CENTRO ESTADUAL DE EDUCAÇÃO TECNOLÓGICA PAULA SOUZA

Faculdade de Tecnologia Baixada Santista Rubens Lara

Curso Superior de Tecnologia em Sistemas para Internet

Impressio

Dominike Righi
Kauan Coli
Rodrigo Oliveira

Santos, SP
2024

Dominike Righi
Kauan Coli
Rodrigo Oliveira

Impressio

Trabalho de Conclusão de Curso apresentado à Faculdade de Tecnologia Rubens Lara, como exigência para a obtenção do Título de Tecnólogo em Sistemas para Internet.

Orientador: Prof. Diego Neri de Souza Felix

Santos, SP
2024

RESUMO

A arte desempenha um papel fundamental na sociedade, refletindo e influenciando aspectos culturais, sociais e emocionais ao longo da história (Gombrich, 2016). Com o advento da era digital, testemunhamos uma transformação na maneira como interagimos e consumimos obras de arte. Nesse contexto, o Impressio surge como uma resposta à necessidade de sistemas eficientes que possibilitem a interação e o compartilhamento de obras. O projeto visa proporcionar uma plataforma dedicada à catalogação e exposição de obras de arte, oferecendo uma experiência imersiva e colaborativa aos seus usuários. O objetivo é desenvolver um sistema intuitivo e funcional, alinhado às demandas contemporâneas de apreciação artística online. Busca-se não apenas divulgar obras de arte, mas também promover a interação e a colaboração entre os usuários. Para alcançar tais objetivos, serão implementadas funcionalidades de cadastro e login de usuários, busca avançada, categorização e interação social. A metodologia adotada será a de desenvolvimento interativo, permitindo ajustes e melhorias ao longo do processo de implementação. O Impressio tem o potencial de se tornar uma plataforma essencial para a apreciação e disseminação cultural no ambiente digital. Ao oferecer uma experiência imersiva e colaborativa, espera-se que o Impressio contribua significativamente para o enriquecimento do cenário artístico online e para a promoção da interação e compartilhamento de obras de arte entre os usuários.

Palavras-chaves: Arte digital. Plataforma online. Experiência colaborativa.

ABSTRACT

Art plays a fundamental role in society, reflecting and influencing cultural, social and emotional aspects throughout history (Gombrich, 2016). With the advent of the digital age, we have witnessed a transformation in the way we interact and consume works of art. In this context, Impressio emerges as a response to the need for efficient systems that enable interaction and sharing of works. The project aims to provide a platform dedicated to the cataloging and exhibition of works of art, offering an immersive and collaborative experience to its users. The objective is to develop an intuitive and functional system, aligned with contemporary demands for online artistic appreciation. The aim is not only to promote works of art, but also to promote interaction and collaboration between users. To achieve these objectives, user registration and login, advanced search, categorization and social interaction functionalities will be implemented. The methodology adopted will be interactive development, allowing adjustments and improvements throughout the implementation process. Impressio has the potential to become an essential platform for cultural appreciation and dissemination in the digital environment. By offering an immersive and collaborative experience, Impressio is expected to significantly contribute to enriching the online art scene and promoting interaction and sharing of artworks among users.

Keywords: Digital art. Online platform. Collaborative experience.

LISTA DE ABREVIATURAS E SIGLAS

JS – JAVASCRIPT 02
WEB – WORLD WIDE WEB 02
SQL – STRUCTURED QUERY LANGUAGE 02
EMAIL – ELECTRONIC MAIL 05
RF – REQUISITO FUNCIONAL 05
RNF – REQUISITO NÃO FUNCIONAL 06
ID – IDENTITY 09
LINK– HYPERLINK 09
MER – MODELO ENTIDADE-RELACIONAMENTO 10
CSS – CASCADING STYLE SHEETS 10
ORM – OBJECT-RELATIONAL MAPPING 12
JPEG – JOINT PHOTOGRAPHIC EXPERTS GROUP 16
PNG – PORTABLE NETWORK GRAPHICS 16

LISTA DE ILUSTRAÇÕES

ILUSTRAÇÃO 01 – TELA INICIAL ARTSTATION 02
ILUSTRAÇÃO 02 – CURTIR IMAGEM ARTSTATION 03
ILUSTRAÇÃO 03 – ABERTURA DE IMAGEM DEVIANTART 03
ILUSTRAÇÃO 04 – TELA EXPLORAR PINTEREST 04
ILUSTRAÇÃO 05 – CÓDIGO FRONT-END 10
ILUSTRAÇÃO 06 – CÓDIGO FRONT-END PARTE 2 11
ILUSTRAÇÃO 07 – CÓDIGO BACK-END 12
ILUSTRAÇÃO 08 – CÓDIGO BACK-END PARTE 2 13
ILUSTRAÇÃO 09 – TELA INICIAL IMPRESSIO 14
ILUSTRAÇÃO 10 – TELA LOGIN IMPRESSIO 14
ILUSTRAÇÃO 11 – TELA GRÁFICO DE RESPOSTAS DOS USUÁRIOS 17
ILUSTRAÇÃO 12 – TELA GRÁFICO DE RESPOSTAS DOS USUÁRIOS PARTE 2 17

SUMÁRIO

1 INTRODUÇÃO 1
1.1 OBJETIVO 1
1.1.1 OBJETIVO GERAL 1
1.1.2 OBJETIVOS ESPECÍFICOS 2
1.2 ESTADO DA ARTE 2
2 DESENVOLVIMENTO 5
2.1 ANÁLISE DO SISTEMA 5
2.1.1 ANÁLISE DE REQUISITOS 5
2.1.2 DIAGRAMA DE CASO DE USO 6
2.1.3 FLUXO DE EVENTOS 7
2.2 BANCO DE DADOS 9
2.3 CAMADA DE NEGÓCIO 10
2.4 CAMADA DE APRESENTAÇÃO 13
3 RESULTADO 16
3.1 TESTES DE DESENVOLVIMENTO 16
3.2 TESTES FINAIS COM USUÁRIOS 16
3.3 CONCLUSÃO 16
REFERÊNCIAS BIBLIOGRÁFICAS 19
APÊNDICE A 20
APÊNDICE B 21

1 INTRODUÇÃO

No contexto da arte, suas diversas formas e manifestações desempenham um papel crucial na sociedade, refletindo e influenciando aspectos culturais, sociais e emocionais ao longo da história, como exposto em *A História da Arte* de E.H. Gombrich (2016). Com o advento da era digital, observamos uma transformação significativa na maneira como interagimos e consumimos obras de arte. Em *Software Takes Command*, Lev Manovich (2013) explora como plataformas *online* se tornaram espaços essenciais para apreciação e disseminação cultural, transformando a relação entre arte e tecnologia.

Além disso, *The Wealth of Networks: How Social Production Transforms Markets and Freedom*, de Yochai Benkler (2006), destaca a importância de sistemas eficientes que possibilitem não apenas a organização, mas também a interação e o compartilhamento de obras de arte de forma acessível e dinâmica. Com relação à interação e colaboração nas plataformas digitais, Castells (2017) em *A Sociedade em Rede* discute o impacto das redes na criação de uma cultura colaborativa, que analisa o papel das plataformas na construção da colaboração cultural. Nesse cenário, o projeto *Impressio* emerge como uma plataforma dedicada à catalogação e exposição de obras de arte, oferecendo uma experiência imersiva e colaborativa aos seus usuários.

1.1 OBJETIVO

Este capítulo apresenta a base conceitual e técnica do projeto, destacando as metas a serem alcançadas e as estratégias utilizadas para o desenvolvimento. Portanto, o conteúdo está organizado com uma estrutura busca conectar a visão geral do sistema às etapas práticas de sua realização, garantindo que o desenvolvimento técnico e funcional esteja alinhado com os objetivos propostos.

1.1.1 OBJETIVO GERAL

Desenvolver o *Impressio* como um sistema intuitivo, alinhado às demandas contemporâneas de apreciação artística online, proporcionando um ambiente virtual que centralize, organize, promova a interação e a colaboração entre os usuários.

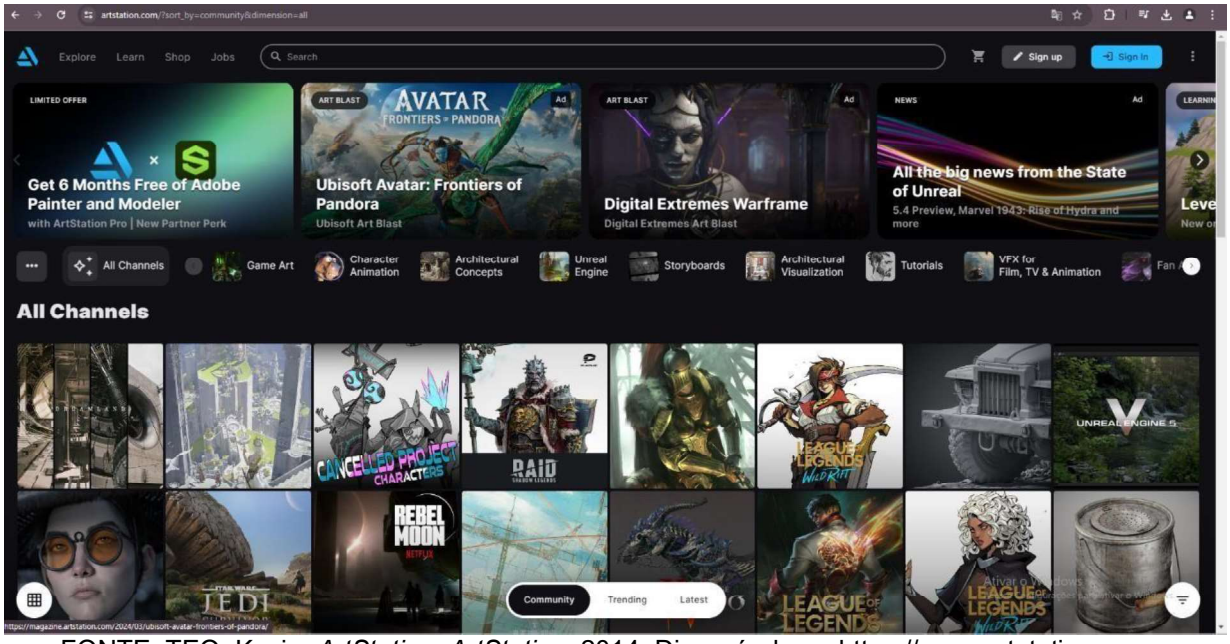
1.1.2 OBJETIVOS ESPECÍFICOS

Para implementação do projeto, será usado a biblioteca *React.js* para criação das funcionalidades de cadastro e *login* de usuários, devido à sua capacidade de construir interfaces interativas e responsivas, como abordado em *Desenvolvimento Web Moderno com React e Redux*, por Ojeda e Cáceres (2021), para uma experiência de usuário fluida e envolvente, garantindo uma experiência de usuário satisfatória. Para integrar recursos sociais como *upload* de imagens, avaliações, validar cadastro de usuários, visando promover interação e compartilhamento de percepções sobre as obras de arte. Para isso, usaremos *C#* para o *back-end* e para armazenamento de todas as informações de usuários e publicações será usado o banco de dados *PostgreSQL*, escolhido por sua robustez e confiabilidade em sistemas de grande escala, conforme analisado em *The Design of Postgres* (1986).

1.2 ESTADO DA ARTE

Para o desenvolvimento do *Impressio*, foi realizada uma revisão de outras plataformas com funcionalidades similares. No caso do *ArtStation*, criado por Leonard Teo em 2014, analisamos características como a disposição das obras na tela inicial:

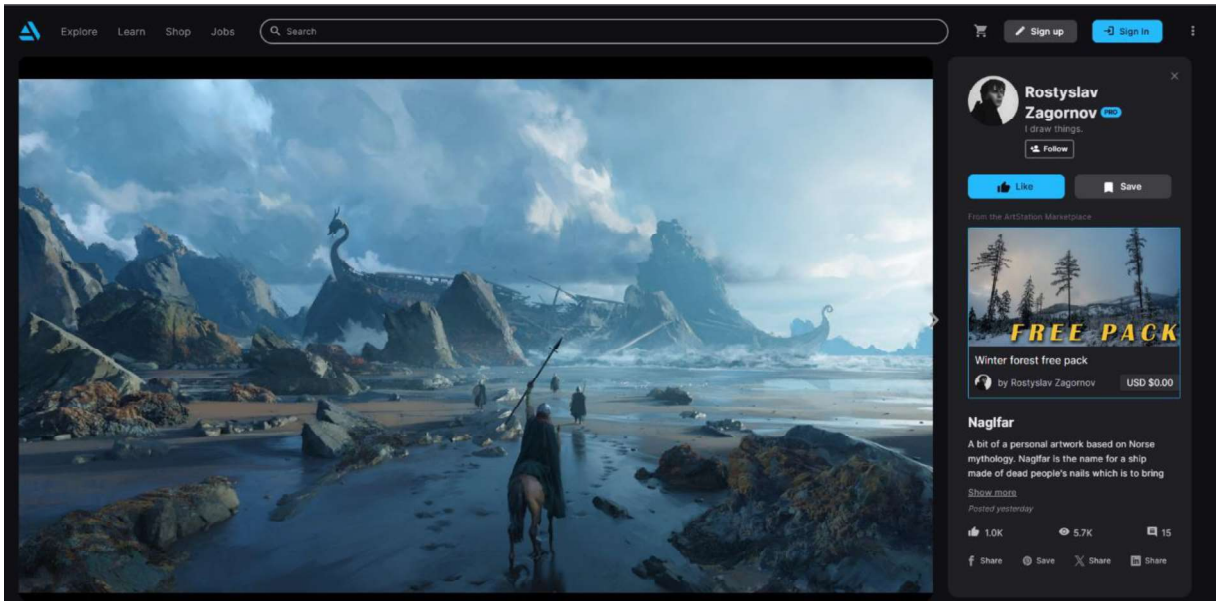
ILUSTRAÇÃO 01 – Tela Inicial *Artstation*



FONTE: TEO, Kevin. *ArtStation*. *ArtStation*, 2014. Disponível em: https://www.artstation.com.

Outro ponto analisado é o acesso as imagens, onde é possível avaliar a imagem escolhida com uma curtida.

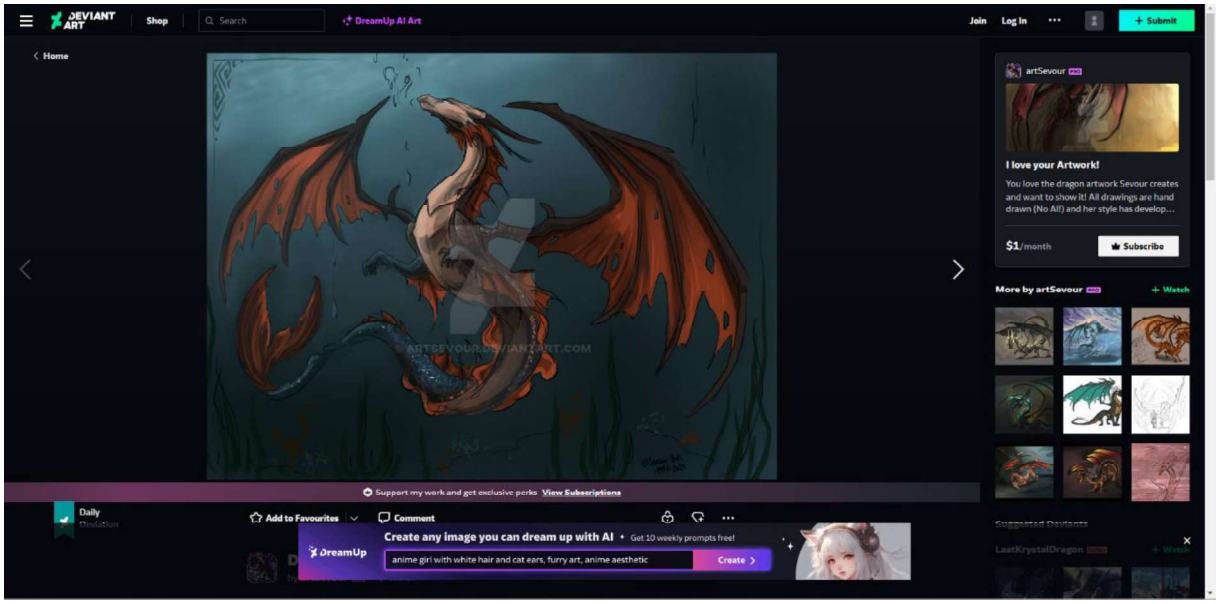
ILUSTRAÇÃO 02 – Curtir imagem *Artstation*



FONTE: TEO, Kevin. ArtStation. ArtStation, 2014. Disponível em: https://www.artstation.com.

No *DeviantArt*, foi analisada a abertura de imagem, onde a escolhida é ampliada e exibe informações da obra.

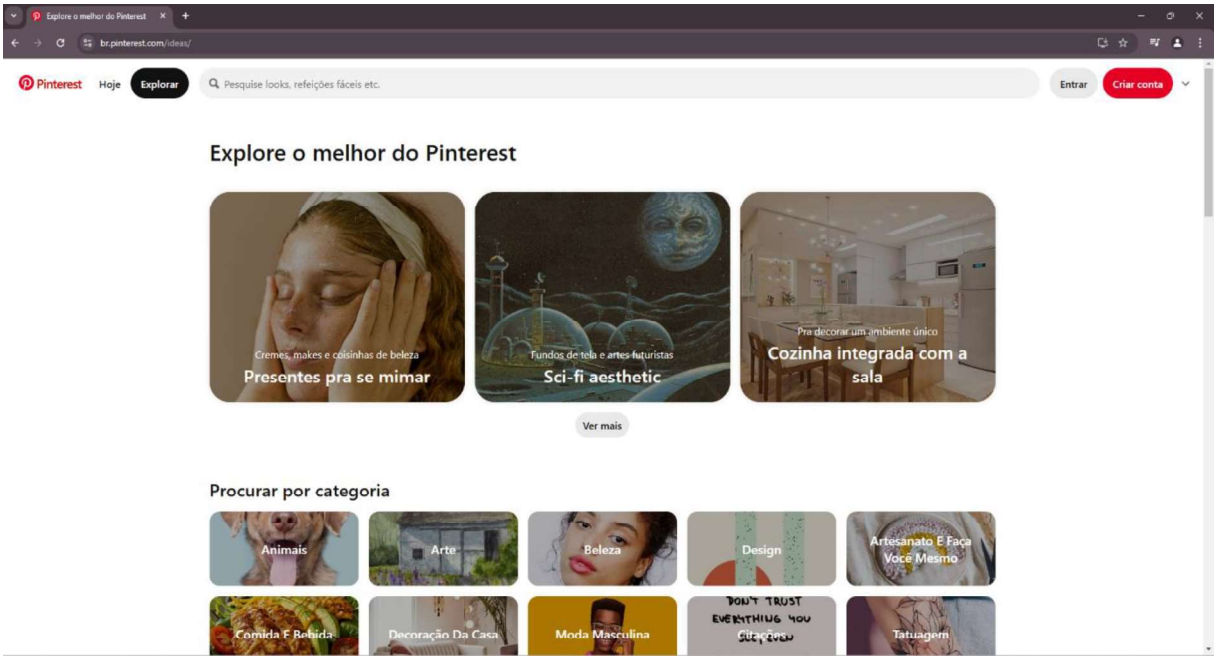
ILUSTRAÇÃO 03 – Abertura de imagem *DeviantArt*



FONTE: OTIS, Angelo; DARREN, Eric. *DeviantArt. DeviantArt*, 2000. Disponível em: https://www.deviantart.com.

No *Pinterest*, foi analisado a ferramenta de explorar outras obras, onde cataloga diferentes tipos de imagens para o usuário selecionar:

ILUSTRAÇÃO 04 – Tela Explorar *Pinterest*



FONTE: SILBERMANN, Ben; SHARP, Evan; SCIARRA, Paul. *Pinterest. Pinterest*, 2010. Disponível em: https://www.pinterest.com.

2 DESENVOLVIMENTO

O desenvolvimento do sistema *Impressio* foi conduzido com base em uma série de análises em outras aplicações com propósitos similares, como foi citado no estado da arte. Além disso, foram feitas definições técnicas, organizadas em etapas claras para garantir a funcionalidade e a experiência de uso esperada. O processo foi dividido em diversas fases abordadas a seguir, que incluem a análise do sistema, a definição dos requisitos, o desenho do banco de dados, a implementação da camada de negócio e da camada de apresentação. Essas fases foram fundamentais para a construção do sistema como um todo, desde a modelagem até a implementação.

2.1 ANÁLISE DO SISTEMA

De acordo com o livro *The Unified Modeling Language User Guide* de Grady Booch, James Rumbaugh e Ivar Jacobson (1998), a *Unified Modeling Language* ajuda a visualizar a estrutura e o comportamento de um sistema, facilitando a comunicação entre os membros da equipe de desenvolvimento e outras partes interessadas. Com isso, dos mecanismos de análise disponíveis para o desenvolvimento do sistema, resolvemos dar prosseguimento usando-a. Seu uso serve para representar o *design* de sistemas de *software* através de diferentes tipos de diagramas, como diagramas de caso de uso, diagramas de classes, diagramas de sequência, entre outros.

2.1.1 ANÁLISE DE REQUISITOS

Os principais requisitos funcionais para o *Impressio* são:

[RF01] Cadastro de Usuários: Permitir que novos usuários se cadastrem com nome, *e-mail* e senha.

[RF02] *Login* de Usuários: Permitir que usuários cadastrados façam *login* utilizando *e-mail* e senha.

[RF03] *Upload* de Obras de Arte: Permitir que usuários façam *upload* de imagens de suas obras de arte.

[RF04] Avaliações de Obras: Permitir que usuários avaliem as obras de arte.

[RF05] Pesquisa de Obras: Permitir que os usuários pesquisem obras de arte porpalavras-chave.

[RF06] Perfis de Usuários: Permitir que os usuários vejam e editem seus perfis, incluindo suas informações pessoais e suas obras de arte publicadas.

Os requisitos não funcionais incluem aspectos como desempenho, segurança, usabilidade e compatibilidade. Exemplos de requisitos não funcionais são:

[RNF01] Segurança: O sistema deve garantir a segurança das informações dos usuários, utilizando criptografia para senhas e dados sensíveis.

[RNF02] Performance: O sistema deve ser capaz de suportar vários usuários simultâneos sem degradação significativa de desempenho.

[RNF03] Usabilidade: A interface do usuário deve ser responsiva e fácil de usar, garantindo uma experiência satisfatória para todos os tipos de usuários.

[RNF04] Compatibilidade: O sistema deve ser compatível com os principais navegadores.

2.1.2 DIAGRAMA DE CASO DE USO

Os principais casos de uso do *Impressio* incluem: Cadastro de Usuários, *Upload* de Obras de Arte, Comentários em Obras e Pesquisa de Obras.

Cada caso de uso será representado no diagrama com a identificação dos atores envolvidos (usuários, administradores etc.) e as interações que ocorrem entre esses atores e o sistema. O diagrama de caso de uso para o sistema pode ser visualizado no Apêndice A.

2.1.3 FLUXO DE EVENTOS

Esse detalhamento é essencial para compreender completamente como cada funcionalidade deve ser implementada dentro do *Impressio*. Por exemplo:

Caso de Uso: Cadastro de Usuários
Fluxo Principal:

1. O usuário acessa a página de cadastro.
2. O usuário insere suas informações (nome, *e-mail*, senha) e clica no botão de cadastro.
3. O sistema valida os dados inseridos:
    a. Verifica se o *e-mail* já está em uso.
    b. Verifica se todos os campos obrigatórios foram preenchidos.
    c. Verifica se a senha atende aos requisitos de complexidade.
4. Se todos os dados forem válidos, o sistema cria uma nova conta de usuário.
5. O sistema envia um *e-mail* de confirmação para o usuário.
6. O sistema redireciona o usuário para a página de *login* com uma mensagem de confirmação.

Fluxo Alternativo:

E-mail já em uso:

O sistema exibe uma mensagem de erro ao usuário solicitando um *e-mail* diferente.

Campos obrigatórios não preenchidos:

O sistema exibe uma mensagem de erro ao usuário indicando quais campos precisam ser preenchidos.

Senha não atende aos requisitos:

O sistema exibe uma mensagem de erro ao usuário informando os requisitos de complexidade da senha.

Caso de Uso: *Upload* de Obras de Arte
Fluxo Principal:

1. O usuário autenticado acessa a página de *upload* de obras de arte.
2. O usuário insere as informações da obra (título, descrição, imagem) e clica

no botão de *upload*.

3. O sistema valida os dados inseridos:
    a. Verifica se todos os campos obrigatórios foram preenchidos.
    b. Verifica se o arquivo de imagem está em um formato suportado.
4. Se todos os dados forem válidos, o sistema salva a obra de arte no banco de dados.
5. O sistema exibe uma mensagem de sucesso ao usuário e redireciona para a página da obra de arte.

Fluxo Alternativo:

Campos obrigatórios não preenchidos:

O sistema exibe uma mensagem de erro ao usuário indicando quais campos precisam ser preenchidos.

Formato de imagem não suportado:

O sistema exibe uma mensagem de erro ao usuário informando os formatos de imagem suportados.

Caso de Uso: Pesquisa de Obras

Fluxo Principal:

1. O usuário acessa a página principal do *Impressio*.
2. O usuário insere uma palavra-chave no campo de pesquisa e clica no botão de buscar.
3. O sistema busca as obras correspondentes no banco de dados com base na palavra-chave.
4. O sistema exibe as obras encontradas na página.

Fluxo Alternativo:

Nenhuma obra corresponde à pesquisa:

O sistema exibe uma mensagem informando que nenhuma obra foi encontrada com a palavra-chave inserida.

Este detalhamento do fluxo de eventos cobre os principais casos de uso para o projeto, fornecendo uma visão clara de como os usuários interagem com o sistema e como o sistema deve responder a essas interações.

2.2 BANCO DE DADOS

No projeto, as tabelas do banco são definidas por:

T_USUARIO:

Armazena os dados dos usuários cadastrados na plataforma. Cada usuário possui um identificador único (ID_USUARIO), um endereço de e-mail (EMAIL_USUARIO), uma senha (SENHA) para autenticação, e uma data de nascimento (DATA_NASCIMENTO). Além disso, inclui um apelido ou nome de usuário público (APELIDO), o nome completo do usuário (NOME_USUARIO), uma breve biografia (BIOGRAFIA_USUARIO), e uma imagem de perfil (IMAGEM_USUARIO). Há também um campo (PUBLICO) para indicar se o perfil do usuário é público ou privado.

T_OBRA_ARTE:

Contém informações sobre as obras de arte publicadas na plataforma. Cada obra possui um identificador único (ID_OBRA_ARTE) e está associada a um usuário específico (ID_USUARIO), que representa o autor ou o usuário que fez o upload. A tabela armazena o *link* da imagem da obra de arte (IMAGEM_OBRA_ARTE), uma descrição breve sobre a obra (DESCRICAO_OBRA_ARTE), e um campo (PUBLICO) que indica se a obra é pública ou privada.

T_OBRA_FAVORITADA:

Registra as obras de arte que foram marcadas como favoritas pelos usuários. Cada favorito possui um identificador único (ID_OBRA_FAVORITADA) e associa uma obra de arte (ID_OBRA_ARTE) a um usuário (ID_USUARIO). Dessa forma, é possível identificar quais obras foram favoritadas por quais usuários, permitindo uma interação e personalização no sistema.

T_OBRA_VOTO:

Armazena as avaliações que os usuários fazem nas obras de arte, permitindo uma classificação das obras. Cada registro de voto possui um identificador único (ID_OBRA_VOTO) e relaciona um usuário (ID_USUARIO) a uma obra de arte específica (ID_OBRA_ARTE). O campo VOTO indica a avaliação dada pelo usuário, permitindo saber se a obra foi avaliada positivamente ou negativamente.

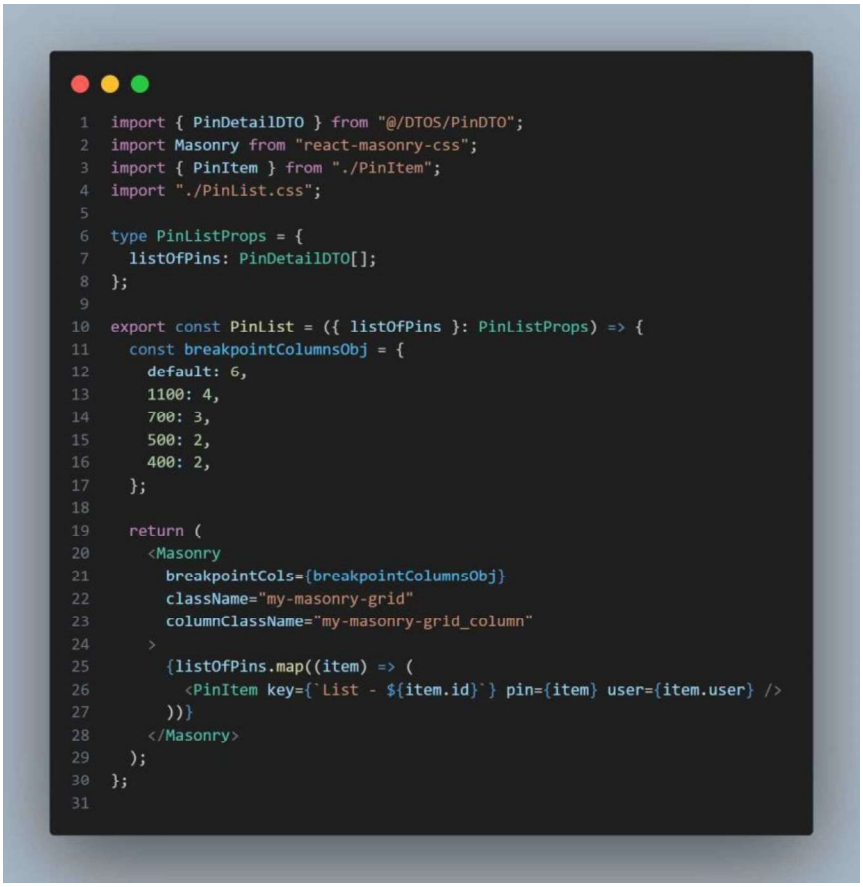
O MER pode ser visualizado no Apêndice B.

2.3 CAMADA DE NEGÓCIO

Para criação da tela Inicial, foi utilizada a estilização "Masonry" com a biblioteca do *React-Masonry-Css* para criação de *layout* líquido e flexível, com as imagens se auto adaptando aos espaços vazios na tela. Foram utilizados *breakpoints* para definir a quantidade de colunas de acordo com o tamanho da tela.

Código abaixo:

ILUSTRAÇÃO 05 – Código *Front-End*

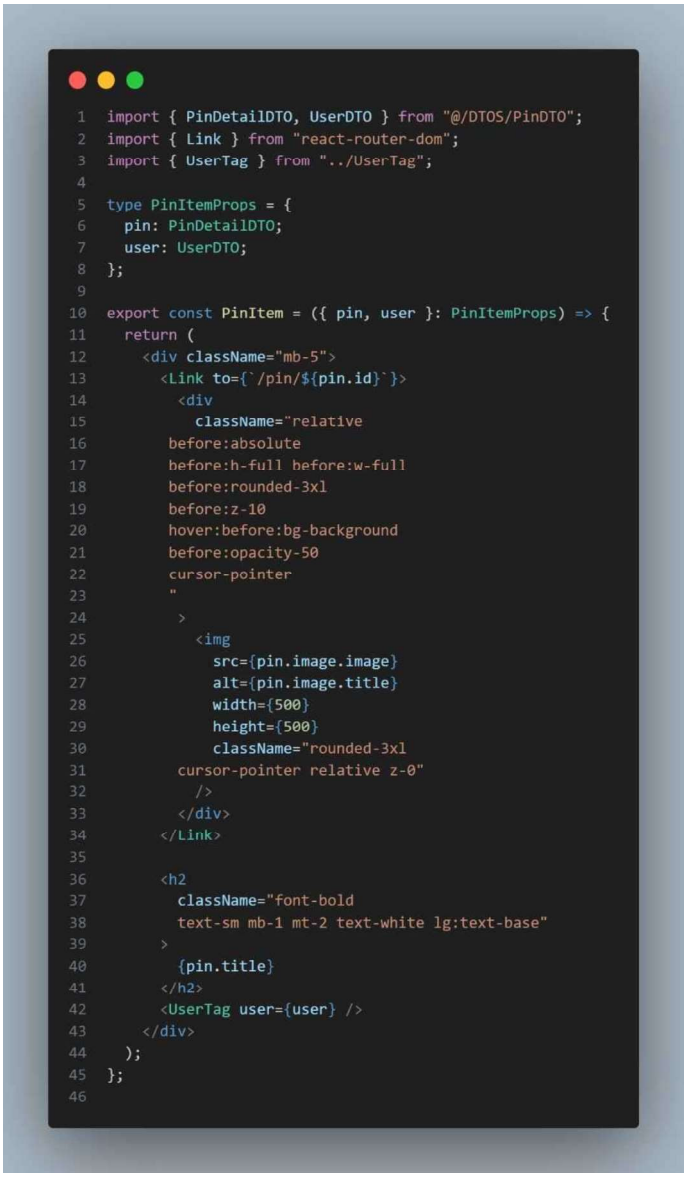


```tsx
import { PinDetailDTO } from "@/DTOS/PinDTO";
import Masonry from "react-masonry-css";
import { PinItem } from "./PinItem";
import "./PinList.css";

type PinListProps = {
  listOfPins: PinDetailDTO[];
};

export const PinList = ({ listOfPins }: PinListProps) => {
  const breakpointColumnsObj = {
    default: 6,
    1100: 4,
    700: 3,
    500: 2,
    400: 2,
  };

  return (
    <Masonry
      breakpointCols={breakpointColumnsObj}
      className="my-masonry-grid"
      columnClassName="my-masonry-grid_column"
    >
      {listOfPins.map((item) => (
        <PinItem key={`List - ${item.id}`} pin={item} user={item.user} />
      ))}
    </Masonry>
  );
};

```

FONTE: Elaboração própria.

Para o código do componente da imagem com título e *tag* de usuário, utilizado *TailwindCSS* para realizar a estilização e responsividade do componente, tornando-o reutilizável em toda a aplicação:

ILUSTRAÇÃO 06 – Código *Front-End* parte 2



```
import { PinDetailDTO, UserDTO } from "@/DTOS/PinDTO";
import { Link } from "react-router-dom";
import { UserTag } from "../UserTag";

type PinItemProps = {
  pin: PinDetailDTO;
  user: UserDTO;
};

export const PinItem = ({ pin, user }: PinItemProps) => {
  return (
    <div className="mb-5">
      <Link to={`/pin/${pin.id}`}>
        <div
          className="relative
       before:absolute
       before:h-full before:w-full
       before:rounded-3xl
       before:z-10
       hover:before:bg-background
       before:opacity-50
       cursor-pointer
       "
        >
          <img
            src={pin.image.image}
            alt={pin.image.title}
            width={500}
            height={500}
            className="rounded-3xl
        cursor-pointer relative z-0"
          />
        </div>
      </Link>

      <h2
        className="font-bold
        text-sm mb-1 mt-2 text-white lg:text-base"
      >
        {pin.title}
      </h2>
      <UserTag user={user} />
    </div>
  );
};
```

FONTE: Elaboração própria.

O resultado pode ser visto na Camada de Apresentação.

A camada de *back-end* do projeto desenvolvida em *.NET Core*, lida com a lógica de negócio e fornece dados ao *front-end*. O arquivo ObterObraArteQuery.cs é responsável por buscar e paginar as obras de arte com filtros dinâmicos, como

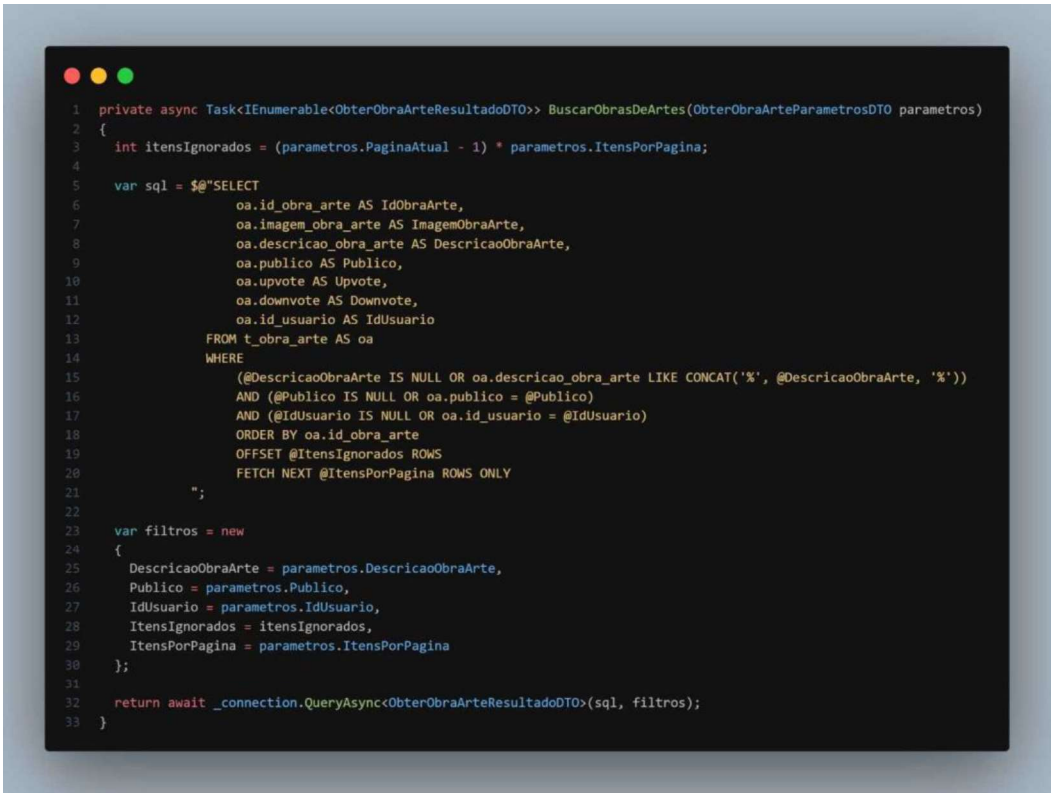
descrição e visibilidade pública. Utilizando o *Dapper*, uma biblioteca leve para mapeamento objeto-relacional (*ORM*), ele executa consultas *SQL* que retornam apenas os dados necessários, melhorando o desempenho da aplicação. Esses dados paginados são então enviados ao *front-end* para serem exibidos de forma otimizada na tela de Exploração, proporcionando uma experiência de navegação rápida e responsiva ao usuário.

ILUSTRAÇÃO 07 – Código *Back-End*

```
public async Task<PaginacaoResposta<ObterObraArteResultadoDTO>> ObterObraDeArte(ObterObraArteParametrosDTO parametros)
{
  var query = await BuscarObrasDeArtes(parametros);
  var paginacao = new PaginacaoResposta<ObterObraArteResultadoDTO>(registros: query);

  if (!parametros.Paginar)
  {
      return paginacao;
  }

  var totalDeItens = query.Count();

  paginacao.PreencherPropriedades(
    totalDeItens: totalDeItens,
    paginaAtual: parametros.PaginaAtual,
    itensPorPagina: parametros.ItensPorPagina
  );

  return paginacao;
}
```

FONTE: Elaboração própria

ILUSTRAÇÃO 08 – Código *Back-End* parte 2



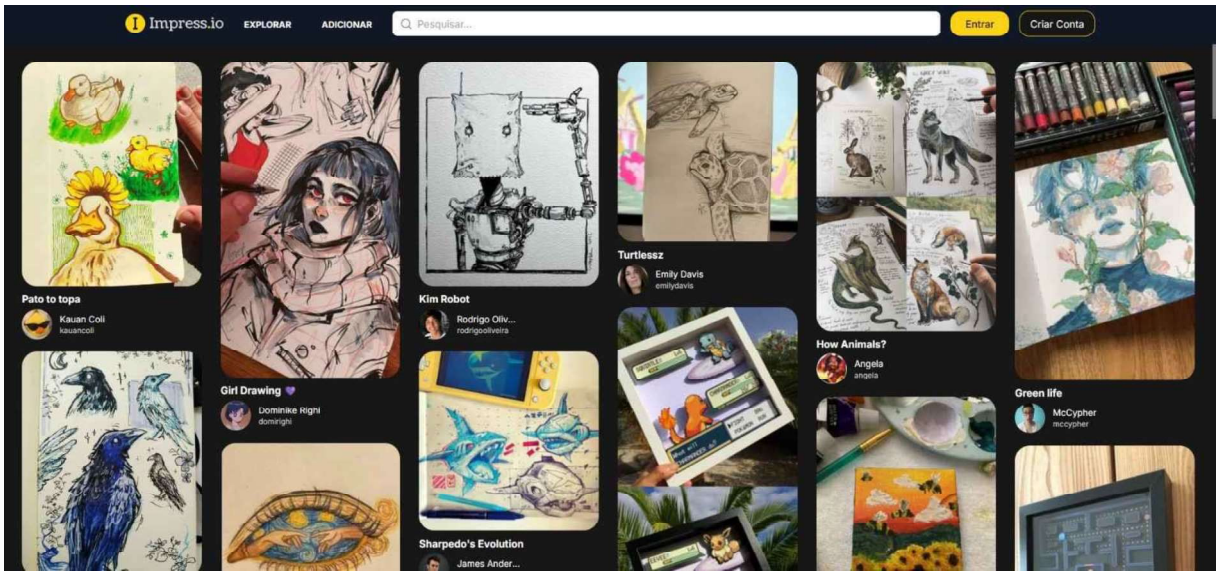
```
private async Task<IEnumerable<ObterObraArteResultadoDTO>> BuscarObrasDeArtes(ObterObraArteParametrosDTO parametros)
{
  int itensIgnorados = (parametros.PaginaAtual - 1) * parametros.ItensPorPagina;

  var sql = $@"SELECT
                oa.id_obra_arte AS IdObraArte,
                oa.imagem_obra_arte AS ImagemObraArte,
                oa.descricao_obra_arte AS DescricaoObraArte,
                oa.publico AS Publico,
                oa.upvote AS Upvote,
                oa.downvote AS Downvote,
                oa.id_usuario AS IdUsuario
            FROM t_obra_arte AS oa
            WHERE
                (@DescricaoObraArte IS NULL OR oa.descricao_obra_arte LIKE CONCAT('%', @DescricaoObraArte, '%'))
                AND (@Publico IS NULL OR oa.publico = @Publico)
                AND (@IdUsuario IS NULL OR oa.id_usuario = @IdUsuario)
                ORDER BY oa.id_obra_arte
                OFFSET @ItensIgnorados ROWS
                FETCH NEXT @ItensPorPagina ROWS ONLY
          ";

  var filtros = new
  {
    DescricaoObraArte = parametros.DescricaoObraArte,
    Publico = parametros.Publico,
    IdUsuario = parametros.IdUsuario,
    ItensIgnorados = itensIgnorados,
    ItensPorPagina = parametros.ItensPorPagina
  };

  return await _connection.QueryAsync<ObterObraArteResultadoDTO>(sql, filtros);
}
```

FONTE: Elaboração própria.

2.4 CAMADA DE APRESENTAÇÃO

A camada de apresentação do *Impressio* é construída utilizando *React.js*, com o objetivo de criar uma interface interativa e responsiva. Foram desenvolvidas diversas telas, cada uma com características específicas para garantir uma navegação fluida e uma experiência de usuário satisfatória.

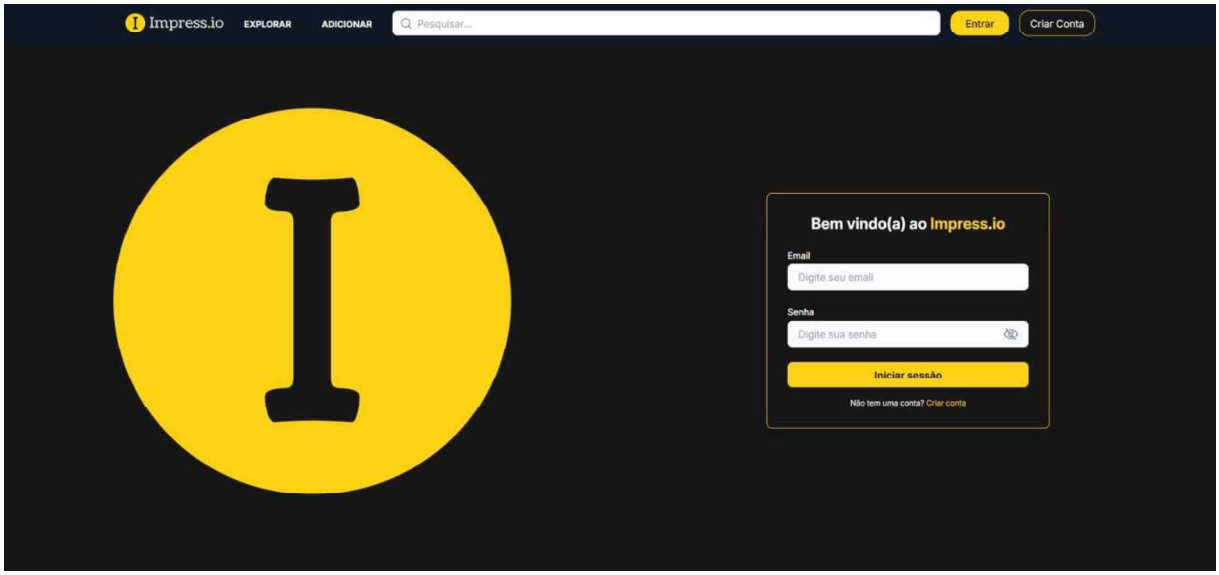
ILUSTRAÇÃO 09 – Tela Inicial *Impressio*



FONTE: Elaboração própria.

Tela Principal: A tela principal exibe as obras de arte em destaque e recentes, permitindo que o usuário explore as publicações. O sistema de busca possibilita a navegação por nome da obra ou do perfil. O *design* dessa tela enfatiza a imagem da obra e facilita a visualização das informações principais, como o nome do artista.

ILUSTRAÇÃO 10 – Tela *Login Impressio*



FONTE: Elaboração própria.

Tela de Cadastro e Login: Essa tela e visualizada ao usuário querer se cadastrar no sistema. Elas apresentam um *layout* simples e claro, com campos de

entrada bem definidos e botões de ação. As validações de campo são realizadas em tempo real, notificando o usuário sobre campos obrigatórios e erros de preenchimento.

3 RESULTADO

Durante a fase de testes do sistema *Impressio*, foram realizados diversos testes de desenvolvimento e testes finais com usuários para validar a conformidade do sistema com os requisitos especificados no início do projeto.

3.1 TESTES DE DESENVOLVIMENTO

Os testes de desenvolvimento focaram nas funcionalidades principais, garantindo que cada módulo estivesse funcionando corretamente antes da integração total. Os testes foram realizados em ciclos iterativos, utilizando técnicas de teste de caixa branca e caixa preta. Abaixo estão os resultados obtidos:

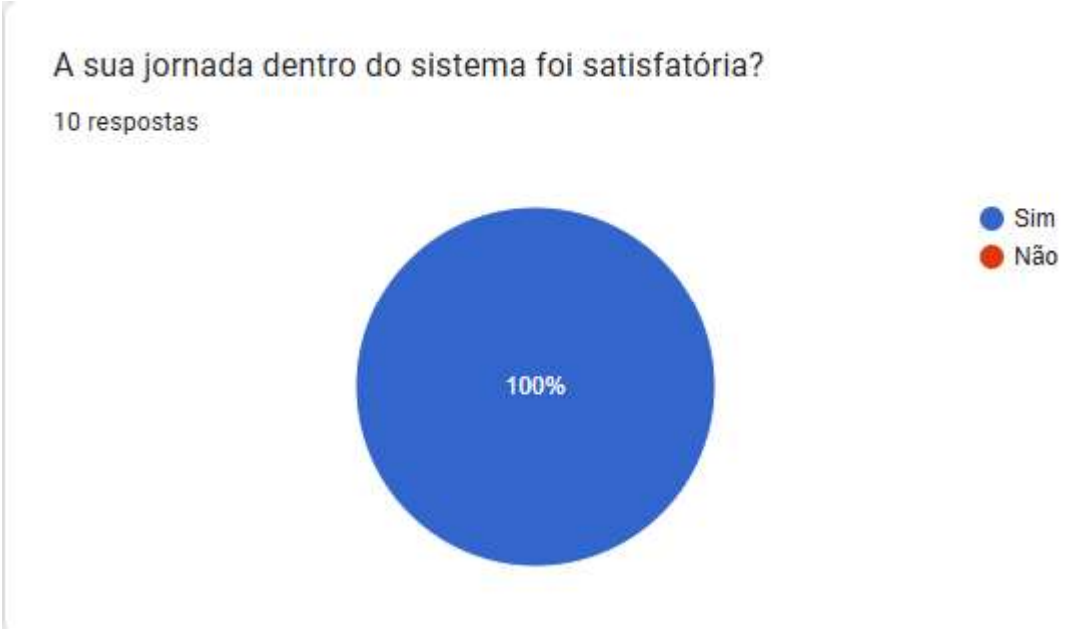
- **Cadastro de Usuários:** Todos os cenários de cadastro foram testados, incluindo *e-mails* duplicados, senhas inválidas e preenchimento incompleto de campos. O sistema respondeu corretamente exibindo as mensagens de erro e confirmando os cadastros válidos.
- **Login de Usuários:** O *login* foi testado com credenciais válidas e inválidas, e o sistema bloqueou acessos indevidos corretamente, garantindo a segurança dos dados dos usuários.
- **Upload de Obras de Arte:** O sistema conseguiu validar e armazenar com sucesso as informações das obras de arte enviadas pelos usuários, mesmo quando testado com diferentes formatos de imagem, como *Joint Photographic Experts Group* (*JPEG*) e *Portable Network Graphics* (*PNG*).
- **Pesquisa:** A pesquisa por palavras-chave apresentou os resultados corretos, retornando obras relacionadas ao termo pesquisado.

3.2 TESTES FINAIS COM USUÁRIOS

Após a fase de testes internos, uma amostra de dez usuários foi selecionada para testar o sistema em um ambiente simulado de uso real. Esses testes finais foram conduzidos para avaliar a usabilidade e a performance do sistema sob condições de uso comum.

O feedback dos usuários foi positivo, relatando que a interface era intuitiva e fácil de navegar, com destaque para a funcionalidade de *upload* de obras, que era simples e direta. O gráfico abaixo mostra a satisfação dos usuários:

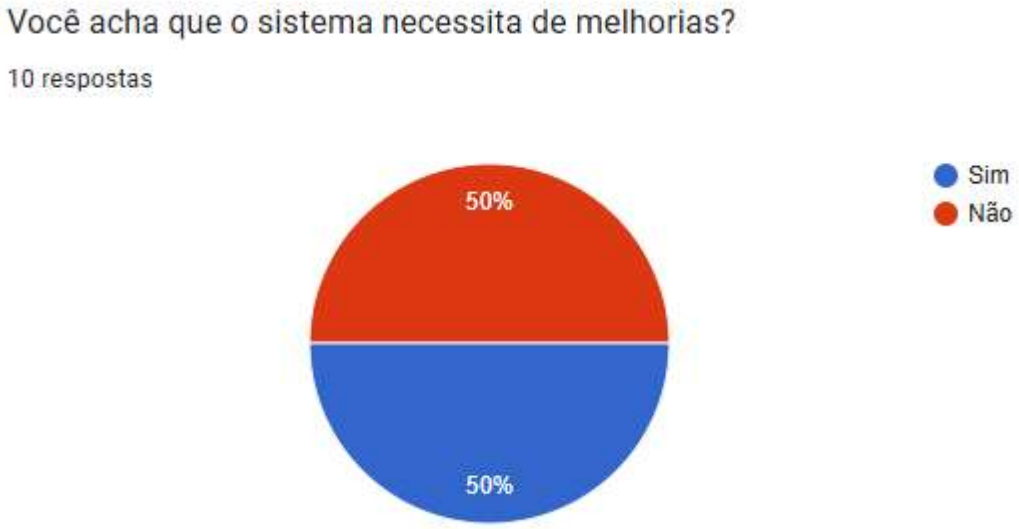
ILUSTRAÇÃO 11 – Gráfico de respostas dos usuários



FONTE: Elaboração própria.

Outro dado importante coletado foi se na opinião dos usuários o sistema necessita de melhorias. Foi obtido o seguinte resultado:

ILUSTRAÇÃO 12 – Gráfico de respostas dos usuários parte 2



FONTE: Elaboração própria.

Diante do resultado, foi questionado as sugestões dos entrevistados e depois discutido a melhor forma de ajustar o projeto. Mudanças foram feitas nas mensagens

de erro, na responsividade e na disposição de alguns botões para melhorar ainda mais a experiência do usuário.

A criptografia de senhas e o uso de tokens de sessão contribuíram para proteger os dados dos usuários, satisfazendo o requisito de segurança (RNF01).

3.3 CONCLUSÃO

A conclusão do desenvolvimento e dos testes do *Impressio* evidencia o sucesso em atender os objetivos propostos. O sistema conseguiu entregar um ambiente virtual colaborativo e intuitivo, alinhado às necessidades contemporâneas de apreciação e interação com obras de arte. A experiência dos usuários foi positiva, destacando-se a navegação fluida, a organização das funcionalidades e a simplicidade no uso das principais ferramentas, como *upload* de obras e pesquisa. Esses resultados reforçam o compromisso do projeto com a funcionalidade, usabilidade e segurança.

Apesar do cumprimento dos objetivos iniciais, algumas oportunidades de melhoria foram identificadas, como a inclusão de um sistema de notificações, uma validação das imagens que estão sendo enviadas, garantindo que o foco da aplicação se mantenha. Esses recursos podem ser explorados em versões futuras, ampliando o valor da plataforma e aprimorando ainda mais a experiência dos usuários.

Em suma, o *Impressio* se posiciona como uma solução promissora para a catalogação e exposição de obras de arte no ambiente digital, contribuindo para uma nova era de interação e apreciação artística *online*.

REFERÊNCIAS

BENKLER, Yochai. The Wealth of Networks: How Social Production Transforms Markets and Freedom. Yale University Press, New Haven, 2006.

BOOCH, Grady; RUMBAUGH, James; JACOBSON, Ivar. The Unified Modeling Language User Guide. Addison-Wesley, Reading, 1998.

CASTELLS, Manuel. A Sociedade em Rede. São Paulo: Paz e Terra, 2017.

GOMBRICH, Ernst. A História da Arte. LTC, Rio de Janeiro, 2016.

MANOVICH, Lev. Software Takes Command. Bloomsbury Academic, New York, 2013.

OJEDA, Victor; CÁCERES, Antonio. Desenvolvimento Web Moderno com React e Redux. Rio de Janeiro, Novatec, 2021.

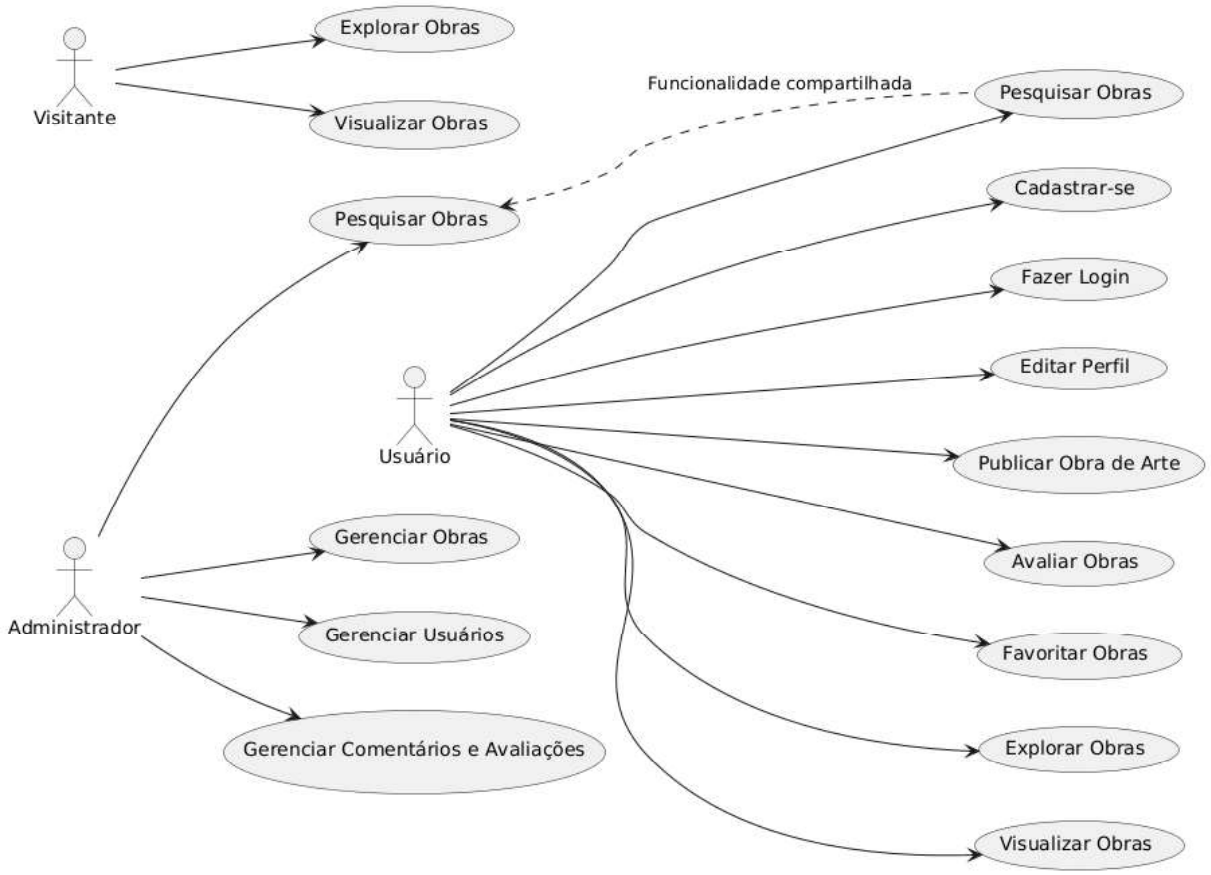
OTIS, Angelo; DARREN, Eric. DeviantArt. DeviantArt, 2000. Disponível em: https://www.deviantart.com.

SILBERMANN, Ben; SHARP, Evan; SCIARRA, Paul. Pinterest. Pinterest, 2010. Disponível em: https://www.pinterest.com.

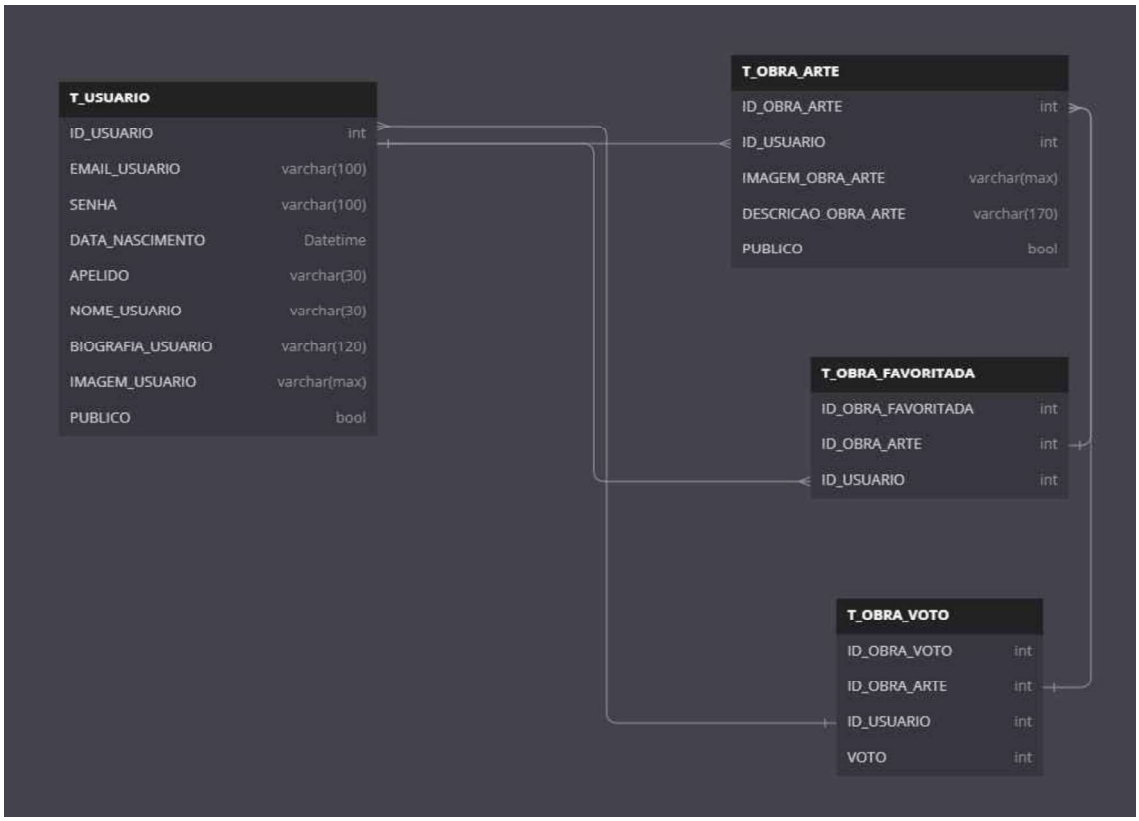
STONEBRAKER Michael; ROWE, Lawrence; HONG, Gary; KELLER, Arthur; LINDSEY Bob; O'NEIL, Elizabeth; ROSENBERG, Anne; WONG, Eugene; HELD, George. The Design of Postgres. Berkeley, University of California, 1986.

TEO, Kevin. ArtStation. ArtStation, 2014. Disponível em: https://www.artstation.com.

APÊNDICE A


Visitante
Explorar Obras
Visualizar Obras
Pesquisar Obras
Funcionalidade compartilhada
Pesquisar Obras
Usuário
Cadastrar-se
Fazer Login
Editar Perfil
Publicar Obra de Arte
Avaliar Obras
Favoritar Obras
Explorar Obras
Visualizar Obras
Administrador
Gerenciar Obras
Gerenciar Usuários
Gerenciar Comentários e Avaliações

APÊNDICE B


T_USUARIO
ID_USUARIO int
EMAIL_USUARIO varchar(100)
SENHA varchar(100)
DATA_NASCIMENTO Datetime
APELIDO varchar(30)
NOME_USUARIO varchar(30)
BIOGRAFIA_USUARIO varchar(120)
IMAGEM_USUARIO varchar(max)
PUBLICO bool
T_OBRA_ARTE
ID_OBRA_ARTE int
ID_USUARIO int
IMAGEM_OBRA_ARTE varchar(max)
DESCRICAO_OBRA_ARTE varchar(170)
PUBLICO bool
T_OBRA_FAVORITADA
ID_OBRA_FAVORITADA int
ID_OBRA_ARTE int
ID_USUARIO int
T_OBRA_VOTO
ID_OBRA_VOTO int
ID_OBRA_ARTE int
ID_USUARIO int
VOTO int