



INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO PIAUÍ
CAMPUS PICOS
CURSO ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

GABRIEL MARCOS BATISTA GUIMARÃES

**GAMEMATE: Frontend para aplicativo mobile para unificação e organização
das bibliotecas digitais de jogos**

PICOS
2025

GABRIEL MARCOS BATISTA GUIMARÃES

GAMEMATE: Frontend para aplicativo mobile para unificação e organização das
bibliotecas digitais de jogos

Relatório técnico apresentado como exigência
para aprovação na disciplina Trabalho de
Conclusão de Curso do Curso de Tecnologia em
Análise e Desenvolvimento de Sistemas do
Instituto Federal de Educação, Ciência e
Tecnologia do Piauí, Campus Picos.

Orientador(a): Prof. MSc. Jader Anderson Oliveira
de Abreu

PICOS

2025

GABRIEL MARCOS BATISTA GUIMARÃES

GAMEMATE: Frontend para aplicativo mobile para unificação e organização das
bibliotecas digitais de jogos

Relatório técnico apresentado como exigência
para aprovação na disciplina de Trabalho de
conclusão de curso do Curso de Elaboração de
Tecnologia em Análise e Desenvolvimento de
Sistemas do Instituto Federal de Educação,
Ciência e Tecnologia do Piauí, Campus Picos.

Aprovada em: 08/01/2026.

BANCA EXAMINADORA

Prof. MSc. Jader Anderson Oliveira de Abreu
Instituto Federal do Piauí (IFPI)

Prof. MSc. Aislan Rafael Rodrigues de Sousa
Instituto Federal do Piauí (IFPI)

Prof. MSc. João Paulo Lima do Nascimento
Instituto Federal do Piauí (IFPI)

RESUMO

Com o crescimento no mercado de jogos digitais e a popularização de plataformas como Steam, Epic Games e GOG, as bibliotecas dos jogadores tornaram-se fragmentadas, dificultando o gerenciamento e o acompanhamento de seus títulos.

Para solucionar essa questão, este trabalho propõe o desenvolvimento do GameMate, o frontend de um aplicativo mobile que centraliza e organiza as bibliotecas digitais de jogos. O objetivo geral é criar, utilizando a tecnologia Flutter, uma interface que resolva problemas de usabilidade e de integração visual de dados, melhorando a experiência do usuário no controle de sua coleção. O projeto resultou em um protótipo funcional que valida a hipótese de que uma aplicação móvel pode facilitar a organização dessas bibliotecas.

Palavras-chave: Jogos digitais; Aplicativo móvel; Flutter; Gestão de bibliotecas; Front-end.

ABSTRACT

With the growth of the digital game market and the popularity of platforms such as Steam, Epic Games, and GOG, players' libraries have become fragmented, making it difficult to manage and track their titles. To address this issue, this work proposes the development of GameMate, the frontend of a mobile application that centralizes and organizes digital game libraries. The general objective is to create, using Flutter technology, an interface that solves usability and visual data integration problems, improving the user experience in collection control. The project resulted in a functional prototype that validates the hypothesis that a mobile application can facilitate the organization of these libraries.

Keywords: Digital games; Mobile application; Flutter; Library management; Frontend.

LISTA DE ILUSTRAÇÕES

Figura 1	- Gerenciamento de conta e perfil	17
Figura 2	- Gerenciamento da biblioteca e contas externas	18
Figura 3	- Gerenciamento da biblioteca de jogos	19
Figura 4	- Modelo de Entidades (Banco de Dados)	20
Figura 5	- Diagrama Geral da Arquitetura do Sistema	21
Figura 6	- Diagrama de Entidade-Relacionamento	22
Figura 7	- Mapa do site	23
Figura 8	- Tela de login	25
Figura 9	- Tela de cadastro	26
Figura 10	- Tela principal do aplicativo	27
Figura 11	- Tela de pesquisa	28
Figura 12	- Tela de jogo	29
Figura 13	- Tela de biblioteca.....	30
Figura 14	- Tela de perfil.....	31
Figura 15	- Diagrama de implantação.....	33

LISTA DE QUADROS

Quadro 1 – Requisitos Funcionais (RF).....	13
--	----

SUMÁRIO

1 INTRODUÇÃO.....	9
1.1 OBJETIVOS.....	10
1.1.1 GERAL.....	10
1.1.2 ESPECÍFICOS.....	11
2 TECNOLOGIAS ENVOLVIDAS.....	11
3 MODELAGEM DO PROJETO.....	12
3.1 LEVANTAMENTO DE REQUISITOS.....	12
3.2 PERSONAS.....	14
3.3 MAPA DE EMPATIA.....	15
3.4 DIAGRAMAS DE CASO DE USO.....	16
3.4.1 GERENCIAMENTO DE CONTA E PERFIL.....	17
3.4.2 GERENCIAMENTO DA BIBLIOTECA E CONTAS EXTERNAS.....	17
3.4.3 DESCOBERTA DE JOGOS.....	19
3.5 DIAGRAMAS DE CLASSE.....	19
3.6 ARQUITETURA DO SISTEMA.....	21
3.7 DIAGRAMA DE ENTIDADE-RELACIONAMENTO.....	21
3.8 MAPA DO SITE.....	22
3.8 INTERFACE.....	24
4 SOFTWARE.....	32
4.1 IMPLANTAÇÃO.....	32
4.2 TESTES.....	33
4.2.1 DESCRIÇÃO DO MÉTODO SUS.....	33
4.2.1.1 CÁLCULO DA PONTUAÇÃO.....	34
4.2.3 APLICAÇÃO DO TESTE.....	34
4.2.4 RESULTADO E INTERPRETAÇÃO DA PONTUAÇÃO DO SUS.....	35
5 CONSIDERAÇÕES FINAIS.....	36
REFERÊNCIAS.....	38

1 INTRODUÇÃO

O mercado de jogos digitais apresenta crescimento constante nos últimos anos, alcançando cerca de 300 milhões de jogadores apenas em *PCs* e consoles (*GamesIndustry.biz*, 2025), impulsionado pela popularização de plataformas de distribuição como *Steam*, *Epic Games* e *GOG*. Este crescimento resultou na fragmentação das bibliotecas digitais dos usuários, que se distribuem em diversas plataformas, tornando difícil o gerenciamento e o acompanhamento de conquistas, estatísticas e progresso nos jogos adquiridos.

A fragmentação das bibliotecas digitais, associada à diversidade de plataformas, amplia a necessidade de soluções que centralizem essas informações e permitam ao jogador gerenciar suas coleções de forma mais prática. Segundo Fragmentação no mercado de jogos digitais (2014), essa fragmentação cria desafios para os usuários ao tentarem administrar suas bibliotecas de forma eficiente, dada a variedade de lojas e sistemas de distribuição existentes.

Além disso, é fundamental considerar aspectos de usabilidade e experiência do usuário (*UX*) em soluções voltadas ao público *gamer*, garantindo interfaces intuitivas, consistentes e que reduzam a sobrecarga cognitiva (Borges et al., 2022). (Nakamura et al., 2023) destacam que a avaliação de *UX* em aplicações móveis deve considerar fatores como usabilidade, *design* visual e a interação com os elementos da interface, impactando diretamente na satisfação do usuário durante o uso.

Nesse contexto, destaca-se também a importância de rastrear e organizar a biblioteca de jogos adquiridos para facilitar o acesso às informações dos títulos, progresso e estatísticas. O trabalho *Game Tracking* (s.d.) demonstra a relevância de sistemas que auxiliam o gerenciamento das coleções digitais de jogos, evidenciando as dificuldades enfrentadas pelos jogadores ao gerenciar suas bibliotecas em plataformas diversas.

Dessa forma, identifica-se uma lacuna relacionada à ausência de uma solução prática que permita o gerenciamento unificado das bibliotecas de jogos em dispositivos móveis, considerando as necessidades de usabilidade, experiência do usuário e acessibilidade. Parte-se da hipótese de que o desenvolvimento de uma

aplicação *mobile* pode contribuir para facilitar a organização das bibliotecas digitais dos jogadores, possibilitando o controle de forma centralizada, intuitiva e visual. O foco recai sobre os desafios reais de *frontend* no ambiente *mobile*, como a criação de uma interface que integre múltiplas plataformas de forma coesa e funcional, garantindo usabilidade, responsividade e clareza visual ao usuário.

Assim, este trabalho propõe o desenvolvimento de uma aplicação *mobile* utilizando *Flutter*, uma tecnologia multiplataforma que permite a criação de interfaces consistentes e adaptáveis para *Android*. A proposta visa, mais do que simplesmente reduzir a fragmentação de bibliotecas de jogos digitais, enfrentar os desafios inerentes ao *frontend* na construção de uma experiência de usuário fluida e agradável. Isso inclui a padronização da exibição de dados vindos de *APIs* heterogêneas (*Steam*, *Epic*, *GOG*), o desenvolvimento de filtros dinâmicos e responsivos, e a criação de componentes visuais interativos que refletem o status e progresso dos jogos na coleção do usuário.

A estrutura do projeto é dividida em três camadas principais: a camada de dados (responsável pela comunicação com as *APIs* externas), a camada de lógica (gestão de estado e controle de fluxo) e a camada de apresentação (interface do usuário). O foco deste trabalho está nesta última, abordando os desafios visuais e interativos da construção da interface em *Flutter*. Serão desenvolvidos componentes como carrosséis de jogos, modais de vinculação entre plataformas, listas dinâmicas filtráveis e dashboards com indicadores visuais. A proposta é estudar e aplicar boas práticas de *design* de interfaces e UX especificamente voltadas para o perfil *gamer mobile*.

1.1 OBJETIVOS

Para orientar o desenvolvimento da pesquisa e a construção da solução proposta neste trabalho, definiram-se os seguintes objetivos:

1.1.1 GERAL

Desenvolver o *frontend* de uma aplicação *mobile* utilizando *Flutter* que centralize e apresente visualmente as bibliotecas digitais de jogos de diferentes plataformas, resolvendo problemas de usabilidade e integração visual de dados fragmentados, com o objetivo de melhorar a experiência do usuário no controle de

sua coleção.

1.1.2 ESPECÍFICOS

- Definir a tecnologia de *frontend* a ser utilizada no desenvolvimento do aplicativo *mobile*.
- Projetar e implementar as telas e fluxos de navegação do aplicativo, abrangendo autenticação, biblioteca de jogos, pesquisa e perfil.
- Implementar a estrutura de gerenciamento de estado e componentes reutilizáveis para manter consistência visual e facilitar a manutenção.
- Levantar e definir os requisitos funcionais e não funcionais necessários para o desenvolvimento de uma aplicação de gerenciamento de bibliotecas de jogos digitais.
- Desenvolver a aplicação *mobile* utilizando *Flutter*, assegurando responsividade, consistência visual e facilidade de uso.

2 TECNOLOGIAS ENVOLVIDAS

O desenvolvimento do GameMate é voltado para a construção de uma interface *mobile* intuitiva e responsiva, com o uso de tecnologias modernas que facilitam a integração com serviços externos e a exibição visual dos dados de forma unificada e organizada para o usuário. A seguir, são descritas as principais ferramentas e recursos utilizados:

Mobile:

- *Flutter*: *Framework* principal utilizado, na versão 3.24, para a construção da interface do aplicativo. Sua estrutura baseada em *widgets* permite o desenvolvimento de componentes visuais reutilizáveis, com foco em responsividade, fluidez e adaptabilidade a diferentes tamanhos de tela.
- *Dart*: Linguagem de programação oficial do *Flutter*, na sua versão 3.5.0, utilizada tanto para estruturar a interface quanto para controlar a lógica de navegação e comunicação com *APIs* externas. Com ela, é possível consumir dados das plataformas e organizá-los visualmente no app.
- *GetX*: Gerenciador de estado, rotas e dependências utilizado no projeto *Flutter*. Foi escolhido por sua simplicidade, baixo consumo de memória, e por proporcionar uma estrutura clara, reativa e escalável para a aplicação.

Design:

- Figma: Ferramenta de *design* utilizada para a criação das interfaces e fluxos da aplicação. Serviu como base visual para a construção das telas no *Flutter*, garantindo consistência na experiência do usuário e facilitando o trabalho entre *design* e desenvolvimento.

APIs e Serviços Externos:

- *Firebase Authentication*: Serviço de autenticação utilizado diretamente no app *Flutter* para gerenciar cadastro e *login* de usuários de forma segura e integrada.
- *IGDB API*: Fonte de dados sobre jogos (títulos, gêneros, avaliações, etc.), utilizada no *backend* para enriquecer o conteúdo exibido no app.
- *Steam Web API*: Responsável pela autenticação via *Steam* e pela obtenção da biblioteca de jogos do usuário, integrando a plataforma ao sistema.
- *SteamGridDB API*: Fornece imagens de alta qualidade (capas e grids), permitindo uma apresentação visual atrativa da coleção de jogos no aplicativo.

3 MODELAGEM DO PROJETO

Neste capítulo é apresentada a modelagem do projeto GameMate, contemplando as principais decisões técnicas e conceituais adotadas para o desenvolvimento do sistema. São descritos os requisitos identificados, bem como os critérios utilizados para garantir que a aplicação atenda às necessidades dos usuários finais. A modelagem busca assegurar coerência entre os objetivos do projeto, a experiência do usuário e a viabilidade técnica da solução proposta.

3.1 LEVANTAMENTO DE REQUISITOS

O levantamento de requisitos é essencial para definir as funcionalidades e características que o sistema deve apresentar, garantindo que as necessidades dos usuários sejam atendidas. Os requisitos se dividem em funcionais, que especificam as ações que o sistema deve realizar, e não funcionais, que tratam das qualidades e restrições do sistema.

A seguir, apresentam-se os principais requisitos funcionais identificados durante o processo de análise, com foco na interface mobile do GameMate. Estes requisitos foram definidos com base nas necessidades dos usuários finais, visando uma navegação simples, eficiente e intuitiva dentro do aplicativo.

Quadro 1: Requisitos Funcionais (RF)

ID	Nome do Requisito	Descrição
RF01	Cadastro e Autenticação de Usuário	A aplicação deve disponibilizar telas intuitivas para cadastro e <i>login</i> de usuários via <i>Firebase Authentication</i> , com formulários responsivos, <i>feedback</i> visual e transições suaves.
RF02	Gerenciamento de Perfil	O usuário logado deve conseguir visualizar e editar informações do seu perfil (nome, email, estatísticas de jogos) por meio de telas organizadas e com navegação fluida. Também deve ser possível excluir a conta com confirmação via interface.
RF03	Integração com Contas Externas	A interface deve permitir ao usuário vincular e desvincular sua conta <i>Steam</i> através de ações visuais guiadas. A aplicação exibirá estados de vinculação e mensagens de status de forma clara.
RF04	Sincronização e Visualização da Biblioteca	A aplicação deve oferecer uma visualização otimizada e responsiva da biblioteca de jogos sincronizada da conta vinculada. Deve exibir cards com imagens, nomes e status, além de filtros dinâmicos e carregamento progressivo (<i>scroll</i> infinito).
RF05	Gerenciamento Manual de Jogos e <i>Status</i>	O usuário deve conseguir adicionar ou remover jogos manualmente da biblioteca, além de definir e atualizar seus respectivos <i>status</i> (como Jogando, Zerado). A interface deve apresentar botões e seletores intuitivos.
RF06	Busca e Visualização de Jogos	A interface deve permitir buscar jogos através de uma barra de pesquisa funcional, exibindo os resultados em <i>cards</i> . Cada jogo pode ser acessado para uma página de detalhes com dados como capa, resumo, gêneros e plataformas.

Fonte: Elaborado pelo autor (2025).

Estes requisitos funcionais foram definidos com base na proposta central do GameMate: fornecer uma plataforma prática e agradável para organização da biblioteca de jogos, unificando funcionalidades automatizadas com controle manual do usuário.

Além das funcionalidades principais, também foram definidos os Requisitos Não Funcionais, que garantem a qualidade técnica da aplicação. Esses requisitos visam assegurar desempenho adequado, segurança dos dados, usabilidade da interface e compatibilidade com diferentes dispositivos Android.

Entre os requisitos funcionais definidos, o foco principal está no Cadastro e Autenticação de Usuário (RF01) e na Busca e Visualização de Jogos (RF06). Essas funcionalidades representam a porta de entrada e a principal forma de interação do usuário com a aplicação, garantindo que ele possa criar sua conta, acessar o sistema de forma segura e explorar o catálogo de jogos com praticidade. Ao priorizar esses pontos, o GameMate assegura uma experiência inicial positiva e fluida, estimulando o engajamento e facilitando o acesso às demais funcionalidades da plataforma.

3.2 PERSONAS

PERSONA 1: O GAMER QUE “ESTUDA”

- **Nome:** Gustavo Silva
- **Idade:** 21 anos
- **Ocupação:** Estudante de enfermagem
- **Características comportamentais da persona:**
 - Gosta de jogos os quais incluem quase todos os gêneros com exceção de Moba.
 - Gosta mais do gênero FPS e terror.
 - Joga casualmente mais pela parte da noite.
 - Joga apenas como diversão.
 - Está sempre comprando novos jogos em lojas diferentes.
- **Necessidades:**
 - Poder verificar quais jogos possui sem precisar ficar abrindo várias lojas todas as vezes.
 - Saber quais jogos já terminou e quais ainda não.
 - Comparar suas estatísticas com amigos.

PERSONA 2: O DEV QUE TEM TEMPO PARA JOGAR

- **Nome:** Junior Sousa
- **Idade:** 22 anos
- **Ocupação:** Desenvolvedor de Software
- **Características comportamentais da persona:**
 - Gosta de jogos de RPG e ação.
 - Costuma jogar três horas por dia, principalmente à noite, e gosta de jogar em grupos com amigos *online*.
- **Necessidades:**
 - Receber recomendações de jogos que sejam do seu interesse.
 - Registrar e acompanhar seu desempenho nos jogos, comparar sua performance com a de outros jogadores.

3.3 MAPA DE EMPATIA

O mapa de empatia é um material para realizar uma série de perguntas com suas *personas* para ter mais detalhes sobre a mesma e conhecer melhor quem vai utilizar seu *software*. O mapa em questão foi baseado nas duas *personas* citadas acima em um só.

O QUE ELE PENSA E SENTE?

- Preocupação em manter um registro de seus jogos.
- Interesse em descobrir mais informações sobre seus jogos jogados.
- Curiosidade em saber como suas estatísticas se comparam com as de outros usuários.
- Desejo de encontrar novos jogos que correspondam a seus gostos e preferências.

O QUE O USUÁRIO VÊ?

- Ambiente: Casa e rua.
- Pessoas: Amigos, colegas de faculdade.
- Situações: Escolhendo algum jogo, querendo visualizar suas estatísticas ou querendo mostrar/comparar com outras pessoas.

O QUE ELE OUVI?

- De amigos: Conversa sobre jogos sobre suas novas aquisições ou indicações, tempo que jogaram, quantidade de jogos.
- Da mídia: Indicações e análises, novos lançamentos.

O QUE ELE FALA E FAZ?

- Realiza o registro de seus jogos de forma manual usando bloco de notas.
- Acessa com frequência as estatísticas de seus jogos.
- Compara suas estatísticas com as de outros usuários.
- Aceita recomendações de jogos com base em seus gostos.

QUAIS SÃO AS SUAS DORES?

- Perda de informações sobre jogos jogados.
- Dificuldade em encontrar jogos que correspondam a seus gostos e preferências.
- Dificuldade em checar quais jogos possui.

QUAIS SÃO SUAS NECESSIDADES?

- Registro e controle de jogos jogados.
- Saber quais jogos já jogou ou ainda não.
- Acesso a estatísticas detalhadas sobre jogos jogados.
- Comparações com outros usuários da plataforma.
- Recomendações de jogos com base em suas preferências e gostos.

3.4 DIAGRAMAS DE CASO DE USO

Os Diagramas de Casos de Uso são utilizados para visualizar as diferentes maneiras como um ator (usuário ou outro sistema) interage com as funcionalidades do software. No contexto do *frontend*, eles ajudam a compreender como cada funcionalidade apresentada na interface é acessada e utilizada pelo usuário, servindo como um guia para alinhar a experiência visual e o fluxo de navegação às necessidades reais.

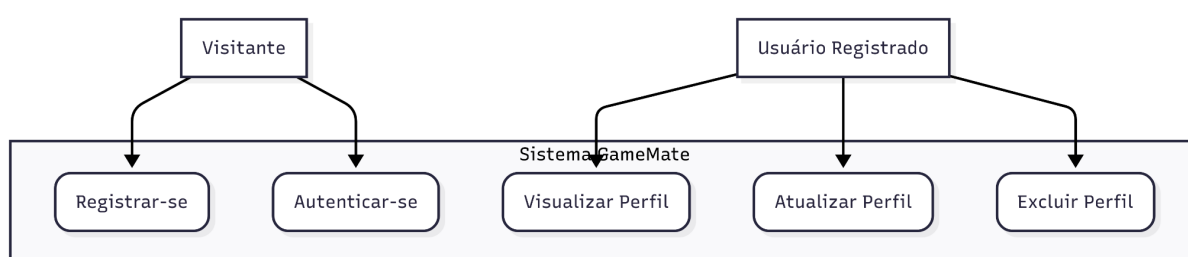
No GameMate, esses diagramas fornecem uma visão de alto nível de como as telas e componentes do aplicativo suportam cada interação, facilitando o planejamento da

interface e garantindo que todas as funcionalidades previstas estejam representadas de forma clara. Para maior organização, dividimos os casos de uso em três diagramas principais: Gerenciamento de Conta, Gerenciamento da Biblioteca e Descoberta de Jogos.

3.4.1 GERENCIAMENTO DE CONTA E PERFIL

Este diagrama representa as funcionalidades essenciais ligadas ao ciclo de vida da conta no sistema. Ele mostra as ações que um Visitante pode realizar para se registrar e tornar-se um Usuário Autenticado, além das funcionalidades de gerenciamento de perfil que passam a estar disponíveis após o *login*. No contexto do *frontend*, essas interações se refletem em telas como cadastro, *login*, edição de informações e atualização de credenciais, evidenciando a jornada do usuário pela interface.

Figura 1: Gerenciamento de conta e perfil



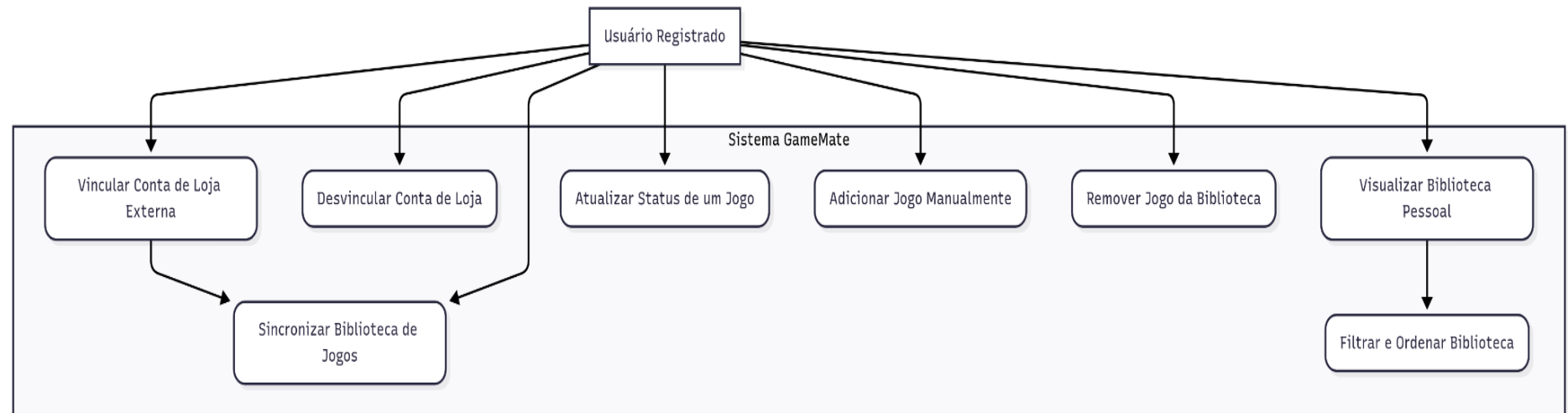
Fonte: O autor

3.4.2 GERENCIAMENTO DA BIBLIOTECA E CONTAS EXTERNAS

Este diagrama foca nas principais funcionalidades disponíveis para um Usuário Registrado no sistema. Ele apresenta como o usuário interage com sua biblioteca pessoal de jogos, gerencia o *status* de cada título e administra a vinculação com contas de lojas externas, como a *Steam*. A sincronização de jogos aparece como resultado direto dessa vinculação, garantindo que o acervo esteja sempre atualizado.

No *frontend*, essas interações se traduzem em telas como a visualização e filtragem da biblioteca, a atualização do status de jogo (por exemplo, “jogando”, “finalizado” ou “planejando jogar”) e a seção de configuração para vinculação e gerenciamento das contas externas. Essas representações visuais reforçam o fluxo descrito no diagrama, permitindo que o usuário perceba de forma clara cada ação disponível.

Figura 2 - Gerenciamento da biblioteca e contas externas



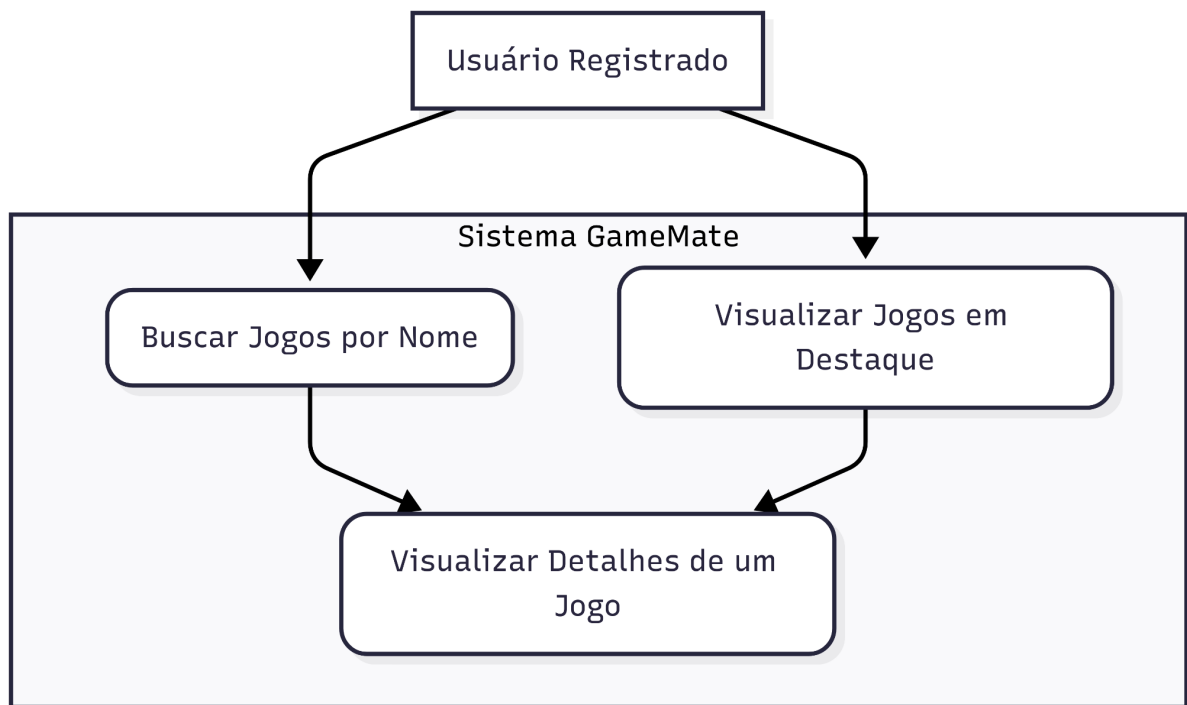
Fonte: O autor

3.4.3 DESCOBERTA DE JOGOS

Este diagrama representa as funcionalidades de exploração do catálogo de jogos da plataforma. Ele permite pesquisar títulos, aplicar filtros e visualizar informações detalhadas sobre cada jogo. Essas interações também podem estar integradas à biblioteca pessoal do usuário, facilitando a descoberta e organização de novos conteúdos.

No *frontend*, essas funcionalidades se traduzem em telas de busca com filtros, listas de resultados e páginas de detalhes dos jogos, oferecendo uma experiência clara e responsiva para o usuário.

Figura 3: Gerenciamento da biblioteca de jogos

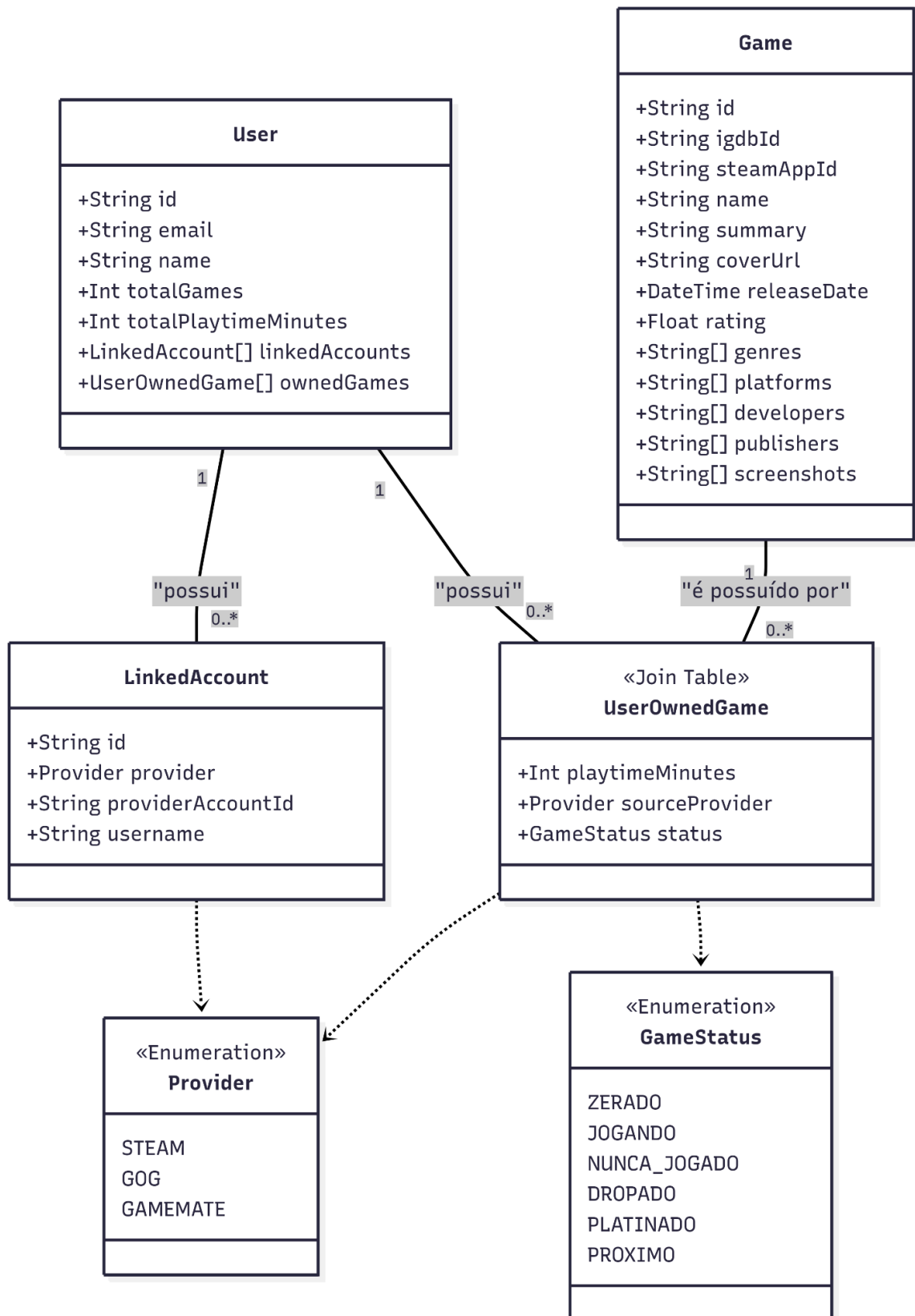


Fonte: O autor

3.5 DIAGRAMAS DE CLASSE

O Diagrama de Classes do Banco de Dados mostra como os dados estão organizados e relacionados no *backend*. Para o *frontend*, compreender esse diagrama é fundamental, pois ele revela quais informações estarão disponíveis, como serão estruturadas e como podemos acessá-las. Isso facilita a integração entre a interface e o sistema, garantindo que os dados sejam exibidos corretamente e que a experiência do usuário seja mais fluida e consistente.

Figura 4: Modelo de Entidades (Banco de Dados)

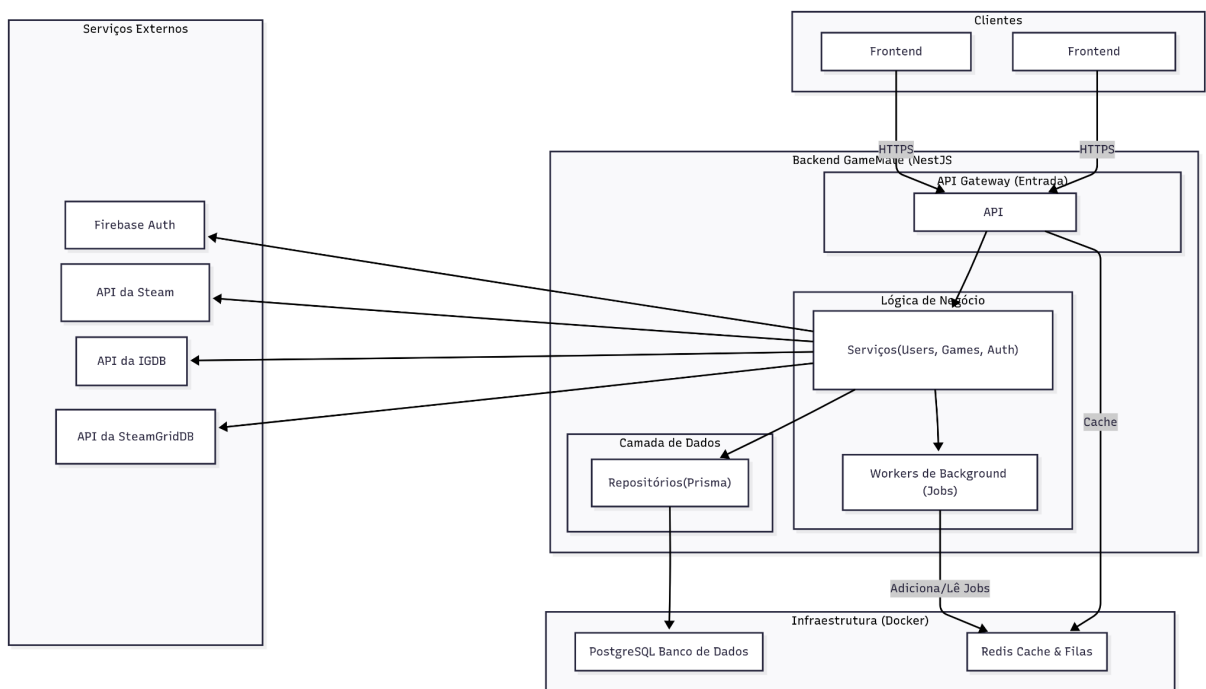


Fonte: O autor

3.6 ARQUITETURA DO SISTEMA

A arquitetura do sistema foi projetada para suportar de forma robusta e escalável as aplicações cliente (*Frontend*). O diagrama abaixo ilustra como o *Frontend* interage com os serviços de *backend* e, indiretamente, com sistemas externos para entregar a experiência completa ao usuário.

Figura 5 - Diagrama Geral da Arquitetura do Sistema

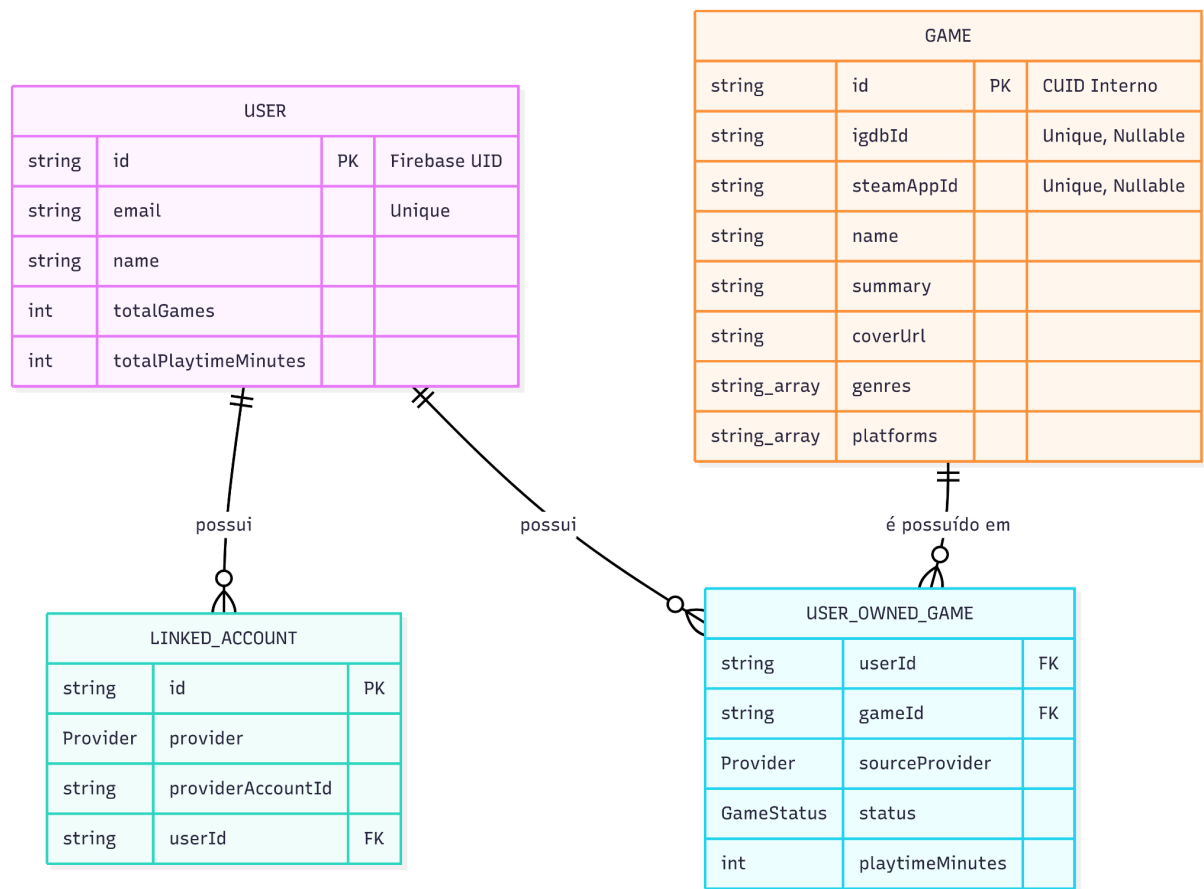


Fonte: O autor

3.7 DIAGRAMA DE ENTIDADE-RELACIONAMENTO

O Diagrama de Entidade-Relacionamento (DER) a seguir modela a estrutura de dados que o *Frontend* consome através da *API* para construir a experiência do usuário no sistema GameMate. Ele define as informações disponíveis para exibição em telas de perfil, bibliotecas de jogos e páginas de detalhes.

Figura 6 - Diagrama de Entidade-Relacionamento

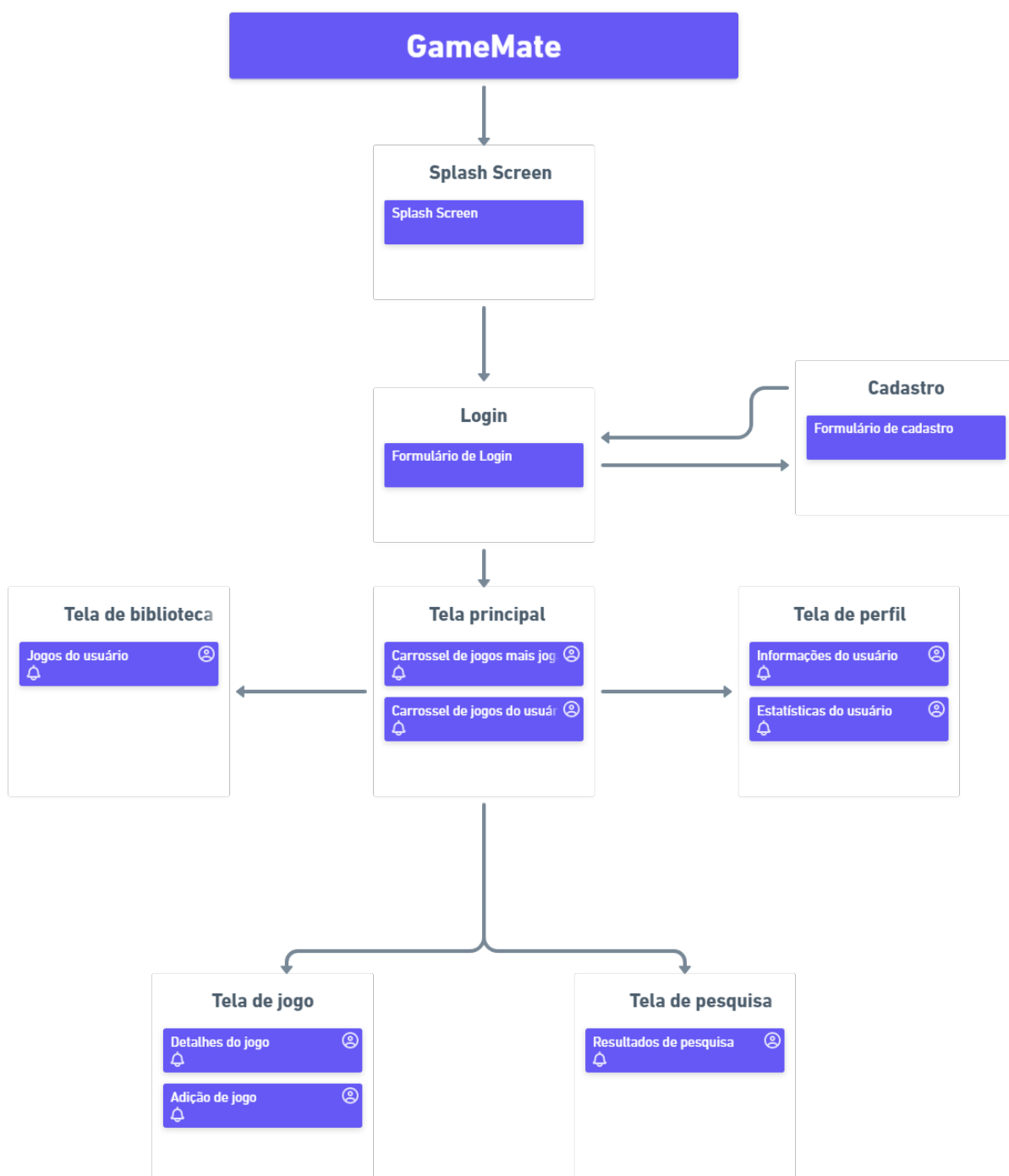


Fonte: O autor

3.8 MAPA DO SITE

O mapa do site apresenta a estrutura de navegação do aplicativo, evidenciando as conexões entre as principais telas e funcionalidades. Esse recurso auxilia na compreensão da arquitetura da aplicação e nos fluxos de interação previstos para o usuário.

Figura 7 - Mapa do site



Fonte: O autor

O fluxo de navegação do aplicativo inicia na tela de *splash*, que apresenta a logo do app enquanto o sistema é carregado. Em seguida, o usuário é direcionado para a tela de *login*, onde pode autenticar-se com suas credenciais ou optar por realizar o cadastro caso ainda não possua uma conta. Ao concluir o cadastro, o sistema retorna para a tela de *login* para que o novo usuário possa realizar o acesso.

Alternativamente, o usuário pode escolher realizar o *login* via Google diretamente na tela de *login*, sendo redirecionado para a tela principal após a autenticação. A partir da tela principal, o usuário pode navegar para a biblioteca de jogos, onde pode acessar a tela de detalhes ao clicar em um jogo específico. A navegação entre a biblioteca, a tela principal e o perfil é feita por meio de uma barra de navegação inferior (*bottom navbar*), facilitando o acesso rápido a essas seções. Além disso, a tela principal dispõe de um ícone que leva à tela de busca, permitindo encontrar novos títulos. O usuário pode também abrir o perfil para gerenciar suas informações pessoais. Essa estrutura proporciona uma navegação clara e intuitiva, facilitando o acesso às funcionalidades mais utilizadas com o mínimo de etapas.

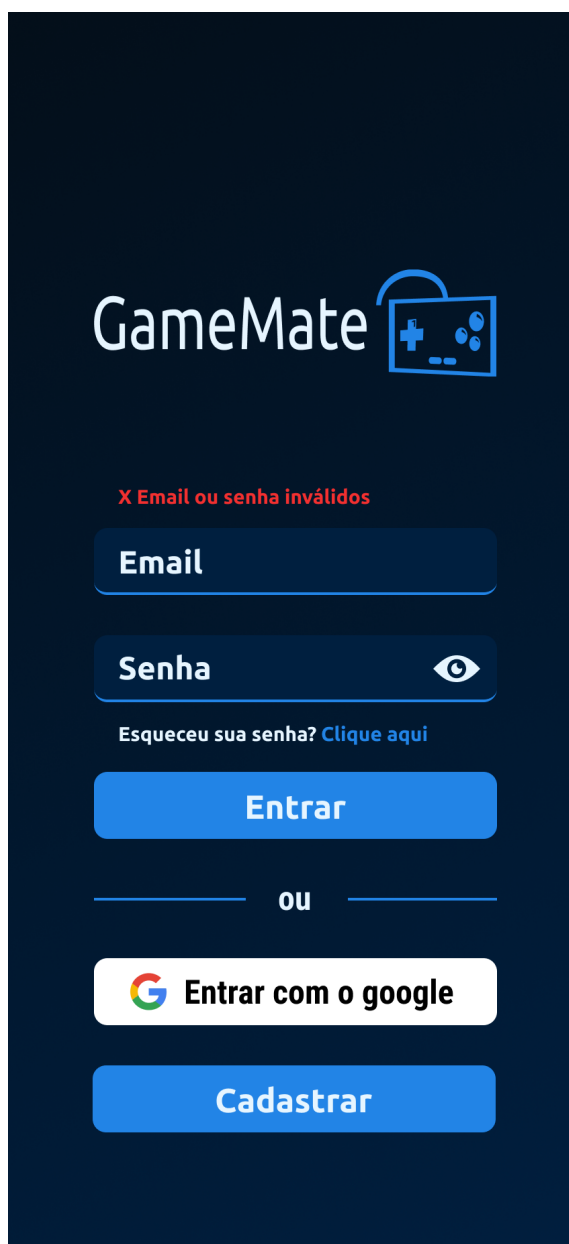
3.8 INTERFACE

Nesta seção, são apresentadas as telas desenvolvidas para o aplicativo, que foram prototipadas no Figma. O objetivo é demonstrar a organização visual e os principais fluxos de navegação pensados para o usuário. Cada imagem a seguir representa um dos principais fluxos de interação da aplicação, sendo acompanhada de uma breve descrição do seu objetivo e funcionalidades.

Figura 8 – Tela de *login*

A tela de *login* é responsável por permitir que o usuário acesse sua conta de forma rápida e segura. Apresenta campos para e-mail e senha, um botão de acesso destacado, opções para cadastro ou recuperação de senha e *login* com o Google, mantendo a navegação intuitiva.

Figura 8 - Tela de login



A tela de login do GameMate apresenta o logotipo da marca no topo, seguido por uma mensagem de erro em vermelho: "X Email ou senha inválidos". Abaixo, há campos de entrada para "Email" e "Senha", este último com ícone de olho para alternar visibilidade. Um link "Esqueceu sua senha? Clique aqui" está posicionado entre os campos. O formulário é concluído por um botão "Entrar" em azul. Abaixo, uma seção separada por uma linha horizontal com o texto "ou" oferece a opção "Entrar com o google" através de um botão branco com o ícone do Google. O botão "Cadastrar" em azul está na base da interface.

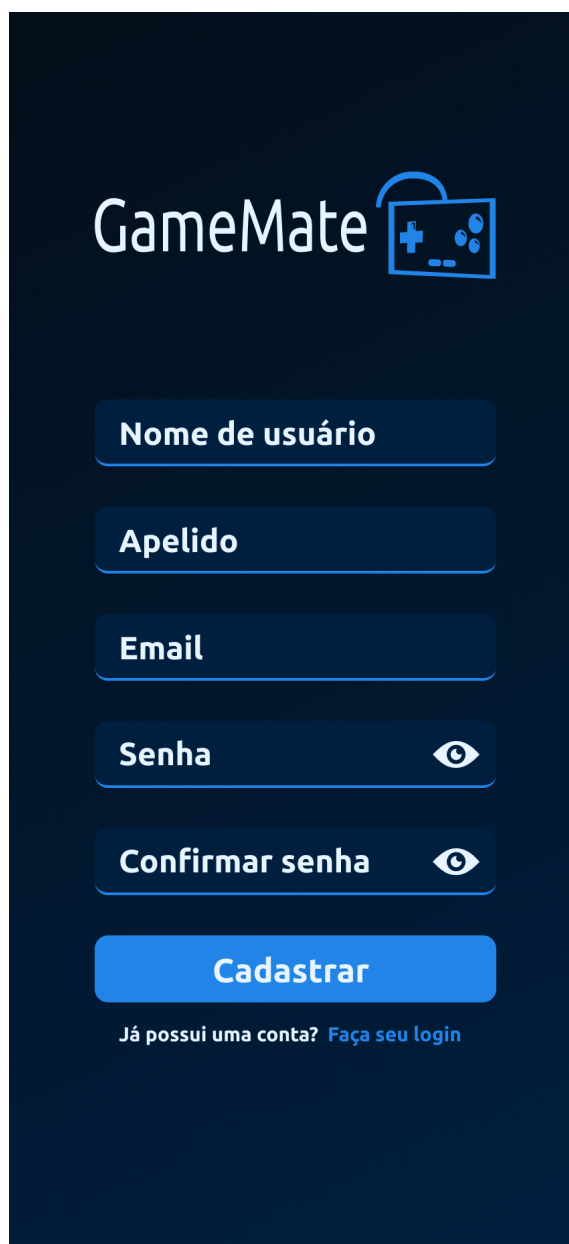
Fonte: O autor

Após a autenticação, o usuário é redirecionado para a tela principal, garantindo uma experiência personalizada e preservando a segurança dos dados.

Figura 9 – Tela de cadastro

A tela de cadastro possibilita que novos usuários criem uma conta na plataforma. Contém campos essenciais como nome, apelido, e-mail, senha e confirmação de senha, organizados de forma clara e objetiva.

Figura 9 - Tela de cadastro

A imagem mostra a tela de cadastro do aplicativo GameMate. No topo, há o logo "GameMate" em branco e um ícone de controle de jogo em azul. Abaixo, há cinco campos de entrada com bordas arredondadas e um brilho azul: "Nome de usuário", "Apelido", "Email", "Senha" e "Confirmar senha". Os campos "Senha" e "Confirmar senha" possuem ícones de olho para alternar a visibilidade. Abaixo dos campos, há um botão azul com o texto "Cadastrar" em branco. No rodapé, há o texto "Já possui uma conta? Faça seu login" em branco, onde "Faça seu login" é um link azul.

Fonte: O autor

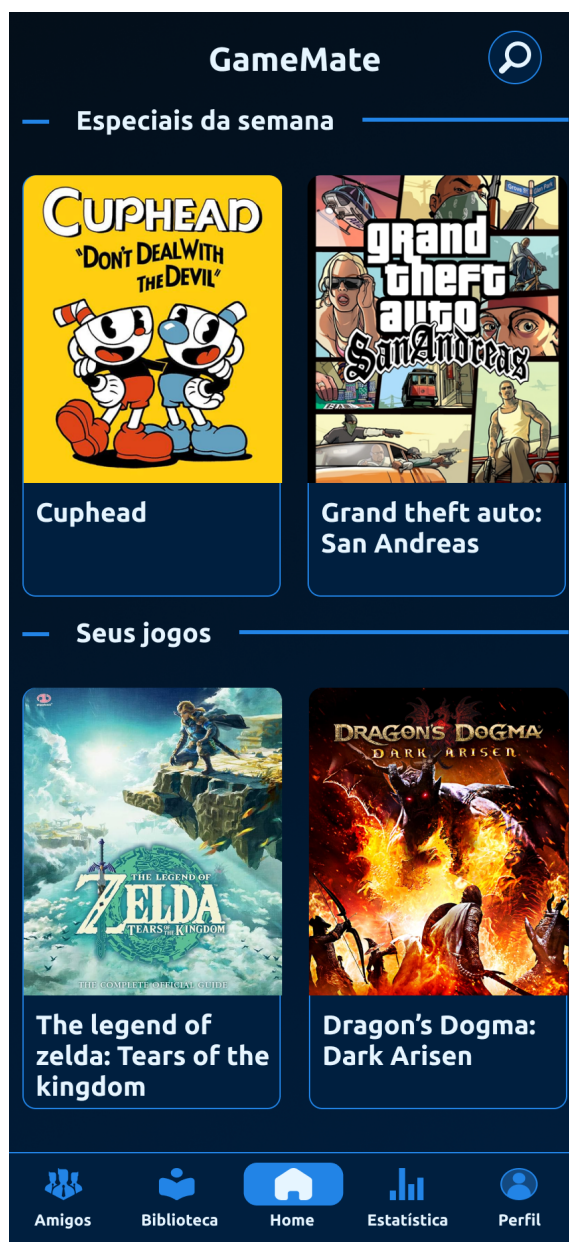
O *layout* segue o padrão visual do aplicativo, garantindo coerência na identidade e facilitando a compreensão dos campos. Ao concluir o preenchimento, o usuário será redirecionado para tela de login para entrar em sua conta e acessar as principais funções do aplicativo.

Figura 10 – Tela principal

A tela principal apresenta uma visão geral personalizada para o usuário. No topo, há um carrossel com os jogos mais jogados da semana, e logo abaixo, outro

carrossel com os jogos presentes na biblioteca do usuário.

Figura 10 - Tela principal do aplicativo



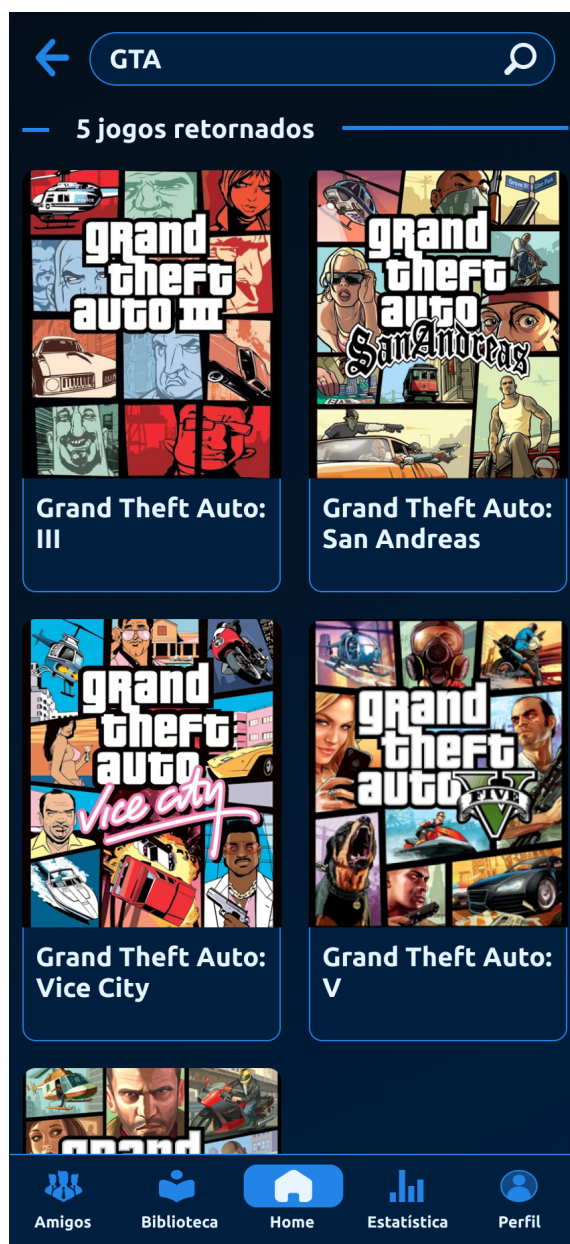
Fonte: O autor

Essa organização visual oferece praticidade, permitindo que o usuário acesse rapidamente seus títulos favoritos e descubra novos conteúdos recomendados.

Figura 11 – Tela de pesquisa

A tela de pesquisa permite ao usuário localizar rapidamente jogos específicos através do nome do mesmo. O campo de pesquisa é de fácil acesso e os resultados são exibidos de forma visual e organizada.

Figura 11 - Tela de pesquisa



Fonte: O autor

A funcionalidade de pesquisa amplia a usabilidade do aplicativo, permitindo descobrir títulos fora da biblioteca e do carrossel principal.

Figura 12 – Tela de detalhes do jogo

Ao selecionar um título, o usuário é levado para a tela de detalhes do jogo, onde pode visualizar informações como capa, descrição, gênero, data de lançamento e adicionar status como por exemplo se já zerou ou não o título.

Figura 12 - Tela de jogo

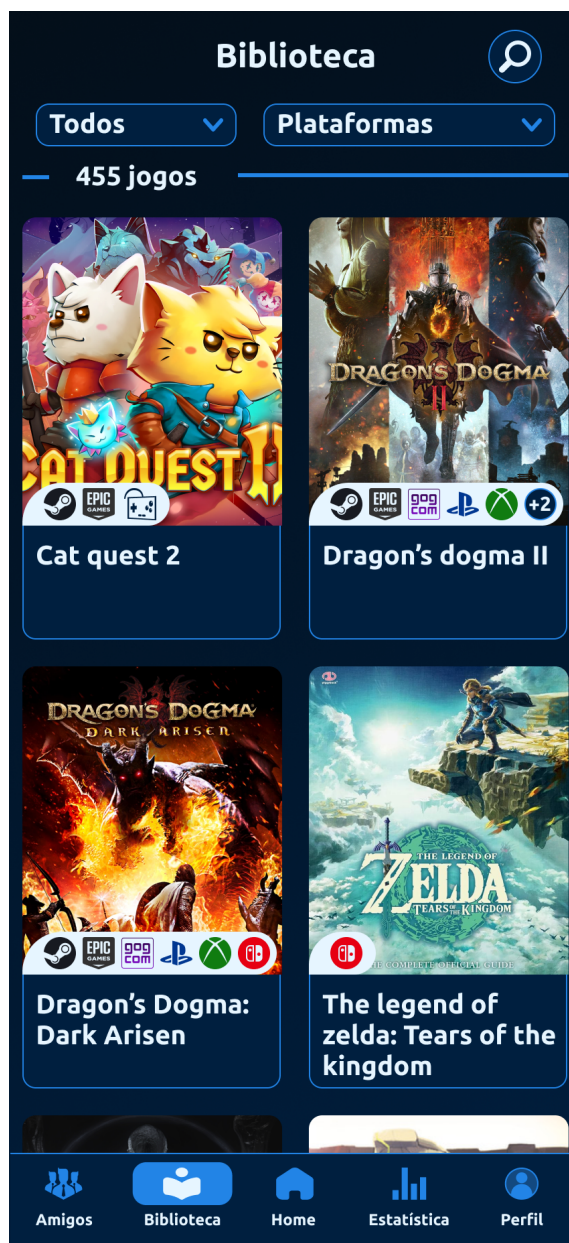


Fonte: O autor

Figura 13 – Tela da biblioteca

A biblioteca reúne todos os jogos adquiridos ou adicionados pelo usuário, organizados de forma visual e acessível. É possível filtrar ou buscar títulos para facilitar a navegação.

Figura 13 - Tela de biblioteca



Fonte: O autor

A disposição dos elementos privilegia a visualização das capas, permitindo que o usuário reconheça rapidamente seus jogos.

Figura 14 – Tela de perfil

A tela de perfil concentra as informações pessoais do usuário, incluindo nome e imagem de perfil, estatísticas. Também oferece opções de edição de perfil,

compartilhamento e vinculação com plataformas de jogos.

Figura 14 - Tela de perfil



Fonte: O autor

O *layout* limpo e organizado facilita a gestão das informações pessoais e mantém a coerência visual com o restante do aplicativo.

4 SOFTWARE

Este capítulo apresenta o processo de implementação do aplicativo GameMate, abordando sua arquitetura, tecnologias utilizadas, forma de distribuição e os tipos de testes aplicados durante o desenvolvimento. O foco principal está na construção da interface e na experiência do usuário, uma vez que o projeto concentra-se na camada de *frontend*. Ainda assim, é apresentada uma visão geral do *backend* responsável por suporte às funcionalidades essenciais do sistema.

4.1 IMPLANTAÇÃO

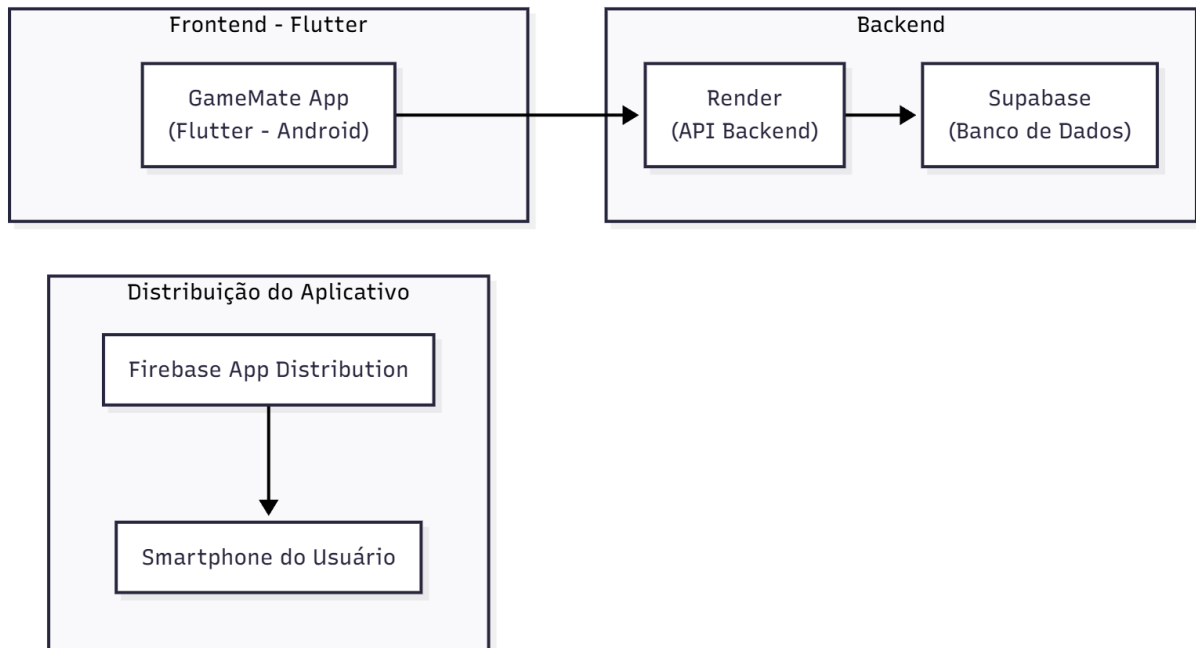
O GameMate foi desenvolvido em *Flutter*, permitindo criar uma interface consistente, fluida e compatível com a maior parte dos dispositivos *Android*. A distribuição do aplicativo para os usuários avaliadores foi realizada por meio do *Firebase App Distribution*, plataforma que facilita o envio de versões de teste. Os usuários receberam um convite por *e-mail*, instalaram o aplicativo diretamente em seus *smartphones* e puderam experimentar as funcionalidades mais recentes.

Embora o projeto tenha foco no *frontend*, o sistema possui um *backend* que dá suporte às funcionalidades de armazenamento e consulta de dados. Esse *backend* está hospedado na Render, enquanto o banco de dados utilizado é o Supabase, responsável por armazenar informações como jogos do usuário e dados de cadastro. A seguir os *links* do repositório e Google *App Distribution*:

<https://github.com/Gabriell1507/gamemate-mobile>

<https://appdistribution.firebase.dev/i/ba2cf1b8862d5fdc>

Figura 15 - Diagrama de implantação



Fonte: O autor

4.2 TESTES

Nessa etapa, o objetivo foi avaliar exclusivamente a usabilidade do GameMate, analisando como os usuários interagem com o sistema e qual era sua percepção geral de uso. Para isso, foi aplicado apenas um método formal de avaliação: o *System Usability Scale* (SUS), desenvolvido por John Brooke em 1986, um instrumento amplamente utilizado para medir a qualidade da experiência do usuário em interfaces digitais.

O teste contou com a participação de 8 usuários, que utilizaram a versão final do aplicativo antes de responder ao questionário. Esse procedimento permitiu que as respostas refletissem a experiência real de uso do GameMate, oferecendo dados confiáveis para a análise de usabilidade.

4.2.1 DESCRIÇÃO DO MÉTODO SUS

O *System Usability Scale* (SUS) é um dos instrumentos mais adotados internacionalmente para avaliar a usabilidade de interfaces digitais. Criado por John Brooke em 1986, o método se baseia em um questionário com 10 afirmações, que o participante responde logo após utilizar o sistema. Cada afirmação é avaliada por meio de uma escala *Likert* de cinco níveis, que varia entre:

- 1 - Discordo totalmente
- 2 - Discordo
- 3 - Neutro
- 4 - Concordo
- 5 - Concordo totalmente

O questionário intercala proposições positivas e negativas, cobrindo aspectos essenciais como facilidade de uso, coerência entre elementos da interface, sensação de confiança e percepção geral da experiência.

4.2.1.1 CÁLCULO DA PONTUAÇÃO

Para transformar as respostas em um índice final de usabilidade, o SUS utiliza um procedimento próprio de normalização:

- Para itens positivos (2, 4, 6, 8, 10), subtraia 1 do valor da resposta.
- Para itens negativos (1, 3, 5, 7, 9), subtraia a resposta de 5.
- Some os valores ajustados de todas as respostas e depois multiplique a soma por 2,5 para escalar o resultado para uma pontuação de 0 a 100.

Interpretação dos Resultados

A literatura especializada geralmente classifica a pontuação do SUS da seguinte maneira:

- Acima de 80 - Usabilidade excelente;
- Entre 68 e 80 - Usabilidade considerada boa ou aceitável;
- Abaixo de 68 - Usabilidade insatisfatória, sugerindo necessidade de melhorias.

4.2.3 APLICAÇÃO DO TESTE

A aplicação do teste SUS foi realizada após os participantes utilizarem o GameMate em sua versão final. Os usuários interagiram livremente com o sistema, explorando suas principais funcionalidades, de modo que suas impressões fossem formadas a partir do uso real do aplicativo.

Em seguida, eles responderam ao questionário SUS, composto pelas dez afirmações tradicionais avaliadas em escala *Likert*. As afirmações aplicadas foram:

1. Eu acho que gostaria de usar o GameMate com frequência.
2. Eu acho o GameMate desnecessariamente complexo.
3. Eu achei o GameMate fácil de usar.
4. Eu acho que precisaria de ajuda de uma pessoa com conhecimentos técnicos para usar o GameMate.
5. Eu acho que as várias funções do GameMate estão bem integradas.
6. Eu acho que o GameMate apresenta muita inconsistência.
7. Eu imagino que as pessoas aprenderão a usar o GameMate rapidamente.
8. Eu achei o GameMate confuso ou atrapalhado de usar.
9. Eu me senti confiante ao usar o GameMate.
10. Eu precisei aprender várias coisas novas antes de conseguir usar o GameMate.

4.2.4 RESULTADO E INTERPRETAÇÃO DA PONTUAÇÃO DO SUS

Após a aplicação do questionário SUS, as respostas dos dez itens foram convertidas em seus respectivos valores ajustados, seguindo o procedimento padrão do método. Os itens ímpares (1, 3, 5, 7 e 9) representam afirmações positivas, enquanto os itens pares (2, 4, 6, 8 e 10) correspondem às afirmações negativas sobre a experiência de uso.

– Itens Ímpares - Afirmações positivas

- Q1: $4,625 - 1 = 3,625$
- Q3: $4,375 - 1 = 3,375$
- Q5: $4,0 - 1 = 3,0$
- Q7: $4,125 - 1 = 3,125$
- Q9: $4,25 - 1 = 3,25$

– Soma dos itens positivos:

$$3,625 + 3,375 + 3,0 + 3,125 + 3,25 = 16,375$$

– Itens Pares - Afirmações negativas

- Q2: $5 - 1,0 = 4,0$
- Q4: $5 - 1,5 = 3,5$

- Q6: $5 - 2,125 = 2,875$
 - Q8: $5 - 1,625 = 3,375$
 - Q10: $5 - 1,0 = 4,0$
- Soma dos itens negativos:
- $$4,0 + 3,5 + 2,875 + 3,375 + 4,0 = 17,75$$
- Soma total dos valores ajustados
- $$16,375 \text{ (ímpares)} + 17,75 \text{ (pares)} = 34,125$$
- Pontuação final SUS
- $$\text{Pontuação SUS} = 34,125 \times 2,5 = 85,31$$

Com uma pontuação de aproximadamente 85, o GameMate apresenta uma usabilidade excelente, superando amplamente a média de 68 pontos, que é considerada o limite mínimo para uma usabilidade aceitável. Esse resultado indica que os usuários perceberam o sistema como intuitivo, eficiente e de fácil aprendizado, com funções bem integradas e experiência de uso satisfatória.

5 CONSIDERAÇÕES FINAIS

O presente trabalho detalha o relatório técnico e desenvolvimento do GameMate, um *frontend* para aplicativo móvel para solucionar um problema que ocorre no universo dos jogos digitais: a fragmentação das bibliotecas de jogos. Com a proliferação de plataformas de distribuição como *Steam*, *Epic Games* e *GOG*, os jogadores acumulam títulos em múltiplos serviços, o que torna o gerenciamento de suas coleções, o acompanhamento de progresso e a visualização de estatísticas uma tarefa complexa e descentralizada. Este projeto partiu da hipótese de que o desenvolvimento de uma aplicação móvel pode facilitar significativamente a organização dessas bibliotecas, oferecendo uma experiência centralizada, intuitiva e visualmente coesa.

Para alcançar o objetivo proposto, foi desenvolvido um protótipo funcional do *frontend* do aplicativo, superando os desafios inerentes à integração de dados fragmentados e à construção de uma experiência de usuário coesa. A escolha tecnológica demonstrou-se acertada, com o emprego do *framework Flutter* para a construção da interface, garantindo responsividade e fluidez. A linguagem *Dart* foi

utilizada para estruturar a interface e a lógica de comunicação com as *APIs* externas , enquanto o gerenciador de estado GetX proporcionou a base para uma arquitetura reativa e escalável. A materialização do *design*, primeiramente prototipado no Figma , resultou em um modelo de interface projetado para interagir de forma eficiente com serviços como Firebase *Authentication* para o *login* e consumir informações de *APIs* como IGDB e *Steam* para apresentar os dados dos jogos de forma rica e visualmente atraente.

Com isso, conclui-se que o desenvolvimento do GameMate apresentou uma solução viável e eficaz para o desafio da fragmentação das bibliotecas de jogos digitais. O protótipo funcional não apenas demonstrou que é possível centralizar e organizar de forma intuitiva coleções dispersas em múltiplas plataformas, como também mostrou que tecnologias modernas como *Flutter*, *Dart* e GetX são adequadas para criar uma experiência fluida, escalável e visualmente atraente. Esse trabalho abre caminho para futuras melhorias e ampliações, fortalecendo a proposta de oferecer aos jogadores uma ferramenta completa para o gerenciamento integrado de suas bibliotecas de jogos.

REFERÊNCIAS

BORGES, B. A. F.; SIDARTA, I. A.; SOUSA, A. M.; COELHO, B.; DARIN, T. Experiência do Usuário em Jogos Digitais: uma catalogação de instrumentos de avaliação. Anais do WIPLAY 2022. Disponível em: <https://sol.sbc.org.br/index.php/wiplay/article/view/7841/7716>. Acesso em: 30 jun. 2025.

BROOKE, J. SUS: a quick and dirty usability scale. November 1995. Disponível em: https://www.researchgate.net/publication/228593520_SUS_A_quick_and_dirty_usability_scale. Acesso em: 25 nov. 2025.

DART. Dart documentation. [S.l.], [s.d.]. Disponível em: <https://dart.dev/docs>. Acesso em: 09 out. 2025.

FIGMA. Figma. [S.l.], [s.d.]. Disponível em: <https://www.figma.com/>. Acesso em: 09 out. 2025.

FLUTTER. Docs | Flutter. [S.l.], [s.d.]. Disponível em: <https://docs.flutter.dev>. Acesso em: 09 out. 2025.

FRAGMENTAÇÃO NO MERCADO DE JOGOS DIGITAIS. In: SIMPÓSIO BRASILEIRO DE JOGOS E ENTRETENIMENTO DIGITAL (SBGAMES), 13., 2014, Porto Alegre. Anais [...]. Porto Alegre: SBGames, 2014. Disponível em: <https://www.sbgames.org/sbgames2014/papers/industry/full/303-industryfullpages.pdf>. Acesso em: 30 jun. 2025.

GAME TRACKING. Repositório Institucional do Centro Paula Souza, [s.d.]. Disponível em: <https://ric.cps.sp.gov.br/bitstream/123456789/10865/1/Monografia%20GameTracking.pdf>. Acesso em: 30 jun. 2025.

GAMESINDUSTRY.BIZ. Global game audience reaches 3.7 billion. GamesIndustry.biz, [S.l.], 02 out. 2025. Disponível em: <https://www.gamesindustry.biz/dfc-global-game-audience-reaches-37-billion>. Acesso em: 02 out. 2025.

GOOGLE. Firebase Authentication | Firebase Documentation. [S.l.], [s.d.]. Disponível em: <https://firebase.google.com/docs/auth?hl=pt-br>. Acesso em: 09 out. 2025.

IGDB. IGDB API Documentation. [S.l.], [s.d.]. Disponível em: <https://api-docs.igdb.com/#getting-started>. Acesso em: 09 out. 2025.

NAKAMURA, W. T.; MARQUES, L. C.; REDMILES, D.; OLIVEIRA, E. H. T.; CONTE, T. Avaliação de UX em aplicações móveis: resumo de uma investigação sobre a influência de diferentes fatores em um app de compras baseado em chatbot. Anais Estendidos do IHC 2023. Disponível em: https://sol.sbc.org.br/index.php/ihc_estendido/article/view/26459/26282. Acesso em: 30 jun. 2025.

PUB.DEV. Get | Flutter package. [S.l.], [s.d.]. Disponível em: <https://pub.dev/packages/get>. Acesso em: 09 out. 2025.

STEAM. Steamworks Web API Documentation. [S.l.], [s.d.]. Disponível em: <https://steamcommunity.com/dev>. Acesso em: 14 out. 2025.

STEAMGRIDDB. SteamGridDB API v2. [S.l.], [s.d.]. Disponível em: <https://www.steamgriddb.com/api/v2>. Acesso em: 14 out. 2025.