



Proj aplic multiplataforma etapa 1- N 705

PROFESSOR ORIENTADOR: Manoel Miqueias Maia

Aluno: Jonarta Santiago Soares- 2327347

Link do Github:

<https://github.com/Jonarta-byte/Proj-aplic-multiplataforma-etapa-1-N-705>

Introdução

Este documento apresenta os requisitos funcionais e não funcionais para o desenvolvimento de um sistema de gestão de bibliotecas públicas. O sistema tem como finalidade otimizar os processos de cadastro, empréstimo, devolução e consulta de acervos, além de apoiar o controle administrativo. O projeto foi pensado partindo do princípio que a Biblioteca pública do município de Redenção o sistema da biblioteca pública é totalmente manual a população não tem acesso de forma digital.

Escopo

O sistema de gestão de bibliotecas públicas será utilizado por bibliotecários, funcionários administrativos e usuários (leitores). Ele deve permitir:

O gerenciamento completo do acervo (livros, revistas, jornais e mídias digitais).



O controle de empréstimos, reservas e devoluções.

A disponibilização de catálogo online para consulta pública.

Requisitos

Requisitos Funcionais (RF)	Requisitos Não Funcionais (RNF)
<p>RF01 - Cadastro de Funcionário: O sistema deve permitir o cadastro de bibliotecários e funcionários com níveis diferentes de acesso.</p> <p>RF02 – Cadastro de Obras: O sistema deve permitir o registro de itens do acervo (livros, revistas, jornais, e-books, CDs/DVDs), autor, título, editora, ano, classificação e número de exemplares.</p> <p>RF03 – Consulta ao Catálogo: O sistema deve permitir que usuários consultem obras disponíveis por título, autor, assunto, ISBN ou palavra-chave.</p> <p>RF04 – Empréstimo de Obras: O sistema deve registrar empréstimos, vinculando usuário, obra e prazo de devolução.</p> <p>RF05 – Devolução de Obras: O sistema deve registrar devoluções, calculando automaticamente possíveis multas por atraso.</p> <p>RF06 – Acesso Online: O sistema deve disponibilizar acesso ao catálogo e à situação do usuário (empréstimos, reservas, multas) via aplicativo.</p>	<p>RNF01 – Desempenho: O sistema deve suportar pelo menos 200 acessos simultâneos sem perda de desempenho.</p> <p>RNF02 – Segurança: Todas as informações pessoais devem ser armazenadas em conformidade com a LGPD, com criptografia de senhas.</p> <p>RNF03 – Disponibilidade: O sistema deve estar disponível 24h por dia, 7 dias por semana, com no mínimo 99% de uptime.</p> <p>RNF04 – Usabilidade: A interface deve ser intuitiva e acessível, seguindo normas de acessibilidade (WCAG 2.1), permitindo uso por pessoas com deficiência.</p> <p>RNF05 – Portabilidade: O sistema deve ser compatível com navegadores modernos e dispositivos móveis (layout responsivo).</p> <p>RNF06 – Integração: O sistema deve permitir integração com bases externas (ex.: ISBN e sistemas de bibliotecas integradas nacionais).</p> <p>RNF07 – Confiabilidade: O sistema deve realizar backups automáticos diários.</p> <p>RNF08 – Manutenibilidade: O código deve seguir padrões de desenvolvimento modular e</p>

	documentado para facilitar manutenção.
--	--

Restrições

- O sistema deverá ser desenvolvido em plataforma API Mobile.
- O sistema deve estar hospedado em servidor da prefeitura ou em serviço de nuvem contratado.

Tecnologias e Ferramentas a serem utilizadas

1. Linguagem + Framework- Node.js + Express.js : Alternativa: Python + FastAPI
2. Banco de Dados- PostgreSQL

Infraestrutura e Hospedagem

1. Serviços em Nuvem: AWS, Azure ou Google Cloud.

Modelo ER

Entidades e Atributos

1. Obra (Livro/Mídia)
 - id obra (PK)
 - título
 - autor
 - editora
 - ano publicação
 - isbn
 - categoria
 - quantidade total
 - quantidade disponível
2. Empréstimo
 - id empréstimo (PK)
 - id_usuario (FK)
 - id obra (FK)
 - data empréstimo
 - data prevista devolução
 - data devolução (null se ainda não devolvido)
3. Reserva

- id reserva (PK)
- id_usuario (FK)
- id obra (FK)
- data reserva
- status (ativa, atendida, cancelada)

Relacionamentos

- Um Usuário pode ter vários Empréstimos.
- Um Usuário pode ter várias Reservas.
- Uma Obra pode estar em vários Empréstimos e Reservas.

Cronograma de desenvolvimento de API

Fase	Atividades	Duração	Responsáveis
Planejamento	- Levantamento de requisitos Definição de escopo e endpoints Escolha de tecnologias e padrões	0,5	PO, Arquiteto, Tech Lead
Design da API	- Modelagem do banco de dados Definição de contratos Criação do diagrama de arquitetura	0,5	Dev Backend, Arquiteto
Configuração do Ambiente	- Configuração do repositório Git CI/CD inicial Configuração do ambiente (Docker, Postman, ferramentas de testes)	1	DevOps, Dev Backend
Desenvolvimento - Iteração 1	- Implementação de endpoints principais (ex.: autenticação, CRUD básico) Testes unitários iniciais	1	Dev Backend
Desenvolvimento - Iteração 2	- Implementação de endpoints complementares (filtros, paginação, integrações externas) Testes de integração	1	Dev Backend
Testes e Validação	- Testes automatizados e manuais Testes de performance e segurança Ajustes de bugs encontrados	1	QA, Dev Backend

Documentação	- Documentar endpoints (Swagger/Postman) - Guia de autenticação e uso - Versionamento da API	0.5	Dev Backend, Tech Writer
Deploy e Monitoramento	- Publicação em ambiente produtivo - Configuração de logs e monitoramento - Entrega para os consumidores da API	0.5	DevOps, Dev Backend