

## UNIVERSIDADE SALVADOR

**ANÁLISE E DESENVOLVIMENTO DE SISTEMAS**

## Bruno do nascimento dos santos - 12723210249 Elane Cristina da Fonseca Souza - 1272223165 Gabriel do Carmo Adriano Maia - 12722133049 Luís Igor Nobre Dos Santos - 1272228572

**Victor Matheus dos Santos Pimenta - 12722133074**

RELATÓRIO A3

Salvador 2023

# Telas do Protótipo:

## Tela Inicial

Na tela inicial do nosso projeto, “Figura 1”, seguimos as heurísticas de Consistência e Padronização, Controle e Liberdade para o Usuário e a heurística de Visibilidade do Status do Sistema, para melhorar e experiência do usuário.

Nessa tela adicionamos os seguintes elementos:

* + Inicio (Para redirecionar para a página inicial).
  + Jogos.
  + Noticias.
  + Comunidade (Página com nosso grupo do facebook e discord).
  + Sobre Nós (Página com um breve resumo do nosso projeto).
  + Fazer Login (Para o usuário entrar em sua conta).
  + Criar Conta (Para que o usuário possa se cadastrar na plataforma).
  + Imagens dos jogos, seus respectivos nomes, uma pequena descrição e uma opção de jogar.

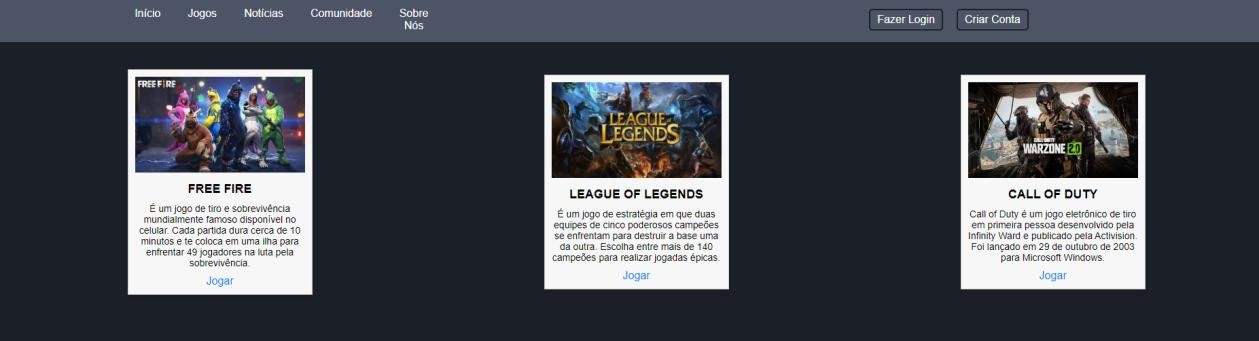


Figura 1: Tela Inicial.

## Login

Na tela de login, “Figura 2”, colocamos alguns elementos, são eles:

* + Botão “Início” no canto superior esquerdo que retorna para a tela inicial.
  + Imagem da logo do nosso site.
  + Campo: Nome de Usuário.
  + Campo: Senha.
  + Opção de recuperação de senha para o usuário.
  + Opção para iniciar o processo de cadastro do usuário, que o leva para uma outra tela que será mostrada em seguida, “Figura 3”.
  + Botão “Entrar” para efetuar a ação de login.

Para fazer essa tela, foi feita uma pesquisa em sites de jogos com o objetivo de manter o padrão e garantir a 4º heurística de Nielsen: Consistência e Padrão.

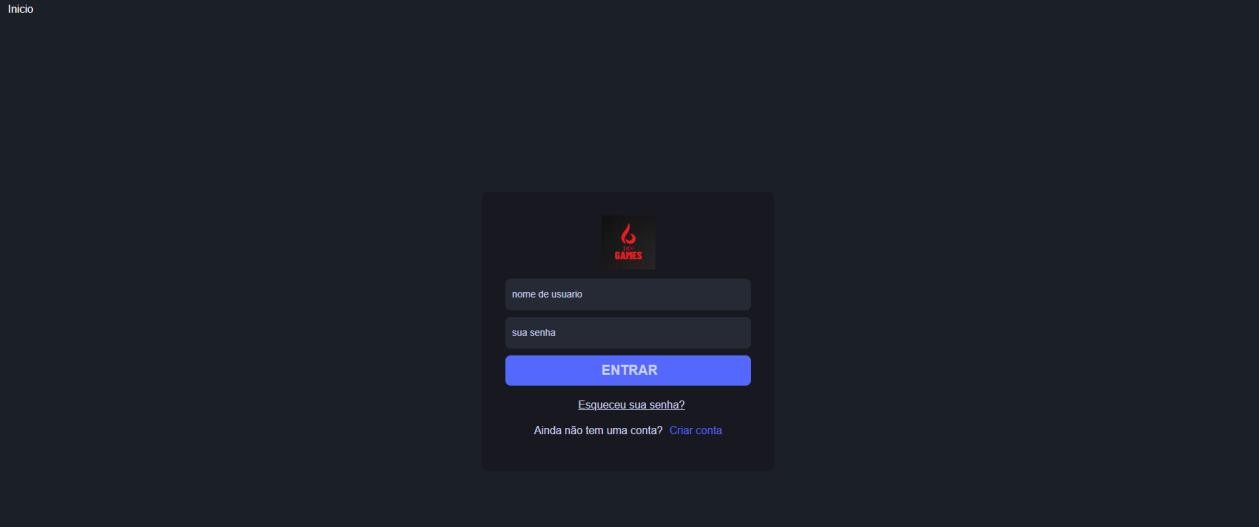


Figura 2: Tela de Login.

## Cadastro de Usuário

Tela de cadastro “Figura 3”, decidimos adicionar os seguintes elementos:

* + Campo: Endereço de email
  + Campo: Nome
  + Campo: Sobrenome
  + Campo: Usuário
  + Campo: Senha
  + Campo: Confirmar senha
  + Botão: Continuar
  + Botão: Voltar

Seguimos o mesmo padrão de telas de cadastros de jogos onlines, também fizemos a validação dos campos no Javascript para exibir mensagens de erro para o usuário. Trazendo a 5ª e 9ª heurísticas, que é Prevenção de Erros e Ajude os Usuários a Reconhecerem, Diagnosticarem e Recuperarem-se de Erros.”Figura 4”.

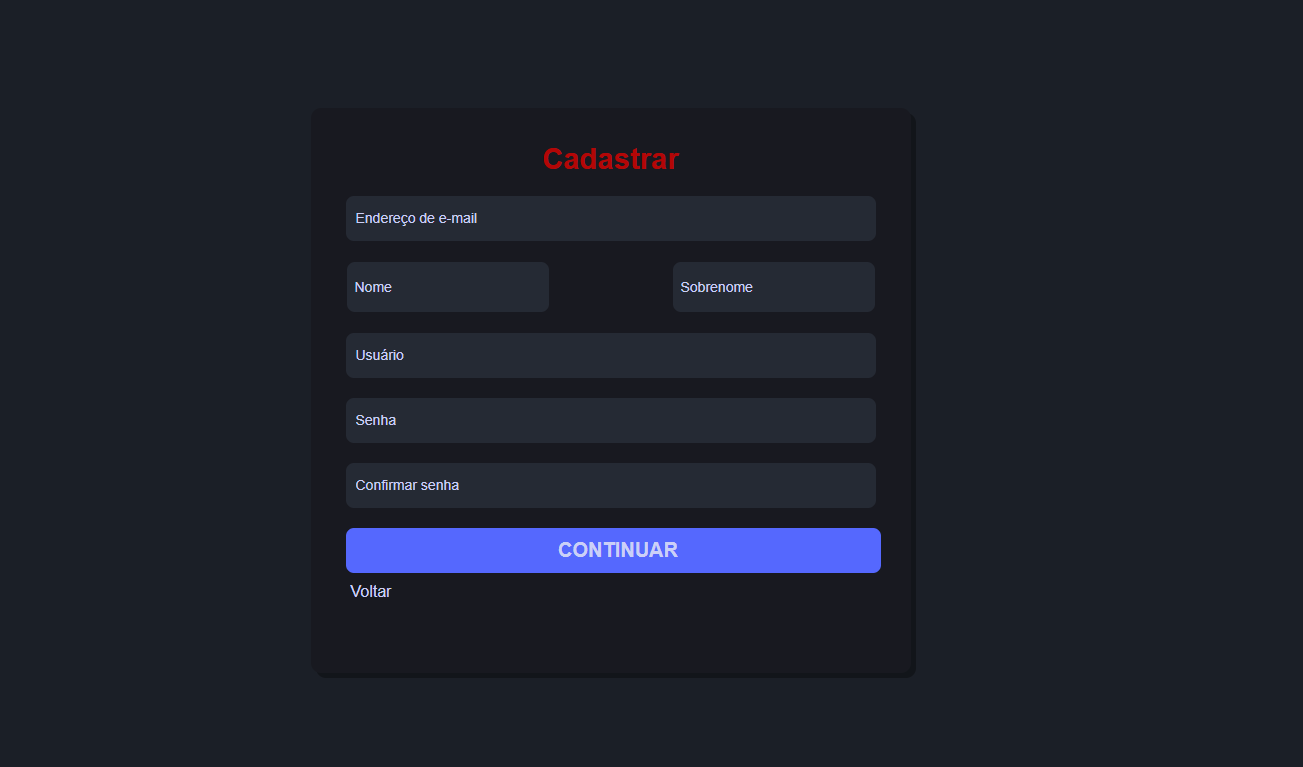


Figura 3: Tela de Cadastro de Usuário.

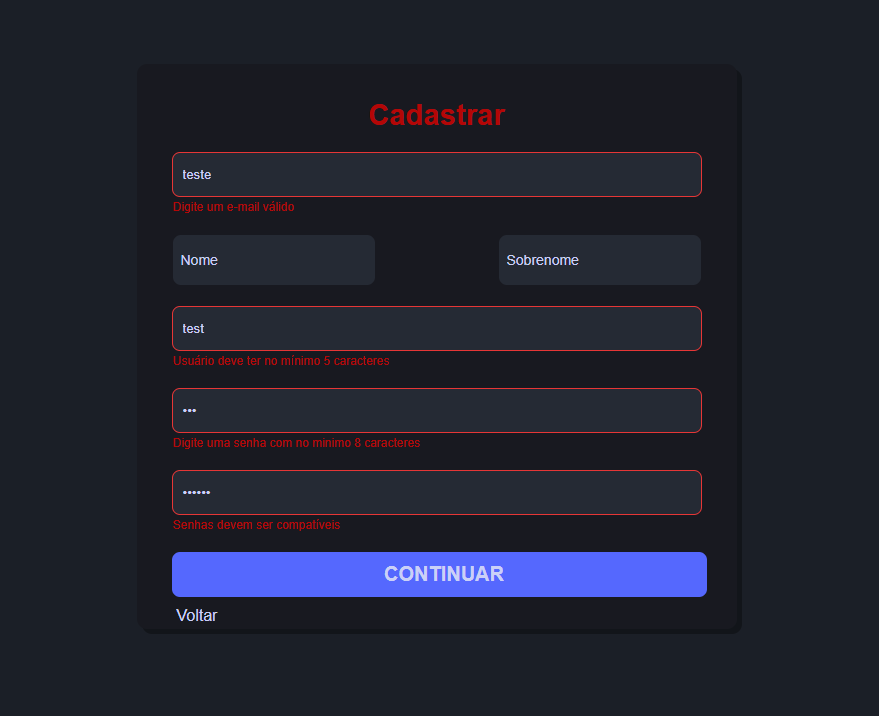


Figura 4:Tela de Cadastro com Erros.

## Recuperação de Senha

Nessa tela seguimos o mesmo padrão da tela de login e de cadastro, a sua finalidade é possibilitar que o usuário possa redefinir sua senha através da confirmação por e-mail. Ela possui os seguintes elementos:

* + Campo: E-mail
  + Botão: Enviar E-mail
  + Uma opção para retornar a tela de login.

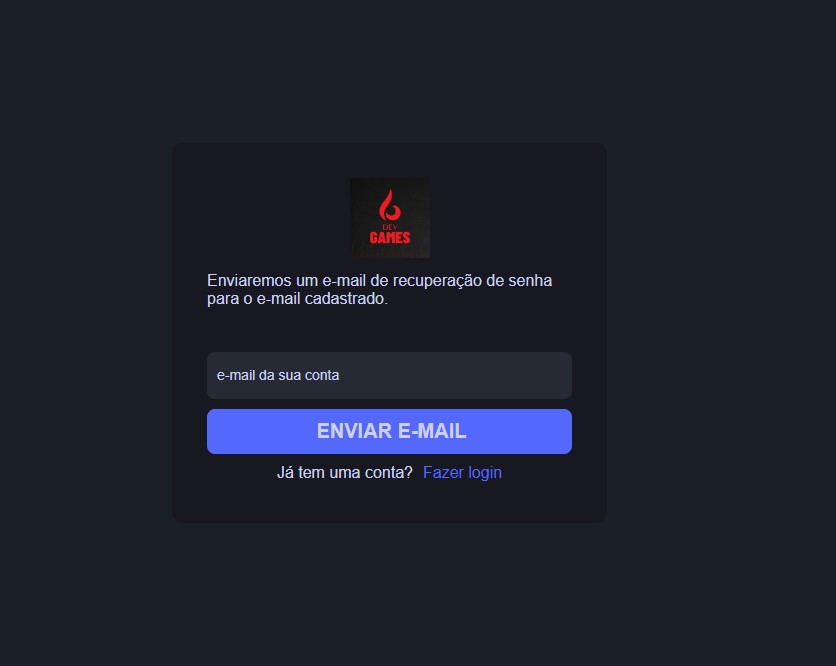


Figura 5: Tela de Recuperação de Senha.

# Figma:

Utilizando o Figma, uma plataforma de design baseada em nuvem, podemos criar uma ilustração em tempo real que representa como o site ficaria tanto em telas de computador quanto em dispositivos móveis. Esta ferramenta permite a visualização instantânea de informações detalhadas sobre o site, jogos e processos de cadastro. Além disso, a flexibilidade do Figma possibilita ajustes contínuos entre as versões de desktop e mobile, proporcionando uma experiência de design colaborativa e eficiente, tendo no total 6 telas ilustrativas na tela de computador, 10 telas ilustrando modo mobile.

## Tela de Login:

A tela de login “Figura 6” é composta pela logo do site “Dev Games”, por uma barra simulando o local onde o usuário irá colocar seu nome, senha e entrar, com informações sobre esqueceu senha e “ainda não tem uma conta?” crie uma, com a cor azul 8ED8ED, com a tela totalmente preta no fundo, uma barra na horizontal com a cor branca com o código FFFFFF 6%.

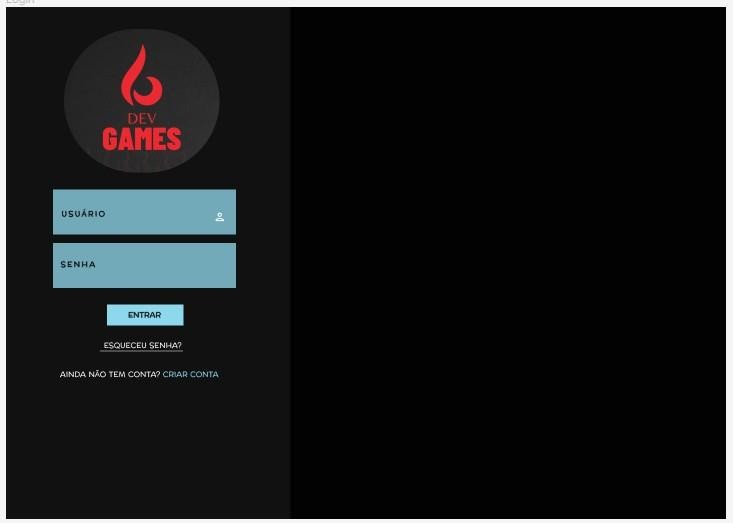


Figura 6: Tela de Login Figma.

## Tela de Cadastro de Usuário:

A segunda tela, “Figura 7”, inclui 5 barras, que são elas: endereço de e-mail, nome, sobrenome, nome de usuário e senha.

As barras são totalmente brancas, execeto em nome de usuário e senha, com uma legenda de cadastro vermelha e a logo na lateral esquerda “Dev Games”.

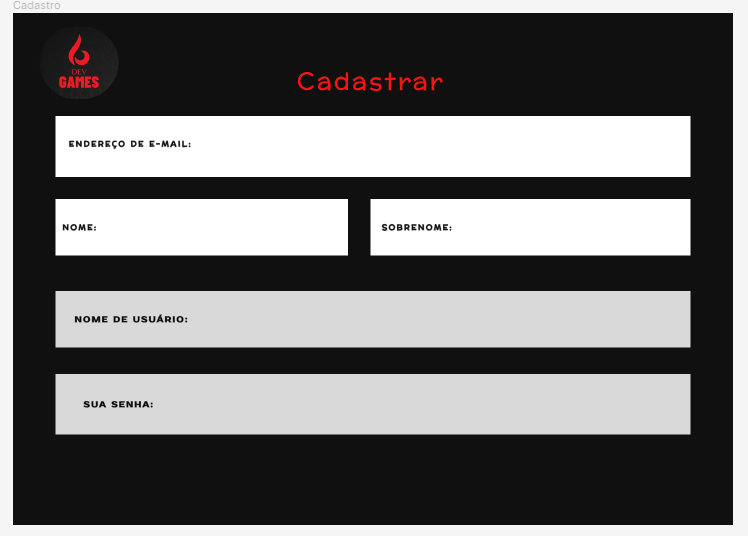


Figura 7: Tela de Cadastro Figma.

## Tela de Recuperação de Senha:

Para recuperação de senha, “Figura 8”, seguimos as seguintes etapas. O usuário inserindo o e-mail cadastrado na plataforma e, em seguida, ele(a) receberá um código de verificação de conta. Ao inserir o código, ele será direcionado para um caminho seguro que permitirá a restauração rápida e eficaz da sua conta.

Tendo uma opção se já tem conta/Fazer login. Essa tela é composta pela logo “Dev Games”, um pequeno texto de instrução para o usuário, uma barra para que ele informe o email para onde será enviando o código e a opção de fazer login.



Figura 8: Tela de Recuperação de Senha Figma.

## Tela com Usuário Logado:

Na tela de usuário logado, “Figura 9”, a logo é acompanhada por botões intuitivos: "Início" para uma visão personalizada, "Jogos" exibindo seis títulos detalhados, "Notícias" para atualizações, "Comunidade" para interação, "Sobre Nós" com um breve resumo do nosso projeto, "Sair" para encerrar a sessão e ao lado um ícone acompanhado do nome do usuário mostrando que ele está logado, trazendo visibilidade do status do sistema. Cada jogo na seção "Jogos" contém um breve resumo sobre o mesmo.

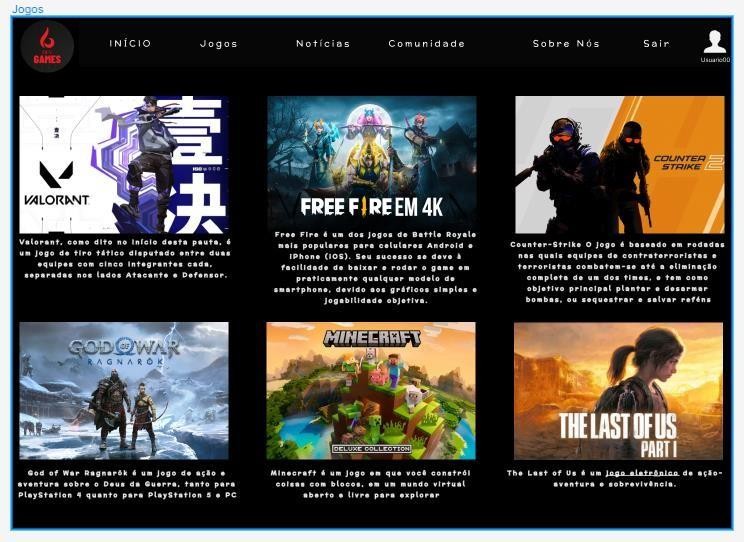


Figura 9: Tela com Usuário Logado.

## Mobile:

Tela de login no ambiente mobile, “Figura 10”, exibindo a logo Dev games, apresentando os campos de usuário e senha para o usuário preencher e fazer login. Podendo se conectar com as seguintes plataformas, Riot, Discord, Google, Facebook.

Figura 10: Tela de Login Mobile.

Agora temos 3 telas, “Figura 11”, a primeira trazendo alguns jogos disponíveis na plataforma, a segunda e terceira faz referência ao lobby de um jogo específico, dando a opção de agendar uma partida.

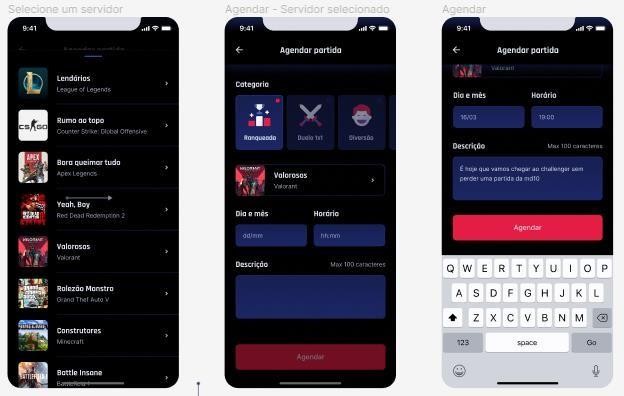


Figura 11: Telas Mobile.

# Wireframes:

Wireframes, usados para as representações visuais simplificadas para o projeto de usabilidade, podendo esboçar a estrutura básica de uma interface de usuário ou produto digital, como sites ou aplicativos. Destacando elementos na página visual e layout de conteúdo e funcionalidades interativas, sem incluir detalhes visuais elaborados. Ajudando na compreensão da estrutura do nosso projeto para ter como teste uma ferramenta eficaz para testar e validar conceitos de design antes do nosso desenvolvimento detalhado para o Figma.

Tendo um esboço de como ficaria a logo na parte ilustrativa do site, a barra de usuário e senha. Trazendo o esboço da tela de Login, “Figura 12”. Podendo fazer algumas alterações de como ficaria.

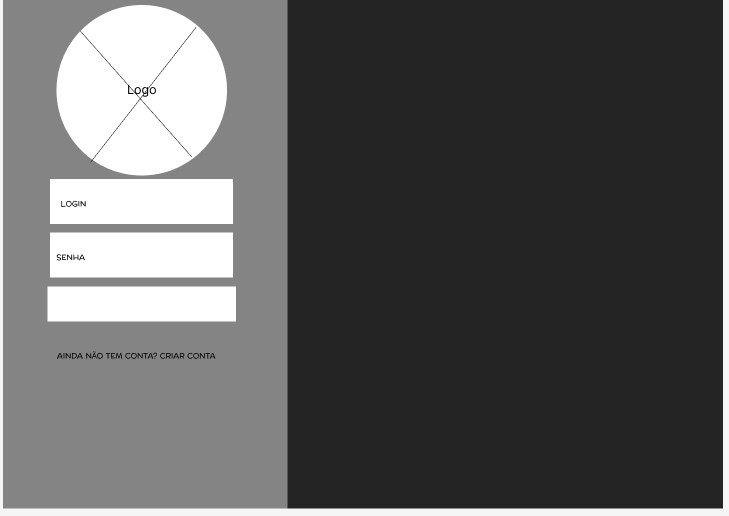


Figura 12: Wireframe Tela de Login.

Tela ilustrada de como ficaria a parte de cadastro na plataforma, com as informações e-mail, nome, sobrenome, usuário, senha e a logo na parte superior esquerda. “Figura 13”.

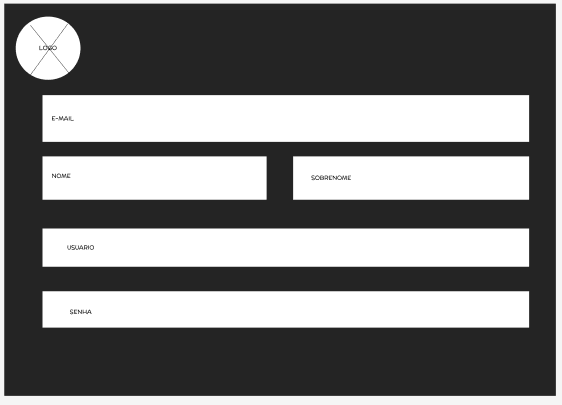


Figura 13: Wireframe Tela de Cadastro do Usuário.

Ilustrando dois tipos de telas uma sendo a tela Splash exibindo a logo.

Segunda tela exibindo o login do usuário e senha, uma pequena imagem breve de jogo, com uns pequenos acessos a serem inseridos na plataforma. ”Figura 14”

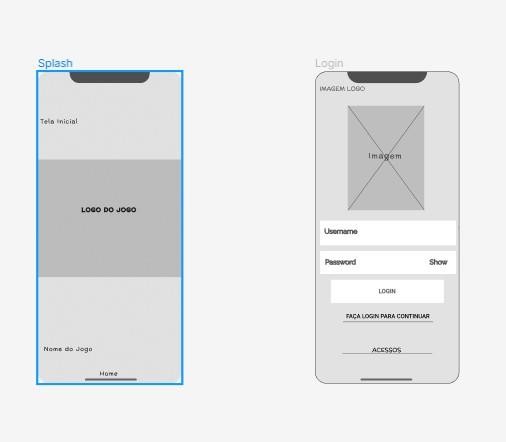


Figura 14: Tela Splash e Tela de Login feita no Wireframe.

Três telas sendo exibida, a primeira sendo os modelos de partida, “Ranqueada, Diversão, Duelo”, a segunda tendo a opção de agendamento, horário e descrição para o jogador, terceira tela exibe diversos jogos na plataforma. “Figura 15”.

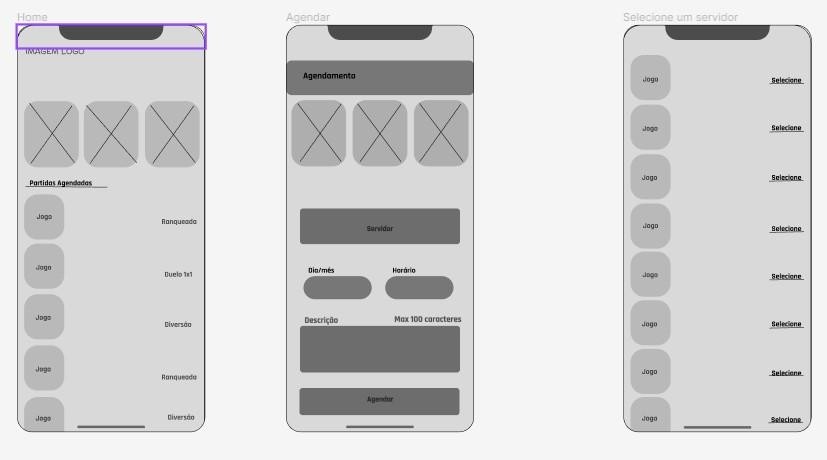


Figura 15: Tela no jogo, Tela de Agendamento, Tela dos Jogos.

**Relatório de Desenvolvimento: Sistema de Gerenciamento de Jogos e Plataforma - P02**

Neste relatório, documentamos o processo de desenvolvimento de um sistema completo para gerenciamento de jogos e plataforma, abrangendo o front-end, back-end, API, banco de dados e operações CRUD. O principal objetivo deste projeto foi criar uma solução eficaz para gerenciar jogos em uma plataforma digital, permitindo aos usuários realizar operações de cadastro, leitura, atualização e exclusão de dados.

Os objetivos deste projeto foram:

Desenvolver um sistema de front-end intuitivo e responsivo para os usuários interagirem com a plataforma.

Implementar um back-end robusto e escalável para processar solicitações do front-end e lidar com a lógica de negócios.

Projetar e desenvolver uma API para permitir a comunicação entre o front-end e o back-end de forma eficiente.

Criar um banco de dados adequado para armazenar informações sobre jogos, usuários e dados relacionados.

Implementar operações CRUD (Create, Read, Update, Delete) para gerenciar os dados dos jogos na plataforma.

Visão Geral da Arquitetura:

Front-end:

O front-end foi desenvolvido utilizando tecnologias/frameworks utilizados: HTML5, CSS3, JavaScript, React.js, visamos fornecer uma experiência amigável ao usuário, com uma interface intuitiva e responsiva.

Back-end e API:

Para o back-end, utilizamos a tecnologia: Node.js, fornecemos uma base sólida para gerenciar as requisições, implementar a lógica de negócios e permitir a interação com o banco de dados. A API foi desenvolvida para atender às demandas do front-end, permitindo a comunicação eficiente entre as camadas. Utilizamos axios, uma biblioteca do javaScript para as requisições HTTP e comunicação do back com o front.

Banco de Dados:

O banco de dados na qual optamos foi o relacional: MySQL, com a finalidade de armazenar as informações sobre jogos, usuários e plataformas.

Desenvolvimento do CRUD

Para possibilitar a gestão dos jogos na plataforma, criamos o CRUD:

Create: Permitindo a adição de novos jogos ao sistema.

Read: Possibilitando a leitura e exibição dos jogos cadastrados.

Update: Permitindo a atualização de informações de jogos existentes.

Delete: Habilitando a remoção de jogos do sistema.

No relatório inicial, apresentamos uma visão geral do que tinhamos em mente para o projeto de desenvolvimento do sistema de gerenciamento de jogos e plataforma final. Abaixo, segue imagens e explicações de algumas das partes principais do nosso "ecossistema".

1. **Tela Inicial:**

Na tela inicial “Figura 16” do nosso projeto, mudamos algumas coisas em relação a primeira tela do nosso protótipo.

Nessa tela optamos por deixar da seguinte forma:

- Noticias

- Comunidade

- Sobre Nós

- Fazer Login (Usuário entrar em sua conta já existente)

- Criar Conta (Usuário se cadastrar na plataforma)

- Carrossel com alguns jogos disponíveis.

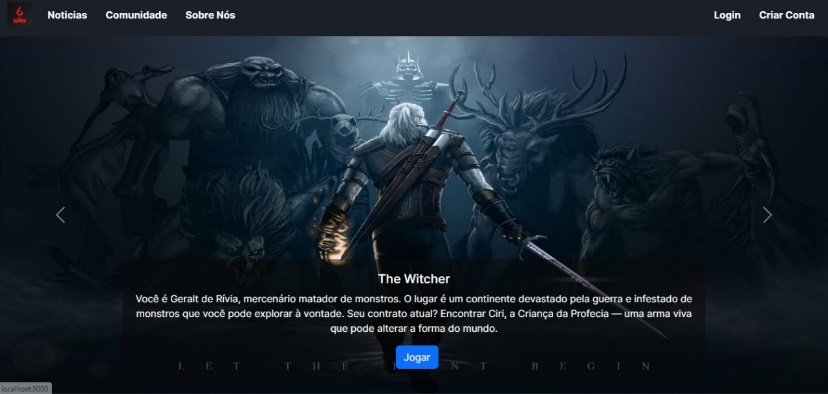


Figura 16: Tela inicial.

1. **Tela Comunidade:**

Na tela inicial, “Figura 17” do nosso projeto, criamos uma comunidade global conforme abaixo:

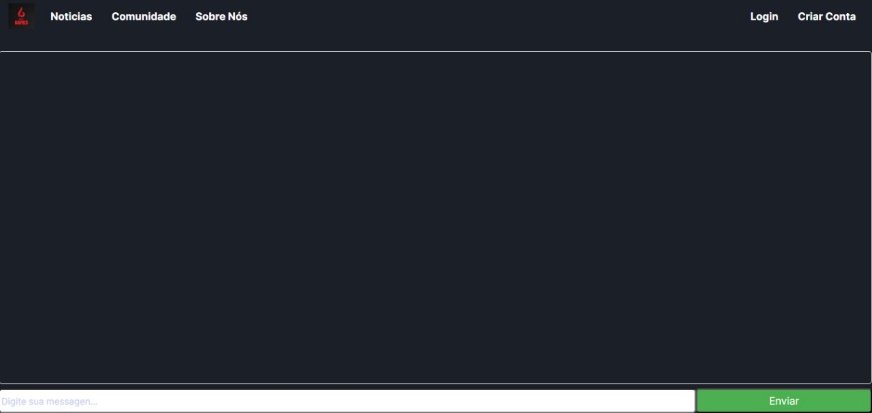


Figura 17: Tela comunidade.

1. **Quem somos nós:**

Na tela de quem somos nós “Figura 18” do nosso projeto, explicamos quem somos e disponibilizamos redes sociais do nosso trabalho.

Nessa tela adicionamos os seguintes elementos:

- Redes sociais (X = antigo Twitter, Discord, Instagram, Facebook e GitHub)

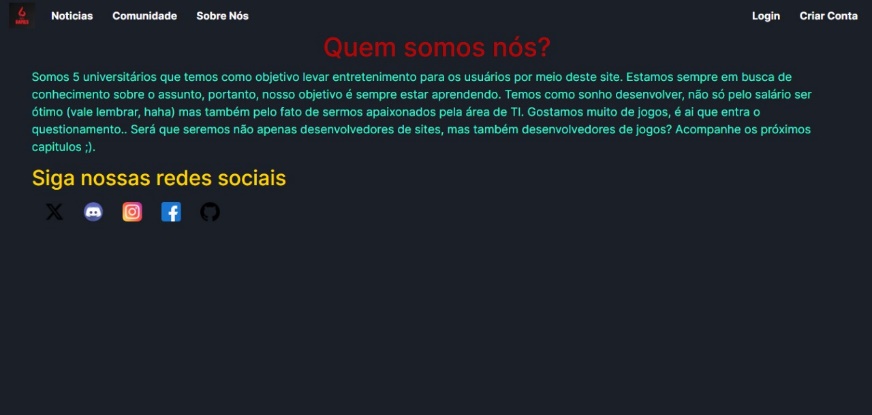


Figura 18: Tela quem somos nós.

1. **Tela Notícias:**

Na tela de notícias “Figura 19” colocamos alguns elementos, são eles:

- Notícias sobre atualizações de jogos

- Informações sobre jogos que estão com campeonatos vigentes

- Informações sobre jogos que estão comemorando +1 ano no mercado.

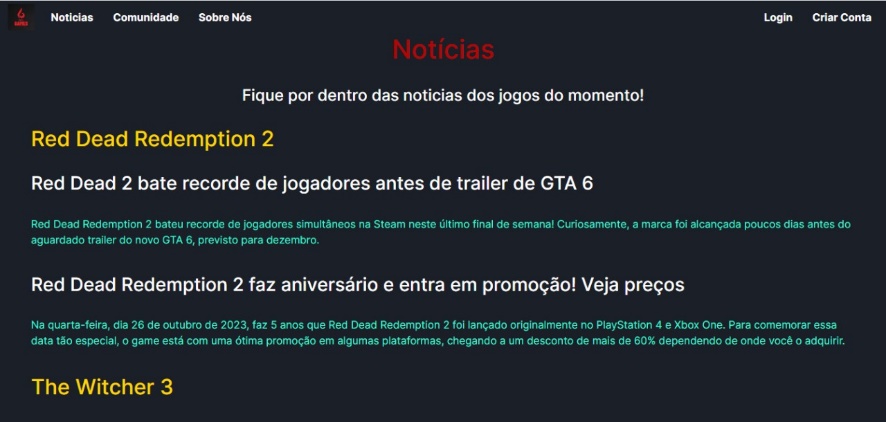


Figura 19: Tela noticias.

1. **Tela Cadastro de Jogos:**

Na tela de jogos, “Figura 20”, já deixamos previamente 10 jogos cadastrados para o usuário, onde ele tem a opção de excluir ou editar. E no botão ao lado (Cadastrar) ele tem a possibilidade de “adicionar”, conforme “Figura 21”.

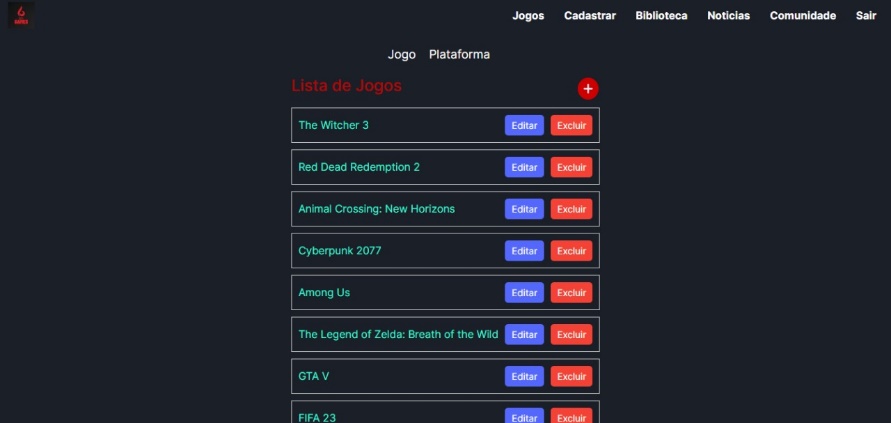


Figura 20: Tela do CRUD.



Figura 21: Tela para adicionar jogos.

1. **API:**

Com a utilização do Node.js para executar o código JavaScript do lado servidor, usamos a API para criação do servidor e gerenciar as rotas. Utilizamos um conjunto de princípios de arquitetura que define um estilo de comunicação entre sistemas distribuídos. Com isso, utilizamos o REST para organizar as rotas de forma lógica e previsível, para permitir acessar, criar, modificar ou deletar recursos. Juntamente com os recursos citados acima utilizamos a biblioteca AXIOS para realizar as requisições HTTP no Node.js e no navegador. Ele também é o responsável por conectar o front – end com o back – end, de forma que, ele funciona no lado do cliente (front) e envia as requisições para o servidor (back) para obter as respostar correspondentes.

Vale ressaltar que as estruturas de rotas são organizadas de acordo com os princípios REST, utilizando métodos HTTP (GET, POST, PATCH, DELETE) para manipularmos os recursos (dados) da API.

Abaixo segue figuras do arquivo do app “Figura 22”, rota game “Figura 23” rota plataforma “Figura 24” e rota user “Figura 25”.



Figura 22: Tela arquivo do app.

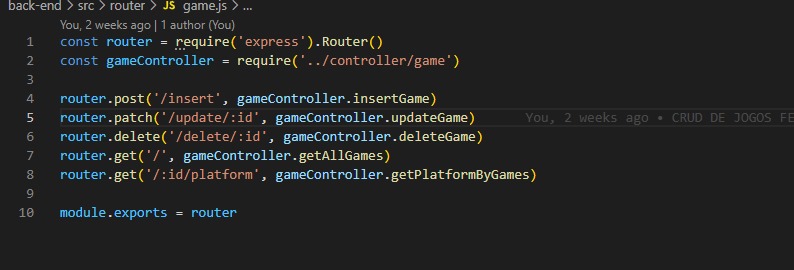


Figura 23: Tela rota game.



Figura 24: Tela rota plataforma.



Figura 25: Tela rota user.