

Documento de Requisitos - Sistema de Gerenciamento de Biblioteca

Versão: 1.4 **Data:** 30 de maio de 2025 **Autores:** Arthur Catarino de Oliveira, Mateus Correa Poddis, Petter Douglas Araujo

1. Introdução

Este documento descreve os requisitos funcionais e não funcionais para o **Sistema de Gerenciamento de Biblioteca**, um sistema web que visa otimizar a gestão de livros, exemplares, usuários e empréstimos em uma biblioteca. O objetivo é aplicar os princípios da Engenharia de Software, incluindo requisitos, projeto, implementação, teste, documentação e gerência do processo.

2. Requisitos Funcionais (RFs)

Os requisitos funcionais descrevem as funcionalidades que o sistema deve executar.

2.1. Módulo de Autenticação e Usuários

- **RF001 - Autenticação de Usuário:** O sistema deve permitir que usuários dos tipos 'leitor', 'bibliotecário' e 'administrador' se autenticuem usando e-mail (ou nome de usuário) e senha.
- **RF002 - Cadastro de Leitor:** O sistema deve permitir que **bibliotecários** cadastrem novos leitores, fornecendo nome, CPF, endereço, telefone e e-mail. O status inicial do leitor deve ser definido como 'ativo'.
- **RF003 - Consulta de Leitor:** O sistema deve permitir que **bibliotecários** busquem leitores cadastrados por nome, CPF ou e-mail.
- **RF004 - Edição de Leitor:** O sistema deve permitir que **bibliotecários** editem as informações cadastrais de um leitor existente.
- **RF005 - Exclusão de Leitor:** O sistema deve permitir que **bibliotecários** solicitem a exclusão de um registro de leitor.
 - **RF005.1 (Regra de Negócio):** O sistema deve **impedir** a exclusão de um leitor que possua empréstimos com status 'ativo' ou 'atrasado'.
- **RF006 - Cadastro de Bibliotecário:** O sistema deve permitir que **administradores** cadastrem novos bibliotecários, fornecendo nome e e-mail. O status inicial deve ser definido como 'ativo'.
- **RF007 - Exclusão de Bibliotecário:** O sistema deve permitir que **administradores** excluam registros de bibliotecários.

- **RF007.1 (Regra de Negócio):** O sistema deve **impedir** a exclusão de um bibliotecário se houver registros de operações (empréstimos, devoluções) associados a ele, para manter a integridade do histórico.
- **RF008 - Consulta de Bibliotecário:** O sistema deve permitir que **administradores** busquem bibliotecários cadastrados.
- **RF009 - Edição de Bibliotecário:** O sistema deve permitir que **administradores** editem as informações cadastrais de um bibliotecário.

2.2. Módulo de Livros e Exemplares

- **RF010 - Cadastro de Livro:** O sistema deve permitir que **bibliotecários** cadastrem um novo título de livro, informando título, autor, ISBN, editora, ano de publicação e gênero.
- **RF011 - Consulta de Livro:** O sistema deve permitir que **bibliotecários e leitores** busquem títulos de livros por título, autor, ISBN ou gênero.
- **RF012 - Edição de Livro:** O sistema deve permitir que **bibliotecários** editem as informações de um título de livro já cadastrado.
- **RF013 - Exclusão de Livro:** O sistema deve permitir que **bibliotecários** solicitem a exclusão de um título de livro.
 - **RF013.1 (Regra de Negócio):** O sistema deve **impedir** a exclusão de um livro se existirem exemplares registrados para ele.
- **RF014 - Cadastro de Exemplar:** O sistema deve permitir que **bibliotecários** registrem um novo exemplar, associando-o a um título de livro existente e gerando um número de registro único.
 - **RF014.1 (Regra de Sistema):** Ao ser cadastrado, todo exemplar deve receber o status inicial 'disponível'.
- **RF015 - Consulta de Exemplar:** O sistema deve permitir que **bibliotecários** consultem exemplares específicos pelo seu número de registro.
- **RF016 - Edição de Status do Exemplar:** O sistema deve permitir que **bibliotecários** alterem o status de um exemplar (ex: 'disponível', 'emprestado', 'danificado').
- **RF017 - Exclusão de Exemplar:** O sistema deve permitir que **bibliotecários** excluam o registro de um exemplar.
 - **RF017.1 (Regra de Negócio):** O sistema deve **impedir** a exclusão de um exemplar que esteja atualmente associado a um empréstimo ativo.

2.3. Módulo de Empréstimos e Devoluções (CRUD 3+ Tabelas)

- **RF018 - Realizar Empréstimo:** O sistema deve permitir que **bibliotecários** registrem um empréstimo, associando um exemplar a um leitor e definindo a data de devolução prevista.
 - **RF018.1 (Regra de Negócio):** O sistema deve **impedir** que um leitor realize um novo empréstimo se ele tiver empréstimos com status 'atrasado'.
 - **RF018.2 (Regra de Negócio):** O sistema deve **permitir** o empréstimo apenas de exemplares com o status 'disponível'.
 - **RF018.3 (Regra de Sistema):** Após a confirmação de um empréstimo, o sistema deve **automaticamente** alterar o status do exemplar para 'emprestado'.

- **RF019 - Registrar Devolução:** O sistema deve permitir que **bibliotecários** registrem a devolução de um exemplar, informando o ID do empréstimo ou do exemplar.
 - **RF019.1 (Regra de Sistema):** Após a confirmação da devolução, o sistema deve **automaticamente** registrar a data de devolução real.
 - **RF019.2 (Regra de Sistema):** Após a confirmação da devolução, o sistema deve **automaticamente** alterar o status do exemplar para 'disponível'.
 - **RF019.3 (Regra de Negócio):** O sistema deve **calcular e registrar** o valor da multa se a data de devolução real for posterior à data de devolução prevista.
- **RF020 - Consultar Empréstimos (Visão Geral):** O sistema deve permitir que **bibliotecários e administradores** consultem a lista completa de empréstimos, com filtros por status (ativo, devolvido, atrasado) e por leitor.
- **RF021 - Alterar Data de Devolução:** O sistema deve permitir que **bibliotecários** alterem a data de devolução prevista de um empréstimo com status 'ativo'.
- **RF022 - Cancelar Empréstimo:** O sistema deve permitir que **bibliotecários** cancelem um registro de empréstimo incorreto.
 - **RF022.1 (Regra de Sistema):** Ao cancelar um empréstimo, o sistema deve **reverter** o status do exemplar associado para 'disponível'.
- **RF023 - Consultar Meus Empréstimos:** O sistema deve permitir que um **leitor** autenticado visualize apenas o seu histórico pessoal de empréstimos (ativos e já devolvidos), incluindo as datas de devolução.

2.4. Módulo de Reservas (CRUD 3+ Tabelas Adicional)

- **RF024 - Realizar Reserva:** O sistema deve permitir que **bibliotecários** registrem uma reserva de um título de livro para um leitor.
 - **RF024.1 (Regra de Negócio):** O sistema só deve permitir a reserva de um livro se **não houver** nenhum exemplar com status 'disponível' no momento.
 - **RF024.2 (Regra de Negócio):** O sistema deve **impedir** a reserva se o leitor já atingiu o limite máximo de 3 reservas ativas.
- **RF025 - Consultar Reservas:** O sistema deve permitir que **bibliotecários** consultem a lista de reservas, com filtros por livro, leitor ou status.
- **RF026 - Atender Reserva:** Quando um exemplar de um livro reservado for devolvido, o sistema deve permitir que o **bibliotecário** atenda à primeira reserva da fila, alterando seu status para 'pronta para retirada'.
- **RF027 - Cancelar Reserva:** O sistema deve permitir que **bibliotecários ou leitores** cancelem uma reserva com status 'ativa'.
 - **RF027.1 (Regra de Sistema):** O sistema deve **automaticamente** cancelar uma reserva com status 'pronta para retirada' se o leitor não retirar o livro dentro de um prazo pré-definido (ex: 3 dias úteis).

3. Requisitos Não Funcionais (RNFs)

Os requisitos não funcionais descrevem qualidades e restrições do sistema.

- **RNF001 - Usabilidade:**
 - **RNF001.1:** Um **bibliotecário** sem treinamento prévio deve ser capaz de realizar um ciclo completo de empréstimo (buscar leitor, buscar livro, registrar empréstimo) em menos de 90 segundos.
 - **RNF001.2:** Um **leitor** logado deve conseguir encontrar seu histórico de empréstimos em no máximo 3 cliques a partir da página inicial.
 - **RNF002 - Segurança:**
 - **RNF002.1 (Autenticação):** As senhas de todos os usuários devem ser armazenadas no banco de dados utilizando um algoritmo de hash unidirecional (ex: bcrypt).
 - **RNF002.2 (Autorização):** O sistema deve implementar controle de acesso baseado em papéis (Role-Based Access Control), garantindo que as permissões de leitores, bibliotecários e administradores sejam estritamente aplicadas em todas as operações.
 - **RNF002.3 (Proteção de Dados):** O sistema deve possuir mecanismos para proteger contra ataques comuns de segurança web, como Injeção de SQL e Cross-Site Scripting (XSS).
 - **RNF003 - Performance:**
 - **RNF003.1:** O tempo de resposta do servidor para qualquer requisição de consulta (busca de livro, consulta de empréstimo) deve ser inferior a 500 milissegundos para um único usuário.
 - **RNF003.2:** O sistema deve manter um tempo de resposta inferior a 3 segundos para 95% das requisições de CRUD quando submetido a uma carga de 20 usuários simultâneos.
 - **RNF004 - Confiabilidade:**
 - **RNF004.1:** Todas as operações de banco de dados que envolvem múltiplas etapas (ex: registrar empréstimo e alterar status do exemplar) devem ser executadas dentro de uma transação atômica (ACID) para garantir que, em caso de falha, o sistema retorne ao estado anterior sem corromper os dados.
 - **RNF005 - Manutenibilidade:**
 - **RNF005.1:** O código-fonte do backend deve aderir a um padrão de codificação específico da linguagem (ex: PSR-12 para PHP, PEP 8 para Python).
 - **RNF005.2:** A suíte de testes de unidade deve cobrir no mínimo 70% das regras de negócio críticas do sistema.
-

4. Modelo de Dados Sugerido (Esboço Atualizado)

O modelo de dados permanece o mesmo, pois as alterações se referem principalmente às regras de negócio e permissões de acesso, que são controladas na camada de aplicação. A tabela **Usuarios** já suporta o campo **senha_hash** para todos os tipos de usuário.

Tabelas:

1. **Usuarios**

- `id_usuario` (PK, INT)
- `nome` (VARCHAR)
- `email` (VARCHAR, UNIQUE)
- `senha_hash` (VARCHAR)
- `tipo_usuario` (ENUM: 'leitor', 'bibliotecario', 'administrador')
- `cpf` (VARCHAR, UNIQUE, para leitores)
- `endereco` (VARCHAR, para leitores)
- `telefone` (VARCHAR, para leitores)
- `data_cadastro` (DATE/DATETIME)
- `status_ativo` (BOOLEAN)

2. Livros

- `id_livro` (PK, INT)
- `titulo` (VARCHAR)
- `autor` (VARCHAR)
- `isbn` (VARCHAR, UNIQUE)
- `editora` (VARCHAR)
- `ano_publicacao` (INT)
- `genero` (VARCHAR)

3. Exemplos

- `id_exemplar` (PK, INT)
- `id_livro` (FK para Livros, INT)
- `numero_registro` (VARCHAR, UNIQUE, ex: "ISBN-001-A")
- `status_exemplar` (ENUM: 'disponível', 'emprestado', 'reservado', 'danificado')

4. Empréstimos

- `id_emprestimo` (PK, INT)
- `id_usuario` (FK para Usuarios, INT - o leitor)
- `id_exemplar` (FK para Exemplos, INT - o exemplar emprestado)
- `data_emprestimo` (DATE/DATETIME)
- `data_devolucao_prevista` (DATE/DATETIME)
- `data_devolucao_real` (DATE/DATETIME, NULL até a devolução)
- `status_emprestimo` (ENUM: 'ativo', 'atrasado', 'devolvido')
- `valor_multa` (DECIMAL, NULL se não houver multa)

5. Reservas

- `id_reserva` (PK, INT)
- `id_usuario` (FK para Usuarios, INT - o leitor que fez a reserva)
- `id_livro` (FK para Livros, INT - o título reservado)
- `data_reserva` (DATE/DATETIME)

- `data_expiracao` (DATE/DATETIME - pode ser NULL se não tiver expiração definida)
 - `status_reserva` (ENUM: 'ativa', 'pronta para retirada', 'atendida', 'cancelada', 'expirada')
-

5. Plano de Testes (Adequado para Express e React)

Esta seção descreve a estratégia de testes para garantir a qualidade, funcionalidade e robustez do Sistema de Gerenciamento de Biblioteca.

5.1 Testes de Backend (JavaScript/Express)

Os testes de backend terão foco em validar a lógica de negócio, a integração entre os componentes da aplicação e a segurança dos endpoints da API.

Testes de Unidade:

- **Ferramentas: Jest.** É o framework de testes mais popular no ecossistema JavaScript, oferecendo uma solução completa com executor de testes (test runner), biblioteca de asserções e funcionalidades de "mock" (simuladores de dependências) integradas.
- **Foco Principal:**
 - Testar os **módulos de serviço** (`services`) de forma isolada para validar todas as regras de negócio.
 - Lógica de empréstimos (verificação de disponibilidade de exemplares, validação de status do leitor, cálculo de multas).
 - Lógica de reservas (verificação de elegibilidade para reservar, controle de limite de reservas).
 - Testar **módulos utilitários** (`utils`) e **middlewares de validação** (ex: usando Joi ou express-validator) para garantir que funcionam como esperado.

Testes de Integração:

- **Ferramentas: Jest** (como orquestrador), **Supertest** (para realizar requisições HTTP à API do Express) e um banco de dados em memória como o **SQLite** (para bancos SQL) ou **MongoDB-in-Memory-Server** (para bancos NoSQL). Isso evita a dependência de um banco de dados externo durante a execução dos testes.
- **Foco Principal:**
 - Validar a interação entre as camadas da aplicação: **Rotas (Routes/Controllers)**, **Serviços (Services)** e a camada de **Acesso a Dados (Models/Repositories)**.
 - Testar as operações CRUD completas, garantindo que os dados são persistidos e recuperados corretamente no banco de dados de teste.
 - Verificar a lógica de autenticação e autorização de ponta a ponta, testando a proteção de rotas com middlewares de autenticação (ex: validação de tokens)

JWT) e as permissões de cada tipo de usuário (leitor, bibliotecário, administrador).

5.2 Testes de Frontend (React)

Os testes de frontend garantirão que a interface do usuário se comporte como esperado, seja acessível e se comunique corretamente com o backend.

Testes de Componente:

- **Ferramentas:** **Jest** em conjunto com a **React Testing Library (RTL)** para testar os componentes React da forma como o usuário os utiliza.
- **Foco Principal:**
 - Testar componentes individuais (botões, formulários, modais) de forma isolada.
 - Verificar se a renderização condicional está correta (ex: exibir botões de "editar/excluir" apenas para administradores).
 - Validar a manipulação de estado e os eventos do usuário dentro dos componentes.

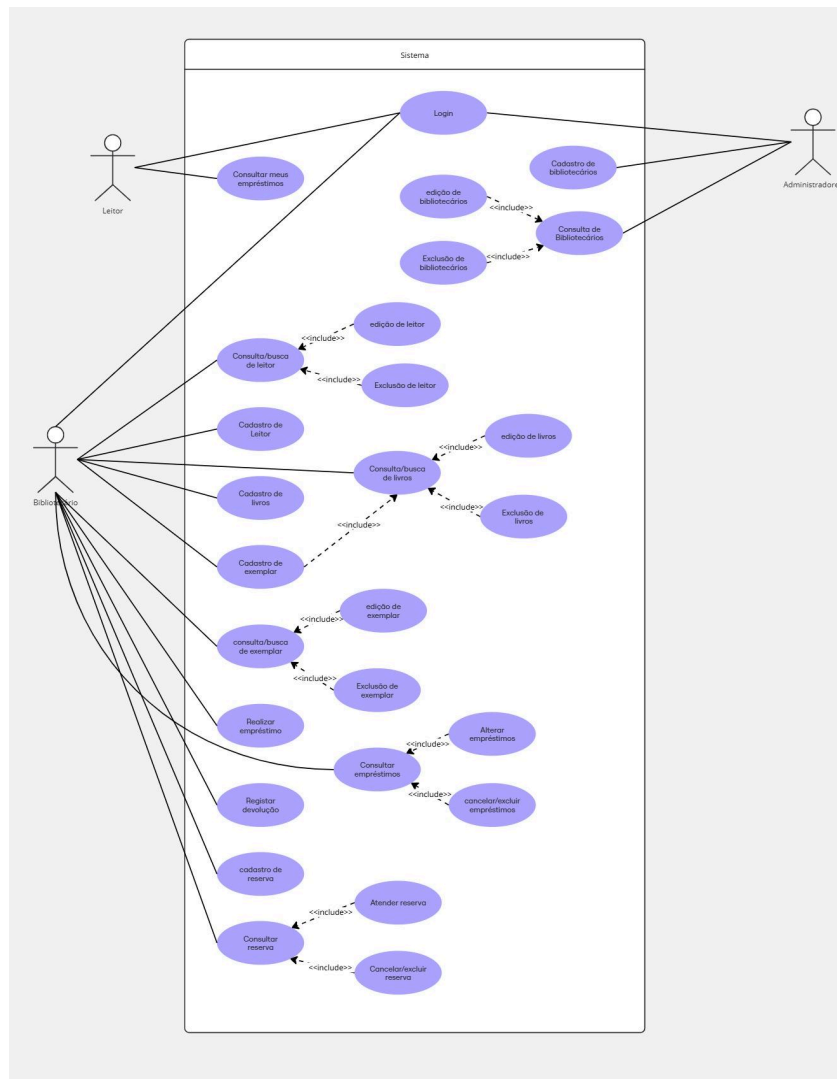
Testes End-to-End (E2E):

- **Ferramentas:** **Cypress** ou **Playwright**. Ambas são ferramentas modernas e robustas para testar aplicações web completas no navegador.
- **Foco Principal:** Simular a jornada completa do usuário no navegador. Os cenários permanecem os mesmos:
 - Login bem-sucedido para os três perfis de usuário.
 - Verificação de que o leitor logado acessa apenas suas próprias informações.
 - Cenário completo de um bibliotecário realizando um empréstimo e uma devolução.
 - Cenário de um administrador cadastrando um novo bibliotecário.
 - Cenário de consulta de livros e de reservas.

5.3 Conceitos e Métricas

- **Cobertura de Teste (Test Coverage):** Utilizaremos o **relatório de cobertura nativo do Jest** (que usa a biblioteca Istanbul.js internamente) tanto para o backend (Express) quanto para o frontend (React). A meta será atingir no mínimo **70% de cobertura** nas regras de negócio críticas.
 - **Testes de Regressão:** Toda a suíte de testes (unidade, integração e E2E) será executada automaticamente em um ambiente de Integração Contínua (CI/CD) após cada nova modificação no código principal, garantindo que novas funcionalidades não quebrem as já existentes.
-

6. Diagrama de Caso de Uso



Nome do caso de uso	Cadastro de Leitor
Descrição	Permite o registro de novos leitores no sistema, armazenando suas informações pessoais e status.
Atores	Bibliotecários
Dados consumidos	Nome, CPF, endereço, telefone, e-mail, status
Dados produzidos	Novo registro na tabela de leitores
Prioridade	Alta
Pré-condição	O bibliotecário precisa estar autenticado com as devidas permissões.
Pós-condição	Um novo leitor é cadastrado no sistema com status "ativo".

Fluxo Principal	
Ação do ator	Resposta do sistema
1. Acessa a função "Cadastrar Leitor" no menu do sistema 2. Preenche todos os dados obrigatórios do leitor. 3. Clica no botão "Salvar" para submeter o formulário.	
Fluxo Alternativo	
3a. Dados inválidos 4a. CPF já cadastrado	

Nome do caso de uso	Consultar leitor
Descrição	Permite localizar leitores cadastrados no sistema por nome, CPF ou e-mail.
Atores	Bibliotecários
Dados consumidos	Filtro de busca (nome, CPF, e-mail)
Dados produzidos	Lista de leitores correspondentes
Prioridade	Alta
Pré-condição	O bibliotecário precisa estar autenticado no sistema.
Pós-condição	O sistema exibe uma lista de leitores que correspondem aos critérios de busca.
Fluxo Principal	
Ação do ator	Resposta do sistema
1. Acessa a página "Consultar Leitores". 2. Insere o termo de busca em um ou mais campos e clica em "Buscar".	
Fluxo Alternativo	
3a. Nenhum resultado encontrado	

Nome do caso de uso	Editar leitor
Descrição	Permite modificar as informações cadastrais de um leitor já existente.
Atores	Bibliotecários
Dados consumidos	CPF ou ID do leitor, dados a serem alterados

Dados produzidos	Registro do leitor atualizado	
Prioridade	Média	
Pré-condição	O bibliotecário está autenticado e o leitor a ser editado já existe no sistema.	
Pós-condição	As informações do leitor são atualizadas no banco de dados.	
Fluxo Principal		
Ação do ator		Resposta do sistema
<div>1. Busca e localiza o leitor desejado (conforme o caso de uso "Consultar Leitor").</div> <div>2. Clica na opção "Editar" correspondente ao registro do leitor.</div> <div>3. Altera os dados desejados no formulário.</div> <div>4. Clica no botão "Salvar Alterações".</div>		
Fluxo Alternativo		
4a. Dados inválidos		

Nome do caso de uso	Excluir leitor	
Descrição	Permite solicitar a remoção de um leitor do sistema.	
Atores	Bibliotecários	
Dados consumidos	ID do leitor	
Dados produzidos	Registro de leitor removido	
Prioridade	Média	
Pré-condição	O bibliotecário está autenticado. O leitor a ser excluído está cadastrado.	
Pós-condição	O registro do leitor é removido do sistema, caso as regras de negócio permitam.	
Fluxo Principal		
Ação do ator		Resposta do sistema
<div>1. Busca e localiza o leitor desejado.</div> <div>2. Clica na opção "Excluir" correspondente ao registro do leitor.</div> <div>3. Confirma a exclusão.</div>		
Fluxo Alternativo		
3a. Exclusão bloqueada		

Nome do caso de uso	Registrar devolução
Descrição	Permite que bibliotecários registrem a devolução de um exemplar, atualizando o status do empréstimo e do exemplar, e calculando multas se necessário.
Atores	Bibliotecários
Dados consumidos	ID do exemplar, ID do empréstimo, data de devolução
Dados produzidos	Status do exemplar atualizado, data de devolução real registrada, multa (se aplicável)
Prioridade	Alta
Pré-condição	O bibliotecário está autenticado. Existe um empréstimo com status "ativo" ou "atrasado".
Pós-condição	O status do empréstimo é alterado para "devolvido", a data de devolução real é registrada , o status do exemplar é alterado para "disponível" e, se aplicável, a multa é registrada.
Fluxo Principal	
Ação do ator	Resposta do sistema
1. Acessa a função "Registrar Devolução". 2. Informa o número de registro do exemplar e clica em "Buscar". 3. Confirma a devolução do item.	
Fluxo Alternativo	
4a. Devolução com atraso 2a. Exemplar não encontrado ou sem empréstimo ativo	

Nome do caso de uso	Consultar Empréstimos	
Descrição	Exibe todos os empréstimos realizados, permitindo a filtragem por diversos critérios como status ou leitor.	
Atores	Bibliotecários, Administradores	
Dados consumidos	Parâmetros de busca (nome, status, data)	
Dados produzidos	Lista de empréstimos encontrados	
Prioridade	Alta	
Pré-condição	O usuário precisa estar autenticado com o perfil de bibliotecário ou administrador.	
Pós-condição	Os empréstimos que correspondem aos filtros são	

	apresentados na tela.
Fluxo Principal	
Ação do ator	Resposta do sistema
<ol style="list-style-type: none"> 1. O usuário (Bibliotecário ou Administrador) acessa a tela de "Consulta de Empréstimos". 2. O sistema exibe os filtros de busca disponíveis (ex: por status, nome do leitor, data). 3. O usuário informa os filtros desejados e aciona a busca. 4. O sistema realiza a consulta e exibe a lista de empréstimos encontrados, detalhando as informações principais de cada um. 	
Fluxo Alternativo	
4a. Nenhum resultado encontrado: No passo 4, se a consulta não retornar nenhum registro, o sistema exibe uma mensagem informativa, como: "Nenhum empréstimo encontrado para os filtros aplicados."	

Nome do caso de uso	Alterar empréstimos
Descrição	Permite alterar a data de devolução prevista de um empréstimo que ainda está ativo.
Atores	Bibliotecários
Dados consumidos	ID do empréstimo, nova data de devolução
Dados produzidos	Registro de empréstimo atualizado
Prioridade	Média
Pré-condição	O empréstimo deve estar com status "ativo".
Pós-condição	A nova data de devolução prevista é registrada no empréstimo.
Fluxo Principal	
Ação do ator	Resposta do sistema
<ol style="list-style-type: none"> 1. O bibliotecário localiza o empréstimo com status "ativo" que deseja alterar. 2. O bibliotecário seleciona a opção para editar os dados do empréstimo. 3. O sistema exibe os detalhes do empréstimo, permitindo a edição do campo "Data de Devolução Prevista". 4. O bibliotecário insere a nova data e confirma a alteração. 5. O sistema valida se a nova data é válida (ex: se é posterior à data do empréstimo e à data atual). 6. O sistema salva a nova data_devolucao_prevista no registro do empréstimo. 7. O sistema exibe uma mensagem de sucesso, como "Data de devolução alterada com sucesso!". 	

Fluxo Alternativo
5a. Data Inválida: No passo 5, se a data informada for inválida (ex: anterior à data em que o empréstimo foi realizado), o sistema bloqueia a alteração e exibe uma mensagem de erro.

Nome do caso de uso	Cadastrar reserva
Descrição	Permite registrar uma reserva para um livro que não possui exemplares disponíveis no momento.
Atores	Bibliotecários
Dados consumidos	Identificação do leitor, título do livro
Dados produzidos	Registro na tabela reservas com status inicial
Prioridade	Alta
Pré-condição	O leitor e o livro estão cadastrados. O bibliotecário está autenticado.
Pós-condição	Uma nova reserva é criada e associada ao leitor e ao livro, com status "ativa".

Fluxo Principal	
Ação do ator	Resposta do sistema
1. Acessa a funcionalidade de reserva, busca e seleciona o leitor e o título do livro desejado. 2. Clica em "Confirmar Reserva".	

Fluxo Alternativo
2a. Livro disponível para empréstimo 3a. Leitor atingiu o limite de reservas