

Requisitos do Sistema

Requisitos Funcionais:

1. **Cadastro de Itens:**
 - O sistema deve permitir o registro de novos itens com os seguintes campos obrigatórios: nome, código, descrição, categoria, fornecedor, e preço de custo.
 - O sistema deve permitir a edição e exclusão de itens cadastrados.
2. **Controle de Estoque:**
 - O sistema deve registrar a entrada e saída de itens, atualizando automaticamente os níveis de estoque.
 - O sistema deve manter um histórico de todas as movimentações de estoque, incluindo data, quantidade e tipo de movimentação (entrada ou saída).
3. **Consultas e Relatórios:**
 - O sistema deve fornecer uma funcionalidade de busca para localizar itens por nome, código ou categoria.
 - O sistema deve gerar relatórios sobre o status do inventário, incluindo itens em estoque, itens com baixa quantidade e histórico de movimentações.
4. **Alertas e Notificações:**
 - O sistema deve permitir a configuração de alertas para itens com estoque abaixo de um nível mínimo definido pelo usuário.
 - O sistema deve enviar notificações automáticas quando um item atingir o nível de estoque baixo ou estiver próximo do vencimento.
5. **Integração:**
 - O sistema deve oferecer APIs para integração com outros sistemas empresariais, como sistemas de vendas, compras e contabilidade.

Requisitos Não Funcionais:

1. **Usabilidade:**
 - O sistema deve ter uma interface amigável e intuitiva, acessível para usuários com diferentes níveis de experiência.
 - O sistema deve estar disponível em múltiplos dispositivos, incluindo desktops, tablets e smartphones.
2. **Desempenho:**
 - O sistema deve ser capaz de processar e atualizar o inventário em tempo real, com respostas rápidas para entradas, saídas e consultas de itens.
 - O sistema deve suportar um grande volume de dados sem degradação de performance.
3. **Segurança:**
 - O sistema deve garantir a segurança dos dados armazenados, implementando medidas de criptografia e proteção contra acessos não autorizados.

- O sistema deve permitir a gestão de permissões de acesso, garantindo que apenas usuários autorizados possam realizar alterações no inventário.
- 4. **Confiabilidade:**
 - O sistema deve ser robusto e resistente a falhas, garantindo a integridade dos dados mesmo em casos de interrupções inesperadas.
 - O sistema deve realizar backups regulares dos dados para evitar perdas em caso de falhas críticas.
- 5. **Manutenibilidade:**
 - O sistema deve ser fácil de manter e atualizar, com uma arquitetura modular que permita a adição de novas funcionalidades sem impactar o desempenho existente.
 - O sistema deve incluir documentação completa para desenvolvedores e usuários finais, facilitando a resolução de problemas e a implementação de melhorias.

Requisitos de Implementação:

1. **Plataforma:**
 - O sistema deve ser desenvolvido para a web, utilizando tecnologias modernas como HTML, CSS, JavaScript (preferencialmente com frameworks como React ou Angular), e uma linguagem de backend como Python, Java ou Node.js.
 - O sistema deve ser compatível com os principais navegadores (Chrome, Firefox, Safari, Edge).
2. **Banco de Dados:**
 - O sistema deve utilizar um banco de dados relacional (como MySQL, PostgreSQL) ou um banco de dados NoSQL (como MongoDB) para armazenar os dados do inventário.
3. **Hospedagem:**
 - O sistema deve ser hospedado em um servidor confiável, com suporte a escalabilidade para atender ao crescimento do volume de dados e número de usuários.
4. **API:**
 - O sistema deve expor uma API RESTful para integração com outros sistemas, permitindo operações de leitura, escrita, atualização e exclusão de dados do inventário.

Requisitos de Teste:

1. **Testes Unitários:**
 - O sistema deve incluir testes unitários para validar a funcionalidade de cada componente individualmente.
2. **Testes de Integração:**
 - O sistema deve incluir testes de integração para garantir que os diferentes módulos funcionem corretamente em conjunto.
3. **Testes de Performance:**

- O sistema deve ser submetido a testes de performance para garantir que possa lidar com altas cargas de dados e acessos simultâneos sem degradação de performance.

4. Testes de Segurança:

- O sistema deve ser submetido a testes de segurança para identificar e corrigir vulnerabilidades potenciais.