

# Documento de Arquitetura - Sistema de Coleta de Resíduos em Cidades Inteligentes

João Gabriel Tavares Felix Monteiro  
João Victor Rosa Couto e Silva  
Karlla Loane Santos Lima  
Murilo Henrique de Sousa Freua

**Goiânia**  
**2024**

# Histórico da Revisão

Data	Versão	Descrição	Autor
31 de outubro de 2024	1.0	Versão inicial	Murilo, Karlla, João Gabriel, João Victor

# Sumário

<b>1. Arquitetura do Sistema.....</b>	<b>3</b>
1.1 Propósito e Contexto.....	3
1.2 Stakeholders Principais e Preocupações.....	3
1.3 Visão Arquitetural do Sistema.....	4
1.3.1 Descrição das Camadas.....	4
1.3.1 Comunicação entre as Camadas.....	6
<b>2. Arquitetura do Software.....</b>	<b>7</b>
2.1 Visões Arquiteturais.....	7
2.1.1 Visão Lógica.....	7
2.1.2 Visão de Desenvolvimento.....	8
2.1.3 Visão Física.....	8
2.2 Decisões Arquiteturais.....	8
2.3 Interfaces e Protocolos.....	10
2.4 Requisitos de Qualidade.....	10

# 1. Arquitetura do Sistema

## 1.1 Propósito e Contexto

O propósito do sistema é otimizar a coleta de resíduos em ambientes urbanos inteligentes, promovendo uma operação eficiente, sustentável e adaptada às necessidades dinâmicas da cidade. Para isso, o sistema utiliza sensores instalados em lixeiras que monitoram continuamente o nível de resíduos, enviando essas informações para uma camada de processamento centralizada. Com base nesses dados, são geradas rotas otimizadas para os agentes de coleta, que visualizam os trajetos em dispositivos móveis.

No contexto de uma cidade inteligente, esse sistema se integra ao ecossistema urbano ao possibilitar a coleta em tempo real de dados de resíduos, alinhando-se com os Objetivos de Desenvolvimento Sustentável (ODS) 11 - Cidades e Comunidades Sustentáveis e 13 - Ação Contra a Mudança Global do Clima. Esse contexto também permite que a solução interaja com outros sistemas de gestão urbana, como monitoramento ambiental, caso seja necessário. Dessa forma, o sistema não apenas contribui para a eficiência operacional, mas também apoia a governança sustentável e a qualidade de vida dos cidadãos.

## 1.2 Stakeholders Principais e Preocupações

- **Agentes de Coleta:** necessitam de rotas ajustadas automaticamente para que a coleta seja otimizada em tempo real e requerem uma interface móvel intuitiva.
- **Supervisores:** desejam uma visão consolidada e histórica das coletas para análise de desempenho e tendências.
- **Gestão Municipal:** esperam que o sistema contribua com métricas de economia e indicadores de sustentabilidade.
- **Cidadãos:** esperam que a coleta seja realizada de forma eficaz para manter a limpeza da cidade.

## **1.3 Visão Arquitetural do Sistema**

### **1.3.1 Descrição das Camadas**

O sistema é composto por três camadas principais, conforme ilustrado na figura 1:

#### **A. Camada de Dispositivos IoT**

Esta camada compreende os sensores instalados nas lixeiras e os dispositivos móveis dos agentes de coleta. Os sensores monitoram continuamente o nível de resíduos e enviam essas informações em tempo real, enquanto os dispositivos móveis permitem que os agentes visualizem e sigam as rotas otimizadas.

##### **Componentes:**

- a. Sensores nas lixeiras
- b. Dispositivos móveis dos agentes

#### **B. Camada de Processamento**

Responsável por processar e armazenar os dados recebidos dos dispositivos IoT. Nessa camada, serviços de processamento analisam os dados dos sensores e aplicam algoritmos de otimização de rotas. Além disso, essa camada inclui funcionalidades de analytics e geração de relatórios, armazenamento de dados históricos e regras de negócio que determinam a priorização das rotas.

##### **Componentes:**

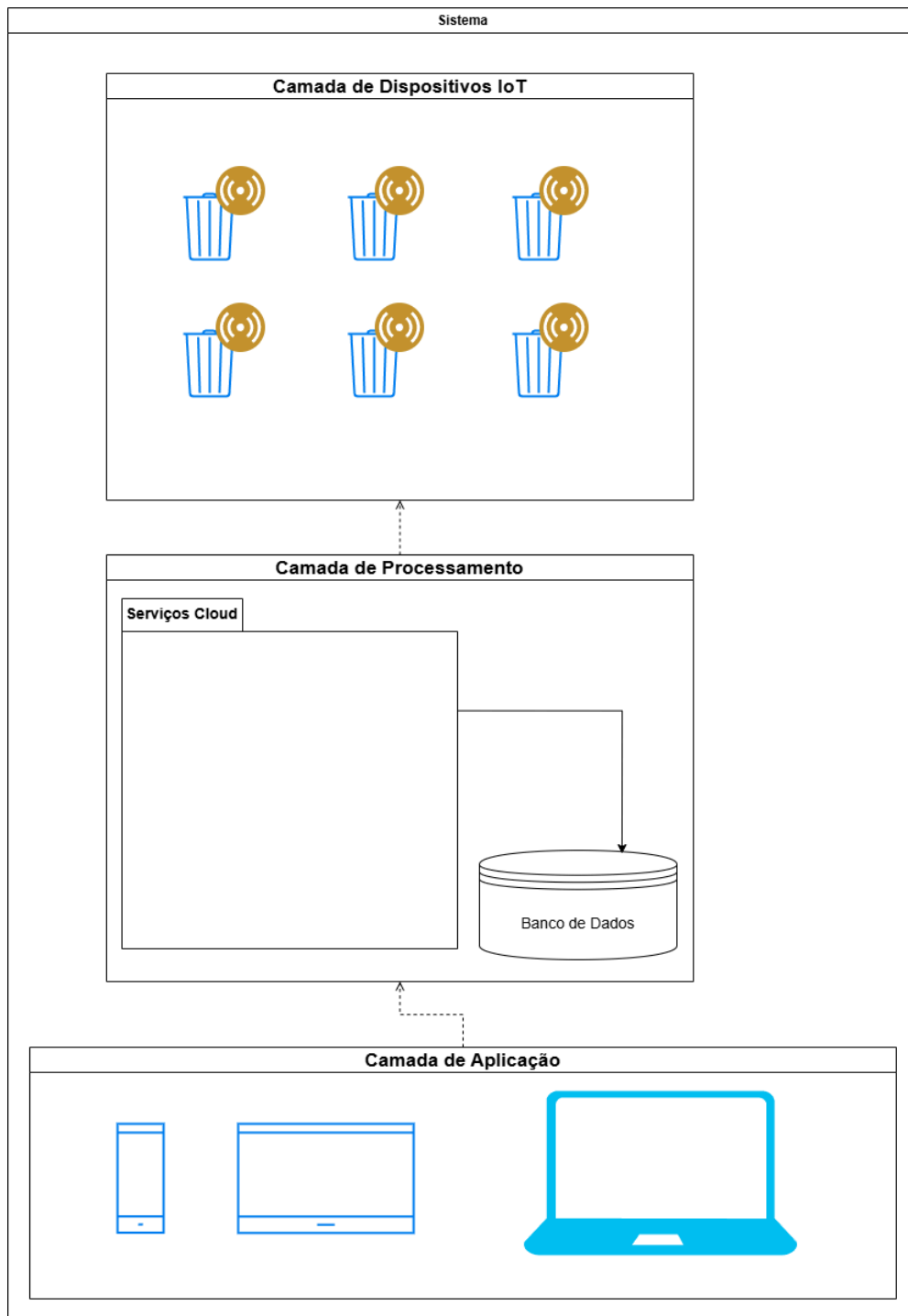
- a. Serviços de processamento de dados em tempo real
- b. Algoritmos de otimização de rotas
- c. Analytics de dados e relatórios
- d. Regras de negócio
- e. Persistência de Dados

### **C. Camada de Aplicação**

Nessa camada, estão os aplicativos e interfaces de usuário que permitem o acesso e a interação com o sistema. O aplicativo móvel, usado pelos agentes, exibe rotas em tempo real e notifica os agentes sobre ajustes dinâmicos. O portal web permite que supervisores monitorem a operação, acompanhem as rotas, analisem tendências e visualizem relatórios de desempenho.

#### **Componentes:**

- a. Aplicativo móvel para agentes
  - i. Smartphones
  - ii. Tablets
- b. Portal web para supervisores
  - i. Desktop/Laptop



*Figura 1: Arquitetura Geral do Sistema*

### 1.3.1 Comunicação entre as Camadas

A comunicação entre as camadas é centralizada e coordenada pela Camada de Processamento. Os sensores da Camada de Dispositivos IoT enviam dados para o sistema de processamento usando o protocolo MQTT. A Camada de

Processamento, então, realiza análises e cálculos com base nesses dados e, uma vez processados, envia rotas otimizadas e notificações diretamente para o aplicativo móvel dos agentes na Camada de Aplicação, utilizando APIs REST para comunicação síncrona e WebSocket para atualizações em tempo real. O portal web na Camada de Aplicação também se conecta à Camada de Processamento via APIs REST para visualizar dados históricos, tendências e relatórios.

## **2. Arquitetura do Software**

### **2.1 Visões Arquiteturais**

#### **2.1.1 Visão Lógica**

##### **Padrão Arquitetural Principal: Serviços**

O sistema adota uma arquitetura baseada em serviços, onde cada componente funcional é isolado em um serviço específico para garantir a escalabilidade, a manutenção simplificada e a independência de cada módulo.

##### **Componentes Principais:**

- **Serviço de Gerenciamento de Sensores:** responsável por coletar dados dos sensores instalados nas lixeiras e manter o monitoramento em tempo real do nível de resíduos. Esse serviço gerencia a conexão com os dispositivos IoT usando o protocolo MQTT, processa as informações de nível de resíduos e as encaminha para o serviço de processamento e otimização de rotas.
- **Serviço de Otimização/ Descobrimiento de Rotas:** utiliza os dados de preenchimento das lixeiras e a localização dos agentes de coleta para calcular rotas otimizadas. Esse serviço envia as rotas para o aplicativo dos agentes e atualiza as rotas em tempo real quando necessário.
- **Serviço de Relatórios e Analytics:** gera relatórios detalhados sobre desempenho da coleta e gráficos de tendência de produção de lixo em bairros e regiões.



- **Serviço de Autenticação e Autorização:** controle de acesso ao sistema.

### 2.1.2 Visão de Desenvolvimento

Padrões e Frameworks:

- Simulações: Node-RED
- Backend: Java Spring + PostgreSQL + AWS
- Frontend: React
- Mobile: Flutter + SQLite

### 2.1.3 Visão Física

- Implantação em nuvem com containers
- Edge computing para processamento local dos dados dos sensores
- CDN (Content Delivery Network) para distribuição de conteúdo estático

## 2.2 Decisões Arquiteturais

As decisões arquiteturais do sistema foram fundamentadas para garantir escalabilidade, desempenho em tempo real e eficiência operacional, considerando os requisitos específicos de uma solução de coleta de resíduos inteligente e urbana. Cada escolha reflete um balanço entre benefícios e possíveis trade-offs, com foco na sustentabilidade e flexibilidade do sistema.

### 1. Uso de Arquitetura Orientada a Serviços

- Justificativa: uma arquitetura orientada a serviços proporciona alta escalabilidade e manutenção simplificada. Cada funcionalidade principal do sistema, como o gerenciamento de sensores, otimização de rotas, e analytics, é implementada como um serviço independente, possibilitando que esses componentes possam ser escalados individualmente conforme a demanda

- Trade-off: maior custo de infraestrutura e necessidade de containerização para isolamento e portabilidade.

## 2. Processamento em Tempo Real

- Justificativa: o sistema requer respostas imediatas para otimização de rotas, visando adaptar os trajetos dos agentes de coleta em tempo real com base nos dados recebidos dos sensores nas lixeiras.
- Trade-off: maior consumo de recursos e complexidade, exigindo uma infraestrutura robusta.

## 3. Edge Computing

- Justificativa: reduz a latência ao processar os dados dos sensores de forma local, próxima ao ponto de coleta, antes de enviá-los para a nuvem. Isso permite uma resposta mais rápida às mudanças de estado das lixeiras e reduz o consumo de banda.
- Trade-off: necessidade de gerenciamento de dispositivos edge.

## 4. Justificativas das Tecnologias Seleccionadas

- Node-RED para Simulações: Node-RED permite simular a comunicação de sensores de forma visual e modular, ideal para criar e testar fluxos de dados IoT de maneira rápida e flexível.
- Java Spring e PostgreSQL para o Backend: Java Spring foi escolhido pela robustez e extensibilidade, com suporte a serviços web e microserviços, enquanto o PostgreSQL oferece eficiência e escalabilidade em bancos de dados relacionais.
- AWS para a Nuvem: A infraestrutura em nuvem da AWS proporciona escalabilidade, resiliência e suporte para containerização com ferramentas integradas como Elastic Container Service (ECS) e Lambda, ideais para orquestração e edge computing.
- React para o Frontend: React é uma biblioteca robusta para interfaces dinâmicas, proporcionando uma experiência de usuário fluida e responsiva no portal web dos supervisores.

- Flutter e SQLite para o Mobile: Flutter é multiplataforma, o que simplifica o desenvolvimento e manutenção do aplicativo para agentes de coleta, enquanto o SQLite proporciona um banco de dados local leve, útil para armazenar dados temporariamente em dispositivos móveis durante a coleta.

## **2.3 Interfaces e Protocolos**

### **1. Interfaces Externas**

- API do Google Maps para visualização e ajuste de rotas
- Interfaces com sistemas municipais existentes

### **2. Protocolos de Comunicação**

- MQTT para comunicação com sensores
- REST para APIs de serviço
- WebSocket para atualizações em tempo real

## **2.4 Requisitos de Qualidade**

### **1. Desempenho**

- Tempo de resposta inferior a 2 segundos para geração de rotas
- Processamento em tempo real de dados dos sensores

### **2. Disponibilidade**

- 99.9% de uptime para serviços críticos

### **3. Segurança**

- Uso de JWT
- Criptografia end-to-end
- Auditoria de acessos

### **4. Escalabilidade**

- Horizontal para serviços de processamento
- Vertical para bancos de dados