

# Levantamento de Requisitos Funcionais e Não-Funcionais

## 1. Introdução

Este documento detalha os requisitos funcionais e não-funcionais do sistema de vendas de peças e serviços automotivos e de ar-condicionado veicular. Os requisitos foram levantados com o objetivo de garantir que o sistema atenda às necessidades de gestão de estoque, venda de peças, agendamento de serviços, faturamento e geração de relatórios.

---

## 2. Requisitos Funcionais

### 2.1 Módulo de Cadastro de Peças e Serviços

#### 1. Cadastro de Peças

- O sistema deve permitir que o usuário cadastre novas peças com informações como:
  - Nome da peça.
  - Código do produto (SKU).
  - Descrição detalhada.
  - Quantidade em estoque.
  - Preço de compra e preço de venda.
  - Imagem da peça (opcional).
  - Categoria e fornecedor.
- A funcionalidade deve incluir:
  - Opções para criar, atualizar e excluir registros de peças.
  - Validações para campos obrigatórios.

#### 2. Cadastro de Serviços

- O sistema deve permitir o cadastro de diferentes tipos de serviços automotivos, como:
  - Manutenção de ar-condicionado.
  - Troca de óleo.
  - Revisão completa.
  - Alinhamento e balanceamento.
- Cada serviço deve ter informações como:
  - Nome do serviço.
  - Descrição detalhada.
  - Preço do serviço.
  - Tempo estimado de execução.

## **2.2 Módulo de Controle de Estoque**

### **1. Controle de Quantidade**

- O sistema deve atualizar automaticamente o estoque de peças ao final de uma venda.
- Em caso de quantidade baixa de uma peça (configurável pelo usuário), o sistema deve enviar um alerta para reposição.

### **2. Movimentação de Estoque**

- O sistema deve registrar todas as movimentações de entrada e saída de peças.
- Relatório de movimentação de estoque, com histórico de alterações e ajustes manuais de quantidade.

### **3. Relatório de Estoque**

- Relatórios detalhados de estoque, incluindo:
  - Quantidade atual de cada peça.
  - Data da última atualização de estoque.
  - Informações sobre produtos com quantidade abaixo do mínimo estabelecido.

## **2.3 Módulo de Vendas e Faturamento**

### **1. Processo de Venda**

- O sistema deve permitir que o vendedor selecione um cliente e adicione peças e/ou serviços ao pedido.
- O sistema deve calcular automaticamente o valor total do pedido, aplicando impostos e descontos, se aplicável.
- Deve ser possível registrar o pagamento e gerar uma nota fiscal.
- Após a conclusão de uma venda, o estoque deve ser atualizado automaticamente.

### **2. Emitir Faturas e Notas Fiscais**

- O sistema deve emitir faturas com informações detalhadas do pedido, incluindo itens, quantidades e valores.
- Deve haver integração com uma API para geração de nota fiscal eletrônica (NF-e), onde aplicável.
- O sistema deve armazenar o histórico de notas fiscais emitidas.

### **3. Histórico de Vendas**

- O sistema deve armazenar o histórico de vendas e permitir consulta por data, cliente ou vendedor.

### **4. Cancelamento de Vendas**

- Deve ser possível cancelar uma venda, com ajuste automático do estoque e atualização dos relatórios financeiros.

## **2.4 Módulo de Agendamento de Serviços**

### **1. Agendamento de Serviços**

- O sistema deve permitir que os clientes ou funcionários agendem serviços, selecionando data e horário disponíveis.
- Cada agendamento deve conter:

- Nome do cliente.
  - Serviço a ser realizado.
  - Veículo (marca, modelo, ano e placa).
  - Data e hora do agendamento.
  - O sistema deve verificar a disponibilidade de horário e evitar conflitos de agenda.
- 2. **Notificações de Agendamento**
  - O sistema deve enviar notificações (e-mail e SMS) de confirmação do agendamento para o cliente.
  - Notificações de lembrete devem ser enviadas ao cliente 24 horas antes do horário agendado.
- 3. **Histórico de Serviços Realizados**
  - O sistema deve manter um histórico de todos os serviços realizados em cada veículo, associando-os ao cliente e ao veículo.

## 2.5 Módulo de Relatórios e Dashboards

1. **Relatórios Financeiros**
    - Relatórios de faturamento, com visão geral das vendas e recebimentos em determinado período.
    - Relatório de despesas, com custo de aquisição de peças e margem de lucro.
  2. **Relatório de Desempenho de Estoque**
    - Relatório de peças mais vendidas e menos vendidas.
    - Análise de tempo de permanência de itens em estoque.
  3. **Relatório de Atendimento**
    - Relatório de serviços prestados, permitindo análise por tipo de serviço, por veículo e por cliente.
  4. **Dashboard Resumido**
    - Dashboard com gráficos de vendas, desempenho de estoque e agenda de serviços.
- 

## 3. Requisitos Não-Funcionais

### 3.1 Requisitos de Desempenho

1. **Tempo de Resposta**
  - O sistema deve responder a todas as operações de CRUD (Create, Read, Update, Delete) em até 3 segundos.
  - Relatórios complexos e consultas de grande volume devem ser carregados em até 5 segundos.
2. **Capacidade de Usuários Simultâneos**
  - O sistema deve suportar pelo menos 100 usuários simultâneos sem degradação perceptível de desempenho.

### 3.2 Requisitos de Segurança

#### 1. Autenticação e Autorização

- O sistema deve implementar autenticação baseada em papéis (e.g., Administrador, Vendedor, Técnico) para controlar o acesso às funcionalidades.
- Todas as senhas e dados sensíveis devem ser armazenados de forma criptografada.

#### 2. Proteção contra Ataques

- Implementar medidas de segurança para proteger contra ataques de SQL Injection, XSS e CSRF.
- A comunicação entre frontend e backend deve ser feita por meio de HTTPS.

### 3.3 Requisitos de Escalabilidade

#### 1. Escalabilidade Horizontal e Vertical

- O sistema deve ser projetado para crescer em capacidade, suportando tanto a adição de novos servidores quanto o aumento de recursos em servidores existentes.
- O backend deve ser modular para suportar o aumento do volume de transações, principalmente em consultas e emissão de relatórios.

### 3.4 Requisitos de Disponibilidade

#### 1. Alta Disponibilidade

- O sistema deve ser configurado para garantir uma disponibilidade de 99,9%, minimizando o tempo de inatividade.
- Deve haver um processo de backup automático diário para garantir a recuperação dos dados em caso de falhas.

### 3.5 Requisitos de Usabilidade

#### 1. Interface Intuitiva

- A interface deve ser projetada de forma a ser intuitiva e fácil de usar, mesmo para usuários com baixo conhecimento técnico.
- A navegação deve ser clara e padronizada, com botões de ação visíveis e feedback para ações executadas pelo usuário.

#### 2. Design Responsivo

- O sistema deve ser responsivo, ajustando-se automaticamente a diferentes tamanhos de tela, incluindo dispositivos móveis, tablets e desktops.

### 3.6 Requisitos de Manutenibilidade

#### 1. Código Modular

- O código do sistema deve ser modular e documentado para facilitar futuras manutenções e atualizações.
- O sistema deve seguir padrões de desenvolvimento (como MVC no .NET e Componentização no Angular).

#### 2. Logs e Monitoramento

- O sistema deve registrar logs detalhados de erros e eventos importantes para facilitar a análise de problemas.
  - Deve haver monitoramento automático para alertar sobre problemas de performance e uso excessivo de recursos.
- 

## **4. Conclusão**

Este documento fornece uma visão detalhada dos requisitos funcionais e não-funcionais do sistema. Ele servirá como base para o desenvolvimento do projeto, garantindo que o sistema atenda aos padrões de qualidade e às expectativas do cliente. Os requisitos aqui descritos poderão ser refinados ao longo do processo de desenvolvimento, conforme novas necessidades forem identificadas ou ajustes se tornarem necessários.