

# RELATÓRIO TÉCNICO DE DESENVOLVIMENTO

**Projeto:** Portal do Paciente - FUNAD

**Desenvolvedor:** Dahyany Kelly Almeida Souza

**Data:** 30/01/2026

## 1. OBJETIVO

O presente documento descreve a implementação do sistema web "Portal do Paciente", desenvolvido como parte do processo seletivo. O objetivo principal foi criar uma aplicação funcional para gestão de consultas médicas (CRUD), garantindo a integridade dos dados, segurança básica e uma interface amigável.

## 2. ARQUITETURA E TECNOLOGIAS

O projeto foi desenvolvido utilizando a arquitetura cliente-servidor, sem o uso de frameworks pesados, para demonstrar conhecimento sólido nas linguagens nativas:

- **Back-end:** PHP 7.4+ (Utilizando **PDO** para abstração e segurança na conexão com o banco).
- **Banco de Dados:** MySQL/MariaDB (Modelagem relacional).
- **Front-end:** HTML5 (Semântico), CSS3 (Responsivo) e **JavaScript Vanilla** (Interatividade).
- **Servidor Local:** Apache (via pacote XAMPP).

## 3. DETALHAMENTO DAS OPERAÇÕES CRUD

O sistema cobre o ciclo completo de vida da informação (Create, Read, Update, Delete) aplicado aos agendamentos e usuários:

### CREATE (Criação)

- **Cadastro de Usuários:** Permite que novos pacientes se registrem no sistema. Inclui validação para evitar campos vazios e duplicidade de e-mail/CPF.
- **Agendamento:** O usuário logado pode solicitar uma nova consulta. O sistema captura automaticamente o ID do usuário da sessão (`$_SESSION`), vinculando o agendamento ao paciente correto sem necessidade de digitação manual.

## **READ (Leitura)**

- **Login e Autenticação:** O sistema verifica as credenciais no banco de dados.
- **Painel de Controle:** Após o login, o usuário visualiza uma lista (tabela) contendo apenas os seus próprios agendamentos. Foi utilizada uma consulta SQL com cláusula WHERE id\_usuario = ? para garantir a privacidade dos dados.

## **UPDATE (Atualização)**

- **Edição de Agendamento:** Permite alterar a data, hora ou especialidade de uma consulta já marcada.
- **Preenchimento Automático:** Ao clicar em editar, o formulário já vem preenchido com os dados atuais do banco, facilitando a visualização para o usuário antes da alteração.

## **DELETE (Exclusão)**

- **Cancelamento:** Permite remover um agendamento do banco de dados.
- **Segurança de UX:** Foi implementada uma confirmação via **JavaScript** (confirm()) antes da exclusão efetiva, prevenindo cliques acidentais.

## **4. DIFERENCIAIS TÉCNICOS E UX**

Além dos requisitos funcionais, foram implementadas melhorias de experiência do usuário:

1. **JavaScript Client-Side:** Script personalizado para **máscara de CPF** (insere pontos e traço automaticamente) e validação de campos no front-end.
2. **Segurança de Sessão:** Todas as páginas internas (painel.php, agendar.php, etc.) possuem verificação de sessão ativa. Se um usuário tentar acessar direto pela URL sem logar, é redirecionado para a tela de login.
3. **Relatórios de Impressão:** Funcionalidade que gera uma versão limpa da tabela de agendamentos e aciona a impressora do sistema (window.print()), removendo menus e cores desnecessárias para economizar tinta.

## **5. MODELAGEM DE DADOS**

O banco funad\_db foi estruturado em duas tabelas principais com relacionamento:

- **Tabela usuarios:** Armazena dados cadastrais (Chave Primária: id).

- **Tabela agendamentos:** Armazena as consultas. Possui uma **Chave Estrangeira** (**id\_usuario**) que referencia a tabela de usuários, garantindo a integridade referencial.

## 6. CONCLUSÃO

O sistema atende a todos os requisitos propostos, entregando uma solução leve, funcional e de fácil manutenção. O código foi comentado e organizado para facilitar a leitura por outros desenvolvedores.