

Documento de Requisitos – Sistema re-maid

Versão: 1.0

Data: 10 de Outubro de 2025

1. Contexto e Objetivos de Negócio

- **Visão geral:** O projeto visa desenvolver o sistema **re-maid**, uma plataforma de chamados (ticketing system) para otimizar e organizar o suporte técnico interno. A plataforma permitirá a abertura, gerenciamento e acompanhamento de solicitações de forma centralizada, melhorando a comunicação entre colaboradores, técnicos e a administração.
- **Objetivos mensuráveis:**
 - Reduzir em 50% o tempo médio para a primeira resposta a um chamado.
 - Aumentar em 40% a taxa de resolução de chamados no primeiro contato.
 - Centralizar 100% das solicitações de suporte, eliminando o uso de canais não oficiais (e-mail, mensagens instantâneas).

2. Escopo

- **Incluído:**
 - Gestão completa de chamados (criação, atribuição, edição, consulta e exclusão).
 - Gerenciamento de usuários (colaboradores e técnicos) e dispositivos.
 - Sistema de permissões baseado em três perfis: Usuário Padrão, Técnico e Administrador.
 - Consulta de histórico de chamados e dispositivos por usuário.
- **Limitações conhecidas:** A versão inicial não incluirá um aplicativo móvel nem notificações por e-mail/push, focando-se na interface web.

3. Stakeholders e Usuários

- **Stakeholders:** Diretoria da empresa, Departamento de TI, Gerentes de área, Colaboradores.
- **Usuários/Personas:**
 - **Usuário Padrão (Colaborador):** Precisa de uma forma rápida e simples para reportar problemas (ex: "minha impressora não funciona") e acompanhar o andamento da solução.
 - **Técnico:** Precisa visualizar os chamados atribuídos a ele, atualizar o status e registrar as soluções aplicadas.
 - **Administrador:** Precisa de uma visão completa do sistema para gerenciar contas de usuários, dispositivos e ter controle total sobre todos os chamados.

4. Requisitos Funcionais (RF)

- **RF-001 - Autenticação de Usuários:** O sistema deve permitir que usuários (padrão, técnico, admin) façam login de forma segura.
- **RF-002 - Abrir Chamado (Usuário Padrão):** O usuário deve poder criar um novo chamado, descrevendo o problema e associando-o, se possível, a um dispositivo.
- **RF-003 - Consultar Meus Chamados (Usuário Padrão):** O usuário deve poder visualizar uma lista de todos os chamados que abriu e seus respectivos status.
- **RF-004 - Consultar Histórico de Dispositivos (Usuário Padrão):** O usuário deve poder ver um histórico dos dispositivos que já utilizou.
- **RF-005 - Consultar Chamados Atribuídos (Técnico):** O técnico deve poder visualizar uma dashboard com todos os chamados que estão sob sua responsabilidade.
- **RF-006 - Editar Chamado (Técnico/Admin):** O técnico deve poder alterar o status, adicionar comentários, reatribuir e fechar um chamado. O admin tem permissão de edição total.
- **RF-007 - Gerenciamento de Colaboradores (Admin):** O admin deve poder criar, ler, editar e deletar contas de usuários padrão.
- **RF-008 - Gerenciamento de Técnicos (Admin):** O admin deve poder criar, ler, editar e deletar contas de técnicos.
- **RF-009 - Gerenciamento de Dispositivos (Admin):** O admin deve poder criar, ler, editar e deletar registros de dispositivos da empresa.
- **RF-010 - Gerenciamento Total de Chamados (Admin):** O admin deve poder criar, ler, editar e deletar qualquer chamado no sistema.

5. Requisitos Não Funcionais (RNF)

- **RNF-Desempenho:** As páginas principais do sistema (dashboards, listas) devem carregar em menos de 3 segundos. A abertura de um chamado deve ser processada em menos de 2 segundos.
- **RNF-Confabilidade:** Disponibilidade mínima de 99.5% do sistema em horário comercial.
- **RNF-Segurança:** O tráfego de dados deve ser protegido com criptografia (HTTPS/TLS). Senhas devem ser armazenadas de forma segura (hashed).
- **RNF-Usabilidade:** A interface deve ser intuitiva, permitindo que um novo usuário abra um chamado sem necessidade de treinamento.

6. Requisitos de Dados (RD)

- **Fontes:** Os dados serão gerados primariamente dentro do próprio sistema (dados de chamados, usuários, dispositivos).
- **Qualidade esperada:** Consistência e integridade dos dados garantidas pela estrutura do banco de dados (chaves primárias, estrangeiras e constraints).
- **Privacidade:** Dados de um usuário só podem ser consultados por ele mesmo, por técnicos designados para seus chamados ou por administradores, em conformidade com as políticas de privacidade internas.

7. Arquitetura e Integrações

- **Linguagens:** PHP (Back-end) e JavaScript (Front-end).
- **Banco de Dados:** PostgreSQL versão 17.
- **Sistema Operacional:** Linux.
- **Hospedagem:** Servidor dedicado/VPS na Hetzner.
- **Segurança e CDN:** Proteção contra ataques (DDoS, WAF) e cache de conteúdo estático gerenciado pelo Cloudflare.
- **Cache e Performance:** Varnish para cache de requisições HTTP e Redis para cache de objetos e sessões.
- **Protocolo de Acesso:** Acesso ao servidor será realizado exclusivamente via SSH.
- **Controle de Versão:** O código-fonte será versionado e armazenado no Git.

8. Riscos, Assunções e Dependências

- **Riscos:** Atraso na configuração da infraestrutura de hospedagem; dificuldade na integração entre front-end e back-end; adoção lenta da plataforma pelos colaboradores.
- **Assunções:** A equipe possui o conhecimento técnico necessário em PHP, JavaScript e PostgreSQL; a infraestrutura da Hetzner atenderá aos requisitos de desempenho.
- **Dependências:** Disponibilidade dos serviços da Hetzner e Cloudflare.

9. Critérios de Aceitação e Plano de Validação

- Um usuário padrão consegue se logar, abrir um chamado e consultar seu histórico com sucesso.
- Um técnico consegue visualizar os chamados atribuídos e alterar seus status.
- Um administrador consegue criar um novo usuário, um novo dispositivo e visualizar todos os chamados do sistema.
- Os testes de CRUD (Criar, Ler, Editar, Deletar) para cada módulo funcionam conforme especificado nos Requisitos Funcionais.

10. Cronograma e Marcos (Milestones)

- **Fase 1 (Mês 1-2):** Configuração da infraestrutura (Hetzner, Cloudflare) e desenvolvimento do back-end (API, lógica de negócio).
- **Fase 2 (Mês 3-4):** Desenvolvimento do front-end e integração com o back-end.
- **Fase 3 (Mês 5):** Fase de testes internos e correções.
- **Fase 4 (Mês 6):** Implantação (Rollout) para um departamento piloto.
- **Fase 5 (Mês 7):** Lançamento para toda a empresa.