshot displays the 'Bookaholics' dashboard. On the left, a sidebar under 'KLUBINHOS' lists navigation options: Home, Enquete, Reunião, Calendário, and Perfil. Below the sidebar is a profile card for 'Liz Clara Catarina Viana Bookaholics'. The main content area is titled 'Bookaholics' and features a post creation prompt: 'No que você está pensando?' with a 'Postar' button. Below this are three posts:

- Post 1: By Tiago Luiz Campos, discussing 'Harry Potter e a Pedra Filosofal' by J.K. Rowling.
- Post 2: By Tiago Luiz Campos, discussing 'Fundação' by Isaac Asimov.
- Post 3: By Liz Clara Catarina Viana, discussing 'O Nome do Vento' by Patrick Rothfuss.

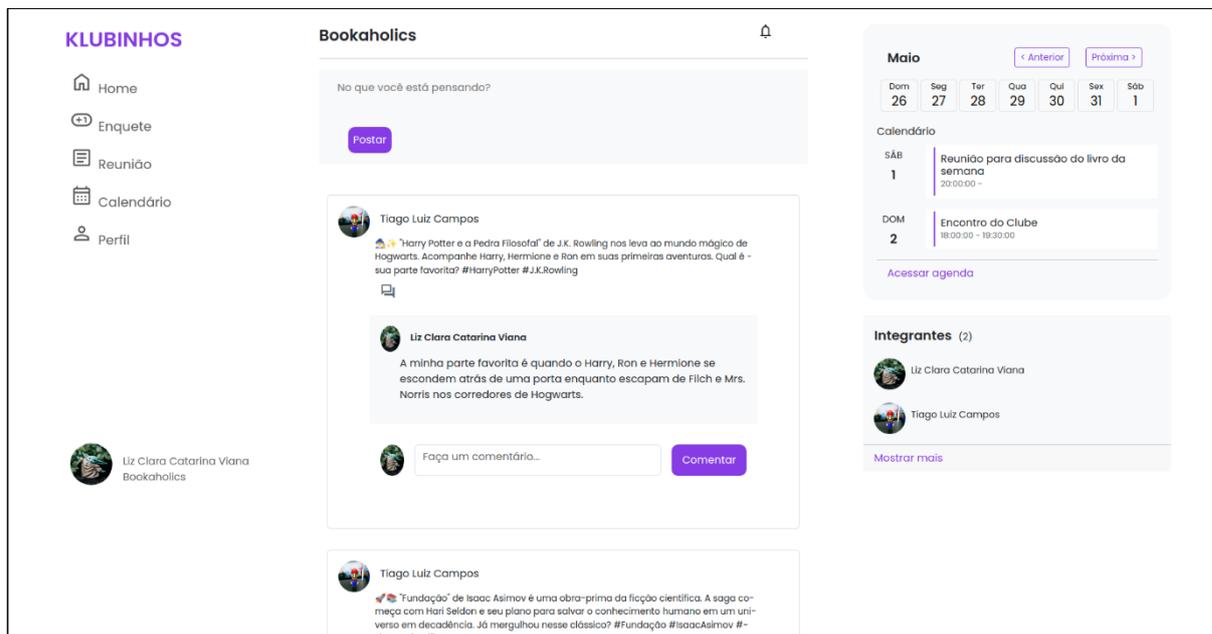
On the right side, there is a calendar for the month of May. The calendar shows two events:

- SAB 1: Reunião para discussão do livro da semana (20:00:00 -)
- DOM 2: Encontro do Clube (18:00:00 - 18:30:00)

Below the calendar is a section for 'Integrantes (2)' listing two members: Liz Clara Catarina Viana and Tiago Luiz Campos, with a 'Mostrar mais' link.

Fonte: Autoral, 2024

**Figura 21:** Tela de Dashboard com comentário

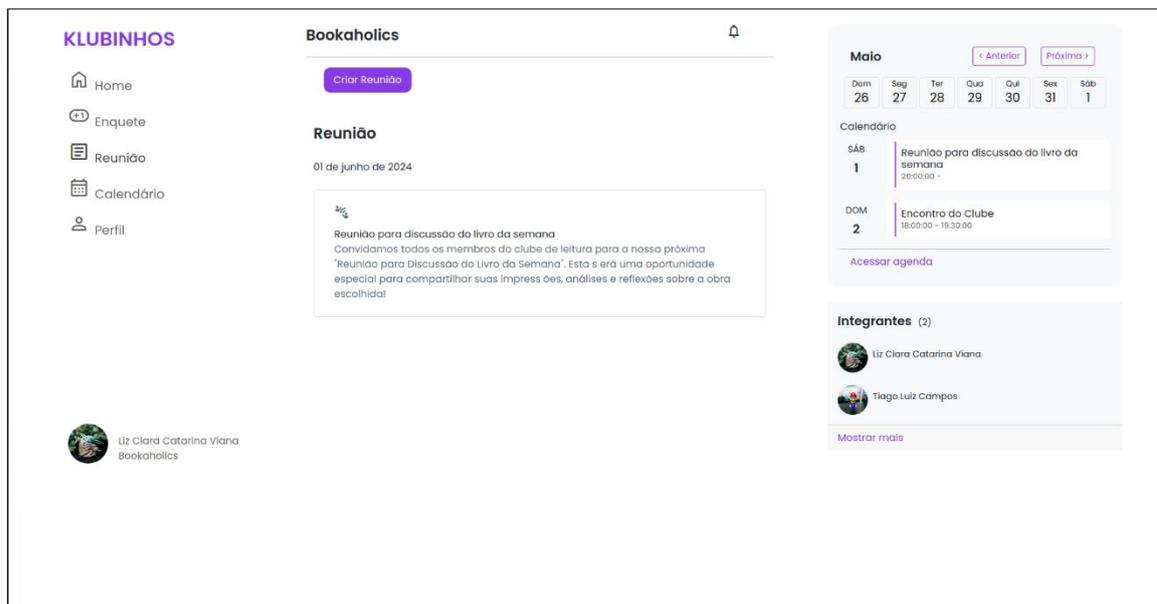


Fonte: Autoral, 2024

### 8.3.8 Telas de Reunião

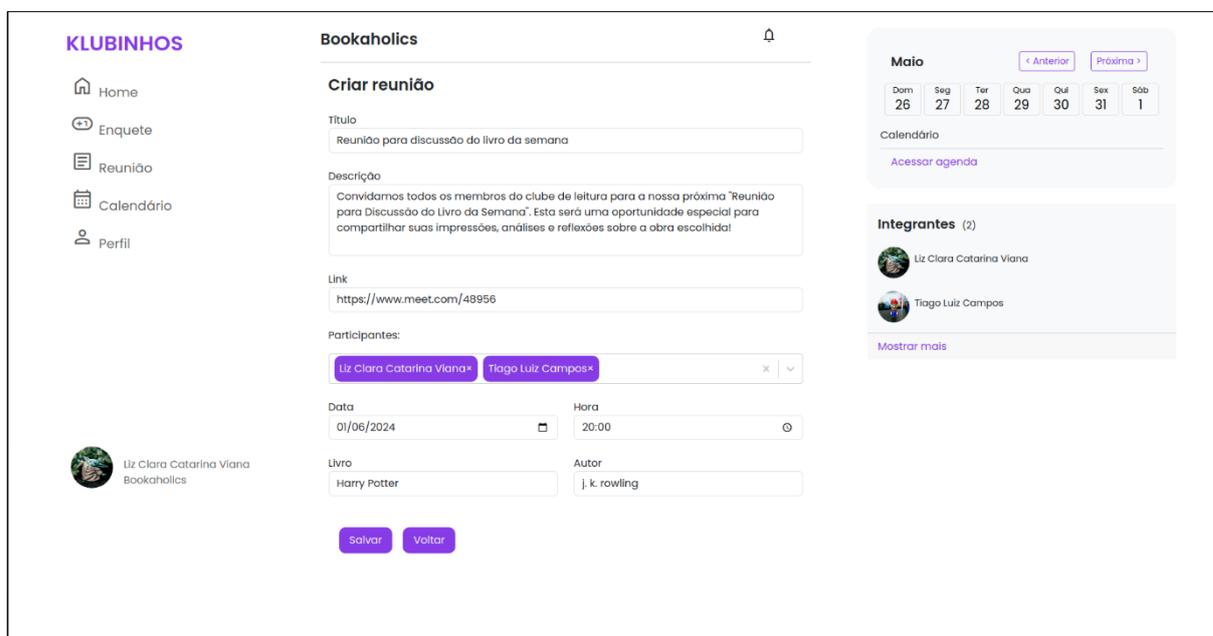
Na área de reuniões (Figuras 22 a 24), os usuários podem visualizar a lista de reuniões criadas, criar uma reunião ou editar uma reunião já criada.

**Figura 22:** Tela de Lista de Reuniões



Fonte: Autoral, 2024

**Figura 23:** Tela de Criação de Reunião



Fonte: Autoral, 2024

**Figura 24:** Tela de Edição de Reunião

**KLUBINHOS**

Home  
Enquete  
Reunião  
Calendário  
Perfil

**Bookaholics**

**Editar reunião**

Título  
Reunião para discussão do livro da semana

Descrição  
Convidamos todos os membros do clube de leitura para a nossa próxima "Reunião para Discussão do Livro da Semana". Esta será uma oportunidade especial para compartilhar suas impressões, análises e reflexões sobre a obra escolhida!

Link  
https://www.meet.com/48956

Data  
01/06/2024

Hora  
20:00

Livro  
Harry Potter

Autor  
j. k. rowling

Salvar Voltar

**Maio**

Dom	Sáb	Ter	Qua	Qui	Sex	Sáb
26	27	28	29	30	31	1

Calendário

SÁB 1  
Reunião para discussão do livro da semana  
20:00:00 -

DOM 2  
Encontro do Clube  
18:00:00 - 19:30:00

[Acessar agenda](#)

**Integrantes (2)**

- Liz Clara Catarina Viana
- Tiago Luiz Campos

[Mostrar mais](#)

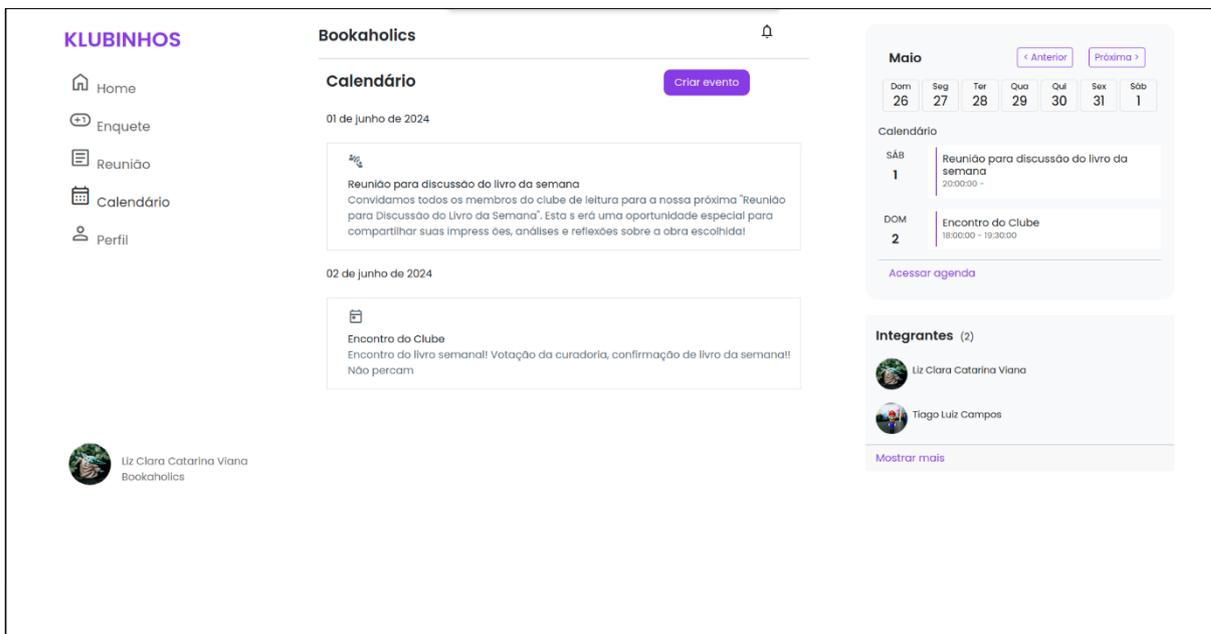
Liz Clara Catarina Viana  
Bookaholics

Fonte: Autoral, 2024

### 8.3.9 Telas do Calendário

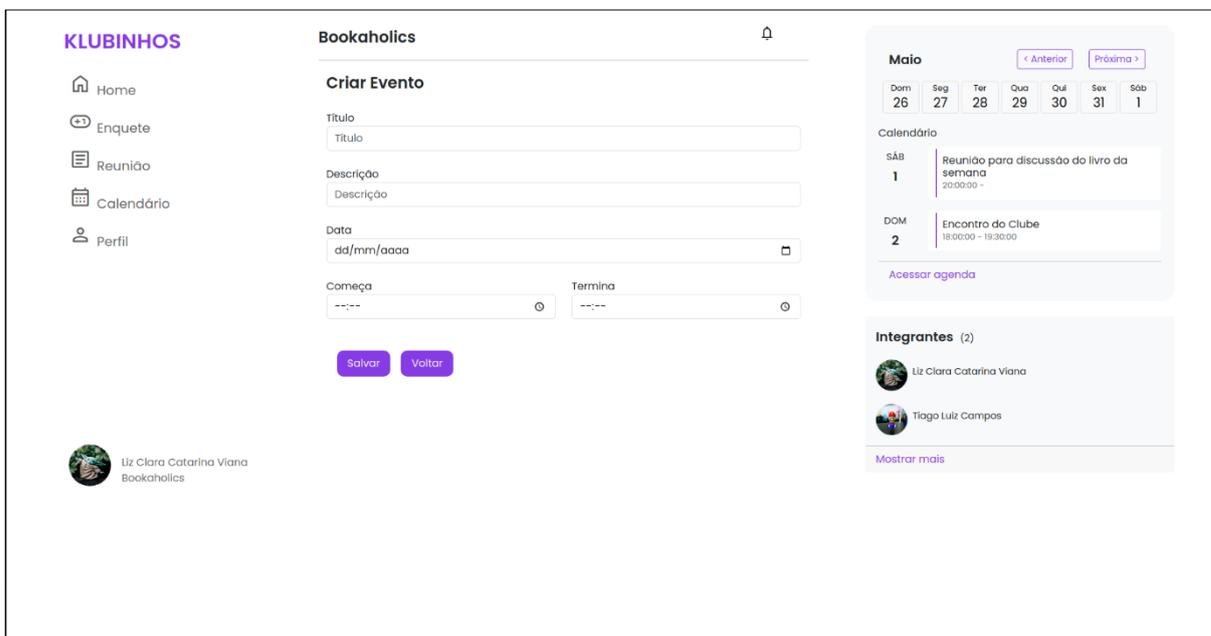
Na área do calendário (Figuras 25 a 27), os usuários podem visualizar a lista de eventos e reuniões criados, podem criar um evento ou editar um evento já criado.

**Figura 25:** Tela de Lista de Eventos e Reuniões



Fonte: Autoral, 2024

**Figura 26:** Tela de Criação de evento



Fonte: Autoral, 2024

**Figura 27:** Tela de Edição de evento

**KLUBINHOS**

Home  
Enquete  
Reunião  
Calendário  
Perfil

**Bookaholics**

**Editar Evento**

Título  
Encontro do Clube

Descrição  
Encontro do livro semanal! Votação da curadoria, confirmação de livro da semana! N

Data  
02/06/2024

Começa  
18:00

Termina  
19:30

Salvar Voltar

**Maio**

Dom	Sáb	Ter	Qua	Qui	Sex	Sáb
26	27	28	29	30	31	1

Calendário

SÁB 1  
Reunião para discussão do livro da semana  
20:00:00 -

DOM 2  
Encontro do Clube  
18:00:00 - 19:30:00

[Acessar agenda](#)

**Integrantes (2)**

Liz Clara Catarina Viana

Tiago Luiz Campos

[Mostrar mais](#)

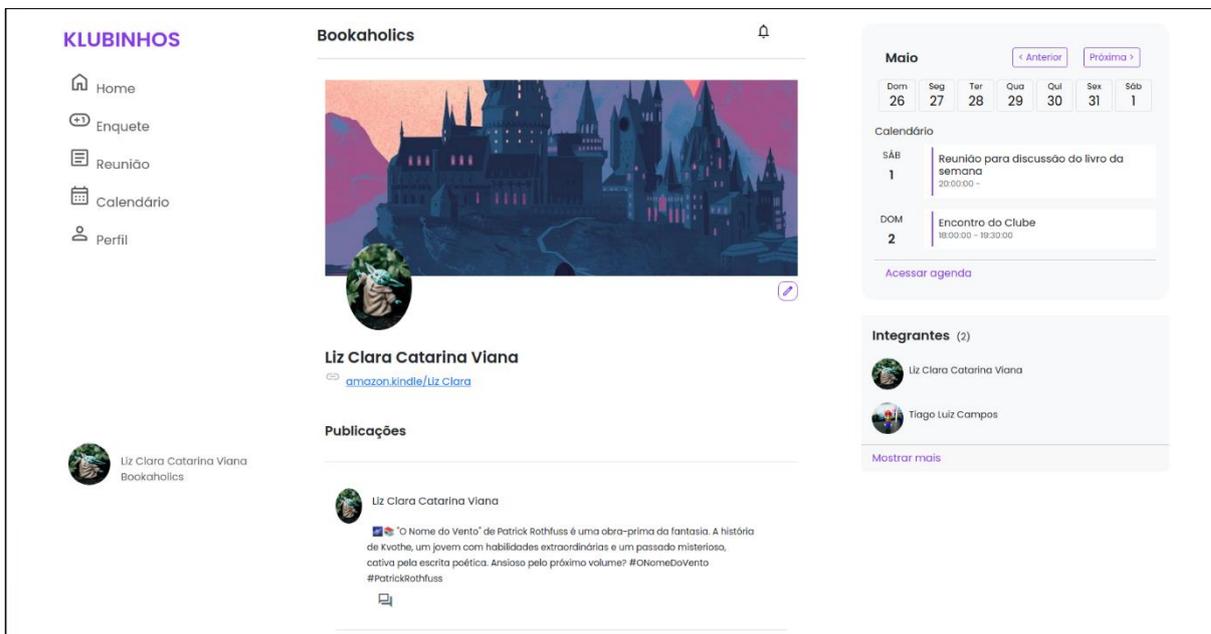
Liz Clara Catarina Viana  
Bookaholics

Fonte: Autoral, 2024

### 8.3.10 Telas de Perfil

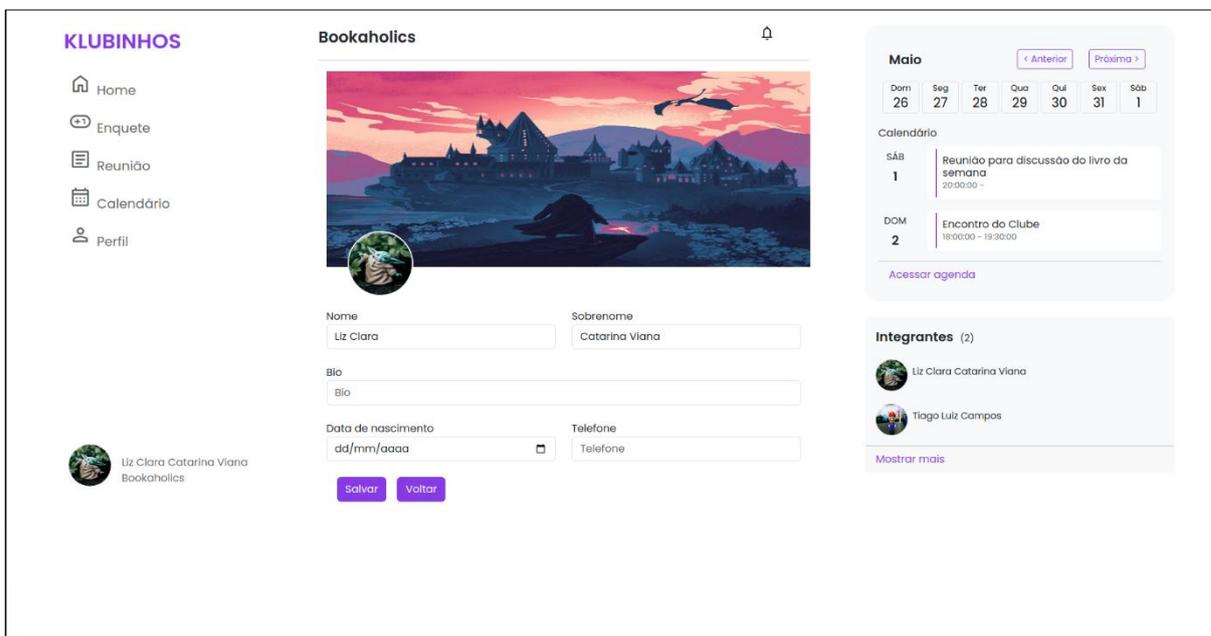
Na área do perfil do usuário (Figuras 28 e 29), é exibido a foto de capa, foto do usuário, bio e suas publicações, além disso, o usuário pode realizar edições em seu perfil.

Figura 28: Tela do perfil do usuário



Fonte: Autoral, 2024

Figura 29: Tela de edição do perfil



Fonte: Autoral, 2024

## 9 CONSIDERAÇÕES FINAIS, RECOMENDAÇÕES E LIMITAÇÕES

A aplicação foi desenvolvida conforme o planejado, dentro do tempo estipulado e conta com suas funcionalidades operando como esperado. Entretanto, algumas fases de teste ainda são necessárias antes da ampla divulgação, para garantir a qualidade do produto.

Como o projeto apresentou um tempo de desenvolvimento restrito para a finalização da primeira versão da aplicação, alguns recursos de interesse não puderam ser adicionados ao escopo e implementados. Sendo assim, para as próximas versões da aplicação, o foco será dado em melhorar o desempenho das funcionalidades existentes e, também, implementar novas funcionalidades como:

- Possibilidade de participar ou gerenciar mais de um clube ao mesmo tempo.
- Capacidade de divulgar e localizar clubes de leitura públicos com base em temas específicos ou pontos de interesse.
- Integração com outras redes sociais.
- Páginas individuais para cada leitura.
- Possibilidade de direcionamento para páginas de compra dos livros.
- Tradução para o idioma inglês.

Com esses novos recursos espera-se abranger um público ainda maior e atrair mais pessoas para a plataforma, ajudando a popularizar ainda mais o acesso à clubes de leituras e incentivar o hábito de leitura no Brasil e no mundo.

Caso a plataforma consiga atingir um público considerável, será estudada a possibilidade de implementação de recursos avançados que só estarão disponíveis mediante à pagamento único ou até mesmo a implementação de uma assinatura periódica paga.

## REFERÊNCIAS

ALURA. Seja um artesão da web com Laravel: o framework MVC do PHP. Disponível em: <<https://www.alura.com.br/artigos/seja-um-artesao-da-web-com-laravel-o-framework-mvc-do-php>> . Acesso em: 7 jun. 2024.

AMAZON WEB SERVICES. Qual é a diferença entre um modelo de dados lógico e um modelo de dados físico? Disponível em: <<https://aws.amazon.com/pt/compare/the-difference-between-logical-and-physical-data-model/>>. Acesso em: 6 junho de 2024.

ANDERSON, D. Kanban: Mudança Evolucionária de Sucesso Para Seu Negócio de Tecnologia. Blue Hole Press. 2011

APACHE FRIENDS. About the XAMPP project. Disponível em: <<https://www.apachefriends.org/about.html>>. Acesso em: 7 jun. 2024.

APPMASER. Arquitetura baseada em componentes React. Disponível em: <https://appmaster.io/pt/blog/arquitetura-baseada-em-componentes-react>. Acesso em: 9 jun. 2024.

BOOCH, G., RUMBAUGH, J., & JACOBSON, I. UML: Guia do Usuário (2nd ed.). Rio de Janeiro: Elsevier. 2006

CHACON, Scott; STRAUB, Ben. Pro git: Everything you need to know about GIT. 2 ed. [S.L.]: Apress, 2014.

COMPOSER. Download. Disponível em: <<https://getcomposer.org/download/>>. Acesso em: 9 maio 2024.

COODESH. O que é arquitetura MVC?. Disponível em: <https://coodesh.com/blog/dicionario/o-que-e-arquitetura-mvc/>. Acesso em: 9 jun. 2024.

COSTA, Hugo. Introdução REST. Medium, 11 jun. 2017. Disponível em: <<https://medium.com/trainingcenter/introdu%C3%A7%C3%A3o-rest-7ec4cefd16c8>>. Acesso em: 9 jun. 2024.

DEV MEDIA. Serviços RESTful: verbos HTTP. Disponível em: <https://www.devmedia.com.br/servicos-restful-verbos-http/37103>. Acesso em: 9 jun. 2024.

DIGITALPRODUCTSDEV. Vite.js: o build tool que vai vitaminar sua experiência de desenvolvimento. Medium, 19 jul. 2021. Disponível em:

<https://medium.com/digitalproductsdev/vite-js-o-build-tool-que-vai-vitaminar-sua-experi%C3%A7%C3%A3o-de-desenvolvimento-9361ab983266>. Acesso em: 9 jun. 2024.

DRAW.IO. Disponível em: <<https://www.drawio.com/>>. Acesso em: 25 maio 2024.

ECHOS. Fases de um projeto. Disponível em: <https://escoladesignthinking.echos.cc/blog/2019/11/fases-de-um-projeto/>. Acesso em: 9 jun. 2024.

ELY, J. Como gerir o tempo e a rotina de trabalho? Uma proposta de aplicação da técnica Kanban e do aplicativo Trello para o planejamento pessoal. Saber Humano: Revista Científica da Faculdade Antonio Meneghetti, 2024.

FIGMA. Figma. Disponível em: <<https://www.figma.com/pt-br/>>. Acesso em: 17 maio 2024.

GITHUB. GitHub. Disponível em: <<https://github.com/>>. Acesso em: 24 maio 2024.

GODADDY. GoDaddy. Disponível em: <<https://www.godaddy.com/pt-br>>. Acesso em: 26 maio 2024.

GOMES, Vinnicius. Arquitetura para aplicações React. Medium, 21 jul. 2020. Disponível em: <https://vinniciusgomes.medium.com/arquitetura-para-aplicacoes-react-d77361bf2d49>. Acesso em: 9 jun. 2024.

GUIMARÃES, F. Benefícios e vantagens da utilização do Laravel framework no desenvolvimento de sistemas web. Medium, 11 jul. 2019. Disponível em: <https://medium.com/@f.guimaraes/benef%C3%ADcios-e-vantagens-da-utiliza%C3%A7%C3%A3o-do-laravel-framework-no-desenvolvimento-de-sistemas-web-7239bc495d06>. Acesso em: 9 jun. 2024.

INSTITUTO PRÓ-LIVRO. Retratos da leitura no Brasil 5. 2021. Disponível em: <<https://www.prolivro.org.br/pt/pesquisa/retratos-da-leitura-no-brasil/>>. Acesso em: 27 maio 2024.

LARAVEL. Laravel Documentation. Disponível em: <<https://laravel.com/docs/11.x>>. Acesso em: 7 maio 2024.

LOCAWEB. Tudo sobre linguagem de programação PHP. Disponível em: <https://www.locaweb.com.br/blog/temas/codigo-aberto/tudo-sobre-linguagem-de-programacao-php/>. Acesso em: 9 jun. 2024.

LUCIDCHART. O que é UML? Disponível em: <<https://www.lucidchart.com/pages/pt/o-que-e-uml>>. Acesso em: 28 maio 2024.

NODE.JS. Disponível em: <<https://www.opus-software.com.br/node-js/>> Acesso em: 05 de novembro de 2023.

NOTION. Disponível em: <<https://www.notion.so/help/start-here>> Acesso em: 27 maio 2024.

OBJECTIVE. Testes de software. Disponível em: <https://www.objective.com.br/insights/testes-de-software/>. Acesso em: 9 jun. 2024.

PHP. PHP: Hypertext Preprocessor. Disponível em: <<https://www.php.net/>>. Acesso em: 14 maio 2024.

PHPMYADMIN. phpMyAdmin. Disponível em: <<https://www.phpmyadmin.net/>>. Acesso em: 13 maio 2024.

PROGRAMMER, Augusto. UML: Diagrama de Caso de Uso e Diagrama de Classe. Blog, 10 mar. 2015. Disponível em: <https://augustoprogrammer.wordpress.com/2015/03/10/uml-diagrama-de-caso-de-uso-e-diagrama-de-classe/>. Acesso em: 9 jun. 2024.

REACT. Disponível em: <<https://pt-br.reactjs.org/>> Acesso em: 16 de abril de 2024.

SINDICATO NACIONAL DOS EDITORES DE LIVROS. Pesquisa Produção e Vendas do Setor Editorial Brasileiro. 2023. Disponível em: <[https://snel.org.br/wp/wp-content/uploads/2024/05/apresentacao\\_imprensa\\_V4.pdf](https://snel.org.br/wp/wp-content/uploads/2024/05/apresentacao_imprensa_V4.pdf)>. Acesso em: 15 de maio de 2024.

STACK OVERFLOW. Stack Overflow Developer Survey 2023. Disponível em: <<https://survey.stackoverflow.co/2023/>>. Acesso em: 26 maio 2024.

TECNOBLOG. O que é CSS (Cascading Style Sheets)?. Disponível em: <https://tecnoblog.net/responde/o-que-e-css-cascading-style-sheets/>. Acesso em: 9 jun. 2024.

TYPESCRIPT. TypeScript: JavaScript With Syntax For Types. Disponível em: <<https://www.typescriptlang.org/>>. Acesso em: 22 maio 2024.

VISUAL STUDIO CODE. Code Editing. Redefined. Disponível em: <<https://code.visualstudio.com/>>. Acesso em: 19 maio 2024.

VITE. Guide. Disponível em: <<https://vitejs.dev/guide/>>. Acesso em: 23 maio 2024.

ZANARDI, Isabelle; MESQUITA, Maria; SCAVACIN, Vitor. #Booktok movimentou mercado literário. *Emerge Mag*, 6 set. 2023. Disponível em: <<https://emergemag.com.br/booktok-movimentou-mercado-editorial-e-literario/>>. Acesso em: 27 maio de 2024.

## APÊNDICE A – QUESTIONÁRIO

## Clube de leitura

Prezado(a) participante,

Este questionário faz parte de um projeto de trabalho de conclusão de curso do curso de Análise e Desenvolvimento de Sistemas da FATEC. Buscamos entender melhor o interesse e a experiência das pessoas em clubes de leitura, visando o desenvolvimento de uma rede social para esse fim.

Agradecemos sua participação e suas respostas serão tratadas com confidencialidade.

\* Indica uma pergunta obrigatória

1. Gostaria de se identificar?

2. Qual a sua faixa etária? \*

Menos de 18 anos

18 - 24 anos

25 - 34 anos

35 - 44 anos

45 - 54 anos

55 anos ou mais

3. Com que frequência você costuma ler livros? \*

Entre 1 - 5 por ano

Entre 6 - 10 por ano

Mais de 10 por ano

Não leio livros

4. Você costuma discutir livros com outras pessoas ou ler comentários sobre eles?

Sim

Não

Às vezes

5. Você tem interesse em fazer parte de uma comunidade online de leitores? \*

Sim

Não

Talvez

6. Você sente que faltam opções de plataformas colaborativas de estímulo à leitura?

Sim

Não

Não sei

7. Você utiliza alguma plataforma online focada na discussão de livros? (Ex: Skoob, Goodreads, etc.)

Sim

Não

8. Você participa de algum clube de leitura? \*

Sim, participo atualmente. *Pular para a pergunta 9*

Já participei, mas não participo de nenhum no momento. *Pular para a pergunta 9*

Não, nunca participei de nenhum clube de leitura. *Pular para a pergunta 15*

As perguntas a seguir são referentes à sua experiência atual ou prévia com clubes de leitura. Caso você participe ou tenha participado de mais de um clube, responda de acordo com o clube que você participa/participou de forma mais assídua ou por mais tempo.

9. As reuniões do clube de leitura que você participa ou participou ocorriam de forma:

Presencial

Online

Mista (Presencial e Online)

10. Aproximadamente quantas pessoas participam ou participavam do seu clube de leitura?

Até 10 pessoas

- Entre 11 a 30 pessoas
- 31 a 50 pessoas
- 51 a 100 pessoas
- Mais de 100 pessoas

11. Com que frequência ocorrem ou ocorriam os encontros do clube de leitura? \*

- Semanalmente
- Quinzenalmente
- Mensalmente
- A cada bimestre ou trimestre
- A cada semestre
- Anualmente

12. O que mais te atrai em participar de um clube de leitura? (Selecione todos que se aplicarem)

*Marque todas que se aplicam.*

- Aprofundar a compreensão dos livros
- Conexão com outras pessoas que compartilham interesses literários
- Descobrir novos livros e autores
- Compartilhar experiências de leitura
- Ampliar o conhecimento sobre diferentes gêneros literários
- Participar de eventos literários exclusivos para membros do clube
- Trocar ideias e opiniões sobre temas relevantes abordados nos livros
- Desenvolver habilidades de análise crítica através das discussões
- Ter acesso a recursos exclusivos, como entrevistas com autores e material complementar

Outro:

13. A escolha dos livros a serem lidos pelo clube de leitura é/era feita de que modo? \*

- Decisão única dos organizadores
- Votações presenciais
- Votações online

Sorteio presencial

Sorteio online

Outro:

14. As reuniões do clube de leitura ocorrem/ocorriam de que forma?

Presencial

Plataforma de videoconferências (Zoom, Skype, Meet, etc.)

Plataforma de Lives (Twitch, Youtube, Instagram, Tiktok, etc)

Fóruns ou grupos de discussão (Reddit, Facebook, Whatsapp, etc.)

Outro:

*Pular para a pergunta 17*

As perguntas abaixo são referentes ao seu interesse em clubes de leitura.

15. Você tem interesse ou curiosidade de participar de um clube de leitura?

Sim

Não *Pular para a pergunta 23*

Talvez

16. Por quais motivos você nunca participou de um clube de leitura? \*

*Marque todas que se aplicam.*

Não gosto de ler

Não vejo necessidade

Não conheço pessoas interessadas

Nunca pensei sobre o assunto

Falta de tempo

Não encontrei um clube de leitura que se encaixasse nos meus interesses literários

Preferência por ler de forma individual, sem participar de grupos de discussão

Falta de confiança para compartilhar minhas opiniões sobre livros com outras pessoas

Desconforto em participar de grupos sociais ou eventos com desconhecidos

Outro:

17. Você prefere participar de um clube de leitura presencial ou online? \*

Presencial

Online

Indiferente

18. Que tipo de plataforma você preferiria para participar de um clube de leitura online?\*

Aplicativo móvel

Site na web

Indiferente

19. Que tipo de recursos online você gostaria que uma plataforma de clube de leitura oferecesse? (Selecione todas as alternativas que se aplicarem)

*Marque todas que se aplicam.*

Fóruns de discussão

Calendário de eventos

Enquetes para escolhas de livros

Históricos de leituras

Avaliações e ranqueamento de livros

Chat ao vivo para discussões em tempo real

Biblioteca virtual para compartilhar e acessar livros

Recursos para compartilhar resenhas e recomendações de livros

Desafios de leitura

Loja para aquisição de livros

Outro:

20. Você gostaria que o clube de leitura oferecesse atividades além das discussões de livros, como eventos culturais ou oficinas literárias?

Sim

Não

Indiferente

21. Você gostaria que o clube de leitura tivesse algum sistema de recompensa ou reconhecimento para os membros mais participativos?

Sim

Não

Talvez, dependendo do tipo de recompensa

22. Você estaria disposto a pagar uma taxa de associação para participar de um clube de leitura?

Sim

Não

Talvez, dependendo do valor

#### Finalização

23. Você acredita que uma plataforma de clubes de leitura poderia ajudar a manter ou estimular seu hábito de leitura?

Sim

Não

Talvez

24. Gostaria de deixar algum comentário ou observação?

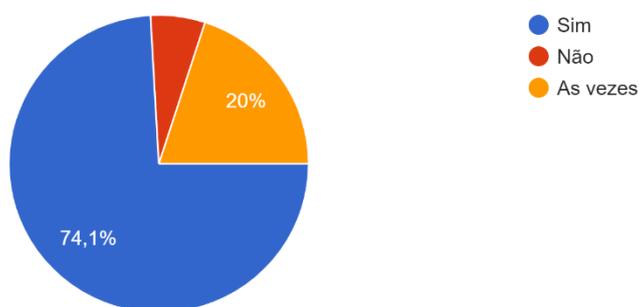
## APÊNDICE B – RESULTADO DO QUESTIONÁRIO

O resultado do questionário das perguntas 4 até 23 estão apresentados nas Figuras 30 até 49.

**Figura 30.** Respostas da questão 4.

Você costuma discutir livros com outras pessoas ou ler comentários sobre eles?

85 respostas

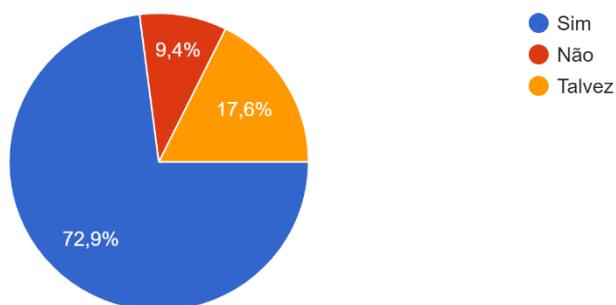


Fonte: Google Forms, 2024

**Figura 31.** Respostas da questão 5.

Você tem interesse em fazer parte de uma comunidade online de leitores?

85 respostas

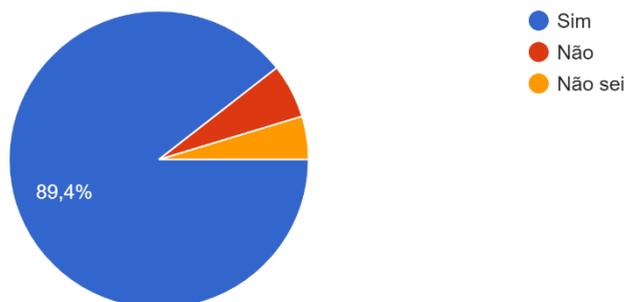


Fonte: Google Forms, 2024

**Figura 32.** Respostas da questão 6.

Você sente que faltam opções de plataformas colaborativas de estímulo à leitura?

85 respostas

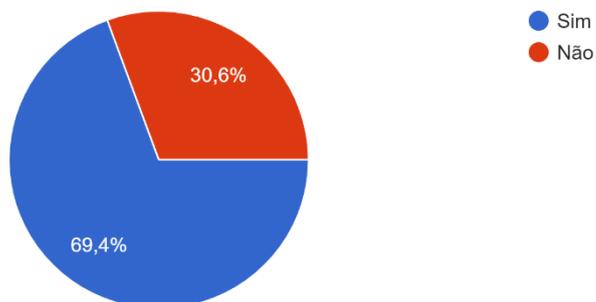


Fonte: Google Forms, 2024

**Figura 33.** Respostas da questão 7.

Você utiliza alguma plataforma online focada na discussão de livros? (Ex: Skoob, Goodreads, etc.)

85 respostas



Fonte: Google Forms, 2024

**Figura 34.** Respostas da questão 8.

Você participa de algum clube de leitura?

85 respostas

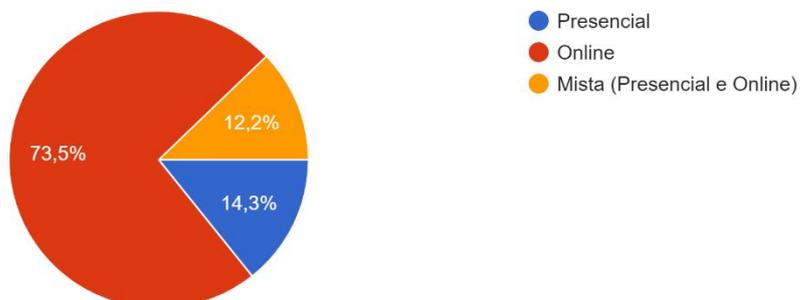


Fonte: Google Forms, 2024

**Figura 35.** Respostas da questão 9.

As reuniões do clube de leitura que você participa ou participou ocorriam de forma:

49 respostas

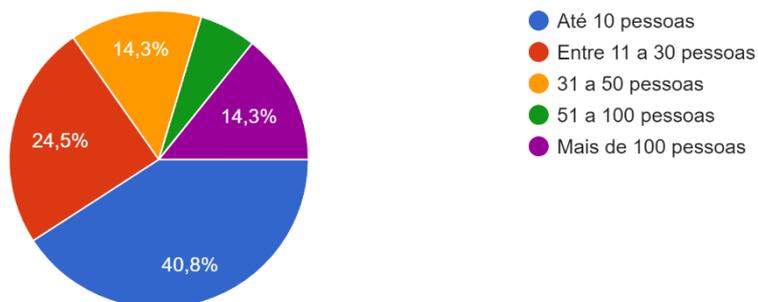


Fonte: Google Forms, 2024

**Figura 36.** Respostas da questão 10.

Aproximadamente quantas pessoas participam ou participavam do seu clube de leitura?

49 respostas

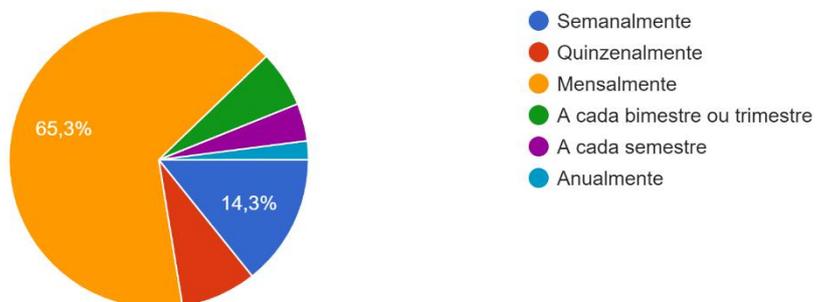


Fonte: Google Forms, 2024

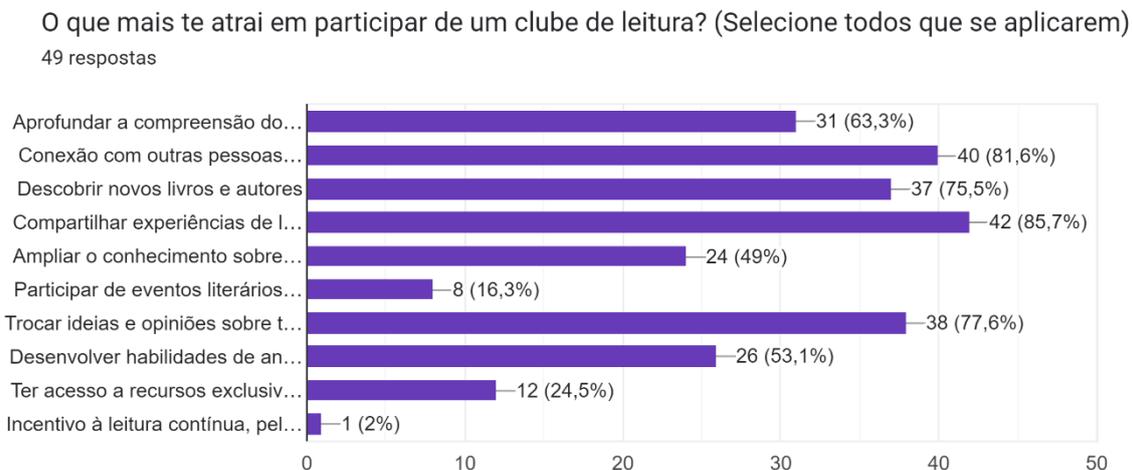
**Figura 37.** Respostas da questão 11.

Com que frequência ocorrem ou ocorriam os encontros do clube de leitura?

49 respostas



Fonte: Google Forms, 2024

**Figura 38.** Respostas da questão 12.

Fonte: Google Forms, 2024

**Figura 39.** Respostas da questão 13.

A escolha dos livros a serem lidos pelo clube de leitura é/era feita de que modo:

49 respostas

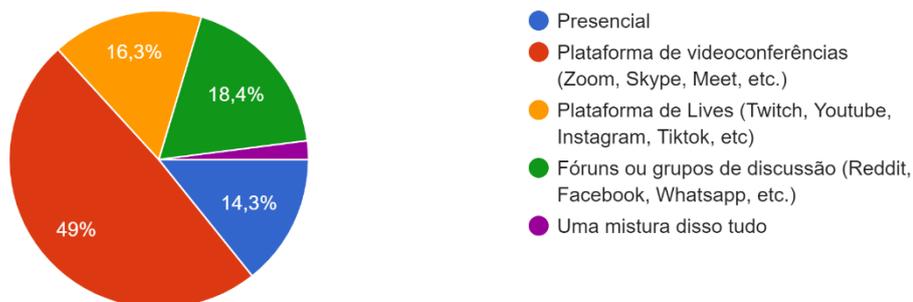


Fonte: Google Forms, 2024

**Figura 40.** Respostas da questão 14.

As reuniões do clube de leitura ocorrem/ocorriam de que forma?

49 respostas

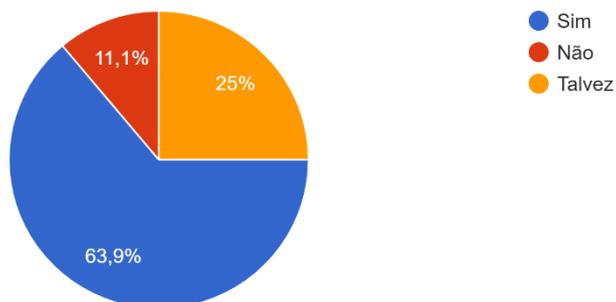


Fonte: Google Forms, 2024

**Figura 41.** Respostas da questão 15.

Você tem interesse ou curiosidade de participar de um clube de leitura?

36 respostas

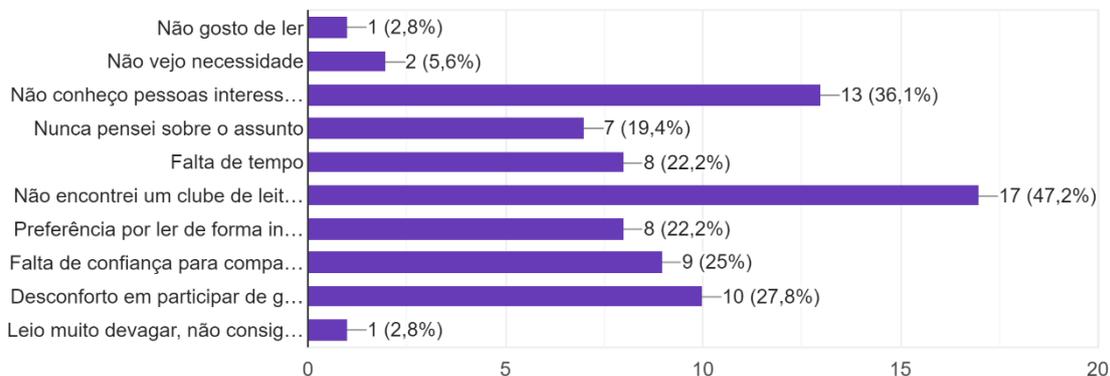


Fonte: Google Forms, 2024

**Figura 42.** Respostas da questão 16.

Por quais motivos você nunca participou de um clube de leitura?

36 respostas

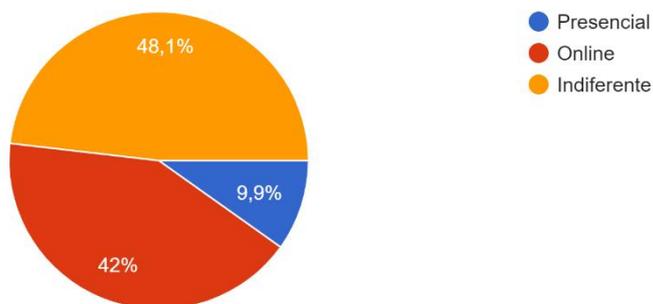


Fonte: Google Forms, 2024

**Figura 43.** Respostas da questão 17.

Você prefere participar de um clube de leitura presencial ou online?

81 respostas

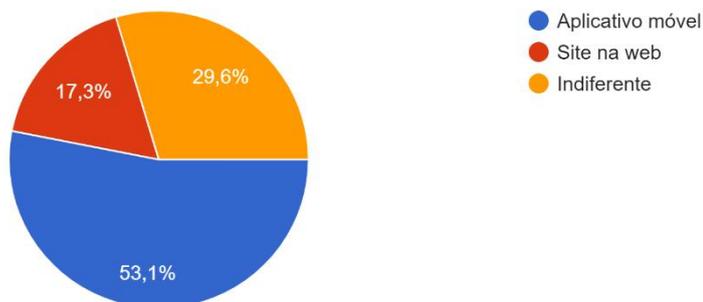


Fonte: Google Forms, 2024

**Figura 44.** Respostas da questão 18.

Que tipo de plataforma você preferiria para participar de um clube de leitura online?

81 respostas



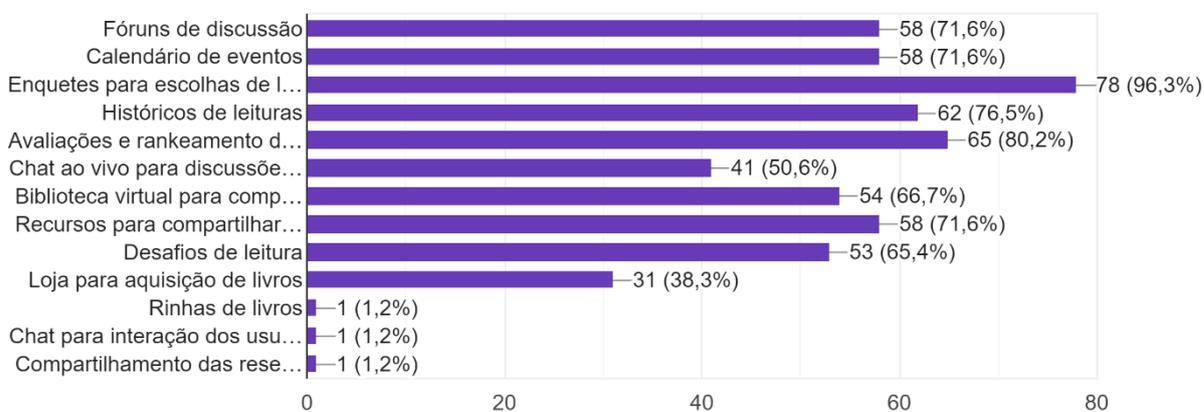
Fonte: Google Forms, 2024

**Figura 45.** Respostas da questão 19.

Que tipo de recursos online você gostaria que uma plataforma de clube de leitura oferecesse?

(Selecione todas as alternativas que se aplicarem)

81 respostas

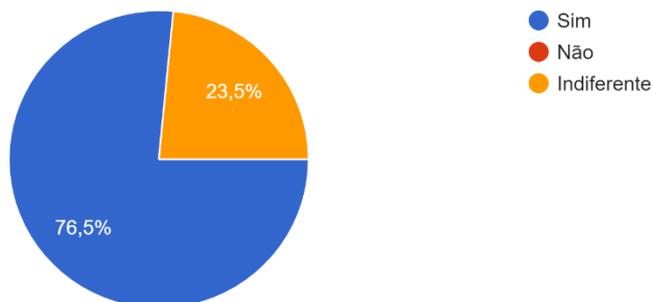


Fonte: Google Forms, 2024

**Figura 46.** Respostas da questão 20.

Você gostaria que o clube de leitura oferecesse atividades além das discussões de livros, como eventos culturais ou oficinas literárias?

81 respostas

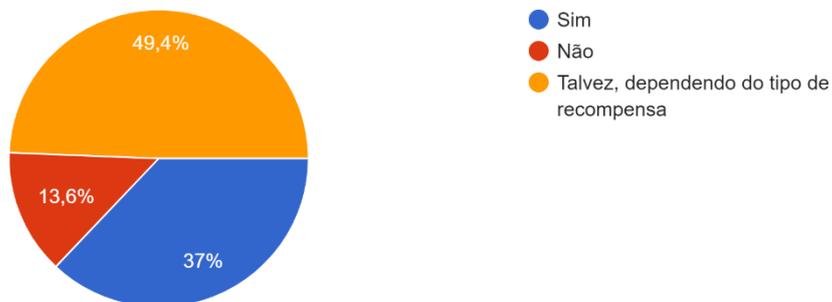


Fonte: Google Forms, 2024

**Figura 47.** Respostas da questão 21.

Você gostaria que o clube de leitura tivesse algum sistema de recompensa ou reconhecimento para os membros mais participativos?

81 respostas

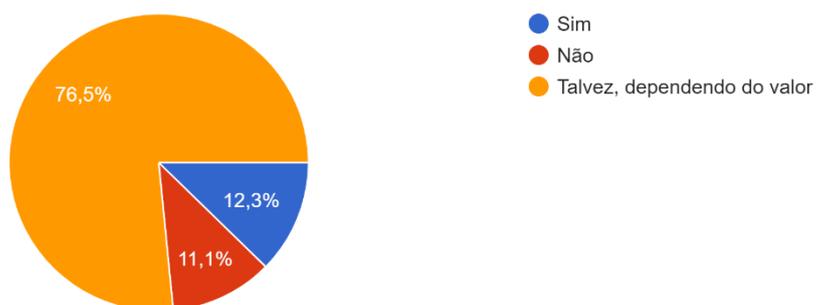


Fonte: Google Forms, 2024

**Figura 48.** Respostas da questão 22.

Você estaria disposto a pagar uma taxa de associação para participar de um clube de leitura?

81 respostas

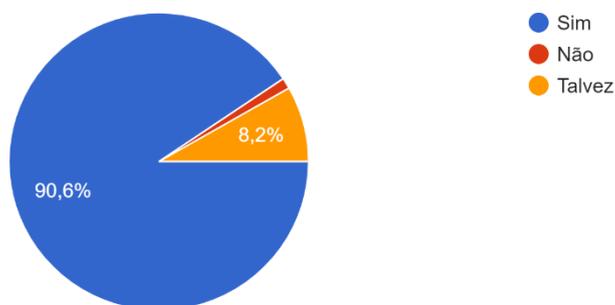


Fonte: Google Forms, 2024

**Figura 49.** Respostas da questão 23.

Você acredita que uma plataforma de clubes de leitura poderia ajudar a manter ou estimular seu hábito de leitura?

85 respostas



Fonte: Google Forms, 2024

As respostas da questão 24, “Gostaria de deixar algum comentário ou observação?” estão descritos abaixo:

*“Adorei as perguntas, muito bem elaboradas e já fiquei na expectativa da criação de um novo aplicativo direcionado à isto. 🍷”*

*“Não participo de clube de leitura, mas sempre participo de todas as maratonas literárias do Geek Freak e utilizo o app Maratona.app para organizar, mas sinto falta de conseguir ver comentários sobre os livros e até recomendações de livros parecidos com os que eu favoritei. Algo na linha de "Outros leitores que favoritaram esse livro, também favoritaram esse e esse", para que eu possa encontrar mais livros parecidos com os que eu amei.”*

*“Acredito muito que um clube de leitura não precisa ser, necessariamente, um clube onde todos estão a ler o mesmo livro. Exemplo, eu, enquanto usuária, posso criar um clube de leitura para um livro x e deixar disponível para que outras pessoas também se associem a minha escolha para ler junto comigo. Porque às vezes, o livro escolhido pelo clube é um livro que eu quero ler, mas não naquele momento, então seria, do meu ponto de vista, mais benéfico ao incentivo da leitura, que não precisa ser algo "imposto".”*

*“Eu gosto de ler de forma particular e observar posts e reações de outras pessoas sobre o que eu leio, mas quando se trata de interagir eu prefiro não”*

*“Dica de inspiração: a plataforma The StoryGraph tem um sistema muito bacana para leituras conjuntas onde eles permitem comentários relacionados a páginas de livros e as pessoas conforme vão chegando nessas páginas vão podendo ver esses comentários”*

*“Mais opções no perfil do usuário, assim como no Tumblr; acredito que muitos se preocupam com a parte da interface/interatividade, então chamaria bastante atenção do público caso houvesse mais ferramentas na aba do perfil.”*

*“Ótima pesquisa”*

*“Vim pelo livraria em casa do Paulo Ratz 😊”*

*“Gostei muito da ideia, espero que isso possa se tornar realidade <3”*

*“Incentivo à leitura é sempre válido e se tiver com quem trocar ideias sobre o livro é enriquecedor; aprimora o senso crítico, dá oportunidade de conhecer outros pontos de vista e tudo isso se divertindo.”*

*“Uma plataforma intuitiva, bonita, organizada e com banco de dados extenso é algo que, certamente, facilitaria e estimularia a leitura de um maior número de pessoas!”*

## APÊNDICE C – REQUISITOS FUNCIONAIS

### **REQ-001 Cadastro de Usuário**

O sistema permitirá o cadastro do usuário, fornecendo informações como email, nome, sobrenome e senha, como também a alteração de seu registro cadastral, realizando a validação através de um e-mail de confirmação.

### **REQ-002 Criação e participação do clube do livro**

O sistema permitirá o cadastro do clube do livro, fornecendo o nome do clube, além da alteração de seu registro cadastral, ou a participação em um clube do livro através da pesquisa pelo nickname único, sendo necessária a aprovação por um administrador.

### **REQ-003 Gerenciar postagens**

O sistema permitirá com que os usuários criem, consultem e excluam postagens, além de comentar e curtir as postagens de outros usuários.

### **REQ-004 Gerenciar reunião**

O sistema oferece aos administradores a capacidade de gerenciar as reuniões e eventos do clube do livro, podendo criar novas reuniões, definindo detalhes como data, hora, link e descrição. Além disso, os administradores podem consultar, editar e excluir reuniões conforme necessário.

### **REQ-005 Gerenciar agenda de eventos**

O sistema fornecerá um calendário para visualização dos eventos relacionados a encontros do clube do livro e metas de leitura.

**REQ-006 Gerenciar enquete**

O sistema permitirá a criação, consultas, edição e exclusão de enquetes.

**REQ-007 Votos**

O sistema permitirá votações em enquetes pelos usuários.

**REQ-008 Avaliações de livros**

O sistema permitirá a possibilidade de avaliações dos livros discutidos no clube.

**REQ-009 Gerar relatórios**

O sistema deverá fornecer relatórios ao administrador referente aos livros lidos durante o ano, a quantidade de livros e votos.

**REQ-010 Gerenciar integrantes**

O sistema permitirá que os administradores realizem a consulta, aprovação, inclusão e exclusão de integrantes no clube do livro.

**REQ-011 Consultar integrantes**

O sistema permitirá que os usuários visualizem todos os integrantes participantes do clube do livro.

**REQ-012 Gerenciar administrador**

O sistema permitirá que os administradores realizem a promoção e remoção de integrantes como administradores do clube do livro.

## APÊNDICE D – REQUISITOS NÃO FUNCIONAIS

### **REQ-001 Segurança**

O sistema deve ser protegido contra acesso não autorizado de usuários não cadastrados no sistema.

### **REQ-002 Portabilidade**

O sistema deve ser capaz de rodar em diferentes plataformas com alterações mínimas.

### **REQ-003 Implementação**

O sistema deve ser implementado na linguagem PHP com Laravel e Javascript com React.

### **REQ-004 Implementação**

O sistema deverá se comunicar com o banco SQL Server.

### **REQ-005 Usabilidade**

O sistema deverá ser de fácil aprendizado e compreensão.

### **REQ-006 Confiabilidade**

O sistema deve se manter estável e disponível durante seu funcionamento.

### **REQ-007 Manutentabilidade**

O sistema deverá ser de fácil manutenção e correção de erros e falhas de funcionalidade.

### **REQ-008 Entrega**

O sistema deverá emitir registros e relatórios de livros lidos, quantidade de livros e votos.

**REQ-009 Segurança**

O sistema deverá garantir proteção e sigilo dos dados de pagamentos.

**REQ-010 Usabilidade**

O sistema deverá ter um baixo tempo de resposta.

**REQ-011 Escalabilidade**

O sistema deverá garantir a capacidade de lidar com um aumento no número de membros, postagens e interações sem comprometer o desempenho.

## APÊNDICE E – MODELO FÍSICO

Script SQL responsável pela criação das tabelas que atendem o modelo lógico dos dados. Este script define a estrutura das tabelas, tipos de dados e restrições necessárias.

```
CREATE TABLE Clube ( id_clube INT PRIMARY KEY, nm_nome
VARCHAR(40), dt_criacao DATE, ds_descricao VARCHAR(255), im_clube BLOB );
```

```
CREATE TABLE Agenda ( id_agenda INT PRIMARY KEY, nm_titulo
VARCHAR(40), ds_descricao VARCHAR(255), dt_data DATE, hr_comeca TIME,
hr_termina TIME );
```

```
CREATE TABLE Livro ( id_livro INT PRIMARY KEY, nm_titulo VARCHAR(40),
nm_autor VARCHAR(40), nm_genero VARCHAR(20), nm_curadoria VARCHAR(30),
ds_descricao VARCHAR(255), cd_num_paginas INT, vl_preco DECIMAL(4,2),
mm_mes_referencia DATE, nm_status VARCHAR(30), cd_nota_livro INT );
```

```
CREATE TABLE Usuario ( id_usuario INT PRIMARY KEY, ic_tipo CHAR(4),
nm_nome VARCHAR(40), dt_nascimento DATE, nm_email VARCHAR(60),
cd_telefone VARCHAR(11), nm_senha VARCHAR(10), ds_descricao
VARCHAR(255), im_usuario BLOB, dt_dataIngresso DATE, ic_situacao CHAR(1),
id_clube INT, FOREIGN KEY (id_clube) REFERENCES Clube(id_clube) );
```

```
CREATE TABLE Reuniao ( id_reuniao INT PRIMARY KEY, ds_participantes
TEXT, dt_reuniao DATE, hr_reuniao TIME, ds_reuniao VARCHAR(255), nm_link
VARCHAR(255), ds_comentario VARCHAR(255), id_clube INT, id_agenda INT,
FOREIGN KEY (id_clube) REFERENCES Clube(id_clube), FOREIGN KEY
(id_agenda) REFERENCES Agenda(id_agenda) );
```

```
CREATE TABLE Postagem ( id_postagem INT PRIMARY KEY, dt_data
DATETIME, ds_texto VARCHAR(255), im_postagem BLOB, id_clube INT, id_usuario
```

```
INT, FOREIGN KEY (id_clube) REFERENCES Clube(id_clube), FOREIGN KEY (id_usuario) REFERENCES Usuario(id_usuario) );
```

```
CREATE TABLE Comentario ( id_comentario INT PRIMARY KEY, dt_data DATETIME, ds_texto VARCHAR(255), id_postagem INT, id_usuario INT, FOREIGN KEY (id_postagem) REFERENCES Postagem(id_postagem), FOREIGN KEY (id_usuario) REFERENCES Usuario(id_usuario) );
```

```
CREATE TABLE Curtida ( id_curtida INT PRIMARY KEY, qt_curtidas INT, id_usuario INT, id_postagem INT, FOREIGN KEY (id_usuario) REFERENCES Usuario(id_usuario), FOREIGN KEY (id_postagem) REFERENCES Postagem(id_postagem) );
```

```
CREATE TABLE Voto ( id_voto INT PRIMARY KEY, qt_voto INT, id_usuario INT, FOREIGN KEY (id_usuario) REFERENCES Usuario(id_usuario) );
```

```
CREATE TABLE Enquete ( id_enquete INT PRIMARY KEY, nm_titulo VARCHAR(255), ds_descricao VARCHAR(255), ic_status CHAR(1), id_agenda INT, FOREIGN KEY (id_agenda) REFERENCES Agenda(id_agenda) );
```

```
CREATE TABLE Opcao ( id_opcao INT PRIMARY KEY, ds_descricao VARCHAR(255), id_votos INT, id_livro INT, id_enquete INT, FOREIGN KEY (id_votos) REFERENCES Voto(id_voto), FOREIGN KEY (id_livro) REFERENCES Livro(id_livro), FOREIGN KEY (id_enquete) REFERENCES Enquete(id_enquete) );
```

```
CREATE TABLE ReuniaoUsuario ( id_reuniao INT, id_usuario INT, PRIMARY KEY (id_reuniao, id_usuario), FOREIGN KEY (id_reuniao) REFERENCES Reuniao(id_reuniao), FOREIGN KEY (id_usuario) REFERENCES Usuario(id_usuario) );
```

```
CREATE TABLE ReuniaoLivro ( id_reuniao INT, id_livro INT, PRIMARY KEY
(id_reuniao, id_livro), FOREIGN KEY (id_reuniao) REFERENCES
Reuniao(id_reuniao), FOREIGN KEY (id_livro) REFERENCES Livro(id_livro) );
```

Após a criação das tabelas, o próximo passo é a inserção de dados com o objetivo de popular as tabelas, para a realização de consultas e testes como forma de validação do modelo criado.

```
INSERT INTO Usuario (nm_nome, dt_nascimento, nm_email, nm_senha,
cd_telefone, celular) VALUES ('Vitória Isis Viana', '1996-01-26',
'vitoria.isis.viana@hotmail.com', '7xQlsl47LQ', '(77) 2698-7414', '(77) 98331-5076'),
('Bryan Kaique Vieira', '1996-04-11', 'bryan_kaique_vieira@centroin.com.br',
'0q9xFdQKkD', '(69) 3854-5200', '(69) 98524-4929'), ('Hugo Fernando Davi
Cavalcanti', '1996-01-09', 'hugo.fernando.cavalcanti@deskprint.com.br',
'ynps6zpDeN', '(86) 2674-3840', '(86) 98952-1410'), ('Luís Augusto Kevin Rezende',
'1996-05-11', 'luisaugustorezende@carreiradasilva.com', 'QjC2ZCgCGO', '(88) 2699-
5352', '(88) 99657-7335');
```