# **ZELOS**

# **Documento de Requisitos do Sistema (DRS)**

## **1. Introdução**

### **1.1 Objetivo**

Este documento tem como objetivo descrever os requisitos necessários para o desenvolvimento do sistema **Zelos**, que será responsável por controlar os chamados de manutenção e suporte dentro da escola SENAI Armando de Arruda Pereira, utilizando o número de patrimônio dos itens da escola como identificador principal para os chamados.

### **1.2 Escopo**

O sistema **Zelos** visa gerenciar a solicitação, acompanhamento e resolução de chamados relacionados aos itens da escola, como móveis, equipamentos e outros recursos, identificados pelo número de patrimônio. O sistema permitirá a criação, atribuição, atualização e fechamento de chamados, fornecendo visibilidade aos responsáveis.

### **1.3 Definições, Acrônimos e Abreviações**

* **Patrimônio**: Código único que identifica um item específico da escola.
* **Chamado**: Solicitação de manutenção ou suporte relacionado a um item da escola.
* **Administrador**: Usuário com privilégios de gerenciamento e configuração do sistema.

## **2. Visão Geral do Sistema**

### **2.1 Funcionalidades**

A seguir, estão as funcionalidades principais que o sistema deverá suportar:

* **Criação de Chamados**: Registrar novos chamados a partir do número de patrimônio.
* **Acompanhamento de Chamados**: Acompanhar o status de chamados criados, com possibilidade de atualizações.
* **Fechamento de Chamados**: Encerrar chamados quando a solicitação for atendida.
* **Relatórios**: Gerar relatórios com o histórico de chamados e manutenção de itens.

### **2.2 Usuários e Permissões**

O sistema será utilizado pelos seguintes tipos de usuários:

* **Administrador**: Pode criar, atualizar, atribuir, fechar chamados e gerar relatórios.
* **Usuário Comum**: Pode criar chamados e visualizar o status de chamados associados ao número de patrimônio que gerou a solicitação.

### **2.3 Arquitetura do Sistema**

O sistema será baseado na arquitetura **cliente-servidor**, utilizando as seguintes tecnologias:

* **Frontend**: Framework javascript (React.JS, Next.js, etc.) para, com design responsivo.
* **Backend**: Node.js, com Express.
* **Banco de Dados**: MySQL.

## **3. Requisitos Funcionais**

### **3.1 Criação de Chamados**

* **Descrição**: O sistema deve permitir a criação de novos chamados informando o número de patrimônio do item ou, caso não seja possível identificar pelo patrimônio, se faz necessário uma descrição específica.
* **Entrada**: Número de patrimônio ou descrição de item, descrição do problema e tipo de chamado.
* **Saída**: Chamado criado com ID único e status inicial aberto.
* **Regras de Negócio**: Cada número de patrimônio pode gerar múltiplos chamados, mas um mesmo número de patrimônio não pode estar associado a dois chamados de mesmo tipo simultaneamente. Caso o usuário tente gerar um chamado de mesmo tipo ele deverá ser alertado.

### **3.2 Acompanhamento de Chamados**

* **Descrição**: O usuário poderá acompanhar os chamados que gerou, verificando o status e as atualizações feitas.
* **Entrada**: ID do chamado ou número de patrimônio.
* **Saída**: Status atual do chamado (ex: "Em andamento", "Concluído").
* **Regras de Negócio**: O status do chamado só pode ser alterado por um administrador ou técnico responsável.

### **3.3 Fechamento de Chamados**

* **Descrição**: O administrador ou técnico responsável poderá fechar um chamado quando o problema for resolvido.
* **Entrada**: ID do chamado, resolução do problema.
* **Saída**: Chamado fechado com data de resolução.
* **Regras de Negócio**: O chamado só pode ser fechado se o status estiver como "Em andamento" ou "Aguardando aprovação".

**4. Requisitos Não Funcionais**

### **4.1 Performance**

* O sistema deverá ser capaz de processar até **500 chamados** simultaneamente.
* O tempo de resposta para criação e atualização de chamados será de no máximo **2 segundos**.

### **4.2 Segurança**

* O sistema deve garantir a **autenticação** segura dos usuários, utilizando **JWT** para sessões de usuário.
* Todos os dados sensíveis, como informações pessoais e detalhes de chamados, devem ser **criptografados**.

### **4.3 Usabilidade**

* O sistema deve ser intuitivo e de fácil navegação para usuários com pouca experiência técnica.
* O layout deve ser responsivo, com adaptações para dispositivos móveis e desktops.

### **4.4 Disponibilidade**

* O sistema deve ter uma disponibilidade de **99%** durante o horário de funcionamento da escola (**segunda a sexta-feira, das 8h às 18h**).

## **5. Requisitos Técnicos**

### **5.1 Tecnologia**

O sistema será desenvolvido utilizando as seguintes tecnologias:

* **Frontend**: React.js
* **Backend**: Node.js (Express)
* **Banco de Dados**: PostgreSQL

### **5.2 Integração com Outros Sistemas**

* O sistema pode integrar-se a um sistema de inventário para buscar os números de patrimônio e suas descrições.

### **5.3 Desempenho e Escalabilidade**

* O sistema deverá ser capaz de escalabilidade horizontal para lidar com picos de uso, garantindo a continuidade da operação durante o horário de maior tráfego.

## **6. Restrições**

### **6.1 Tecnológicas**

* O sistema deverá ser desenvolvido para **plataforma web** e deverá ser compatível com os navegadores **Google Chrome** e **Mozilla Firefox**.

### **6.2 Orçamentárias**

* O orçamento para o desenvolvimento do sistema é de **R$ 15.000**, com prazo de entrega até **6 meses após a assinatura do contrato**.

### **6.3 Legais**

* O sistema deve estar em conformidade com a **LGPD** (Lei Geral de Proteção de Dados Pessoais).

## **7. Critérios de Aceitação**

### **7.1 Critérios de Aceitação de Funcionalidades**

Cada funcionalidade será considerada aceita se:

* A criação, acompanhamento e fechamento de chamados estiverem funcionando corretamente.
* O sistema apresentar desempenho adequado mesmo com o volume de dados previstos.
* A segurança e integridade dos dados forem mantidas.

### **7.2 Critérios de Aceitação do Sistema**

* O sistema será considerado aceito quando atender aos requisitos descritos neste documento e todas as funcionalidades estiverem operacionais sem falhas críticas.

## **8. Anexos**

### **8.1 Diagramas**

* **Diagrama de Arquitetura**: [Incluir diagrama aqui]
* **Diagrama de Casos de Uso**: [Incluir diagrama aqui]

### **8.2 Glossário**

* **Patrimônio**: Código único que identifica um item da escola.
* **Chamado**: Solicitação de manutenção ou suporte relacionado ao patrimônio.