



UNIVERSIDADE FEDERAL DO CEARÁ
CAMPUS RUSSAS
CURSO DE GRADUAÇÃO EM ENGENHARIA DE SOFTWARE

BRAYAN ANDERSON VIEIRA SANTOS

SISTEMA DE GERENCIAMENTO DE ATIVOS DE ZELADORIA URBANA

RUSSAS

2026

BRAYAN ANDERSON VIEIRA SANTOS

SISTEMA DE GERENCIAMENTO DE ATIVOS DE ZELADORIA URBANA

Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Engenharia de software da Universidade Federal do Ceará, como requisito parcial à obtenção do grau de bacharel em Engenharia de software.

Orientador: Prof. Dr. Mayrton Dias de Queiroz

RUSSAS

2026

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Sistema de Bibliotecas
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

S233s Santos, Brayan Anderson Vieira.
Sistema de Gerenciamento de Ativos de Zeladoria Urbana / Brayan Anderson Vieira Santos. – 2026.
44 f. : il. color.

Trabalho de Conclusão de Curso (graduação) – Universidade Federal do Ceará, Campus de Russas,
Curso de Engenharia de Software, Russas, 2026.
Orientação: Prof. Dr. Mayrton Dias de Queiroz.

1. Zeladoria Urbana. 2. Gestão Urbana. 3. Ativos Públicos. 4. Administração Pública. 5. Sistemas de
Informação. I. Título.

CDD 005.1

BRAYAN ANDERSON VIEIRA SANTOS

SISTEMA DE GERENCIAMENTO DE ATIVOS DE ZELADORIA URBANA

Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Engenharia de software da Universidade Federal do Ceará, como requisito parcial à obtenção do grau de bacharel em Engenharia de software.

Aprovada em: 26/01/2026

BANCA EXAMINADORA

Prof. Dr. Mayrton Dias de Queiroz (Orientador)
Universidade Federal do Ceará (UFC)

Prof. Dr. Cenez Araújo de Rezende
Universidade Federal do Ceará (UFC)

Prof. Dr. Pablo Luiz Braga Soares
Universidade Federal do Ceará (UFC)

RESUMO

Um dos grandes desafios da administração pública é a gestão eficiente de ativos públicos de uso comum no contexto de zeladoria urbana. Dificuldades relacionadas à identificação, manutenção, monitoramento e priorização das demandas relacionadas aos bens públicos impactam diretamente a qualidade do serviço prestado à população e a distribuição eficiente de recursos públicos. Diante deste cenário, o objetivo deste trabalho consiste em identificar uma alternativa capaz de auxiliar os profissionais da gestão pública no gerenciamento dos ativos nas cidades, dessa forma, foi proposto um *framework* para auxiliar na criação, verificação e conclusão das ordens de serviço dos ativos. O sistema permite o registro, geolocalização, classificação, gerenciamento de ordens de serviço em tempo real de ativos como pavimento, logradouros, árvores, galerias subterrâneas, bueiros, hidrantes, entre outros ativos quaisquer que impactem a mobilidade urbana, serviços essenciais como distribuição de água e luz, prevenção e minimização de desastres ambientais e eventos climáticos excepcionais. Utilizando tecnologias atuais como banco de dados relacionais, interfaces web e integração com mapas interativos, a solução facilita a interação das informações entre os gestores, população e equipes de campo. O artefato incorpora funcionalidades para geração de ordens de serviço, controle de *status* das manutenções, emissão de relatórios e análise de desempenho e auditoria pública. A proposta visa manter a transparência, reduzir o tempo de respostas às demandas de zeladoria urbana e otimizar a sustentabilidade operacional da gestão pública. Os resultados obtidos através do desenvolvimento de um módulo e pesquisas referentes a sistemas similares demonstram viabilidade na execução da operação do sistema, promovendo melhorias na qualidade de vida da população, em aspectos como segurança, mobilidade e saúde.

Palavras-chave: zeladoria urbana; gestão urbana; ativos públicos; administração pública; sistemas de informação.

ABSTRACT

One of the major challenges in public administration is the efficient management of public assets intended for common use within the context of urban caretaking. Difficulties related to the identification, maintenance, monitoring, and prioritization of public asset demands directly impact the quality of services provided to the population and the efficient allocation of public resources. This work proposes the development of a software system for the management of urban assets, focusing on municipal public caretaking in small cities. The system enables the registration, geolocation, classification, and real-time management of service orders for assets such as pavement, public roads, trees, underground galleries, storm drains, fire hydrants, among other assets that impact urban mobility, essential services such as water and electricity distribution, and the prevention and mitigation of environmental disasters and exceptional weather events. By using modern technologies such as relational databases, web interfaces, and integration with interactive maps, the solution facilitates the exchange of information between managers, the population, and field teams. Additionally, it incorporates functionalities for generating service orders, tracking maintenance status, issuing reports, and conducting performance analysis and public audits. The proposal aims to ensure transparency, reduce response times to urban caretaking demands, and optimize the operational sustainability of public management. The results obtained through simulated tests and research on similar systems demonstrate the feasibility of the system's operation, promoting improvements in quality of life for the population in areas such as safety, mobility, and health.

Keywords: urban maintenance; urban management; public assets; public administration; information systems.

LISTA DE FIGURAS

Figura 1 – Resultados da Questão de Pesquisa 1.	18
Figura 2 – Resultados da Questão de Pesquisa 2.	19
Figura 3 – Passos realizados durante o desenvolvimento deste trabalho.	21
Figura 4 – Quantidade de artigos obtidos após a pesquisa nas bases de dados, no <i>download</i> e na aplicação dos critérios de inclusão e exclusão.	21
Figura 5 – Fluxograma do planejamento do <i>framework</i> e suas funcionalidades principais.	23
Figura 6 – Aplicativo do cidadão para registro de ocorrências de ordens de serviço.	24
Figura 7 – Visualização das ordens de serviço e suas geolocalizações.	25
Figura 8 – Fragmento C - Novas ordens de serviço.	26
Figura 9 – Fragmento D - Detalhes nova OS, Aprovar/Cancelar.	26
Figura 10 – Fragmento E - Rotas de inspeção.	27
Figura 11 – Fragmento F - Aplicativo do Inspetor.	28
Figura 12 – Fragmento G - Ordem de Serviço (OS) Aprovada/Andamento.	29
Figura 13 – Fragmento H - Aplicativo da Equipe de Campo.	29
Figura 14 – Protótipo da página de edição de telas.	35
Figura 15 – Lista de componentes na página de edição de telas.	36
Figura 16 – Componente de mapa dentro da área de edição na página de edição de telas.	36
Figura 17 – Aba Geral das propriedades do componente na edição de telas.	37
Figura 18 – Aba Dados das propriedades do componente na edição de telas.	38
Figura 19 – Aba Comportamento das propriedades do componente na edição de telas.	38
Figura 20 – Página de visão geral obtida através da utilização do <i>framework</i> proposto.	40

LISTA DE TABELAS

Tabela 1 – Quantidade de artigos em cada base de dados.	17
---	----

LISTA DE ABREVIATURAS E SIGLAS

API	Interface de Programação de Aplicações
CAGECE	Companhia de Água e Esgoto do Ceará
ENEL	Empresa de energia elétrica presente no Brasil
GPS	Sistema de Posicionamento Global
HTML	<i>HyperText Markup Language</i>
IoT	Internet das Coisas
OS	Ordem de Serviço
REST	Transferência de Estado Representacional
RESTful	Estilo arquitetural baseado em REST

SUMÁRIO

1	INTRODUÇÃO	10
1.1	Contextualização	10
1.2	Objetivo geral	11
1.2.1	<i>Objetivos específicos</i>	11
1.3	Justificativa	11
1.4	Organização do trabalho	12
2	FUNDAMENTAÇÃO TEÓRICA	13
2.1	Gestão em cidades	13
2.2	Geolocalização e mapas interativos	14
3	TRABALHOS RELACIONADOS	15
3.1	Contextualização	15
3.2	Planejamento	15
3.2.1	<i>Definição das Questões de Pesquisa</i>	15
3.2.2	<i>Definição da string de busca</i>	16
3.2.3	<i>Definição das bases de dados</i>	16
3.2.4	<i>Definição dos critérios de inclusão e exclusão</i>	16
3.3	Condução	17
3.4	Resultados	17
4	METODOLOGIA	21
4.1	Passo 1 - Levantamento dos trabalhos relacionados	21
4.2	Passo 2 - Planejamento do <i>framework</i>	22
4.3	Passo 3 - Implementação do <i>framework</i>	30
5	RESULTADOS	31
5.1	Configuração de banco de dados	31
5.2	Editor de telas (<i>Screen Editor</i>)	35
5.3	<i>Scripts</i> de automação	38
5.4	Resultado gerado pelo <i>framework</i>	39
6	CONSIDERAÇÕES E TRABALHOS FUTUROS	41
	REFERÊNCIAS	42

1 INTRODUÇÃO

Nesta seção será apresentada a contextualização deste trabalho descrevendo o cenário de análise e os desafios, o objetivo geral e os objetivos específicos e a justificativa que embasa a busca por estratégias para o desenvolvimento de uma solução para o problema desta pesquisa.

1.1 Contextualização

A zeladoria urbana é responsável por garantir a conservação dos ativos públicos que compõe o espaço urbano da cidade, como: vias, calçadas, praças, iluminação pública, bueiros, sinalização, galerias, monumentos, entre outros. Em cidades de pequeno porte, geralmente localizadas no interior dos estados brasileiros, inclusive do nordeste, essa atividade enfrenta sérios desafios estruturais e operacionais. O controle desses ativos se dá, geralmente, de forma manual ou por meio de planilhas, dificultando o monitoramento contínuo, a priorização de demandas e transparência na execução das ordens de serviço e no recebimento das solicitações de serviço.

A ausência de ferramentas digitais que possibilitam a visualização e a geolocalização dos ativos, compromete a eficiência do trabalho das equipes de campo e dos gestores públicos. Tal deficiência impacta diretamente na qualidade dos serviços prestados a população, gera desperdício de recursos públicos e torna ineficiente o processo para zelar os bens públicos, tornando o tempo entre, a identificação e a solução de um problema, cada vez mais distantes, principalmente os problemas de maior incidência, como: buracos em vias, lâmpadas queimadas e equipamentos danificados.

Neste contexto, o avanço das tecnologias de informação e comunicação oferece uma gama de soluções para transformar a maneira como as cidades atendem à população. Ferramentas como o Google Maps, Interface de Programação de Aplicações (API) e OpenStreetMap permitem a integração de dados e localização em tempo real em aplicações para web, tornando possível localizar, mapear e acompanhar a situação dos ativos pela cidade. Com esses avanços tecnológicos, não somente é possível integrar os ativos da cidade em um sistema, mas também integrar os cidadãos à cidade, conectando as pessoas ao seu papel de cidadão, promovendo uma comunicação mais efetiva entre os cidadãos e o poder público com atuação mais ativa e optativa da população. Essas soluções também promovem a transparência entre a população, a gestão pública local e a gestão pública ao nível estadual e nacional. Transparência essa que pode

beneficiar as cidades de menor porte ao solicitar recursos dos governos estadual e federal e ter meios de comprovar em quais setores e serviços está sendo investido, facilitando a prestação de contas aos governos superiores e ao povo que é o "cliente" dessas melhorias.

Um dos conceitos alinhados a esse contexto é o de *smart cities*, ou cidades inteligentes, que surgem como uma alternativa moderna para enfrentar os problemas da urbanização, mesmo em realidades de menor escala (Weiss *et al.*, 2015). Embora o termo seja comumente associado a grandes cidades, seus princípios, como: conectividade, uso eficiente de dados, participação cidadã e automação de processos, podem ser aplicados adaptadamente em cidades de pequeno porte. A implantação de um sistema inteligente é um passo importante para gerenciar os ativos públicos, pois contribui para melhoria da tomada de decisão por parte dos gestores, otimizar os recursos financeiros e oferecer melhores serviços à população.

1.2 Objetivo geral

O objetivo deste trabalho consiste em identificar uma alternativa capaz de auxiliar os profissionais da gestão pública no gerenciamento dos ativos nas cidades, dessa forma, propõe-se um *framework* para auxiliar na criação, verificação e conclusão das ordens de serviço relacionados aos ativos públicos sob controle da zeladoria municipal.

1.2.1 Objetivos específicos

Assim, é possível listar os seguintes objetivos específicos:

- Obter o levantamento dos principais trabalhos alinhados ao problema;
- Identificar as principais etapas relacionadas à criação e gerenciamento das ordens de serviço dos ativos para obter uma proposta de sistema de zeladoria urbana;
- Obter um produto mínimo viável a partir do *framework*, que permita visualizar o fluxo e ciclo de vida completo das Ordens de serviço.

1.3 Justificativa

A partir da contextualização, é possível identificar os desafios da gestão pública na manutenção de ativos públicos, sobre tudo em cidades de pequeno porte e com orçamento limitado. A ausência de um sistema, gera maior tempo de execução de serviço, dificulta o planejamento, desperdiça recursos, gera falhas de comunicação, baixa qualidade do serviço

prestado, falta de histórico das intervenções realizadas e dificuldade de prestação de contas por parte das prefeituras.

Portanto, faz-se necessário a implementação de um sistema informativo capaz de lidar com gestão eficiente de ativos e desenvolvimento de funcionalidades que sejam capazes de suprir esses aspectos apontados. Integrando o processo de realização de serviço a um sistema mais robusto que possibilite a melhoria da zeladoria pública.

1.4 Organização do trabalho

Este trabalho está estruturado em seis capítulos, sendo este o primeiro com elementos pré textuais e introdutórios do trabalho. O capítulo dois trata da fundamentação teórica, reunindo elementos conceituais como: zeladoria urbana, gestão de ativos públicos, cidades inteligentes, geolocalização e mapas interativos, e por último, participação cidadã e transparência pública, todos os elementos explicados e contextualizados. O capítulo três reúne os trabalhos relacionados a este trabalho e o planejamento da revisão de literatura, definindo as questões de pesquisa, *string* de busca inserida nos engenhos de busca, bases de dados utilizadas e os critérios de inclusão ou exclusão dos trabalhos levantados, como a revisão foi conduzida e os resultados com a análise dos trabalhos primários. No capítulo quatro é apresentada a metodologia que possui três passos detalhados, são eles: levantamento dos trabalhos relacionados, planejamento do *framework* a ser desenvolvido e a implementação do *framework* proposto. O capítulo cinco relata os resultados do desenvolvimento deste trabalho. O capítulo seis com as considerações e trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo serão apresentados os principais conceitos alinhados ao projeto e o desenvolvimento do *framework* proposto, abordando os assuntos e conceitos utilizados atualmente para soluções de melhoria de bem-estar da população, como: zeladoria urbana, gestão de ativos públicos, cidades inteligentes, tecnologias de geolocalização, participação cidadã e transparência pública. Nas próximas seções aborda-se cada um dos conceitos relacionados as habilidades de gestão mencionadas.

2.1 Gestão em cidades

Cidades inteligentes: (do inglês, *smart cities*) é definida como um ambiente urbano que utilizam as ferramentas de comunicação e informação para melhorar a infraestrutura urbana, a utilização inteligente dos recursos, a mobilidade, a governança e os serviços públicos que tem como principal beneficiado o cidadão (Weiss *et al.*, 2015). Ainda que o conceito de cidades inteligentes seja amplamente utilizado em grandes centros urbanos e grandes metrópoles, é possível realizar sua adaptação a pequenos municípios brasileiros. O uso de tecnologias em tempo real, automação dos processos, e a participação da população possibilitam a adaptação e a melhoria da qualidade da gestão pública local (Santiago, 2023).

Zeladoria urbana: aborda um conjunto de atividades que visam manter boas condições de uso dos ambientes públicos para a população, realizar serviços essenciais de manutenção e conservação, tais como: manutenção de ruas, poda de árvores, limpeza de vias, conservação de praças e calçadas. Em cidades de pequeno porte, especialmente no interior dos estados do Nordeste brasileiro, é possível encontrar desafios, como: limitações operacionais e de infraestrutura (Santos *et al.*, 2019). Os processos nesses contextos costumam não ser automatizados e sistematizados, o que compromete seriamente o planejamento, priorização das demandas e transparência das ações dos gestores e o possível mau uso de verba pública.

Gestão de ativos públicos: são recursos, direitos e serviços, que são acessíveis e disponíveis para todos os membros de uma sociedade. Com diversas finalidades, como: infraestrutura, serviços essenciais, bens comuns, patrimônio cultural e recursos naturais. Os ativos públicos, quando geridos, permitem obter o controle, inventário, manutenção e planejamento de serviços mais eficientes. O gerenciamento eficaz desses ativos demanda um sistema de informação que seja capaz de permitir o acompanhamento em tempo real, com dados precisos

de localização, conservação e ações realizadas. Soluções informatizadas de gestão de ativos reduzem erros operacionais, previne retrabalhos e promovem melhor uso dos recursos públicos (Storck *et al.*, 2017). Isso é importante no contexto onde há limitação orçamentária, como em municípios de pequenos e médio porte.

Participação cidadã e transparência pública: Um aspecto relevante para cidades inteligentes e gestão pública eficiente é a participação do cidadão comum. Permitir que a população tenha acesso a um canal de comunicação para relatar ocorrências, acompanhar o *status* de sua solicitação e receber um retorno institucional sobre suas ocorrências fortalece a relação entre os cidadãos e a administração pública, promovendo corresponsabilidade na manutenção da cidade (Carvalho, 2024). A disponibilização de relatórios e indicadores dos serviços realizados pela zeladoria contribui para a transparência e controle social. O processo de prestação de contas, com uso de sistemas que gerem dados confiáveis e atualizados, dá suporte a processos de audiência pública.

2.2 Geolocalização e mapas interativos

A gestão de ativos requer a tecnologia de geolocalização para mapear, identificar e acompanhar, em tempo real, a posição física dos elementos da cidade, como: árvores, bueiros, galerias, logradouros, hidrantes, calçadas, monumentos, praças, entre outros. Ferramentas como *Google Maps API* e *OpenStreetMap* têm a capacidade de integração de mapas interativos em sistema web, utilizando elementos gráficos e visuais, viabilizando experiência visual e funcional tanto para os gestores quanto para os cidadãos (Oliveira *et al.*, 2020). Essas ferramentas facilitam o planejamento de rotas, inspeção de ocorrências e a emissão de relatórios por área geográfica.

3 TRABALHOS RELACIONADOS

Neste capítulo serão descritas as fases realizadas durante a revisão da literatura do trabalho.

3.1 Contextualização

Diante do objetivo do trabalho, foi necessário a realização de uma revisão sistemática com o intuito de identificar os principais trabalhos relacionados com essa pesquisa. Esta etapa da pesquisa foi organizada em três fases: Planejamento, Condução e Resultados que serão descritas nas próximas seções.

3.2 Planejamento

Segundo Kitchenham *et al.* (2009), o Planejamento é a fase de definição de cada passo do protocolo, que consiste em: definição das Questões de Pesquisa; definição da *string* de busca; definição das bases de dados e definição dos critérios de inclusão e exclusão. Sendo assim, nas próximas seções, serão descritos os passos desse protocolo.

3.2.1 Definição das Questões de Pesquisa

Inicialmente, são definidas as Questões de Pesquisa (QPs) que serão respondidas durante a fase de condução.

PQ1 - Qual a estratégia adotada no trabalho?

- Criação de um *framework*
- Simulação
- Análise conceitual de algum tema relacionado às cidades inteligentes
- Análise de dados
- Abordagem com internet das coisas

QP2 - Qual(is) o(s) eixo(s) adotado(s) no trabalho?

- Economia
- Educação
- Governança
- Meio ambiente

- Mobilidade
- Segurança
- Saúde
- Cultura
- Infraestrutura
- Tecnologia

3.2.2 Definição da string de busca

Após a definição das questões de pesquisas, foi definida a seguinte *string* de busca:

"cidades inteligentes" AND "framework" AND "serviços" AND "centro urbano"

3.2.3 Definição das bases de dados

De posse da *string* de busca, foram selecionadas as bases de dados, conforme mostra o Quadro 1:

Quadro 1 – Bases de dados selecionadas

Base de dados	Link
Google acadêmico	https://scholar.google.com.br/
Periódicos da capes	https://www.periodicos.capes.gov.br
SBC-OpenLib	https://sol.sbc.org.br

Fonte: Elaborado pela autor (2025).

3.2.4 Definição dos critérios de inclusão e exclusão

Para concluir essa fase do planejamento, foram definidos os critérios de inclusão e exclusão que todos os artigos deverão satisfazer. Os critérios definidos são:

Critérios de inclusão:

- Estudos completos publicados em revistas ou conferências alinhados ao problema de pesquisa;
- Estudos teóricos ou experimentais com o objetivo de apresentar conceitos para o entendimento da área;
- Acessível eletronicamente e ter sido publicado no período de 2015 a 2025.

Critérios de exclusão:

- Estudos que não estejam relacionados ao problema de pesquisa;
- Estudos que não respondem a nenhuma das questões de pesquisa;
- Artigos duplicados, ou seja, aqueles encontrados em mais de uma base de dados;
- Artigos convidados, tutoriais, relatórios técnicos que não passam pelo critério de avaliação das conferências ou revistas;
- Estudos não disponíveis para *download*.

3.3 Condução

Na fase de condução, foi colocado em prática o planejamento definido na seção de Planejamento, dessa forma, a *string* de busca foi inserida em cada base de dados, respeitando as especificações de cada base. De posse dos resultados da busca, foi realizado o *download* dos estudos disponíveis. Em seguida, os trabalhos passaram pelos critérios de inclusão e exclusão. Na Tabela 1 é possível identificar a quantidade de trabalhos em cada base durante a busca, a realização do *download* e a aplicação dos critérios de inclusão e exclusão.

Tabela 1 – Quantidade de artigos em cada base de dados.

Base de Dados	Resultados da busca	<i>Download</i>	Inclusão/Exclusão
Google acadêmico	97	73	5
Periódicos da capes	6	6	1
Biblioteca Digital da Sociedade Brasileira de Computação	11	9	5
Total	114	88	11

Fonte: elaborada pelo autor (2025).

3.4 Resultados

Nesta seção serão descritos os resultados obtidos das duas questões de pesquisa. Na Figura 1, é possível observar a quantidade de trabalhos em cada alternativa da questão de pesquisa 1 que investigou as estratégias adotadas e o escopo envolvido nos trabalhos primários. Entre as estratégias identificadas, destaca-se a criação de *frameworks*, observada em dois estudos: o trabalho de Storck *et al.* (2017) que desenvolveu um *framework* que utiliza mineração de dados para otimizar as futuras redes 5G em cidades inteligentes, e o trabalho de Dias *et al.* (2019), que propõe um *framework front-end* para a criação de painéis de cidades inteligentes, focando em um padrão de *design* e interface web com estruturas reutilizáveis.

Na alternativa sobre a abordagem da estratégia de simulação, foi possível identificar dois trabalhos que remetem a essa estratégia, o primeiro trabalho foi de Pinto (2025), que propõe a criação de uma plataforma de testes para treinar veículos autônomos, realizando simulações no contexto do software criado. O segundo trabalho de Castro *et al.* (2020), que utiliza um software de terceiro para simular um cenário de estacionamento inteligente, detalhando o ambiente físico, conexão entre agentes e os artefatos.

Figura 1 – Resultados da Questão de Pesquisa 1.



Fonte: elaborada pelo autor (2025).

Em Análise conceitual de algum tema relacionado a cidades inteligentes, foi identificada em três trabalhos: o primeiro de Melonio (2021), que analisa acidentes de trânsito na cidade de São Paulo e levanta os fatores que contribuem para tal, possibilitando embasar o tipo de informação com o conceito de cidades inteligentes e a gestão de tais incidentes. O segundo trabalho foi de Costa (2023), onde é possível observar uma proposta de um anel viário utilizando sistemas inteligentes para potencializar intervenções estruturais nos ativos do sistema viário, como: ruas, estradas e avenidas. O terceiro trabalho é de Oliveira *et al.* (2020), onde o autores abordam o crescimento populacional e a necessidade de aprimorar serviços aos cidadãos sobre todo o serviço de segurança, utilizando a gestão de operações no contexto de cidades inteligentes.

Um trabalho foi categorizado na alternativa de análise de dados, a pesquisa de Carvalho (2024), apresentou uma análise de dados ambientais em cidades inteligentes utilizando um protótipo de dispositivo Internet das Coisas (IoT). Assim, o autor usou o dispositivo para coletar dados ambientais georreferenciados em tempo real e a integrar esses dados com uma base de dados e posteriormente realizar a análise dos dados colhidos.

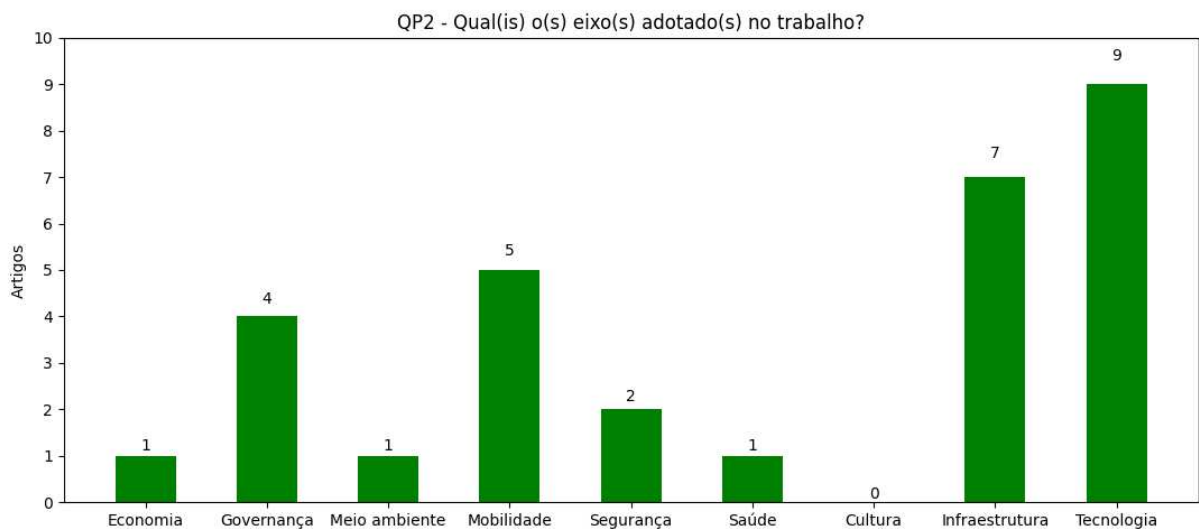
No item, abordagem com internet das coisas (IoT), é possível destacar o trabalho de Botelho *et al.* (2019), o qual propõe melhorias em um contexto de estacionamento utilizando sen-

sores e internet das coisas, que foram adotados para alocar e gerenciar vagas de estacionamento utilizando um ambiente simulado.

Os trabalhos ??) e Andrade *et al.* (2016) abordaram o desenvolvimento de sistemas: Em ??), os autores criaram um sistema para auxiliar na coleta de informações em locais de homicídios que visa modernizar o processo investigativo preliminar da polícia, substituindo métodos manuais por um sistema informatizado moderno. O segundo trabalho de Andrade *et al.* (2016), desenvolveu um sistema que reúne, em tempo real, informações de agentes públicos para combater epidemias de doenças causadas pelo mosquito *Aedes aegypti*.

A questão de pesquisa 2 (QP2), identifica os eixos das cidades inteligentes abordados pelos trabalhos primários. Sendo o eixo de mobilidade o mais frequente e cultura o menos frequente com nenhum estudo apontando sua abordagem. Os eixos de mobilidade, infraestrutura e tecnologia foram apontados pelos trabalhos de Costa (2023), Botelho *et al.* (2019), Melonio (2021) e Castro *et al.* (2020) que discutiram em seus trabalhos os problemas apontados e propuseram soluções variadas. Os eixos de governança, segurança e tecnologia apontados nos trabalhos de Dias *et al.* (2019), Oliveira *et al.* (2020) e Dias *et al.* (2019), discutiram a governança de órgãos públicos e propostas de melhorias para cada tema abordado, dois deles melhorando aspectos de segurança, sendo eles: Oliveira *et al.* (2020) e ??). No gráfico da Figura 2, é possível observar o número de ocorrências de cada eixo nos trabalhos relacionados, em que é possível em um mesmo trabalho um ou mais eixos relacionados.

Figura 2 – Resultados da Questão de Pesquisa 2.



Fonte: elaborada pelo autor (2025).

Com os trabalhos primários obtidos através da pesquisa demonstrada, foi possível

identificar as soluções de diversos segmentos utilizando estratégias variadas de abordagem, como simulações, criações de *frameworks*, análise conceitual e de dados, utilização de internet das coisas, e desenvolvimento de sistemas informativos. Considera-se para este trabalho que a pesquisa foi eficiente e demonstrou vários problemas que tiveram propostas de soluções relevantes. Dessa forma, esta revisão auxiliou na identificação de possíveis caminhos para novos estudos, como a necessidade de criação de um *framework* para auxiliar no gerenciamento de ordens de serviço de ativos das cidades.

4 METODOLOGIA

Diante do objetivo desta pesquisa, foi possível organizar a metodologia deste trabalho em três passos, conforme mostra a Figura 3 e a descrição nas subseções seguintes:

Figura 3 – Passos realizados durante o desenvolvimento deste trabalho.

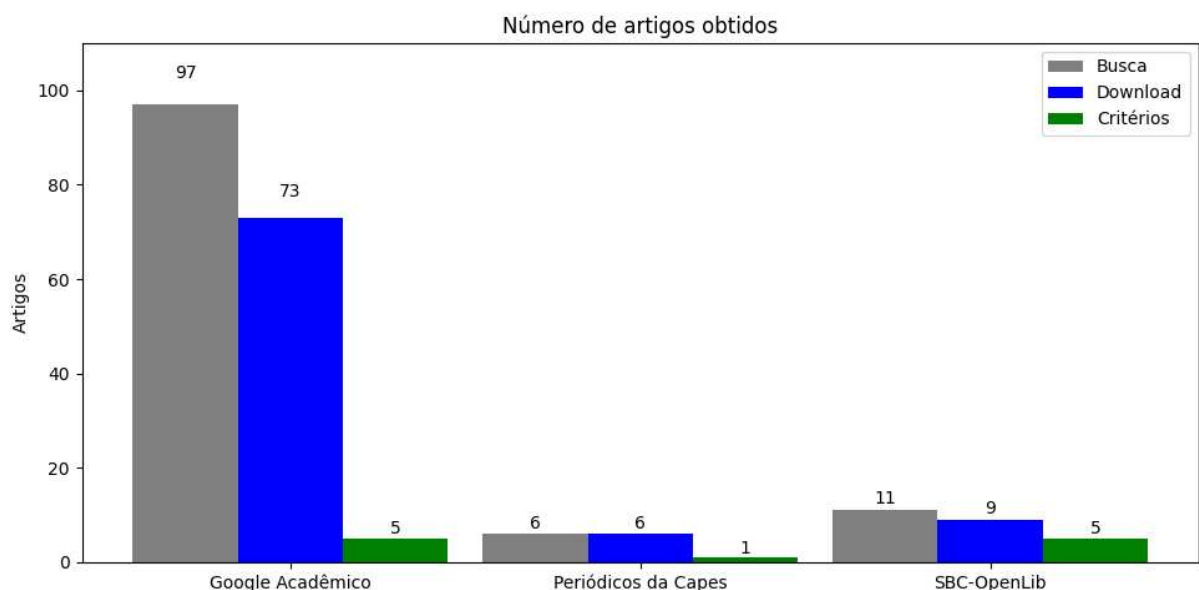


Fonte: elaborada pelo autor (2025).

4.1 Passo 1 - Levantamento dos trabalhos relacionados

O primeiro passo realizado no desenvolvimento da metodologia deste trabalho foi a revisão da literatura com o intuito de identificar os principais trabalhos alinhados a esta pesquisa. A descrição das três fases realizadas durante a revisão pode ser observada no Capítulo 3. Ao realizar a busca dos artigos nas bases de dados, foram encontrados 114 trabalhos. Logo após, foi realizado o *download* desses artigos, em seguida, os mesmos passaram pelos critérios de inclusão e exclusão, restando 11 trabalhos primários, conforme mostra a Figura 4. Assim, foi possível obter uma visão geral dos principais trabalhos relacionados a este estudo.

Figura 4 – Quantidade de artigos obtidos após a pesquisa nas bases de dados, no *download* e na aplicação dos critérios de inclusão e exclusão.



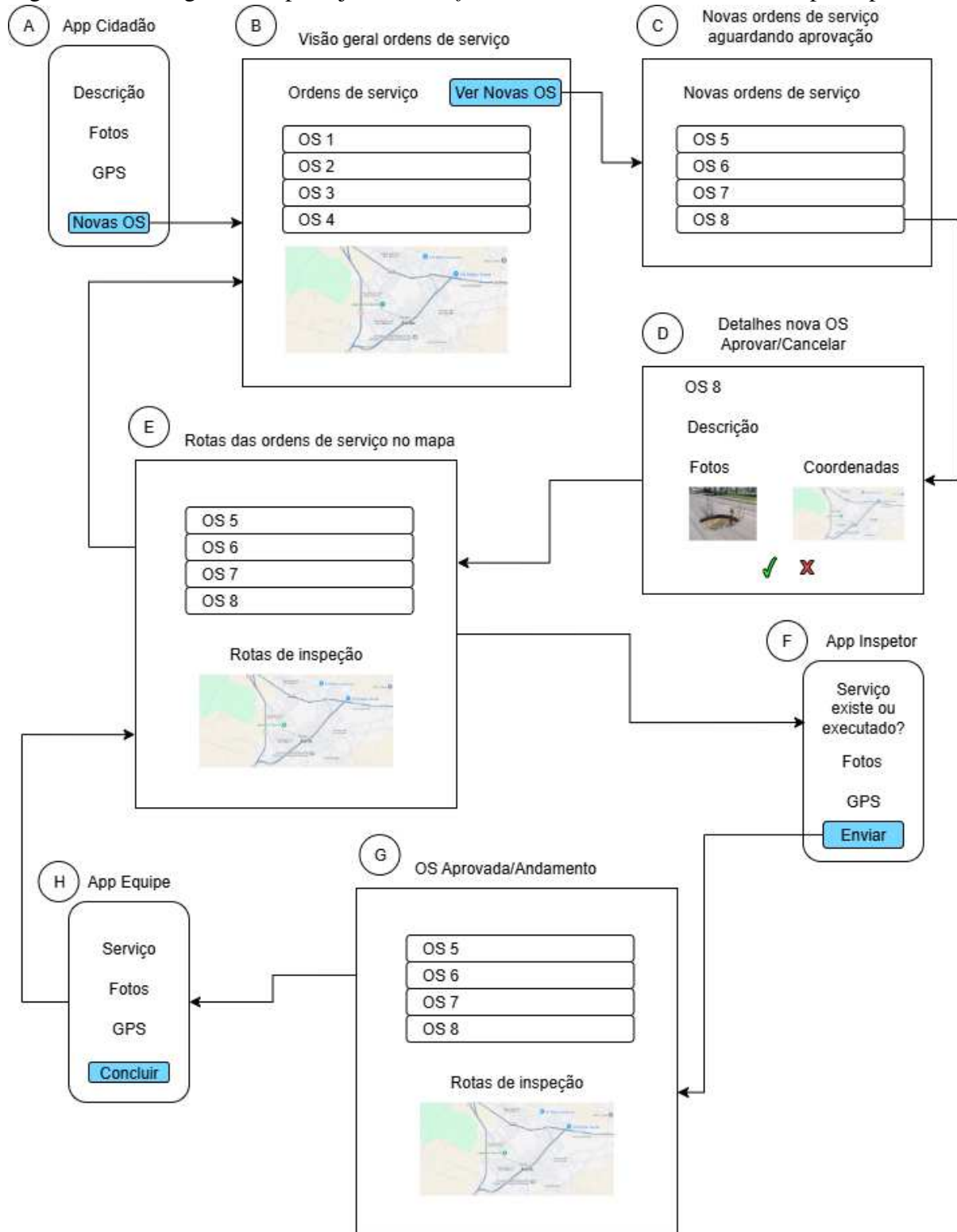
Fonte: elaborada pelo autor (2025).

4.2 Passo 2 - Planejamento do *framework*

Na segunda etapa, foi realizado o planejamento do *framework*, com o intuito de facilitar sua implementação. Para que o produto final seja capaz de cumprir os requisitos funcionais no contexto em que ele será inserido e permitindo que as regras de negócio sejam implementadas facilmente no *software*, ele foi planejado para que a inserção dessas regras sejam com pouca ou nenhuma utilização de códigos de linguagem de programação para o usuário final. Dessa forma, será desenvolvido um sistema web que será integrado aos aplicativos móveis, assim a comunicação será feita com tecnologias atuais de requisição e resposta, como interfaces de programação de aplicações API Transferência de Estado Representacional (REST) e Estilo arquitetural baseado em REST (RESTful). A ideia central é viabilizar a criação de regras de negócio, funcionalidades e objetos de forma interativa e que o usuário final consiga utilizá-lo sem precisar de conhecimentos de codificação e linguagens de programação, permitindo a customização do sistema para garantir que atenda aos seus requisitos funcionais, técnicos e não funcionais. Viabilizando a integração com produtos já consolidados no mercado, porém de forma reduzida e adaptada para o contexto de uso do produto a ser desenvolvido.

Na Figura 5, é possível observar o fluxograma das ordens de serviço desde sua origem até sua conclusão, a integração do sistema web com os aplicativos móveis sendo utilizados para movimentar e atualizar os dados das ordens de serviço geradas. Diante deste fluxograma, nas subseções seguintes serão apresentadas as descrições de cada fragmento/item apresentado na Figura 5.

Figura 5 – Fluxograma do planejamento do *framework* e suas funcionalidades principais.



Fonte: elaborada pelo autor (2025).

Fragmento A - Aplicativo do Cidadão

Ao observar um problema na cidade, como mostra a Figura 6, o cidadão poderá relatá-lo utilizando o aplicativo móvel, que gerará uma ordem de serviço (OS). Para criar uma ordem

de serviço, o usuário irá inserir a descrição do ocorrido e a foto. O Sistema de Posicionamento Global (GPS) capturará as informações de localização e enviará em formato de coordenadas para o sistema. Com a OS gerada, os gestores e fiscais responsáveis poderão visualizá-las no módulo de novas ordens de serviço na Figura 8 (Fragmento C), que permitirá visualizar seus detalhes individualmente, assim, poderão optar por aprovar ou reprovar a solicitação na Figura 9 (Fragmento D). Cidadãos, funcionários da Empresa de energia elétrica presente no Brasil (ENEL) e Companhia de Água e Esgoto do Ceará (CAGECE) e demais atores do sistema podem abrir esse tipo de requisição.

Figura 6 – Aplicativo do cidadão para registro de ocorrências de ordens de serviço.

The screenshot displays the 'App Cidadão' interface for reporting a service order. The form includes the following fields and elements:

- Nome:** Input field with placeholder 'Seu nome'.
- Email:** Input field with placeholder 'Seu email'.
- Local da ocorrência:** Input field with placeholder 'Endereço ou ponto de referência'.
- Tipo de ocorrência:** Dropdown menu with 'Árvore caída' selected.
- Descrição:** Text area with placeholder 'Descreva a ocorrência'.
- Foto:** Section containing a photo of a tilted utility pole. The photo is labeled 'Poste_de_luz_inclinado.jpg' and '6.31 MB'. Below the photo are 'Trocar' and 'Remover' buttons.
- Enviar:** A large blue button at the bottom of the form.
- Bottom Navigation Bar:** Contains icons for 'Home', 'Registrar', and 'Ocorrências'.

Fonte: elaborada pelo autor (2025).

Fragmento B - Visão geral ordens de serviço

No Fragmento B, Figura 7, os gestores, fiscais e demais responsáveis podem visualizar as ordens de serviço gerais, em qualquer fase do processo, seja um novo relato, ordens de serviço em fase de execução e concluídas, sendo necessário filtrá-las para serem visíveis na página web. É possível acessar, através de um botão, somente as novas ordens de serviço pendentes de aprovação. Nessa página, é possível observar um mapa geral com as ordens de serviço da lista colocadas em sua respectiva coordenada, facilitando a visualização geográfica e a densidade por território.

Figura 7 – Visualização das ordens de serviço e suas geolocalizações.

Ordens de Serviço - Visão Geral

ID	Classificação	Descrição	Status	Data Criação	Inspecção	Início	Término	Conclusão	Ações
1001	TAPA BURACO	Buraco na via pública na Rua Francisco Alves	NOVO	09/08/2023	-	-	-	-	Ver no mapa
1002	MANUTENÇÃO DE HIDRANTES	Hidrante com vazamento na Av. Dom Lino	EM ANDAMENTO	04/08/2023	05/08/2023	06/08/2023	-	-	Ver no mapa
1003	PODA E REMOÇÃO DE ÁRVORES	Árvore com risco de queda na Rua Cel. Alexanzito	AGUARDANDO INSPEÇÃO	11/08/2023	-	-	-	-	Ver no mapa
1004	ILUMINAÇÃO PÚBLICA	Lâmpada queimada na Praça da Matriz	NOVO	14/08/2023	-	-	-	-	Ver no mapa
1005	LIMPEZA URBANA	Acúmulo de lixo na Rua 7 de Setembro	CONCLUÍDO	31/07/2023	01/08/2023	02/08/2023	03/08/2023	03/08/2023	Ver no mapa
1006	MANUTENÇÃO DE BUEIRO	Bueiro entupido na Av. Castelo Branco	EM ANDAMENTO	07/08/2023	08/08/2023	09/08/2023	-	-	Ver no mapa
1007	SINALIZAÇÃO VIÁRIA	Placa de trânsito caída na Rua João Pessoa	NOVO	13/08/2023	-	-	-	-	Ver no mapa
1008	MANUTENÇÃO DE CALÇADA	Calçada danificada na Rua da Conceição	AGUARDANDO INSPEÇÃO	12/08/2023	-	-	-	-	Ver no mapa

Fonte: elaborada pelo autor (2025).

Fragmento C - Novas ordens de serviço

A Figura 8 representa parte do sistema web utilizado pelos gestores para visualizar a lista das novas ordens de serviço com a possibilidade de ir para os detalhes daquela ordem de serviço e posteriormente, no Fragmento D, a possibilidade de aprová-las ou reprová-las.

Figura 8 – Fragmento C - Novas ordens de serviço.

Novas OS Aguardando Aprovação

ID	Classificação	Descrição	Data Criação	Prioridade	Ações
1001	TAPA BURACO	Buraco na via pública na Rua Francisco Alves	09/08/2023	ALTA	Ver no mapa Detalhes
1003	PODA E REMOÇÃO DE ÁRVORES	Árvore com risco de queda na Rua Cel. Alexandrino	11/08/2023	MÉDIA	Ver no mapa Detalhes
1004	ILUMINAÇÃO PÚBLICA	Lâmpada queimada na Praça da Matriz	14/08/2023	BAIXA	Ver no mapa Detalhes
1007	SINALIZAÇÃO VIÁRIA	Placa de trânsito caída na Rua João Pessoa	13/08/2023	ALTA	Ver no mapa Detalhes
1008	MANUTENÇÃO DE CALÇADA	Calçada danificada na Rua da Conceição	12/08/2023	MÉDIA	Ver no mapa Detalhes



Fonte: elaborada pelo autor (2025).

Fragmento D - Detalhes nova OS, Aprovar/Cancelar

A Figura 9 representa a nova ordem de serviço, selecionada na página web do fragmento C, e suas informações mais relevantes, como: identificador, descrição, fotos e posição na cidade, contribuindo na análise por parte do responsável para aprovação ou reprovação daquela ordem de serviço.

Figura 9 – Fragmento D - Detalhes nova OS, Aprovar/Cancelar.

Detalhes da Ordem de Serviço

OS 1001 - TAPA BURACO

Descrição: Buraco na via pública na Rua Francisco Alves
 Data Criação: 09/08/2023
 Localização: Rua Francisco Alves, Centro
 Prioridade: **ALTA**
 Status: **NOVO**
 Data Inspeção: Não informada
 Data Início: Não informada
 Data Término: Não informada
 Data Conclusão: Não informada
 Fotos:

[✓ APROVAR](#) [✗ REJEITAR](#) [← VOLTAR À LISTA NOVAS OS](#)

Fonte: elaborada pelo autor (2025).

Fragmento E - Rotas das ordens de serviço no mapa

Na Figura 10 exibe no mapa as rotas pendentes de inspeção, seja de existência de serviço, em que o inspetor irá se deslocar ao local e avaliará se aquela ocorrência realmente existe, ou conclusão, em que o inspetor se deslocará para o local e verificará se a ordem de serviço foi executada (Fragmento F). Essas rotas serão designadas apenas aos inspetores e será possível que o gestor visualize as rotas dos inspetores.

Figura 10 – Fragmento E - Rotas de inspeção.



Fonte: elaborada pelo autor (2025).

Fragmento F - Aplicativo do Inspetor

O fragmento do sistema móvel responsável por exibir as informações de inspeção ao inspetor, na Figura 11, exibe a rota que deve ser feita para atingir todas as ordens de serviço designadas ao inspetor. Ao chegar no local de inspeção, é necessário que o inspetor responda a um breve questionário e envie fotos da ocorrência ou do serviço realizado, aprovando se a ocorrência existe ou, caso seja uma ordem de serviço concluída, aprovando caso a ordem de serviço tenha sido realizada com sucesso.

Figura 11 – Fragmento F - Aplicativo do Inspetor.

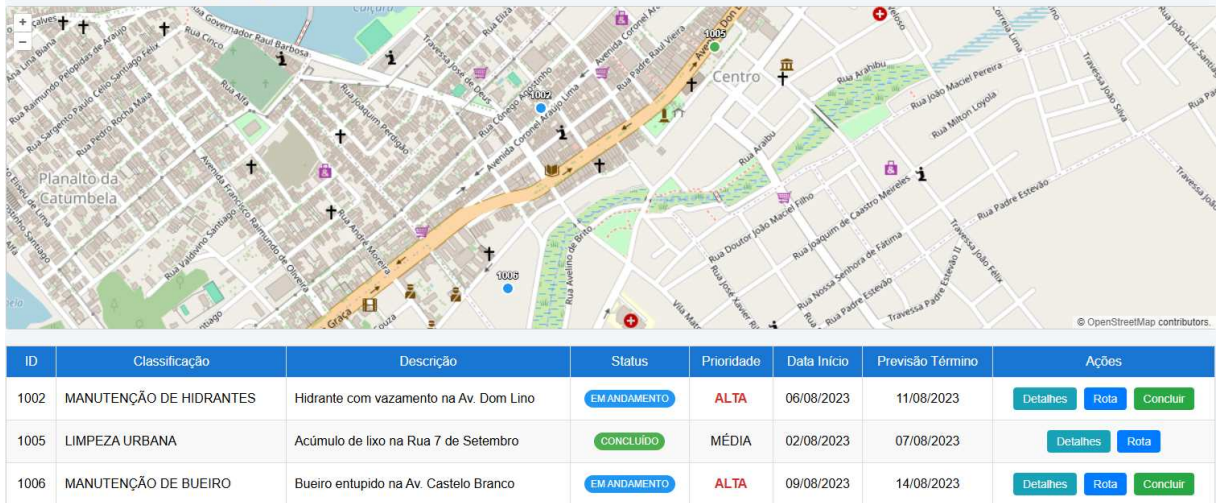


Fonte: elaborada pelo autor (2025).

Fragmento G - OS Aprovada/Andamento

Exibe as ocorrências aprovadas pelo inspetor e as ordens de serviço em execução ou aguardando execução pela equipe no mapa da Figura 12. Nessa parte, as equipes de campo podem visualizar os serviços que foram designadas para realização e cada equipe atuar em seu serviço designado, sabendo antecipadamente o que é necessário realizar antes de chegar ao local da ordem de serviço.

Figura 12 – Fragmento G - OS Aprovada/Andamento.
Ordens de Serviço Aprovadas e Em Andamento



Fonte: elaborada pelo autor (2025).

Fragmento H - Aplicativo da Equipe de Campo

A Figura 13 exibe as informações da ordem de serviço aprovada para a equipe designada e a rota até o local do serviço, o encarregado do serviço envia informações do serviço realizado como descrição, fotos e o aplicativo envia as informações de instância do GPS.

Figura 13 – Fragmento H - Aplicativo da Equipe de Campo.



Fonte: elaborada pelo autor (2025).

4.3 Passo 3 - Implementação do *framework*

A implementação do *framework* consiste-se em desenvolver módulos capazes de serem personalizados conforme a demanda e a necessidade de cada produto a ser gerado. Os módulos do sistema são Configuração de banco de dados, Editor de telas e *Scripts* de automação. Cada módulo tem responsabilidades e características únicas que sustentam um sistema com a proposta de gestão de ativos e dão suporte ao desenvolvimento de produtos para gerir os ativos de zeladoria. No Capítulo 5 os componentes desenvolvidos são descritos e exemplificados utilizando-se de testes para gerar imagens do processo de criação do *framework*.

5 RESULTADOS

A implementação do *framework* proposto tem como objetivo solucionar os desafios da gestão de ativos municipais. Para tal, é necessário utilizar técnicas de desenvolvimento baseadas em arquiteturas que suportam a configuração de regras de negócio para implantação de um processo real em um sistema informatizado. Inicialmente, pretende-se utilizar um sistema voltado à arquitetura de banco de dados, fazendo com que as classes, objetos, configurações de tela e visualização, regras de negócio, entre outros aspectos, estejam presentes no banco de dados do sistema. Ainda é necessário se aprofundar nesse tipo de desenvolvimento para implementar a ideia do projeto em um produto viável. Para isso, realizou-se o desenvolvimento de um produto mínimo viável com parte das funcionalidades do produto final.

A comunicação com o sistema ocorrerá através de requisições de API RESTful a qual suporta diversos tipos de operações com um banco de dados relacional. O intuito é que o usuário administrador do sistema consiga, com pouco esforço, criar classes no banco de dados, criar e manipular regras para essas classes e conseguir inicialmente fazer o relacionamento entre classes. Posteriormente, um módulo da aplicação será responsável por dar suporte à criação de telas e regras de exibição. A implementação da solução possui três partes fundamentais que fazem o sistema conseguir receber as regras de negócio e requisitos personalizados de cada cliente. Os três módulos nativos do sistema que integram o sistema são “Configuração de banco de dados”, “Editor de telas” e “*Scripts* de automação” que serão introduzidos e exemplificados nas próximas seções deste capítulo.

5.1 Configuração de banco de dados

O módulo nativo Configuração de banco de dados possui três submódulos, são eles o Construtor de Objetos, Instanciador de Objetos e Criador de Relacionamentos entre Objetos, são recursos nativos do sistema que são complementares entre eles e fazem parte da configuração de banco de dados. O construtor de objetos tem como função criar as tabelas físicas no banco de dados com as colunas definidas pelo usuário, o instanciador de objetos tem como objetivo popular os dados dos objetos criados no banco e o criador de relacionamentos tem como função relacionar os objetos do banco, portanto trata-se de um banco de dados relacional. Nos serviços, adota-se por convenção o padrão de linguagem inglês, tanto na criação das tabelas quanto na criação dos relacionamentos, porém os objetos podem ter seus atributos preenchidos no idioma

português, pois é o que será apresentado em tela aos usuários do sistema.

1. **Construtor de objeto (*Object Designer*):** Implementação do módulo de banco de dados, em que é possível realizar a construção e manipulação de tabelas no banco através de requisições vindas da interface gráfica, Estas tabelas são os objetos do sistema que possuem por padrão os seguintes atributos: “*name*”, “*tableName*”, “*description*” e “*attributes*”. Os atributos são, respectivamente, o nome do objeto, o nome da tabela como aparece no banco de dados, descrição do objeto, os atributos daquele objeto no formato de lista que possui nome do atributo, tipo de dado, tamanho do dado, e se é possível adicionar um dado nulo. No Código-fonte 1 é possível observar um exemplo de requisição do tipo *POST* no formato *json* para criação de um objeto denominado *WORKORDER* com os campos de objeto preenchido e três atributos dinâmicos de escolha do usuário do sistema, *wonum*, *description*, *status*. Optou-se por colocar apenas três atributos arbitrários, mas em outros cenários e a depender da necessidade, pode-se criar outros atributos seguindo o modelo e replicando para os atributos desejados.

Código-fonte 1 – JSON completo de entrada no serviços de objetos para criação de tabelas.

```

1 /POST - {{base_url}}/objects
2 Request:
3 {
4     "name": "WORKORDER",
5     "tableName": "wo",
6     "description": "Ordens de servico",
7     "attributes": [
8         { "name": "wonum", "type": "string", "size": 30, "nullable
9           ": false },
10        { "name": "description", "type": "text" },
11        { "name": "status", "type": "string", "size": 20 }
12    ]
13 }
14 Response:
15 {
16     "message": "Objeto criado com sucesso",
17     "object": {
18         "id": 6,

```

```

19     "name": "WORKORDER",
20     "table_name": "wo",
21     "description": "Ordens de servico",
22     "created_at": "2025-12-21T14:27:06.314Z"
23 }
24 }

```

Fonte: elaborado pelo autor (2026).

2. **Instâncias de objetos:** O preenchimento dos dados nos objetos do banco de dados é possível através do serviço de gravação de informações nas tabelas de objetos criadas dinamicamente, como mostrado no tópico anterior. Na "URL" percebe-se que os dados a serem gravados no banco são inseridos através da rota "[BASE URL]/records/[NOME DO OBJETO]" sendo "BASE URL" o endereço do servidor e "NOME DO OBJETO" o nome definido do objeto conforme a linha 4 do Código-fonte 1, neste caso seria *WORKORDER*. Dessa forma, não se faz necessária a criação manual de uma rota de serviço na aplicação para cada tabela existente no banco de dados. As rotas e os serviços são disponibilizados automaticamente assim que a respectiva tabela estiver presente na base de dados. Nesse contexto, basta preencher os atributos previamente definidos e realizar o envio da requisição *POST* exemplificado no Código-fonte 2.

Código-fonte 2: JSON completo de entrada no serviços de instância de dados nas tabelas criadas.

```

1 /POST - {{base_url}}/records/WORKORDER
2
3 Request:
4 {
5     "wonum": "OS4045",
6     "description": "PODA DE ARVORE",
7     "status": "NOVO"
8 }
9
10 Response:
11 {
12     "id": 1,
13     "wonum": "OS4045",
14     "description": "PODA DE ARVORE",

```

```

15     "status": "NOVO",
16     "created_at": "2025-12-21T14:32:41.350Z",
17     "updated_at": "2025-12-21T14:32:41.350Z"
18 }

```

Fonte: elaborado pelo autor (2026).

3. **Criação de relacionamentos entre objetos:** O relacionamento de dois objetos pode ser criado a partir do serviço *relationships* com os campos *sourceObject* sendo o objeto de origem e *sourceAttribute* sendo o objeto de destino, como exemplificado no Código-fonte 3.

Código-fonte 3: JSON completo de entrada no serviços de relacionamento entre as tabelas.

```

1  /POST - {{base_url}}/relationships
2
3  Request:
4  {
5     "sourceObject": "WORKORDER",
6     "sourceAttribute": "wonum",
7     "targetObject": "ASSET",
8     "targetAttribute": "assetnum",
9     "relationType": "LOOKUP"
10 }
11
12 Response:
13 {
14     "id": 1,
15     "source_object_id": 6,
16     "source_attribute": "wonum",
17     "target_object_id": 1,
18     "target_attribute": "assetnum",
19     "relation_type": "LOOKUP",
20     "created_at": "2025-12-22T13:08:59.261Z"
21 }

```

Fonte: elaborado pelo autor (2026).

5.2 Editor de telas (*Screen Editor*)

A Figura 14 apresenta um exemplo do objeto *WORKORDER*, no qual foram utilizados campos visuais baseados diretamente nos atributos definidos no banco de dados. A partir desse exemplo, observa-se a implementação do módulo de interface do usuário, responsável por permitir a construção de interfaces personalizadas com base nos objetos previamente criados. Por meio desse recurso, é possível configurar a disposição e a forma de visualização dos campos em tela, de modo que as informações sejam exibidas conforme o *layout* definido pelo usuário. O Editor de Telas (*Screen Editor*) possui uma área de edição que pode ser visualizada na Figura 14 como a parte em fundo branco, ao lado esquerdo ficam os componentes da barra lateral observa-se na Figura 15 um botão para cada componente (“*text*”, “*textarea*”, “*number*”, “*select*”, “*checkbox*” e “*date*”), eles podem ser clicados e arrastados até a área de edição para serem adicionados ao *layout* da tela.

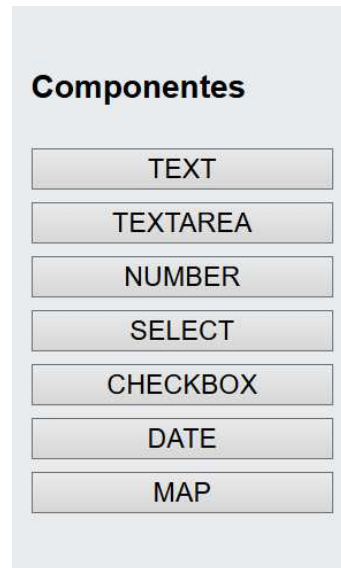
Figura 14 – Protótipo da página de edição de telas.

The screenshot shows the 'Screen Editor' interface. At the top, there's a dark header with the text 'Screen Editor'. Below it, a light blue bar contains 'Dados Gerais'. The main area is split into two columns. The left column, 'Componentes', lists various UI elements: TEXT, TEXTAREA, NUMBER, SELECT, CHECKBOX, and DATE. The right column, 'Informações Básicas', displays a form layout for a 'WORKORDER'. It includes fields for 'Número de OS' (with attribute 'worknum'), 'STATUS' (with attribute 'status'), a large text area for 'Descrição da Ordem de Serviço' (with attribute 'Description'), a 'Novo Campo' field, and a date field for 'Data de Status' (with attribute 'status_date' and format 'dd/mm/aaaa').

Fonte: elaborada pelo autor (2025).

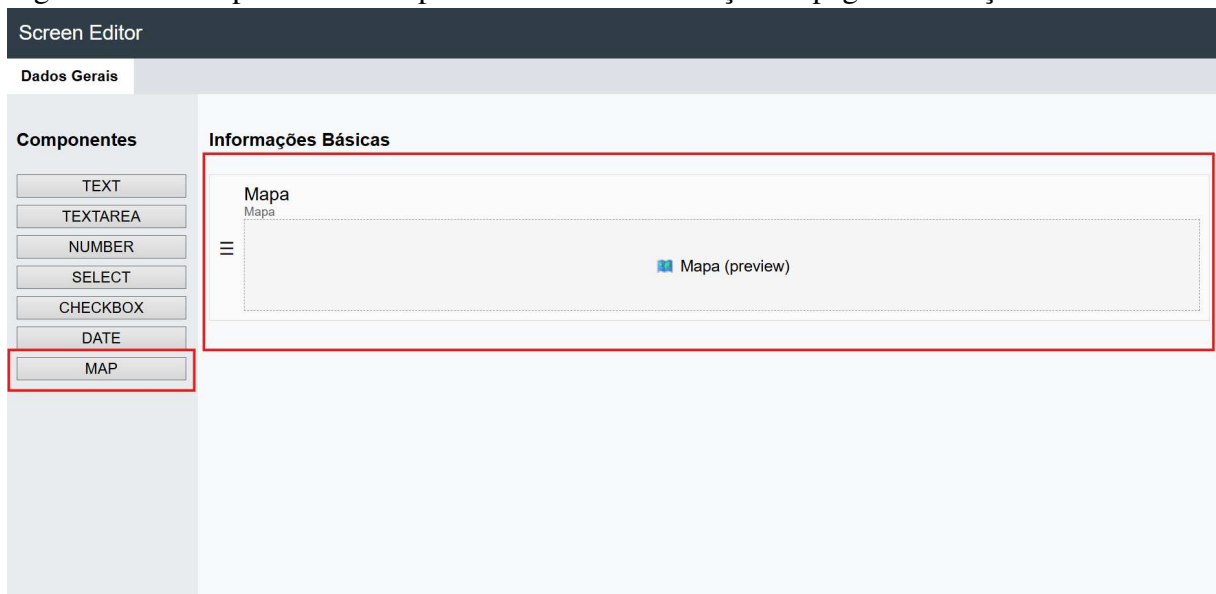
Os componentes são baseados nos componentes padrão do *HyperText Markup Language* (HTML), como textos simples, campos de textos longos, campos de entrada numeral, caixas de seleção, campo entradas de data, como exibido na Figura 16.

Figura 15 – Lista de componentes na página de edição de telas.



Fonte: elaborada pelo autor (2025).

Figura 16 – Componente de mapa dentro da área de edição na página de edição de telas.



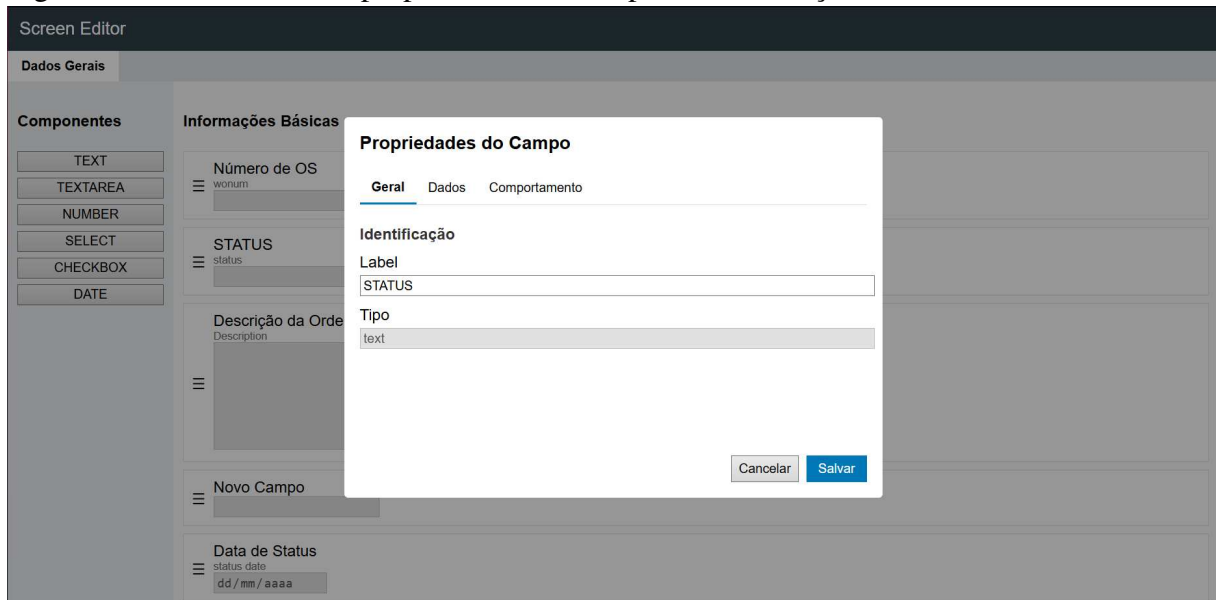
Fonte: elaborada pelo autor (2025).

É importante destacar que o componente de Mapas não é um elemento nativo do HTML, mas um componente personalizado desenvolvido no contexto deste *framework*. Seu objetivo é possibilitar a geração de mapas com posicionamentos geográficos reais, a partir dos objetos do sistema que possuam atributos de latitude e longitude previamente definidos. Este componente foi desenvolvido com a biblioteca gratuita *OpenStreetMap* utilizando *Open Layers* e seus recursos gráficos, com isso ao inserir as latitudes e longitudes do que deseja-se exibir,

cria-se pontos no mapa indicando o local exato das coordenadas apontadas.

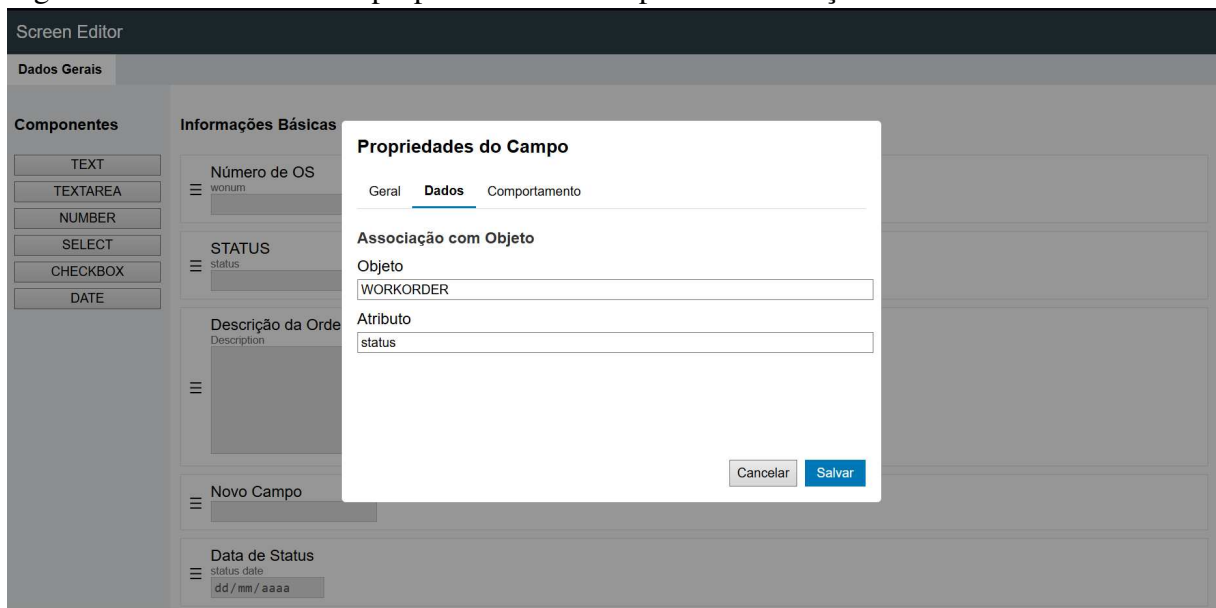
Ao clicar no componente já inserido na área de edição é possível abrir um *modal* com as propriedades do componente, sendo possível editar algumas das propriedades. Na aba “geral” (Figura 17) é possível editar o *label* do campo, que seria o nome do campo em tela e o tipo que já é incluído do componente assim que é inserido na área de edição. Na aba “dados” (Figura 18) é possível incluir o objeto e o atributo a ser exibido no campo, sendo possível visualizar também em caso de relacionamento, utilizando o nome do relacionamento com o objeto de destino, caso exista o objeto e o atributo de destino o dado é exibido no campo. Na aba “comportamento” (Figura 19) é possível determinar as propriedades de “obrigatório” que o campo seja obrigatório o preenchimento e “somente leitura” que o campo não seja possível editar após o preenchimento.

Figura 17 – Aba Geral das propriedades do componente na edição de telas.



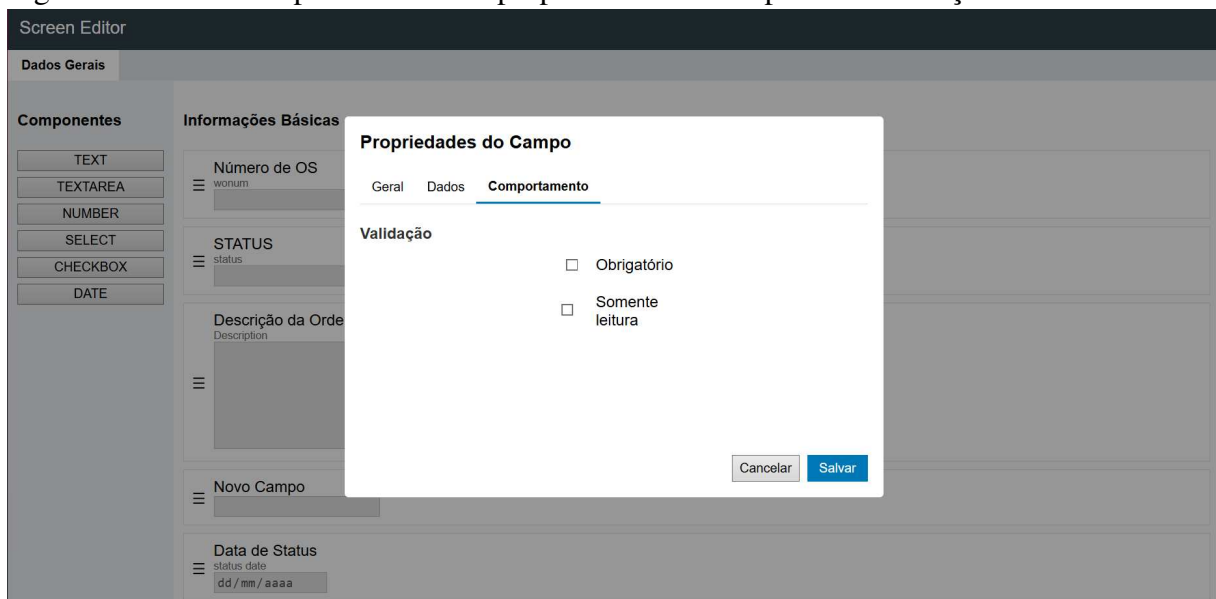
Fonte: elaborada pelo autor (2025).

Figura 18 – Aba Dados das propriedades do componente na edição de telas.



Fonte: elaborada pelo autor (2025).

Figura 19 – Aba Comportamento das propriedades do componente na edição de telas.



Fonte: elaborada pelo autor (2025).

5.3 Scripts de automação

Os *Scripts* de automação são códigos fonte capazes de manipular os dados dos objetos a partir de ações executadas pelo usuário ou pelo próprio sistema. Utiliza-se essas ações como gatilhos para que seja iniciada a execução do *script* daquela determinada tarefa. Exemplos de ações gatilhos são:

1. Ao inicializar o objeto em tela (Carregar totalmente o objeto em tela);
2. Ao editar um atributo do objeto atualmente em tela;

3. Ao clicar para salvar o objeto atualmente em tela;
4. Escalações periódicas, pré definidas com espaço de tempo entre uma execução e outra;
5. Ao realizar algum tipo de integração via API.

A partir dessas ações, executam-se os *scripts* pré-definidos pelo usuário. Os *scripts* seriam produzidos pelo próprio usuário do sistema a partir de programação em blocos, em que cada comando seria um bloco e os dados e campos dos objetos seriam colocados nesses blocos, pré-definidos como condicionais (*if/else*), estruturas de repetição (*for, while, do-while, forEach*) e suporte à escrita de funções. Com esses blocos, o sistema geraria um *script de automação* em linguagem de máquina interpretada, como python ou javascript.

Este módulo possui a necessidade de realizar a integração de código-fonte via interface do usuário a partir de técnicas de linguagem de programação em bloco em que é possível escrever códigos-fonte a partir de uma série de blocos visuais e encaixáveis.

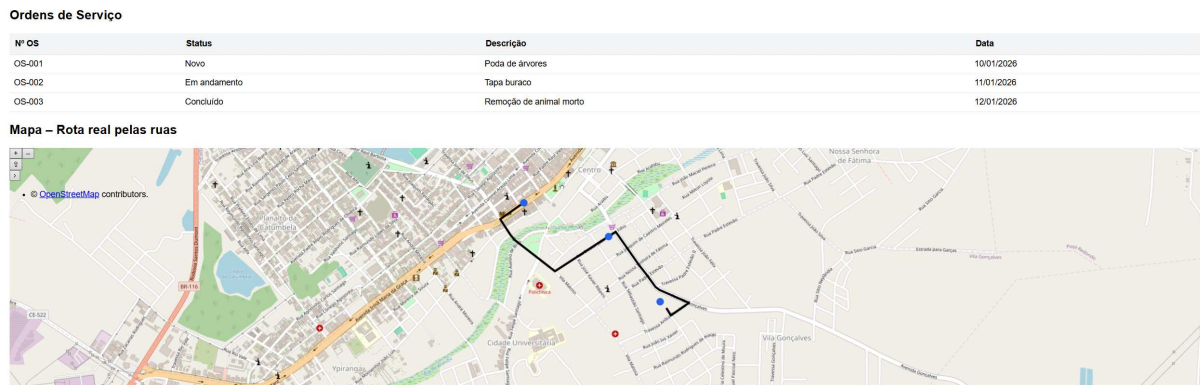
5.4 Resultado gerado pelo *framework*

Os resultados obtidos nos testes estão de acordo com a proposta do escopo e intenção de solução através de uma simulação do produto viável obtido a partir do *framework*. Na Figura 20 pode-se observar os componentes gráficos de configuração de tela citados na seção 5.2, nestes componentes, é possível visualizar os dados de ordens de serviços, informações das colunas e atributos dos objetos do sistema. Neste caso, criaram-se ordens de serviço como um objeto do sistema. Também é possível visualizar as ordens de serviço no componente personalizado de mapa, com a rota desenhada a ser seguida. Utilizando-se dos serviços do *Open Layers* é capaz de criar as rotas no componente de mapas e facilitar o uso do sistema. Observa-se também que a estilização do sistema é um padrão sem cores definidas e componentes estilizados, trata-se apenas de um *design* de interface sem uma personalidade definida, havendo necessidade de estilizar sob demanda a partir dos requisitos do cliente detentor do sistema.

Observa-se que, após a realização dos testes, os serviços implementados passaram a disponibilizar um recurso relevante para o sistema: sempre que uma nova tabela é criada, o serviço de integração é automaticamente ativado. Dessa forma, a tabela recém-criada passa a possuir um *endpoint* próprio para integração. Portanto, é possível criar diversos tipos de objetos no sistema, a partir dos atributos e tipos de atributos, deixando o sistema flexível e dinâmico para cada tipo de cenário e contexto, sendo personalizável e maleável a partir da entrada do cliente destino e suas demandas.

As colunas da tabela no banco de dados são refletidas diretamente nas colunas da interface do usuário na Figura 20, correspondendo aos atributos das ordens de serviço. No componente de mapa da interface do usuário, os pontos marcados representam os dados de localização dessas ordens, definidos, respectivamente, pelos atributos de latitude e longitude.

Figura 20 – Página de visão geral obtida através da utilização do *framework* proposto.



Fonte: elaborada pelo autor (2025).

Com o trabalho finalizado e o produto desenvolvido, espera-se, em caso de implantação em cenário real e produtivo, uma melhoria na qualidade dos serviços da zeladoria municipal prestados à sociedade. Acredita-se no potencial da solução do problema com a utilização do *framework* e de suas funcionalidades para benefício comum aos cidadãos, de forma a impactar positivamente em aspectos sociais e melhorias nos serviços prestados pelo órgão público responsável que possivelmente adquirir este *framework* como um produto em benefício da população.

6 CONSIDERAÇÕES E TRABALHOS FUTUROS

O estudo realizado durante este trabalho conseguiu atender à expectativa na resolução do problema proposto, embora o *framework*, enquanto produto, não tenha sido utilizado em um cenário real produtivo. Foi possível, com os testes e objetos produzidos e citados neste documento, que seria possível implantar o produto para possíveis clientes reais, como prefeituras e demais órgãos públicos relacionados à zeladoria pública. Os recursos presentes permitem criar ordens de serviços, ativos, equipes de campo, suportar parte das demandas de gerenciamento dos demais ativos e seus ciclos de vida, fazendo com que a viabilidade do produto se concretizasse e o potencial inicial pudesse ser provado. Para trabalhos futuros, necessita-se a criação completa do módulo *Scripts* de Automação para traduzir uma entrada lógica produzida a partir de comandos em blocos visuais para linguagem de computadores interpretada com suporte ao paradigma imperativo de linguagem de programação.

REFERÊNCIAS

- Andrade, L. H. de; AMORIM, B.; OLIVEIRA, M. de; ALVES, A. L.; ABRANTE, J. N. de; LEITE, D.; ROCHA, J. H.; BAPTISTA, C. Deuzikachico: o poder da agi no monitoramento e combate de epidemias como a de dengue, zika e chikungunya. In: **Anais do XII Simpósio Brasileiro de Sistemas de Informação**. Porto Alegre, RS, Brasil: SBC, 2016. p. 377–384. ISSN 0000-0000. Disponível em: <https://sol.sbc.org.br/index.php/sbsi/article/view/5985>. Acesso em: 08 ago. 2025.
- Botelho, P.; BORGES, A.; ALVES, G. Proposta de implantação de um sistema ciber-físico para um smart parking baseado em agentes inteligentes. In: **Anais do XIII Workshop-Escola de Sistemas de Agentes, seus Ambientes e Aplicações**. Porto Alegre, RS, Brasil: SBC, 2019. p. 259–264. ISSN 2326-5434. Disponível em: <https://sol.sbc.org.br/index.php/wesaac/article/view/33361>. Acesso em: 08 ago. 2025.
- Carvalho, M. D. S. d. **Smart cities: uso de sensores e dados secundários para cidades inteligentes centradas no cidadão**. Dissertação (Mestrado) — Universidade de Caxias do Sul, 2024.
- Castro, L.; MANOEL, F.; JESUS, V.; PANTOJA, C.; BORGES, A.; ALVES, G. Integrando sistemas multi-agentes embarcados, simulação urbana e aplicações de iot. In: **Anais do XIV Workshop-Escola de Sistemas de Agentes, seus Ambientes e Aplicações**. Porto Alegre, RS, Brasil: SBC, 2020. p. 165–176. ISSN 2326-5434. Disponível em: <https://sol.sbc.org.br/index.php/wesaac/article/view/33389>. Acesso em: 13 ago. 2025.
- Costa, I. S. **Proposição de elementos para uso de sistemas inteligentes de transportes para potencialização de intervenções estruturantes no sistema viário de cidades de médio porte: uma proposta para o novo anel viário de Bom Despacho-MG**. Monografia (Bacharelado em Engenharia Urbana) — Universidade Federal do Tocantins, Ouro Preto - MG, 2023.
- Dias, C.; LOPES, F.; LEITE, J. Smartnode dashboard: um framework front-end baseado em node-red para criação de city dashboards. In: **Anais do II Workshop Brasileiro de Cidades Inteligentes**. Porto Alegre, RS, Brasil: SBC, 2019. ISSN 0000-0000. Disponível em: <https://sol.sbc.org.br/index.php/wbci/article/view/6744>. Acesso em: 08 ago. 2025.
- KITCHENHAM, B.; Pearl Brereton, O.; BUDGEN, D.; TURNER, M.; BAILEY, J.; LINKMAN, S. Systematic literature reviews in software engineering – a systematic literature review. **Information and Software Technology**, v. 51, n. 1, p. 7–15, 2009. ISSN 0950-5849. Special Section - Most Cited Articles in 2002 and Regular Research Papers. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0950584908001390>. Acesso em: 09 ago. 2025.
- MELONIO, A. C. C. **Smart São Paulo: um estudo da mobilidade urbana sob a ótica de Machine Learning e aspectos espaço-temporais**. Dissertação (Mestrado) — Universidade Presbiteriana Mackenzie, 2021.
- Oliveira, V. A. T. *et al.* **Gestão de operações de serviços de emergência no contexto de cidades inteligentes e sustentáveis: o caso da Polícia Militar do Paraná**. Dissertação (Mestrado) — Universidade Tecnológica Federal do Paraná, 2020.

Pinto, N. R. **Desenvolvimento de uma virtual test bed e digital twin para treinamento de veículos autônomos**. Monografia (Bacharelado em Engenharia de Controle e Automação) — Universidade Estadual Paulista (Unesp), Sorocaba - SP, 2025.

Santiago, T. E. T. **Cidades inteligentes, gestão urbana e geotecnologias: proposta de city information modeling (CIM) aplicado ao Município de Madre de Deus-BA**. Dissertação (Mestrado) — UCSal, Universidade Católica do Salvador, 2023.

Santos, L.; NASCIMENTO, L.; NUNES, S. Reclamando app: um aplicativo para auxiliar na reivindicação de problemáticas urbanas. In: **Anais da XIX Escola Regional de Computação Bahia, Alagoas e Sergipe**. Porto Alegre, RS, Brasil: SBC, 2019. p. 184–189. ISSN 0000-0000. Disponível em: <https://sol.sbc.org.br/index.php/erbase/article/view/8976>. Acesso em: 09 ago. 2025.

Storck, C.; SALES, E.; ZÁRATE, L.; FIGUEIREDO, F. Proposta de um framework baseado em mineração de dados para redes 5g. **Revista Eletrônica de Sistemas de Informação**, v. 16, n. 2, 2017. ISSN 1677-3071. Disponível em: <https://www.periodicosibepes.org.br/index.php/reinfo/article/view/2488>. Acesso em: 10 ago. 2025.

Weiss, M. C.; BERNARDES, R. C.; CONSONI, F. L. Cidades inteligentes como nova prática para o gerenciamento dos serviços e infraestruturas urbanos: a experiência da cidade de porto alegre. **Urbe. Revista Brasileira de Gestão Urbana**, SciELO Brasil, v. 7, p. 310–324, 2015.