

Game-Hit

Nome: Lucas Oliveira Silva

RA: 082220019

Nome: Vitoria Kaori Kuriyama

RA: 082220005

Disciplina: Modelagem de Software e Linguagem de Programação II

Turma: EC – 6º semestre

Sumário

1. INTRODUÇÃO	4
1.1 Propósito do Documento	4
1.2 Escopo do Projeto	4
1.3 Visão Geral do Documento	4
2. DESCRIÇÃO GERAL.....	6
2.1 Perspectiva do Produto	6
2.2 Funções do Produto	6
2.3 Características dos Usuários	6
2.4 Restrições	7
2.5 Suposições e Dependências	7
3. REQUISITOS ESPECÍFICOS	8
3.1 Requisitos Funcionais	8
3.2 Requisitos Não-Funcionais	10
3.3 Requisitos de Interface	11
3.4 Requisitos de Dados	13
4. MODELAGEM DE PROCESSOS DE NEGÓCIO (BPMN)	14
5. MODELAGEM UML E DIAGRAMAS.....	14
5.1 Modelo de Domínio (diagrama de classes de alto nível)	14
5.2 Diagrama de Casos de Uso	15
5.3 Diagrama de Sequência - Login.....	15
5.4 Diagrama de Classes Detalhado.....	16
6. PROTÓTIPOS E WIREFRAMES	17
6.1 Wireframes	17
6.2 Story Board	18
6.3 Story Map	18
7. ANÁLISE DE PRIORIDADE DE RECURSOS – MODELO KANO.....	19
8. CASOS DE TESTE FUNCIONAIS.....	20
9. MATRIZES DE RASTREABILIDADE	21
9.1 Matriz de Rastreabilidade: Requisitos vs. Objetivos	21
9.2 Matriz de Rastreabilidade: Requisitos vs. Casos de Teste.....	21
GLOSSÁRIO DE TERMOS E ABREVIACÕES.....	23

REFERÊNCIAS 24

1. INTRODUÇÃO

1.1 Propósito do Documento

Este documento tem como objetivo definir e especificar os requisitos funcionais e não-funcionais do sistema Game-Hit. Ele servirá como base para o desenvolvimento, testes e validação do produto, estabelecendo um acordo entre as partes interessadas sobre o que o sistema deverá fazer.

1.2 Escopo do Projeto

O sistema Game-Hit será uma plataforma web que permitirá os usuários registrarem-se, fazerem login, gerenciar suas credenciais e visualizar informações sobre jogos. O escopo inclui:

- Cadastro de usuários com autenticação via login e senha.
- Diferenciação de permissões entre administrador e usuários normais.
- Interface para visualização e gerenciamento de jogos.

O escopo não inclui:

- Sistema de amizade/funcionalidades sociais entre usuários.
- Integração com redes sociais ou outras plataformas externas.
- Sistema de playlists.

1.3 Visão Geral do Documento

Este documento está organizado da seguinte forma:

- **Seção 1 – Introdução:** Apresenta o propósito, escopo e estrutura geral do documento.
- **Seção 2 – Descrição Geral:** Descreve a perspectiva do sistema, suas funções principais, perfis de usuário, restrições técnicas e premissas.
- **Seção 3 – Requisitos Específicos:** Detalha os requisitos funcionais, não-funcionais, de interface e de dados em um formato estruturado, incluindo critérios de aceitação e dependências.
- **Seção 4 – Modelagem de Processos de Negócio:** Contém o diagrama BPMN colaborativo, representando o fluxo de cadastro de usuário.

- **Seção 5 – Modelagem UML:** Apresenta os diagramas de domínio, casos de uso, sequência (login) e classes detalhadas com atributos e relacionamentos.
- **Seção 6 – Protótipos e Wireframes:** Inclui telas principais do sistema, storyboard ilustrando a navegação e story map com organização funcional e priorização.
- **Seção 7 – Análise de Prioridade de Recursos:** Aplicação do modelo Kano para três funcionalidades-chave, categorizando-as conforme o impacto na satisfação do usuário.
- **Seção 8 – Casos de Teste Funcionais:** Contém cinco testes principais documentados com entrada, procedimento, resultado esperado e critérios de aceitação.
- **Seção 9 – Matrizes de Rastreabilidade:** Mapeia requisitos aos objetivos do sistema e aos casos de teste, assegurando rastreabilidade e cobertura funcional.
- **Seção Final – Glossário e Referências:** Compilação dos termos técnicos e fontes utilizadas no desenvolvimento e documentação do sistema.

2. DESCRIÇÃO GERAL

2.1 Perspectiva do Produto

O Game-Hit é uma aplicação web autônoma projetada para permitir que usuários explorem, gerenciem e acompanhem informações sobre jogos eletrônicos. A solução foi idealizada para promover uma navegação fluida e segura, com ênfase na experiência do usuário final e controle administrativo eficiente.

O sistema é independente, ou seja, não depende de integrações com sistemas externos para suas funcionalidades principais. Ele adota a arquitetura MVC (Model-View-Controller) com separação de camadas, utilizando tecnologia Java (Spring Boot) no backend e HTML/CSS/JavaScript no frontend.

2.2 Funções do Produto

As principais funcionalidades do sistema estão organizadas por perfil de acesso:

Para Usuários Comuns:

- Realizar cadastro e autenticação segura.
- Navegar por uma lista de jogos cadastrados, com filtros por nome e gênero.
- Visualizar detalhes de cada jogo.
- Registrar progresso (ex: jogando, concluído) e fazer anotações pessoais.

Para Administradores:

- Gerenciar todos os registros de jogos: criar, editar e excluir.
- Acessar e editar informações de usuários cadastrados.
- Manter controle sobre a base de dados e integridade das informações.

2.3 Características dos Usuários

O sistema será utilizado pelos seguintes tipos de usuários:

- **Usuário comum:** acessa as funcionalidades básicas (navegação, visualização, progresso).

- **Administrador:** possui permissões avançadas para modificar registros, jogos e usuários.

Os usuários são esperados ter familiaridade básica com navegação web. Não é necessário conhecimento técnico para utilizar a aplicação.

2.4 Restrições

- O sistema deve ser acessível apenas via navegadores modernos e atualizados.
- Deve funcionar corretamente em resoluções de desktop e dispositivos móveis (responsividade).
- As senhas dos usuários devem ser armazenadas com algoritmos de hashing seguros, como BCrypt.
- A arquitetura do sistema deve permitir escalabilidade para novos módulos no futuro.

2.5 Suposições e Dependências

- O sistema supõe que os usuários tenham acesso à internet durante seu uso.
- Depende de um servidor web com suporte a Java (Spring Boot) e banco de dados compatível (ex: H2 ou PostgreSQL).
- O backend deve estar sempre ativo para permitir autenticação e persistência de dados.

3. REQUISITOS ESPECÍFICOS

3.1 Requisitos Funcionais

RF-001

- **Descrição:** O sistema deve permitir que novos usuários se cadastrem fornecendo nome, e-mail e senha.
- **Categoria:** Requisito Funcional
- **Prioridade:** Alta
- **Fonte:** Stakeholder
- **CrITÉrios de Aceitação:**
 - O formulário deve validar o formato do e-mail.
 - O sistema deve impedir cadastro de e-mails duplicados.
 - A senha deve respeitar critérios mínimos de segurança.
 - Após cadastro bem-sucedido, o usuário é redirecionado para a tela de login.
- **Dependências:** Nenhuma

RF-002

- **Descrição:** O sistema deve permitir que usuários façam login utilizando e-mail e senha.
- **Categoria:** Requisito Funcional
- **Prioridade:** Alta
- **Fonte:** Stakeholder
- **CrITÉrios de Aceitação:**
 - O sistema deve autenticar as credenciais fornecidas.
 - Em caso de sucesso, redirecionar o usuário para o painel correspondente.
 - Em caso de falha, exibir mensagem de erro apropriada.
- **Dependências:** RF-001

RF-003

- **Descrição:** O sistema deve permitir que administradores visualizem, criem, editem e excluam informações sobre jogos.
- **Categoria:** Requisito Funcional
- **Prioridade:** Alta
- **Fonte:** Stakeholder
- **Crerérios de Aceitação:**
 - A interface de administração deve conter tabela com jogos e botões de ações.
 - Cada operação deve ser validada e persistida no banco de dados.
 - O sistema deve confirmar cada ação com mensagens de feedback.
- **Dependências:** RF-002

RF-004

- **Descrição:** O sistema deve permitir que usuários visualizem uma lista de jogos com filtros aplicáveis.
- **Categoria:** Requisito Funcional
- **Prioridade:** Alta
- **Fonte:** Documento de especificação
- **Crerérios de Aceitação:**
 - A lista deve mostrar título, gênero e data de lançamento.
 - O filtro por nome e gênero deve funcionar em tempo real.
 - O layout deve ser responsivo.
- **Dependências:** RF-002

RF-005

- **Descrição:** O sistema deve permitir que usuários registrem o progresso dos jogos (ex.: jogando, concluído), notas e comentários.
- **Categoria:** Requisito Funcional
- **Prioridade:** Média
- **Fonte:** Pesquisa com usuários
- **Crerérios de Aceitação:**

- O sistema deve permitir salvar e editar o estado do jogo.
 - O progresso registrado deve ser persistido corretamente e visível no painel do usuário.
- **Dependências:** RF-004

3.2 Requisitos Não-Funcionais

RNF-001

- **Descrição:** O sistema deve ser seguro, protegendo as informações dos usuários.
- **Categoria:** Requisito Não-Funcional (Segurança)
- **Prioridade:** Alta
- **Fonte:** Stakeholder
- **CrITÉrios de Aceitação:**
 - As senhas devem ser armazenadas utilizando algoritmo de hashing (ex.: BCrypt).
 - O sistema deve proteger contra injeção de SQL e outras vulnerabilidades conhecidas.
- **Dependências:** RF-001, RF-002

RNF-002

- **Descrição:** O sistema deve ser responsivo e acessível em dispositivos móveis.
- **Categoria:** Requisito Não-Funcional (Usabilidade)
- **Prioridade:** Alta
- **Fonte:** Boas práticas de desenvolvimento
- **CrITÉrios de Aceitação:**
 - Todos os elementos devem se adaptar corretamente em telas móveis.
 - Não deve haver quebra de layout em dispositivos com resolução reduzida.
- **Dependências:** RF-001, RF-004

3.3 Requisitos de Interface

RI-001

- **Descrição:** O sistema deve fornecer uma interface de cadastro de usuário.
- **Tipo de Interface:** Usuário
- **Prioridade:** Alta
- **Fonte:** Stakeholder / Funcionalidade básica esperada
- **Especificações:**
 - Campo de entrada para nome.
 - Campo de entrada do tipo e-mail com validação.
 - Botão “Cadastrar”.
 - Mensagem de sucesso/erro após tentativa.
- **Dependências:** RF-001

RI-002

- **Descrição:** O sistema deve fornecer uma interface de login.
- **Tipo de Interface:** Usuário
- **Prioridade:** Alta
- **Fonte:** Stakeholder / Requisito essencial de autenticação
- **Especificações:**
 - Campos obrigatórios: e-mail e senha.
 - Botão “Entrar” com envio do formulário.
 - Mensagens de feedback em caso de erro.
- **Dependências:** RF-002

RI-003

- **Descrição:** O sistema deve fornecer uma interface para visualização de jogos.
- **Tipo de Interface:** Usuário
- **Prioridade:** Alta
- **Fonte:** Documento de especificação / funcionalidades esperadas

- **Especificações:**
 - Lista de jogos com dados básicos (título, gênero, data).
 - Filtro por nome e gênero.
 - Design responsivo e intuitivo.
- **Dependências:** RF-004

RI-004

- **Descrição:** O sistema deve fornecer uma interface de gerenciamento de jogos para administradores.
- **Tipo de Interface:** Usuário
- **Prioridade:** Alta
- **Fonte:** Stakeholder
- **Especificações:**
 - Tabela com jogos.
 - Botões para adicionar, editar, excluir.
 - Formulário com campos: título, descrição, gênero, data, imagem.
- **Dependências:** RF-003

RI-005

- **Descrição:** O sistema deve fornecer mensagens de feedback visuais ao usuário (ex.: sucesso, erro, carregamento).
- **Tipo de Interface:** Usuário
- **Prioridade:** Média
- **Fonte:** Boas práticas de UX
- **Especificações:**
 - Exibir mensagens claras após ações como cadastro, login, falhas de validação, etc.
 - Utilizar cores e ícones para reforçar o tipo de mensagem (verde para sucesso, vermelho para erro).
- **Dependências:** RF-001, RF-002, RF-003

3.4 Requisitos de Dados

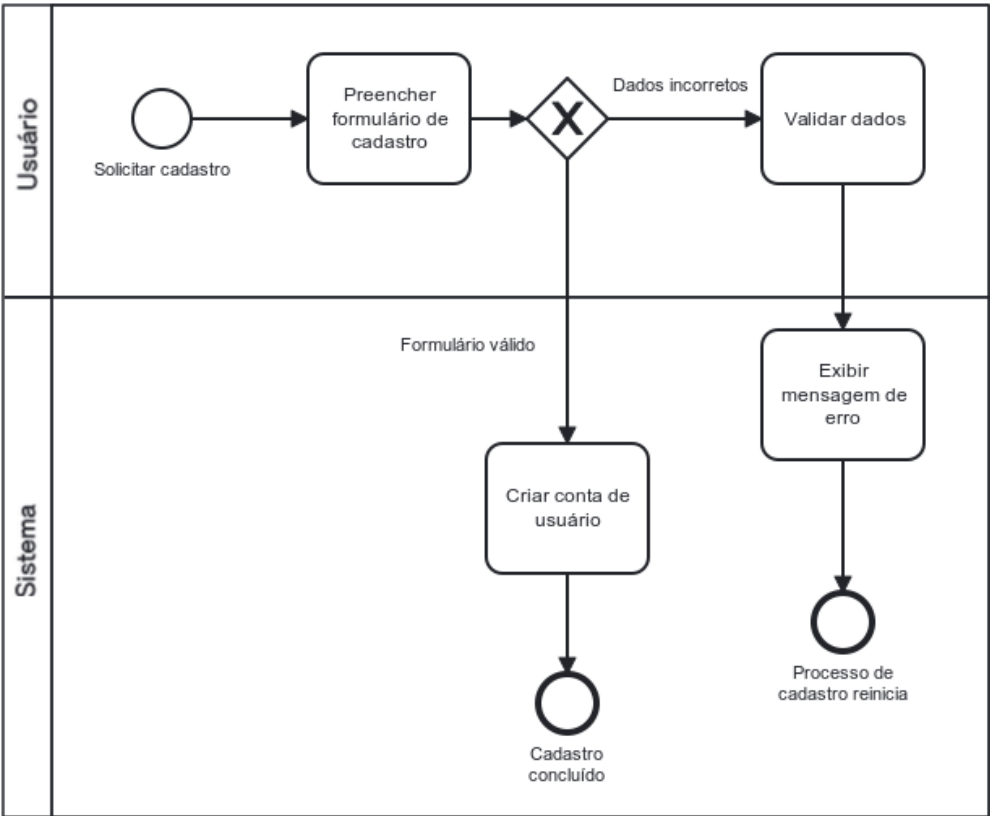
RD-001

- **Descrição:** O sistema deve armazenar os dados dos usuários cadastrados.
- **Prioridade:** Alta
- **Fonte:** Stakeholder
- **Especificações:**
 - **Tipo de dados:** Nome, e-mail, senha (criptografada), tipo de usuário (administrador ou comum)
 - **Volume:** Estimativa inicial de 100–500 usuários
 - **Frequência de acesso:** Alta, principalmente em login e operações de leitura
 - **Requisitos de retenção:** Indeterminado (dados persistem até exclusão solicitada)
- **Dependências:** RF-001, RF-002

RD-002

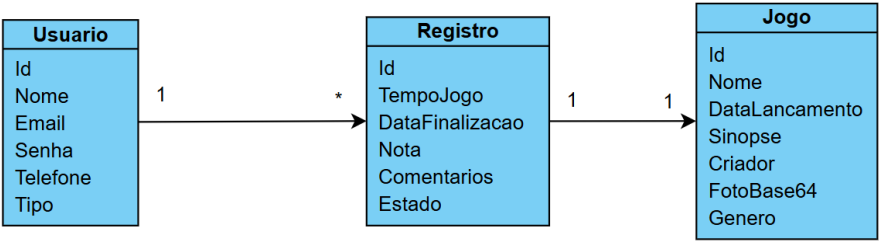
- **Descrição:** O sistema deve armazenar informações sobre jogos cadastrados.
- **Prioridade:** Alta
- **Fonte:** Stakeholder
- **Especificações:**
 - **Tipo de dados:** Título, descrição, gênero, data de lançamento, imagem (opcional)
 - **Volume:** Estimativa inicial de 200–1000 registros
 - **Frequência de acesso:** Alta, especialmente por usuários navegando pelos jogos
 - **Requisitos de retenção:** Permanente até exclusão pelo administrador
- **Dependências:** RF-003

4. MODELAGEM DE PROCESSOS DE NEGÓCIO (BPMN)

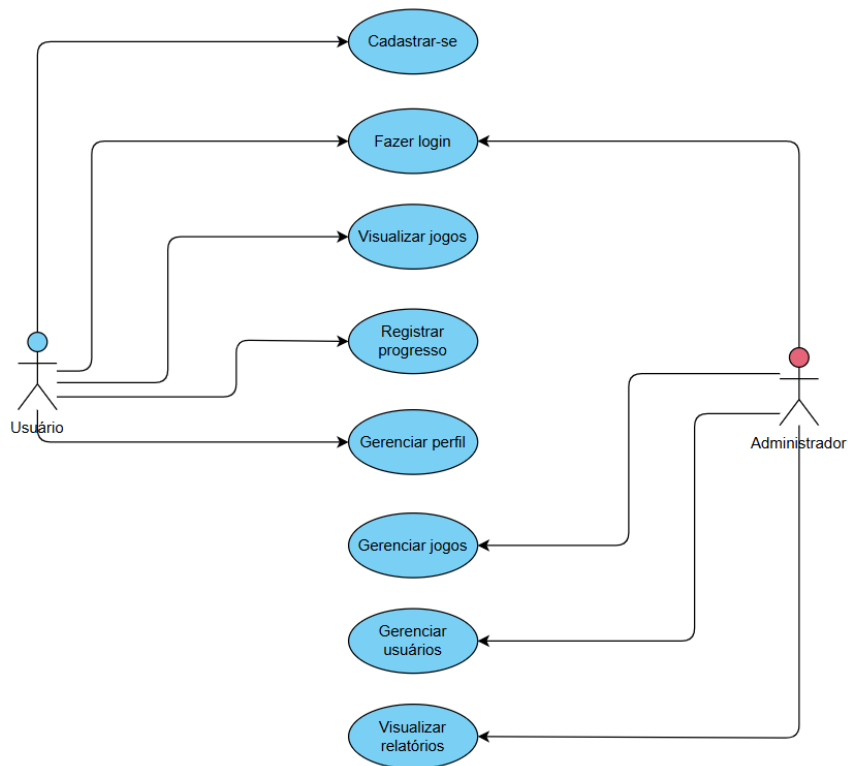


5. MODELAGEM UML E DIAGRAMAS

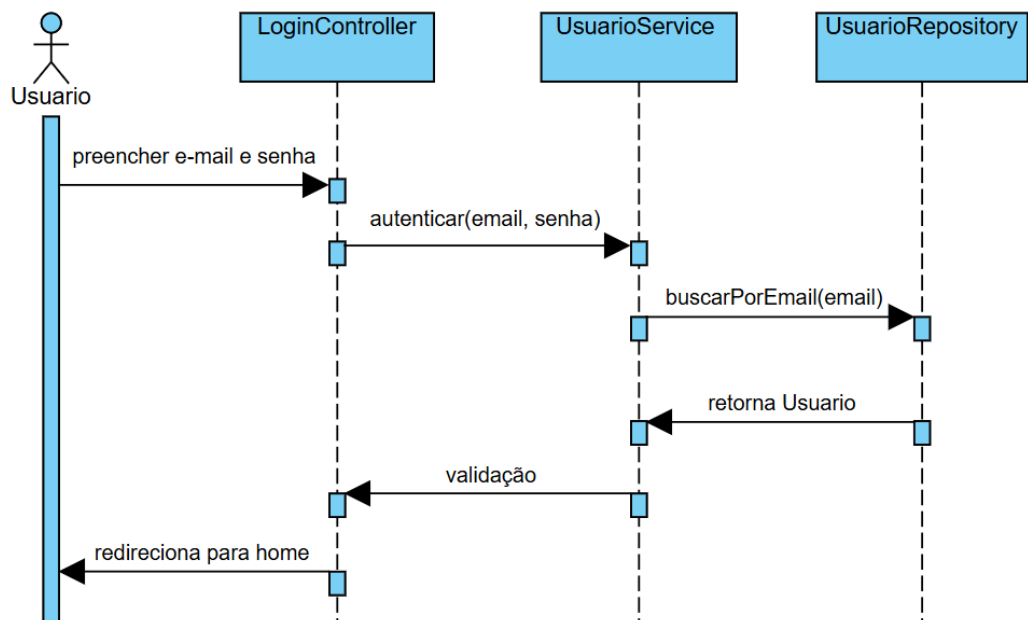
5.1 Modelo de Domínio



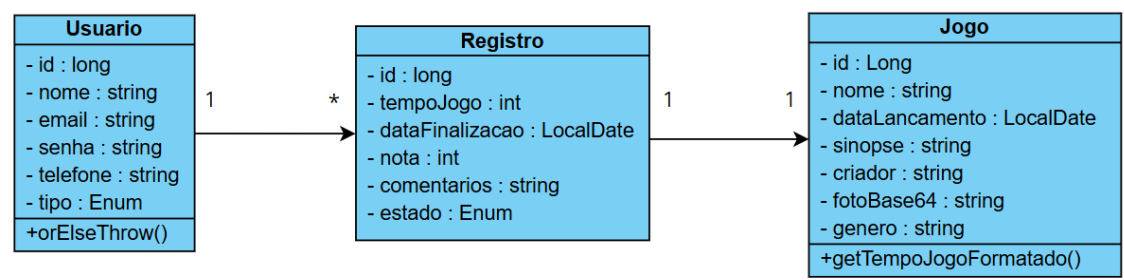
5.2 Diagrama de Casos de Uso



5.3 Diagrama de Sequência - Login

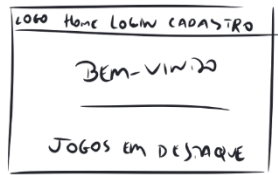


5.4 Diagrama de Classes Detalhado



6. PROTÓTIPOS E WIREFRAMES

6.1 Wireframes



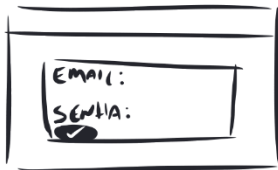
HOME



CADASTROS



LISTAS



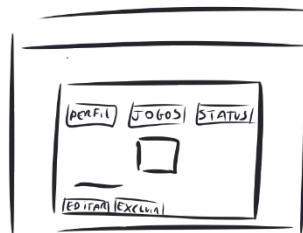
LOGIN



REGISTRO



DASHBOARDS



PERFIL DO
USUARIO

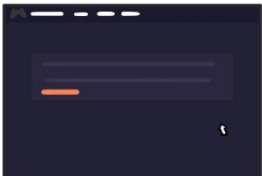
6.2 Story Board



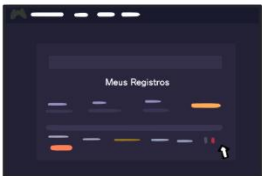
Usuário acessa o site Game-Hit pela primeira vez



Usuário clica em "Cadastrar" e preenche nome, e-mail, telefone e senha



Usuário insere suas credenciais e clica em "Fazer Login"

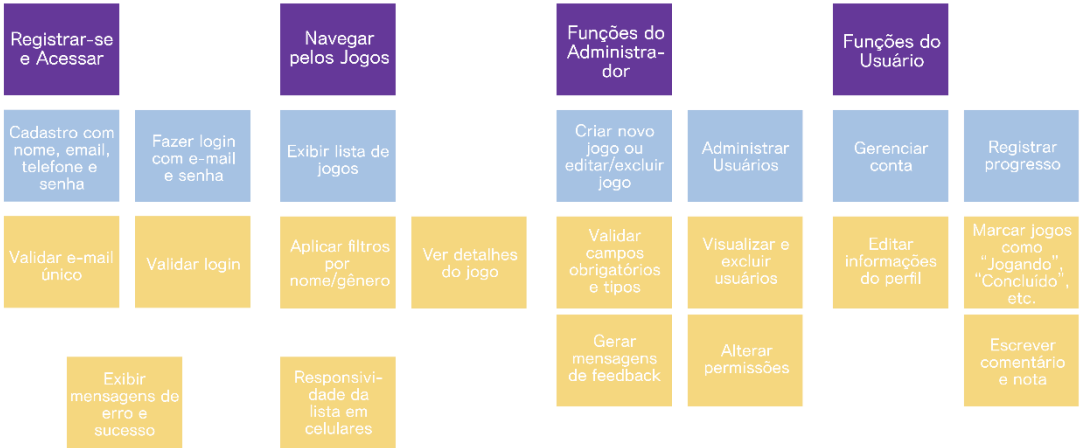


O usuário vê sua lista de jogos onde pode adicionar, editar ou procurar seu registro de jogos



O usuário cadastra seu registro e ele é adicionado para seu perfil

6.3 Story Map



7. ANÁLISE DE PRIORIDADE DE RECURSOS – MODELO KANO

Funcionalidade	Categoria	Justificativa
Cadastro de usuário	Must-Be	Essencial para iniciar uso
Gerenciar jogos (administrador)	Performance	Qualidade e fluidez impactam eficiência
Registro de progresso	Attractive	Interessante, mas não essencial

8. CASOS DE TESTE FUNCIONAIS

ID	Descrição	Procedimento	Resultado Esperado	Critério de Aceitação
CT-001	Cadastro	Nome, e-mail e senha válidos	Cadastro e redirecionamento	E-mail não duplicado
CT-002	Login	E-mail e senha corretos	Acesso ao painel	Redirecionamento correto e sessão iniciada
CT-003	Visualizar jogos	Login como usuário	Lista com filtros funcionando	Interface funcional e dados corretos apresentados
CT-004	CRUD de jogos (administrador)	Informações válidas	Adição, edição e exclusão com sucesso	Persistência das alterações no banco
CT-005	Layout responsivo	Acesso via celular	Interface adaptada sem bugs	Navegação fluida e interface adaptável em diferentes resoluções

9. MATRIZES DE RASTREABILIDADE

9.1 Matriz de Rastreabilidade: Requisitos vs. Objetivos

ID Requisito	Autenticação Segura	Gerenciamento de Jogos	Navegação e Usabilidade	Controle Administrativo
RF-001	X		X	
RF-002	X			
RF-003		X		X
RF-004			X	
RF-005			X	
RNF-001	X			
RNF-002			X	

Legenda dos Objetivos:

- **Autenticação Segura:** Garantir login seguro e proteção de dados.
- **Gerenciamento de Jogos:** Oferecer controle completo sobre o conteúdo.
- **Navegação e Usabilidade:** Garantir boa experiência ao usuário comum.
- **Controle Administrativo:** Permitir ações específicas para administradores.

9.2 Matriz de Rastreabilidade: Requisitos vs. Casos de Teste

ID	CT-001	CT-002	CT-003	CT-004	CT-005
RF-001	X				X
RF-002		X			
RF-003				X	X
RF-004			X		X
RF-005					
RNF-001	X	X			
RNF-002			X	X	X

Legenda:

- **CT-001:** Teste de cadastro de usuário
- **CT-002:** Teste de login
- **CT-003:** Teste de visualização de jogos (usuário comum)
- **CT-004:** Teste de CRUD de jogos (administrador)
- **CT-005:** Teste de layout responsivo

GLOSSÁRIO DE TERMOS E ABREVIATÓES

Termo	Definição
RF	Requisito Funcional
RNF	Requisito Não-Funcional
CRUD	Criar, Ler, Atualizar, Deletar
MVP	Produto Mínimo Viável

REFERÊNCIAS

Spring Boot Documentation – Spring Framework Guides

Disponível em: <https://spring.io/projects/spring-boot>

Oracle – Java SE Documentation

Disponível em: <https://docs.oracle.com/en/java/javase/>

Visual Paradigm Online - Ferramenta para geração de diagramas UML e de sequência

Disponível em: <https://online.visual-paradigm.com/pt/>

BPMN.io - Editor gráfico para diagramas BPMN

Disponível em: <https://bpmn.io>

GitHub – Repositório do Projeto Game-Hit

Disponível em: <https://github.com/Lucas-O-S/Game-Hit>