

# BookShelf (Parte 1) - Instruções para Desenvolvimento

## Visão Geral do Projeto

Você deve desenvolver uma aplicação web moderna para gerenciamento de biblioteca pessoal chamada **BookShelf**. A aplicação permitirá aos usuários catalogar, organizar e acompanhar o progresso de leitura de seus livros.

## Tecnologias Obrigatórias

- **Next.js 15** com App Router
- **React 19**
- **TypeScript**
- **Tailwind CSS**
- **shadcn/ui** (biblioteca de componentes)

## Funcionalidades Obrigatórias

### 1. Dashboard Principal

- Exibir estatísticas gerais da biblioteca:
  - Total de livros cadastrados
  - Livros sendo lidos atualmente
  - Livros já finalizados
  - Total de páginas lidas
- Navegação rápida para outras seções da aplicação
- Design responsivo e atrativo

### 2. Biblioteca (Listagem de Livros)

- Exibir todos os livros cadastrados em formato de cards
- Sistema de busca por título ou autor
- Filtros por gênero literário
- Cada card deve mostrar:
  - Capa do livro (com fallback para imagem padrão)
  - Título e autor

- Ano de publicação
- Sistema de avaliação por estrelas (1-5)
- Gênero como badge
- Botões para visualizar, editar e excluir

### 3. Adicionar Novo Livro

- Formulário completo com os seguintes campos:
  - **Obrigatórios:** Título, Autor
  - **Opcionais:** Total de páginas, página atual, status de leitura, ISBN, URL da capa, gênero, avaliação por estrelas, notas pessoais
- Preview da capa em tempo real quando URL for fornecida
- Barra de progresso do preenchimento do formulário
- Validação antes do envio
- Feedback visual de sucesso/erro

### 4. Visualizar Detalhes do Livro

- Página individual para cada livro
- Exibição completa de todas as informações
- Sinopse detalhada
- Botões para editar ou excluir o livro

### 5. Editar Livro

- Formulário pré-preenchido com dados existentes
- Mesma interface da criação de novos livros
- Atualização em tempo real dos dados

### 6. Excluir Livro

- Dialog de confirmação antes da exclusão
- Prevenção de exclusões acidentais
- Feedback visual claro

## Estrutura de Dados

### Modelo de Livro

Cada livro deve conter as seguintes propriedades:

- id: identificador único (string)
- title: título do livro (string, obrigatório)
- author: autor do livro (string, obrigatório)
- genre: gênero literário (string)
- year: ano de publicação (number)
- pages: total de páginas (number)
- rating: avaliação de 1-5 estrelas (number)
- synopsis: sinopse/descrição (string)
- cover: URL da capa (string)

## Status de Leitura

- QUERO\_LER: Livros na lista de desejos
- LENDO: Livros em progresso
- LIDO: Livros finalizados
- PAUSADO: Leitura temporariamente interrompida
- ABANDONADO: Leitura descontinuada

## Gêneros Disponíveis

Literatura Brasileira, Ficção Científica, Realismo Mágico, Ficção, Fantasia, Romance, Biografia, História, Autoajuda, Tecnologia, Programação, Negócios, Psicologia, Filosofia, Poesia

## Requisitos Técnicos

### 1. Responsividade

- Design mobile-first
- Funcionar perfeitamente em dispositivos móveis, tablets e desktop
- Grid adaptativo para listagem de livros
- Navegação otimizada para diferentes tamanhos de tela

### 2. Acessibilidade

- Usar componentes semânticos
- Suporte a navegação por teclado
- Labels apropriados para formulários
- Contraste adequado de cores

### 3. Performance

- Lazy loading de componentes quando apropriado
- Otimização de imagens
- Estados de loading para operações assíncronas

### 4. Experiência do Usuário

- Feedback visual para todas as ações do usuário
- Design consistente em toda a aplicação
- Tratamento de erros adequado

## Requisitos de Interface

### 1. Design System

- Utilizar a biblioteca shadcn/ui para componentes base
- Manter consistência visual em toda a aplicação
- Usar Tailwind CSS para estilização
- Implementar tema claro e profissional

### 2. Navegação

- Header/navbar com navegação principal
- Breadcrumbs quando apropriado
- Botões de voltar em páginas internas
- Links claros e intuitivos

### 3. Formulários

- Validação em tempo real
- Mensagens de erro claras
- Estados de loading durante submissão
- Campos obrigatórios claramente marcados

### 4. Feedback Visual

- Toasts/notificações para ações importantes
- Estados de loading
- Confirmações para ações destrutivas
- Indicadores de progresso quando apropriado

## Dados Iniciais

A aplicação deve começar com pelo menos 5 livros pré-cadastrados para demonstração, incluindo:

- Livros de diferentes gêneros
- Diferentes anos de publicação
- Variadas avaliações por estrelas
- Sinopses completas
- URLs de capas funcionais

## Dicas Importantes

- Comece pelas funcionalidades básicas antes de partir para recursos avançados
- Teste a aplicação em diferentes dispositivos e tamanhos de tela
- Mantenha o código organizado e bem comentado
- Use o sistema de tipos do TypeScript adequadamente
- Implemente tratamento de erros em todas as operações
- Considere a experiência do usuário em cada decisão de design

## Recursos Recomendados

- Documentação oficial do Next.js 15
- Documentação do shadcn/ui
- Guias de boas práticas do React
- Documentação do Tailwind CSS
- Exemplos de interfaces modernas para inspiração

Boa sorte no desenvolvimento! Lembre-se de que a qualidade da implementação é mais importante que a quantidade de funcionalidades extras.