

# Sprint 1 – Projeto VirtuaLib

## User Stories

1. **US01 – Ver a Lista de Livros**  
*Como usuário, quero visualizar uma lista de todos os livros disponíveis na biblioteca para saber quais posso pegar emprestado.*
2. **US02 – Encontrar um Livro Facilmente**  
*Como usuário, quero buscar livros pelo título ou autor para localizar rapidamente o livro que desejo.*
3. **US03 – Emprestar um Livro**  
*Como usuário, quero marcar um livro como “emprestado” para que fique registrado que ele não está mais disponível.*
4. **US04 – Devolver um Livro**  
*Como usuário, quero marcar um livro como “devolvido” para que ele fique disponível novamente para outros usuários.*
5. **US05 – Ver Detalhes de um Livro**  
*Como usuário, quero visualizar informações detalhadas de um livro (sinopse, ano, autor, categoria) para decidir se desejo pegá-lo emprestado.*

## Requisitos Funcionais (RF)

- **RF01:** O sistema deve permitir ao usuário visualizar uma lista de livros, exibindo status (disponível/emprestado).
- **RF02:** O sistema deve oferecer uma barra de busca por título ou autor.
- **RF03:** O sistema deve permitir ao usuário marcar um livro como emprestado, alterando seu status no catálogo.
- **RF04:** O sistema deve permitir ao usuário marcar um livro como devolvido, alterando seu status no catálogo.
- **RF05:** O sistema deve exibir detalhes completos de um livro selecionado (sinopse, ano, autor, categoria).

## Requisitos Não Funcionais (RNF)

- **RNF01:** A listagem e a busca devem retornar resultados em tempo real (tempo de resposta  $\leq 2$  segundos).

- **RNF02:** A interface deve ser intuitiva e responsiva, permitindo uso em dispositivos móveis e desktop.
- **RNF03:** O sistema deve manter consistência nos dados, garantindo que um livro não apareça como disponível quando estiver emprestado.
- **RNF04:** O sistema deve armazenar os dados em um banco confiável que suporte concorrência de múltiplos usuários.

## **Sprint 1 – Planejamento de Tasks**

### **US01 – Ver a Lista de Livros**

- **Task 1.1:** Implementar listagem básica de livros (front-end).
  - **Dev responsável:** Lucas Yudi
  - **Revisor:** Cayke
- **Task 1.2:** Conectar listagem ao backend (API que retorna todos os livros com status).
  - **Dev responsável:** Lucas Gabriel
  - **Revisor:** Filipe

### **US02 – Encontrar um Livro Facilmente**

- **Task 2.1:** Implementar campo de busca (front-end) com filtro por título/autor.
  - **Dev responsável:** Filipe
  - **Revisor:** Gabryel
- **Task 2.2:** Implementar endpoint de busca otimizada no backend.
  - **Dev responsável:** Cayke
  - **Revisor:** Lucas Yudi

### **US03 – Emprestar um Livro**

- **Task 3.1:** Implementar botão de “emprestar” na interface.

- **Dev responsável: Gabryel**
  - **Revisor: Lucas Gabriel**
- **Task 3.2:** Implementar lógica de atualização de status no backend (emprestado).
  - **Dev responsável: Lucas Yudi**
  - **Revisor: Filipe**

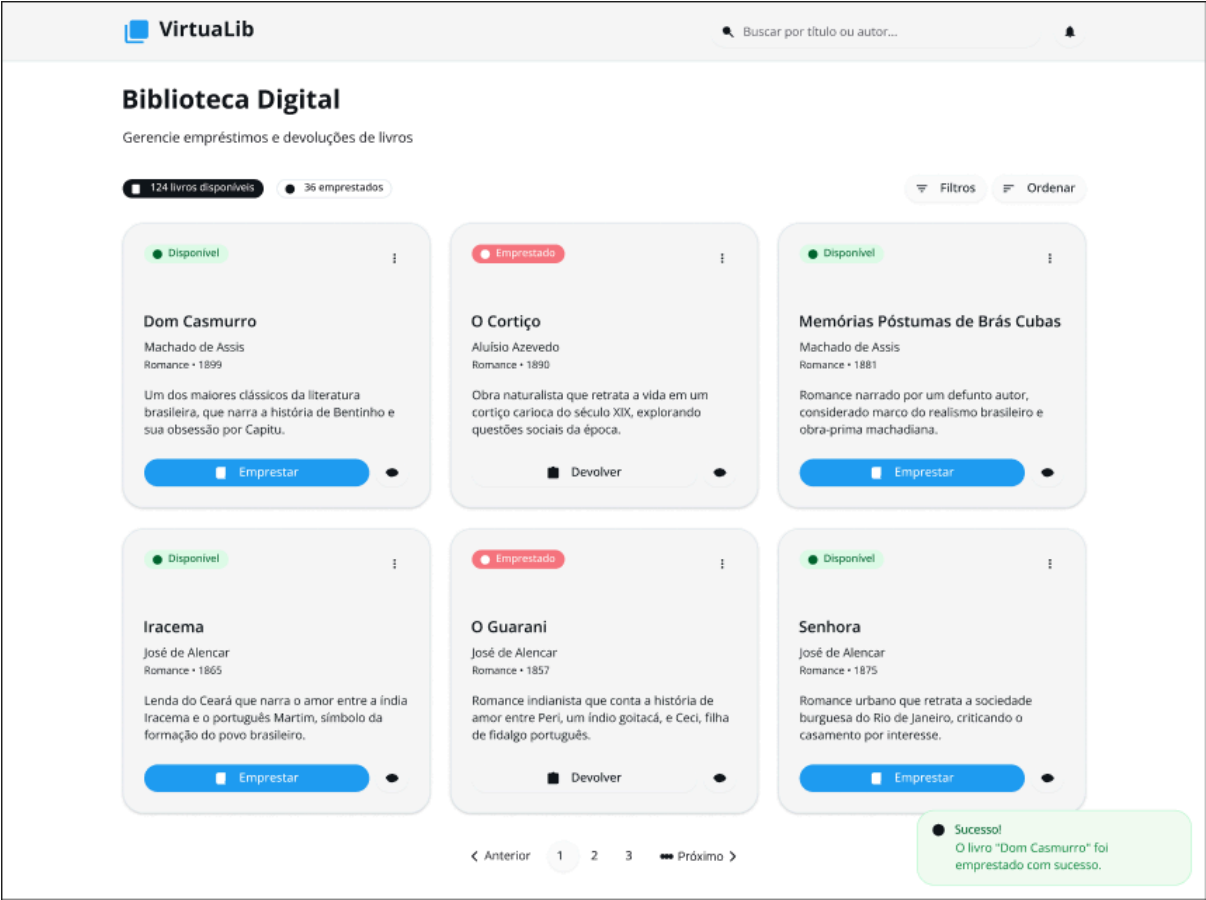
## **US04 – Devolver um Livro**

- **Task 4.1:** Implementar botão de “devolver” na interface.
  - **Dev responsável: Lucas Gabriel**
  - **Revisor: Gabryel**
- **Task 4.2:** Implementar lógica de atualização de status no backend (disponível).
  - **Dev responsável: Cayke**
  - **Revisor: Lucas Yudi**

## **US05 – Ver Detalhes de um Livro**

- **Task 5.1:** Criar página/modal de detalhes com informações do livro.
  - **Dev responsável: Filipe**
  - **Revisor: Cayke**
- **Task 5.2:** Implementar endpoint para retornar detalhes de um livro específico.
  - **Dev responsável: Gabryel**
  - **Revisor: Lucas Gabriel**

# Protótipo Tela Inicial



Figma: [Link](#)

Trello: [Link](#)