

Documento de Especificação de Requisitos de Software (SRS)

OdontoSys Lite v3

Desenvolvido por: Luidy
Curso: Análise e Desenvolvimento de Sistemas

21 de janeiro de 2026

Sumário

| | | |
|----------|--|----------|
| 1 | Introdução | 2 |
| 1.1 | Propósito | 2 |
| 1.2 | Escopo do Produto | 2 |
| 1.3 | Definições e Acrônimos | 2 |
| 2 | Descrição Geral | 2 |
| 2.1 | Perspectiva do Produto | 2 |
| 2.2 | Classes e Características dos Usuários | 2 |
| 3 | Requisitos Funcionais (RF) | 3 |
| 4 | Requisitos Não-Funcionais (RNF) | 3 |
| 5 | Regras de Negócio (RN) | 4 |
| 6 | Modelo de Dados (Lógico) | 4 |
| 6.1 | Entidade: Paciente | 4 |
| 6.2 | Entidade: Agendamento | 5 |
| 6.3 | Entidade: Financeiro | 5 |

1 Introdução

1.1 Propósito

O propósito deste documento é definir os requisitos funcionais e não-funcionais para o sistema **OdontoSys Lite v3**. O sistema visa gerenciar as atividades básicas de uma clínica odontológica de pequeno porte, focando na organização de pacientes, agenda e fluxo de caixa.

1.2 Escopo do Produto

O **OdontoSys Lite v3** é uma aplicação web (Single Page Application - SPA) que opera localmente no navegador do usuário. O sistema permite o cadastro de pacientes, agendamento de consultas e lançamentos financeiros, com controle de acesso diferenciado para Dentistas e Secretárias. Nesta versão, a persistência de dados é realizada via *LocalStorage*, garantindo funcionamento offline e portabilidade imediata sem necessidade de infraestrutura de servidor complexa.

1.3 Definições e Acrônimos

- **SPA:** Single Page Application (Aplicação de Página Única).
- **MVC:** Model-View-Controller (Padrão de Arquitetura de Software).
- **DAO:** Data Access Object (Objeto de Acesso a Dados).
- **CRUD:** Create, Read, Update, Delete (Operações básicas de dados).

2 Descrição Geral

2.1 Perspectiva do Produto

O sistema é independente e autossuficiente. Foi arquitetado em camadas (View, Controller, Service, DAO) para facilitar a manutenção e permitir a futura migração do armazenamento local para uma API REST sem impactar a interface do usuário ou as regras de negócio.

2.2 Classes e Características dos Usuários

- **Administrador (Dentista):** Possui acesso total ao sistema, incluindo visualização e edição de registros financeiros e históricos clínicos sensíveis.
- **Secretária:** Possui acesso restrito. Pode gerenciar pacientes (dados cadastrais) e agenda, mas não tem permissão para visualizar o módulo financeiro ou alterar configurações críticas.

3 Requisitos Funcionais (RF)

Esta seção lista as funcionalidades que o sistema deve fornecer.

| Identificador | Descrição |
|---------------|---|
| RF01 | Autenticação de Usuário O sistema deve permitir que o usuário realize login informando usuário e senha. O sistema deve identificar o perfil (Admin/Secretária) no momento do login. |
| RF02 | Manter Pacientes (CRUD) O sistema deve permitir cadastrar, listar, editar e excluir pacientes. Devem ser armazenados: Nome, CPF, Telefone e Histórico Clínico. |
| RF03 | Auditoria de Registros Ao salvar ou editar um paciente, o sistema deve registrar automaticamente qual usuário realizou a ação e a data/hora da modificação (Rastreabilidade). |
| RF04 | Gerenciar Agenda O sistema deve permitir agendar consultas vinculando um paciente, um dentista, data e hora. Deve permitir o cancelamento de agendamentos. |
| RF05 | Controle Financeiro O sistema deve permitir lançar receitas (procedimentos), informando descrição, valor, forma de pagamento e status (Pago/Pendente). |
| RF06 | Dashboard (Visão Geral) O sistema deve exibir uma tela inicial com indicadores chave: Total de Pacientes, Consultas do Dia e Total em Caixa (apenas recebidos). |
| RF07 | Controle de Acesso (ACL) O sistema deve ocultar o menu e impedir o acesso à funcionalidade "Financeiro" caso o usuário logado pertença ao perfil "Secretária". |

4 Requisitos Não-Funcionais (RNF)

Requisitos de qualidade, desempenho e restrições técnicas.

| Identificador | Descrição |
|---------------|--|
| RNF01 | Arquitetura em Camadas O código deve ser estritamente separado nas camadas: <i>View</i> (Interface), <i>Controller</i> (Orquestração), <i>Service</i> (Regras de Negócio) e <i>DAO</i> (Persistência). |
| RNF02 | Portabilidade O sistema deve funcionar nos navegadores modernos (Chrome, Firefox, Edge) sem necessidade de instalação de plugins. |

| Identificador | Descrição |
|---------------|---|
| RNF03 | Persistência Local Para esta versão, os dados devem ser persistidos utilizando a API <i>LocalStorage</i> do navegador, garantindo que os dados não sejam perdidos ao recarregar a página. |
| RNF04 | Usabilidade A interface deve fornecer feedback visual (Notificações/Toasts) para todas as ações de sucesso ou erro (ex: "Paciente salvo com sucesso", "Login inválido"). |
| RNF05 | Interface Responsiva O layout deve se adaptar para visualização em desktops e notebooks, utilizando CSS Flexbox/Grid. |

5 Regras de Negócio (RN)

Políticas e restrições que governam o comportamento do sistema.

- **RN01 - Obrigatoriedade de Campos:** Não é permitido cadastrar pacientes sem informar Nome e CPF.
- **RN02 - Cálculo de Caixa:** O valor total exibido no Dashboard e no Fluxo de Caixa deve considerar apenas os lançamentos com status "Pago". Lançamentos "Pendentes" não somam ao total.
- **RN03 - Ordenação da Agenda:** A lista de agendamentos deve ser exibida sempre ordenada cronologicamente (data e hora), independentemente da ordem de cadastro.
- **RN04 - Edição de Histórico:** Apenas usuários com perfil "Admin" podem visualizar o campo de Histórico Clínico completo na listagem (visualização restrita).

6 Modelo de Dados (Lógico)

Estrutura dos objetos JSON armazenados no *LocalStorage*.

6.1 Entidade: Paciente

```
{
  "nome": "String",
  "cpf": "String",
  "tel": "String",
  "hist": "String (Histórico Clínico)",
  "editadoPor": "String (Nome do Usuário)",
  "dataEdicao": "ISO Date String"
}
```

6.2 Entidade: Agendamento

```
{  
    "nome": "String (Cópia do nome do paciente)",  
    "dentista": "String",  
    "data": "YYYY-MM-DD",  
    "hora": "HH:MM",  
    "status": "String (ex: 'Agendado')"  
}
```

6.3 Entidade: Financeiro

```
{  
    "desc": "String (Descrição do procedimento)",  
    "valor": "Number (Float)",  
    "data": "YYYY-MM-DD",  
    "forma": "String (Dinheiro, Cartão, etc)",  
    "status": "String (Pago/Pendente)"  
}
```