

# Otaku List

João Vittor Barbosa Andre - 324119383  
Shayene Lorena Meira Figueiredo - 324157119  
João Pedro Cassiano dos Santos - 324127678  
Gabriel Victor Magalhães Costa - 324120093  
Arthur Vitor Machado Bragança - 324143965

**Modelos, métodos e técnicas da engenharia de software**

Belo Horizonte  
2025

# Introdução e Propósito

O objetivo principal deste documento é detalhar os requisitos do sistema **Otaku List**, uma plataforma projetada para resolver a **frustração e a ineficiência** do processo atual de gerenciamento de listas de animes.

O relatório serve como um guia central para a equipe de desenvolvimento alinhar as expectativas entre desenvolvedores, *stakeholders* e o público-alvo. A meta é fornecer uma experiência focada, rápida e visualmente agradável, permitindo ao usuário gerenciar suas listas sem as barreiras encontradas nas ferramentas concorrentes.

## Descrição Geral do Produto

O OtakuList nasceu da necessidade de oferecer uma solução **simples, rápida e agradável de usar**, contrastando com as ferramentas atuais que são frequentemente complicadas, lentas e cheias de bugs.

O produto é uma plataforma de gerenciamento que permite ao usuário organizar facilmente os animes que está assistindo, concluiu ou planeja ver.

### Principais Diferenciais:

- **Foco na Simplicidade:** O sistema é destinado ao público que prefere uma ferramenta que "simplesmente funciona", sem a poluição visual e o excesso de funcionalidades sociais presentes em outras plataformas.
- **Busca e Filtro Avançados:** O sistema possui funcionalidades poderosas de busca e filtro, permitindo refinar resultados por ano de lançamento, plataforma de streaming e combinação de múltiplos gêneros. Além disso, conta com uma busca inteligente capaz de unificar diferentes títulos de uma mesma obra (ex: "Demon Slayer" e "Kimetsu no Yaiba").
- **Gestão Simplificada:** O projeto visa simplificar o processo, permitindo que o usuário adicione e atualize o *status* de seus animes com o mínimo de cliques.

O sistema busca garantir um desempenho rápido e responsivo, atacando diretamente a principal queixa dos usuários (100% dos insatisfeitos) sobre as ferramentas concorrentes: a lentidão e os *bugs*.

# Escopo e Requisitos do Projeto

O escopo define o que está incluído no desenvolvimento do **OtakuNote/Otaku List** e o que foi explicitamente deixado para versões futuras.

## 1. Módulos Principais do Escopo

O projeto foi segmentado nos seguintes módulos principais, focados na organização e descoberta de animes:

- **Cadastro e Gerenciamento de Usuário (Módulo 1):** Permite ao usuário criar conta, editar perfil, configurar preferências e gerenciar sua lista.
- **Consultas e Filtros (Módulo 2):** Permite pesquisar e filtrar animes por título, gênero, ano de lançamento e plataforma de *streaming*.
- **Gerenciamento de Listas (Módulo 3):** Permite adicionar animes às listas personalizadas, classificando-os em status como: Planejo Assistir, Assistindo, Concluído, Pausado ou Dropado.
- **Integração com API AniList (Módulo 4):** O sistema se integra à API AniList para obter informações atualizadas de animes (sinopse, nota, gênero, estúdio, etc.).

## 2. Itens Fora do Escopo

As seguintes funcionalidades e requisitos foram definidos como **fora do escopo inicial** do projeto, priorizando a estabilidade e a experiência de organização central do usuário:

- **Recursos de Rede Social:** Não há desenvolvimento de fóruns, comentários, curtidas ou interações diretas entre usuários.
- **Reprodução de Mídia:** Não há implementação de funcionalidades de *download* ou reprodução direta de animes dentro da plataforma.
- **Aplicativo Nativo:** A primeira versão está restrita ao uso em **navegador web**, não incluindo desenvolvimento de aplicativo nativo para Android/iOS.
- **Gamificação:** Sistemas de medalhas, conquistas ou outros elementos de gamificação não estão no ciclo inicial de entregas.
- **Outros Idiomas:** O sistema é disponibilizado apenas em português na versão inicial.

## 3. Requisitos Funcionais (RFs) Implementados

Os principais requisitos funcionais atendidos no desenvolvimento incluem:

- **Acesso e Segurança:** Permitir **login, logout e cadastro de conta** (RF 01, RF 02).
- **Gestão de Lista:** Habilidade de **adicionar, atualizar o status e excluir** animes da lista do usuário (RF 04, RF 05, RF 07).
- **Filtros e Descoberta:** Disponibilizar filtros avançados para pesquisar animes por **gênero, ano de lançamento e plataforma de streaming** (RF 08).
- **Estatísticas:** Apresentar **estatísticas pessoais** ao usuário, como quantidade de animes assistidos e gêneros mais vistos (RF 09).

- **Usabilidade:** O sistema deve ter uma **interface limpa e intuitiva**, priorizando as informações essenciais (RF 12)

#### **4. Requisitos Não Funcionais (RNFs) e Padrões de Qualidade**

Estes requisitos definem os parâmetros de funcionamento e a qualidade da experiência do usuário:

- **Segurança:** Descreve os requisitos relacionados à integridade e privacidade dos dados, o tratamento de informações confidenciais e a gestão do acesso dos usuários ao sistema.
- **Performance:** Detalha o tempo de resposta do sistema durante a utilização de seus recursos.
- **Usabilidade:** Foca nos requisitos associados à facilidade de uso do sistema.
- **Confiabilidade:** Define os requisitos não funcionais relacionados à frequência de falhas e à robustez do sistema na recuperação dessas falhas.
- **Padrões:** Especifica os padrões e normas a serem seguidos durante o desenvolvimento do sistema.
- **Hardware e Software:** Descreve os componentes de hardware e software que serão empregados pelo sistema.
- **Outros:** Consulte a tabela de requisitos não funcionais presente no material didático do SOL para requisitos adicionais.

# Detalhes de Arquitetura e Tecnologia (Dev)

Esta seção detalha a estrutura técnica e as tecnologias utilizadas no desenvolvimento do OtakuList, conforme especificado no Documento de Requisitos.

## 1. Estrutura e Padrões

O sistema foi desenvolvido seguindo boas práticas de arquitetura, focando em manutenção e escalabilidade.

- **Arquitetura:** Adotamos o padrão arquitetural em camadas, implementando o uso de API RESTful para comunicação entre *frontend* e *backend*.
- **Segurança:** A autenticação é restrita por login e senha, com o uso de **JWT (JSON Web Token)** para segurança das sessões e garantia de acesso apenas por usuários autenticados. As senhas são armazenadas de forma criptografada.
- **Design de Interface (UI/UX):** Foi adotado um design **minimalista e responsivo** para garantir fluidez, facilidade de uso e adaptação a diferentes dispositivos (desktop, tablet, smartphone), atacando a sobrecarga visual das plataformas concorrentes.

## 2. Tecnologias Utilizadas

O projeto utilizou uma *stack* moderna e eficiente:

- **Backend:** Python.
- **Frontend:** HTML, CSS, JavaScript e React (seguindo o padrão SPA - Single Page Application).
- **Banco de Dados:** MySQL.

## 3. Diagramas (Visualização da Estrutura)

Os diagramas abaixo ilustram a estrutura do sistema e as interações entre seus componentes:

### Diagrama de Casos de Uso

Este diagrama mostra as funcionalidades disponíveis e os atores envolvidos:

- **Atores:** Usuário, Administrador e API AniList.
- **Casos de Uso Principais:** Pesquisar Anime, Gerenciar Conta, Visualizar Detalhes do Anime e Gerenciar Listas/Favoritos.

### Diagrama de Classes

Define as entidades e seus relacionamentos no sistema, como a conexão entre **Usuário**, **Anime**, **Avaliação** e a **API AniList** para obtenção de dados atualizados.

## Histórias de Usuário

As Histórias de Usuário representam os requisitos do sistema a partir da perspectiva do usuário, focando no valor que cada funcionalidade entrega. Elas foram classificadas por Prioridade e Complexidade.

ID	Título	Prioridade / Complexidade	História de Usuário	Critérios de Aceite
VS#01	Cadastrar	2 / 1	Permitir que o <b>Usuário</b> personalize o site, acesse histórico e configurações anteriores	Deve permitir <b>Login/Logout</b> , <b>registro</b> de novo usuário (nome, e-mail, senha) e <b>recuperação de senha</b> .
VS#02	Pesquisa e Descoberta	4 / 4	Permitir ao <b>Usuário</b> pesquisar animes por nome, nome alternativo e filtrar por gênero.	Deve permitir a pesquisa por <b>nome em japonês e inglês</b> , exibir o <b>status de exibição</b> , indicar <b>plataformas legais</b> , e permitir <b>avaliação</b> (1 a 5 estrelas).
VS#03	Criação de Listas	3 / 1	Permitir que o <b>Usuário</b> organize e separe animes em diferentes categorias.	Deve permitir a criação de <b>listas ilimitadas e independentes</b> , com opção de nomear/renomear e a função de <b>compartilhar listas</b> (se implementado).
VS#04	Atualização de Fichas	5 / 3	Dar ao <b>Usuário</b> controle total para adicionar, remover e definir o status de um anime na lista.	Deve permitir atualizar o <b>status</b> ('Assistindo', 'Concluído', 'Planejo Assistir' ou 'Abandonado'), <b>remover</b> um anime, e marcar como " <b>Favorito</b> ".

## **Gerenciamento e Planejamento - Quadro Trello**

Conforme a premissa de que a metodologia de desenvolvimento ágil foi utilizada para este projeto, o link abaixo fornece acesso ao quadro de gerenciamento de tarefas:

Link do Quadro Trello: <https://trello.com/b/ttH2HwQF/meu-quadro-do-trello>

## **Link para o Repositório do Projeto**

O código-fonte completo do projeto **OtakuNote/Otaku List** está hospedado em um repositório Git, conforme o padrão de versionamento estabelecido (Restrição RS006). O repositório contém o código de desenvolvimento ([/Dev](#)) e os documentos finais ([/Docs](#)), incluindo este relatório.

Link do Repositório (GitHub): [https://github.com/shayxnx/MVP\\_OtakuList.git](https://github.com/shayxnx/MVP_OtakuList.git)

## **Link para Documentação de Especificações e Requisitos**

Este link direciona para a pasta específica dentro do repositório onde está armazenado este Documento de Especificação de Requisitos (DER).

Link para documentação

[https://github.com/shayxnx/MVP\\_OtakuList/blob/main/README.md](https://github.com/shayxnx/MVP_OtakuList/blob/main/README.md)

Belo Horizonte  
2025