



**UNIVERSIDADE FEDERAL DE CAMPINA GRANDE
CENTRO DE ENGENHARIA ELÉTRICA E INFORMÁTICA
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO**

GUSTAVO FARIAS DE SOUZA SILVA

**MOSQUITO ZERO:
UMA PLATAFORMA WEB PARA GERENCIAMENTO DAS
INSPEÇÕES DE VIGILÂNCIA PARA O CONTROLE DE
VETORES**

CAMPINA GRANDE - PB

2024

GUSTAVO FARIAS DE SOUZA SILVA

**MOSQUITO ZERO:
UMA PLATAFORMA WEB PARA GERENCIAMENTO DAS
INSPEÇÕES DE VIGILÂNCIA PARA O CONTROLE DE
VETORES**

**Trabalho de Conclusão Curso
apresentado ao Curso Bacharelado em
Ciência da Computação do Centro de
Engenharia Elétrica e Informática da
Universidade Federal de Campina
Grande, como requisito parcial para
obtenção do título de Bacharel em
Ciência da Computação.**

Orientador : Professor Dr. Tiago Lima Massoni.

CAMPINA GRANDE - PB

2024

GUSTAVO FARIAS DE SOUZA SILVA

**MOSQUITO ZERO:
UMA PLATAFORMA WEB PARA GERENCIAMENTO DAS
INSPEÇÕES DE VIGILÂNCIA PARA O CONTROLE DE
VETORES**

**Trabalho de Conclusão Curso
apresentado ao Curso Bacharelado em
Ciência da Computação do Centro de
Engenharia Elétrica e Informática da
Universidade Federal de Campina
Grande, como requisito parcial para
obtenção do título de Bacharel em
Ciência da Computação.**

BANCA EXAMINADORA:

**Tiago Lima Massoni
Orientador – UASC/CEEI/UFCG**

**José Antão Beltrão Moura
Examinador – UASC/CEEI/UFCG**

**Francisco Vilar Brasileiro
Professor da Disciplina TCC – UASC/CEEI/UFCG**

Trabalho aprovado em: 15 de Maio de 2024.

CAMPINA GRANDE - PB

RESUMO

Os agentes de vigilância ambiental em saúde exercem um papel fundamental no controle da proliferação dos mosquitos transmissores de doenças como a dengue, a zika e a chikungunya. Esses profissionais atuam no combate das arboviroses através de ações de campo, visitas domiciliares e comunitárias. Essas atividades têm como principal função a identificação e o tratamento de focos de reprodução dos vetores. Atualmente, o registro dos dados das inspeções realizadas pelos agentes de vigilância da cidade de Campina Grande - PB, é feito utilizando um formulário de papel. Porém, esse formato de anotação pode apresentar diversos problemas na organização, na visualização e na segurança das informações. Sendo assim, visando auxiliar o serviço desses profissionais, este trabalho tem como objetivo desenvolver a primeira versão de uma plataforma web responsiva, para o registro e acompanhamento das pesquisas entomológicas realizadas pelo agente, o Mosquito Zero. A primeira versão da aplicação foi testada com alguns agentes de vigilância da cidade de Cabedelo no estado da Paraíba, a fim de obter a avaliação desses usuários referente à usabilidade do sistema.

**MOSQUITO ZERO:
A WEB PLATFORM FOR MANAGEMENT OF SURVEILLANCE
INSPECTIONS FOR VECTOR CONTROL**

ABSTRACT

Environmental health surveillance agents play a fundamental role in controlling the proliferation of disease-transmitting mosquitoes such as dengue, Zika, and chikungunya. These professionals engage in combating arboviruses through field actions, household, and community visits. Their main tasks involve identifying and treating breeding sites of vectors. Currently, data recording from inspections conducted by surveillance agents in the city of Campina Grande - PB, is done using paper forms. However, this format of annotation can pose various problems in organization, visualization, and information security. Therefore, aiming to assist these professionals' service, this work aims to develop the first version of a responsive web platform for recording and monitoring entomological surveys conducted by the agent, Mosquito Zero. The first version of the application was tested with some surveillance agents in the city of Cabedelo in the state of Paraíba, in order to obtain these users' evaluation regarding the system's usability.

Mosquito Zero: Uma Plataforma Web para Gerenciamento das Inspeções de Vigilância para o Controle de Vetores

Gustavo Farias de Souza Silva
Universidade Federal de Campina Grande
Campina Grande, Paraíba, Brasil
gustavo.silva@ccc.ufcg.edu.br

Dr. Tiago Lima Massoni (Orientador)
Universidade Federal de Campina Grande
Campina Grande, Paraíba, Brasil
massoni@dsc.ufcg.edu.br

RESUMO

Os agentes de vigilância ambiental em saúde exercem um papel fundamental no controle da proliferação dos mosquitos transmissores de doenças como a dengue, a zika e a chikungunya. Esses profissionais atuam no combate das arboviroses através de ações de campo, visitas domiciliares e comunitárias. Essas atividades têm como principal função a identificação e o tratamento de focos de reprodução dos vetores. Atualmente, o registro dos dados das inspeções realizadas pelos agentes de vigilância da cidade de Campina Grande - PB, é feito utilizando um formulário de papel. Porém, esse formato de anotação pode apresentar diversos problemas na organização, na visualização e na segurança das informações. Sendo assim, visando auxiliar o serviço desses profissionais, este trabalho tem como objetivo desenvolver a primeira versão de uma plataforma *web* responsiva, para o registro e acompanhamento das pesquisas entomológicas realizadas pelo agente, o Mosquito Zero. A primeira versão da aplicação foi testada com alguns agentes de vigilância da cidade de Cabedelo no estado da Paraíba, a fim de obter a avaliação desses usuários referente à usabilidade do sistema.

Palavras-chave

Dengue, zika, chikungunya, agentes de vigilância, controle de vetores, plataforma *web*, desenvolvimento de aplicativos.

REPOSITÓRIO

<https://github.com/GusttaFS/mosquitozero-frontend>

<https://github.com/GusttaFS/mosquitozero-backend>

1. INTRODUÇÃO

A proliferação de mosquitos transmissores de doenças, como o *Aedes aegypti*, continua sendo um grande problema para a saúde pública. No Brasil, segundo dados disponibilizados pelo Ministério da Saúde [1], os casos prováveis de dengue, entre os meses de janeiro e abril de 2024, já ultrapassam o total de 3 milhões [2]. Para efeito de comparação, em 2023 foram contabilizados cerca de 1,5 milhão de registros durante todo o ano. Na Paraíba, as cidades de João Pessoa e Campina Grande juntas somam mais de 4 mil casos e estão entre os únicos cinco municípios do estado com ao menos um óbito pela doença já confirmado até abril de 2024 [3].

Os agentes de vigilância ambiental em saúde (AVAS), são responsáveis por desempenhar o papel central no controle e na prevenção de arboviroses como a dengue, a zika e a chikungunya [4]. Esses profissionais concentram as atividades sobretudo em ações de campo, visitas domiciliares e comunitárias. Essas visitas têm como finalidade a identificação e o tratamento dos focos de reprodução dos mosquitos

transmissores, com o intuito de prevenir a proliferação em grande escala.

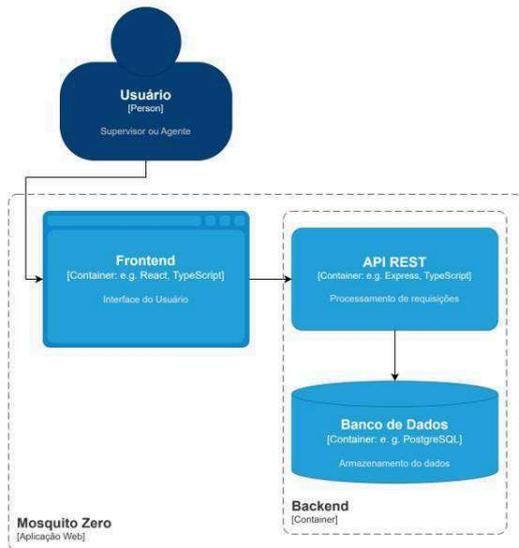
Tradicionalmente, o registro das informações coletadas durante as inspeções é feito utilizando um formulário de papel. Porém, esse formato de anotação pode apresentar diversos problemas tanto de organização quanto de visualização ou até mesmo de segurança. Documentos desse tipo podem crescer consideravelmente devido ao volume de informações obtidas, tornando difícil a organização e análise dos dados posteriormente. Além disso, por se tratar de um formulário físico os dados contidos nele podem ser facilmente perdidos, danificados ou até mesmo fraudados, comprometendo a integridade das informações. Para os supervisores, encarregados de coordenar e acompanhar as atividades dos agentes, a utilização deste formulário também dificulta a observação do progresso das atividades de pesquisa.

Sendo assim, foi desenvolvida a primeira versão do Mosquito Zero¹, uma plataforma *web* para auxiliar o trabalho dos agentes e supervisores no registro, na visualização e no acompanhamento das vigilâncias entomológicas [5]. A versão inicial da aplicação inclui uma API REST (backend) responsável por disponibilizar as funcionalidades de manipulação dos dados que são necessárias para o funcionamento dos recursos oferecidos na interface do usuário (frontend).

O Mosquito Zero tem como inspiração o sistema legado de um aplicativo *mobile* para dispositivos Android e um dashboard *web* do projeto MEWAR, denominado como Zika App [6]. O *Mosquito population modelling for early warning system and rapid health authority response* (MEWAR) [7] é um projeto em colaboração entre diversas instituições que combina saúde pública, tecnologia móvel e modelagem climática para avaliar os impactos das mudanças ambientais na água e seus efeitos na reprodução de mosquitos transmissores de doenças. A aplicação a qual este trabalho toma como base, teve o desenvolvimento iniciando em 2021 e seria utilizada pelos agentes de saúde do município de Recife - PE, a fim de coletar dados de vigilância sobre focos de reprodução. Os dados coletados iriam alimentar os modelos de análise preditiva de proliferação para alertar às autoridades de saúde pública e ambientais e mobilizar a comunidade na prevenção e resposta rápida a surtos das doenças transmitidas pelos vetores.

Como prova de conceito, foi realizado o teste de usabilidade com um supervisor e um agente de vigilância da cidade de Cabedelo - PB, através de uma videoconferência. Durante a avaliação, ambos os usuários expressaram satisfação com a proposta da aplicação, destacando sua utilidade no registro e na

¹ <http://mosquitozero.vercel.app>



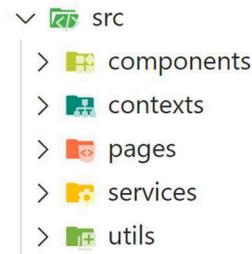
[Figura 2: Diagrama de contêiner C4 do Mosquito Zero]

4.1 Frontend

O frontend do Mosquito Zero foi desenvolvido utilizando a linguagem de programação *TypeScript*² e a biblioteca para construção de interfaces do usuário, *ReactJS*³. Essas tecnologias foram escolhidas devido a facilidade na manutenção e a eficiência que oferecem. Com a tipagem estática do código *JavaScript*, o *TypeScript* auxilia a prevenir erros de compilação durante o processo de desenvolvimento. O *ReactJS*, por sua vez, é conhecido pela abordagem baseada em componentes, que facilita a reutilização dos elementos e contribui na organização do código.

Outras ferramentas também utilizadas para o desenvolvimento do frontend foram o *axios*, o *sass* e o *react-toastify*. A biblioteca *axios*⁴ serve para simplificar o processo de comunicação assíncrona entre o cliente e o servidor oferecendo uma API simples para tratar das requisições e respostas HTTP. *Sass*⁵, *Syntactically Awesome Stylesheets*, é uma extensão do CSS que permite a definição de estilos de forma mais eficiente e modular, auxiliando na manutenção e na organização dos estilos. Por último, *react-toastify*⁶ é uma biblioteca que serve para exibir notificações (*toasts*) na interface, como mensagens de sucesso ou de erro, de maneira simples e responsiva.

A estrutura de pastas do frontend, como apresentada na figura 3, é constituída por 5 módulos principais, o *components*, o *contexts*, o *pages*, o *services* e o *utils*. O módulo *components* possui as estruturas visuais que são utilizadas pelas páginas da interface. O *contexts* é responsável por gerenciar todo o estado da aplicação. No *pages* é feita a renderização completa das páginas do sistema, divididas em duas partes, uma com a interface direcionada para o agente e outra para o supervisor. O *services* mantém a lógica de comunicação com a API. E por último, o *utils* contém os códigos de apoio utilizados entre os módulos.



[Figura 3: Estrutura de pastas do frontend]

4.2 Backend

Para o desenvolvimento da API do Mosquito Zero também foi utilizado a linguagem de programação *TypeScript*. Além disso, foi escolhido o *express*⁷, um framework *web* rápido, minimalista e flexível, para processar as solicitações HTTP, através da definição das rotas de acesso (*endpoints*) e para o envio das respostas às requisições, no formato *JSON* [13].

A API utiliza o JWT⁸ (*JSON Web Tokens*) para a geração e verificação do token necessário para autenticação do usuário no sistema. Um novo token é gerado sempre que o usuário efetuar o login na aplicação. Um dos dados codificados no token é o *type*, responsável por indicar se o usuário que está acessando é um agente ou um supervisor, como exposto na figura 4. Dessa forma, o frontend ao decodificar o token gerado saberá quais páginas e recursos o usuário logado poderá acessar.

```
const token = sign(
  {
    email: user.email,
    type: user.type
  },
  process.env.JWT_SECRET,
  {
    subject: user.id,
    expiresIn: '1d'
  }
);
```

[Figura 4: Código para geração do token de autenticação]

A estrutura de pastas do backend, mostrada na figura 5, é formada principalmente pelos módulos *controllers*, *services*, *middlewares* e *validators*. O *controllers* representa a camada mais externa que recebe as solicitações e extrai os dados necessários para as negociações realizadas pelo *services*, a camada interna responsável pelo acesso ao banco de dados para manipulação das informações.

O módulo *middlewares* contém a função responsável por verificar se o usuário está devidamente autenticado no sistema. Finalmente, o *validators* encapsula as verificações e validações realizadas nos dados recebidos pelo *controllers*. Além disso, o backend possui o módulo *prisma* que inclui as configurações necessárias para modelagem e comunicação do banco de dados. Por fim, o arquivo *routes* mantém a definição de todos os *endpoints* da aplicação.

² <https://www.typescriptlang.org/>

³ <https://react.dev/>

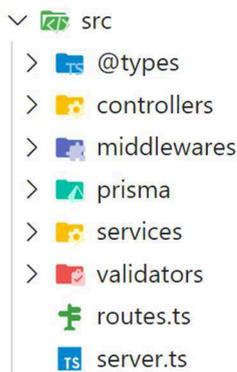
⁴ <https://axios-http.com/>

⁵ <https://sass-lang.com/>

⁶ <https://fkhadra.github.io/react-toastify/introduction>

⁷ <https://expressjs.com/>

⁸ <https://jwt.io/>

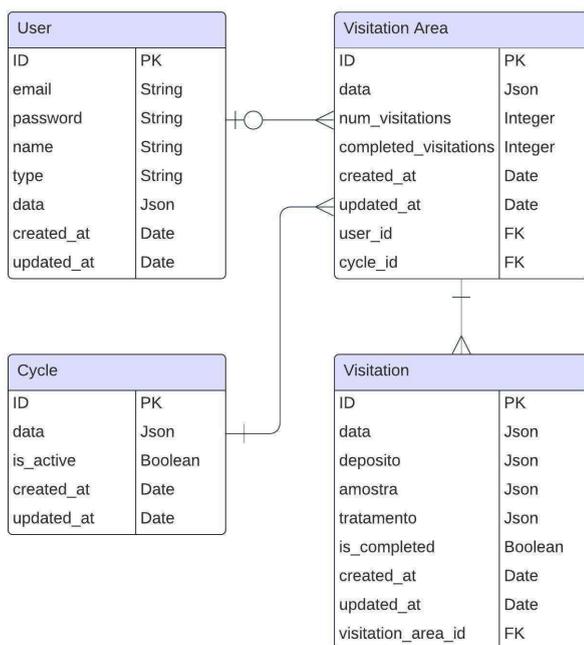


[Figura 5: Estrutura de pastas do backend]

4.2.1 Banco de dados

Foi utilizado o *PostgreSQL*⁹, para o armazenamento dos dados da aplicação, e a ferramenta de mapeamento objeto relacional *prisma* para auxiliar no acesso ao banco e na manipulação dos dados. Com o *prisma*¹⁰ é possível definir os modelos de dados do sistema de forma simples e direta. A ferramenta gera automaticamente as consultas SQL para criação das entidades no banco de dados, com base nos modelos definidos.

A figura 6 mostra o diagrama de relacionamento das entidades definidas no banco de dados do sistema.



[Figura 6: Diagrama de relacionamento das entidades do banco de dados]

Todos os registros possuem um identificador único (*id*) que é utilizado para recuperação dos dados e também para referências internas. Além disso, cada tabela no banco possui atributos específicos para o armazenamento das informações.

A entidade *user* possui os atributos *email*, *password*, *name*, *type* e *data* que correspondem respectivamente ao endereço de e-mail, a senha, o nome, o tipo (agente ou supervisor) e aos dados adicionais do usuário. As senhas cadastradas são

⁹ <https://www.postgresql.org/>

¹⁰ <https://www.prisma.io/>

criptografadas utilizando a biblioteca de hashing *bcrypt*¹¹ antes de serem armazenadas no banco de dados.

O *cycle* possui apenas os atributos: *data*, que armazena os dados gerais do ciclo, como a referência e o ano; e *is active* que indica se o ciclo está ativo ou não. A *visitation area* também possui o atributo *data*, que deve conter as informações gerais referentes a área, e os atributos *num* e *completed visitations* que se referem respectivamente ao número total de visitas e a quantidade de visitas já concluídas na área. Esses dois últimos dados são utilizados para o cálculo de progresso do agente.

Por fim, *visitation* possui os atributos *data*, *deposito*, *amostra*, e *tratamento* que juntos representam os dados do formulário de pesquisa utilizado pelo agente durante as inspeções. A entidade também possui o atributo *is completed* que indica se a visita foi ou não concluída pelo agente.

4.2.2 Endpoints

Os *endpoints* disponíveis na primeira versão da API estão agrupados de acordo com as entidades às quais se referem, sendo elas o usuário (*user*), o ciclo (*cycle*), a área de visita (*visitation area*) e a visita (*visitation*). As operações possíveis nas entidades estão relacionadas sobretudo à criação, recuperação e atualização das informações. A documentação detalhada dos endpoints oferecidos pela API está disponível no repositório do backend da aplicação.

5. DESENVOLVIMENTO

5.1 Requisitos

Após a análise e a compreensão do formulário utilizado e da atividade de pesquisa realizada pelos agentes, foram definidos os requisitos principais para a primeira versão do sistema, como apresentado na tabela 1.

Requisito	Descrição
01	Permitir a listagem das áreas de visita atribuídas ao agente.
02	Permitir a exibição do ciclo atual das atividades.
03	Permitir a listagem dos imóveis a serem inspecionados na área.
04	Permitir a adição de novas visitas na área.
05	Permitir o cadastro das informações conforme o formulário de pesquisa.
06	Permitir a exibição do progresso das atividades de campo.
07	Permitir a consulta dos registros realizados em ciclos anteriores.

[Tabela 1: Requisitos do Mosquito Zero para realização de atividades pelo agente]

Inicialmente o Mosquito Zero iria oferecer apenas os recursos voltados para a atividade de inspeção realizada pelos agentes. Porém, observou-se a necessidade de se produzir também funcionalidades para o supervisor, uma vez que este será o responsável por controlar e fornecer os dados referentes às áreas

¹¹ <https://www.npmjs.com/package/bcrypt>

e aos imóveis a serem visitados pelos agentes. Portanto, foram definidos requisitos adicionais para atender a essa necessidade, conforme mostrado na tabela 2.

Requisito	Descrição
08	Permitir a definição de novos ciclos de atividades.
09	Permitir a listagem dos agentes supervisionados.
10	Permitir a atribuição de áreas de visita aos agentes.
11	Permitir a definição de imóveis a serem visitados na área.
12	Permitir a visualização do andamento das atividades realizadas pelos agentes.
13	Permitir a visualização dos dados coletados durante as inspeções feitas pelo agente.
14	Permitir a consulta dos registros realizados em ciclos anteriores.

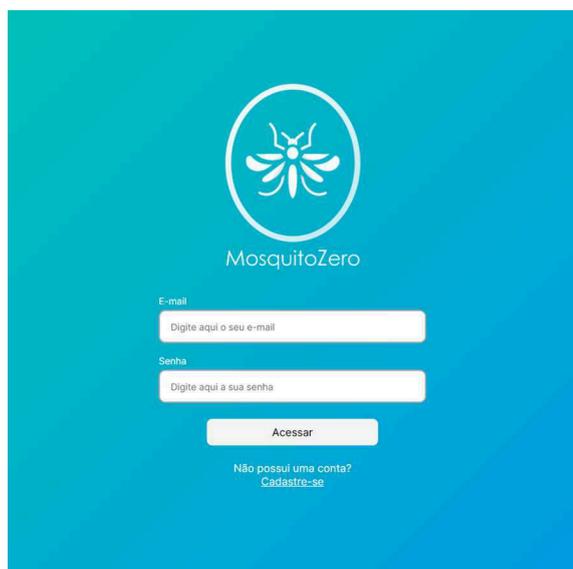
[Tabela 2: Requisitos do Mosquito Zero para realização de atividades pelo supervisor]

Também foi definido os requisitos comuns a ambos os usuários, agente e supervisor, que correspondem às atividades de cadastro (R15) e autenticação (R16) no sistema.

5.2 A aplicação

Para contemplar as funcionalidades a serem disponibilizadas na primeira versão do Mosquito Zero, foi feito o desenvolvimento da aplicação, com base nos requisitos definidos. A interface do sistema é dividida em duas partes: a primeira contendo os recursos oferecidos para o agente e a segunda para o supervisor.

Ao acessar a aplicação, o usuário é direcionado para a tela de login (figura 7), na qual deve informar o endereço de e-mail e a senha de acesso para autenticação no sistema (R16). A partir dessa tela, também é possível acessar a página para o cadastro no sistema (R15), caso o usuário ainda não tenha se registrado.

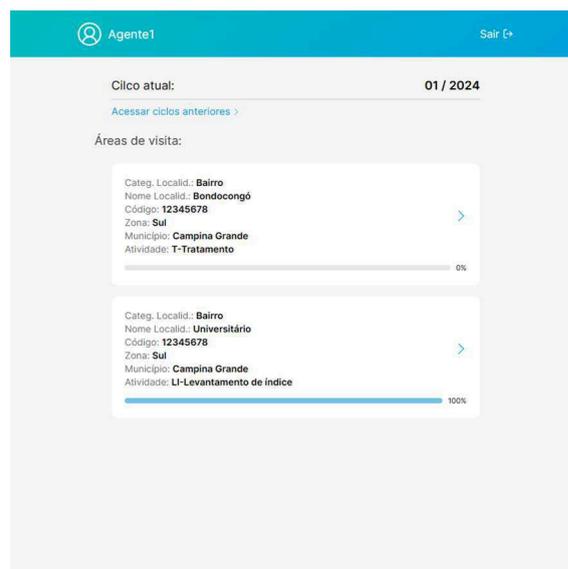


[Figura 7: Tela de login do Mosquito Zero]

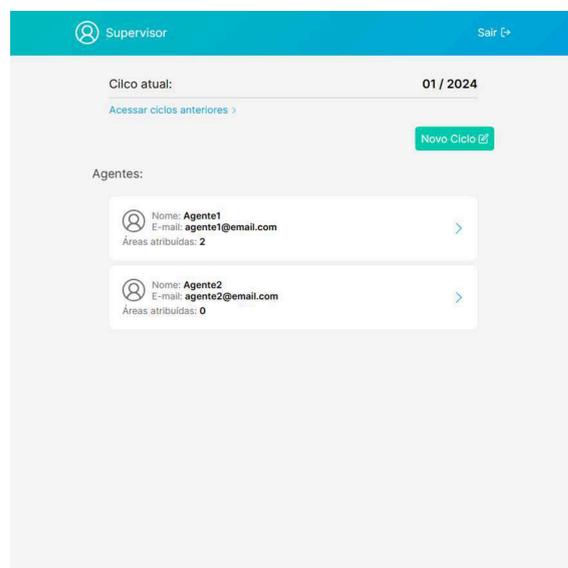
Após a autenticação, o usuário é conduzido para a página inicial que varia de acordo com o cargo, agente ou supervisor, informado durante o registro.

5.2.1 Páginas iniciais

A página principal do agente (figura 8) possui a listagem das áreas a serem visitadas (R01) e progresso atual nas atividades (R06). Por outro lado, a tela inicial do supervisor, lista todos os agentes supervisionados (R09), como mostrado na figura 9. Em ambas as telas é possível visualizar no canto superior, o ciclo atual das atividades de pesquisa (R02). No entanto, a funcionalidade de definição de um novo ciclo está disponível apenas para o supervisor (R08).



[Figura 8: Tela inicial do agente no Mosquito Zero]



[Figura 9: Tela inicial do supervisor no Mosquito Zero]

5.2.2 Páginas disponíveis para o agente

A partir da tela principal o agente poderá verificar a lista de imóveis a serem visitados (R03), ao clicar em uma das áreas atribuídas, ou consultar os registros feitos nos ciclos anteriores (R07), sem a possibilidade de alterar as informações registradas.

Ao selecionar uma das visitas a serem feitas na área, o agente é direcionado para a página contendo o formulário de pesquisa (figura 10), no qual deve inserir os dados da inspeção realizada no imóvel (R05). Além disso, é possível que o agente adicione uma nova visita na área (R04), caso seja necessário.

The screenshot shows a web form for entomological research. At the top, there's a header with 'Agente1' and a 'Sair' button. Below that, a status indicator shows 'Pendente' and a 'Concluída' button. The main title is 'PESQUISA ENTOMOLÓGICA / TRATAMENTO'. The form fields are: 'Nº do Quart:' (input: 1), 'Lado:' (input: Direito), 'Logradouro:' (input: Rua Exemplo de Nome da Rua), 'Número:' (input: 1), 'Compl.:' (input: Casa), 'Horário de entrada:' (input: 12:31), 'Tipo do Imóvel:' (dropdown: Residencial), 'Visita:' (dropdown: Normal), and 'Pendência:' (dropdown). Below these is a table for 'Nº de Depósitos Inspeccionados' with columns A1, A2, B, C, D1, D2, E. The 'Eliminado' row has values 1, 0, 1, 0, 1, 0, 0. The 'Imóveis Inspeccionados' row has values 2, 3.

[Figura 10: Tela do formulário de pesquisa no Mosquito Zero]

5.2.3 Páginas disponíveis para o supervisor

Assim como o agente, o supervisor também poderá consultar as atividades realizadas nos ciclos anteriores (R14). Para cada ciclo é mostrada as áreas de visitas que foram atribuídas a cada um dos agentes, nas quais é possível consultar os imóveis visitados e os dados coletados nas inspeções.

O supervisor, ao selecionar um agente na página principal, é direcionado para tela com os dados gerais das atividades do agente. Nela é possível visualizar as áreas atribuídas, o progresso atual do agente (R12) e a opção para atribuição de uma nova área de visita (R10), a qual é feita a partir do preenchimento das informações referentes à região a ser visitada.

A partir da seleção de uma das áreas de visitas atribuídas ao agente, o supervisor poderá definir os imóveis a serem visitados na área (R11). Também é possível que o supervisor visualize as visitas já definidas e os dados registrados, caso o agente já as tenha concluído (R13).

5.3 Implantação da API

Para a implantação da API, foi criada uma instância no serviço de nuvem EC2¹² da AWS, em que foi configurado o banco de dados *PostgreSQL* juntamente com as ferramentas e os arquivos necessários para execução do backend. Além disso, foi utilizado o gerenciador de processos PM2¹³ para iniciar e manter a aplicação em execução, mesmo com possíveis falhas ou reinicializações que possam ocorrer no servidor.

5.4 Implantação da interface

Finalmente, com o backend já disponível para o acesso remoto, foi feita a implantação do frontend na plataforma de

hospedagem de aplicativos *web*, Vercel¹⁴. Essa plataforma permitiu a disponibilização da interface ao usuário de forma rápida e automatizada, além de garantir a integração contínua a partir do repositório da aplicação.

6. AVALIAÇÃO

6.1 Metodologia

Para a prova de conceito da aplicação, foi realizada uma videoconferência com os potenciais usuários da plataforma, a fim de avaliar a usabilidade e as funcionalidades oferecidas pelo sistema. O teste inicialmente estava programado para ocorrer na cidade de Campina Grande, mas devido à indisponibilidade dos agentes da região, o teste acabou sendo feito com um supervisor e um agente de combate a endemias da cidade de Cabedelo, também localizada no estado da Paraíba.

Durante a avaliação, foi apresentada a ideia geral do Mosquito Zero, bem como a interface e recursos atuais da aplicação. Após a apresentação foi solicitado que os usuários utilizassem o sistema, de acordo com cenários e tarefas propostas para a realização das atividades oferecidas na primeira versão da plataforma. No decorrer do teste, foram feitas anotações em relação ao comportamento dos usuários ao interagir com o sistema e as opiniões a respeito da interface e das funcionalidades do projeto.

Algumas das tarefas solicitadas ao agente, para a realização do teste de usabilidade, foram: consultar as áreas e os imóveis a serem inspeccionados; preencher o formulário de pesquisa e registrar os dados no sistema; e conferir o progresso, as atividades realizadas no ciclo atual e nos anteriores, como também as informações coletadas durante a atividade de campo. Por fim, para o supervisor foram sugeridos: a definição de um novo ciclo; a atribuição das áreas e dos imóveis a serem visitados pelo agente; e a consulta das informações coletadas na atividade de pesquisa no ciclo atual e nos anteriores.

Após a utilização da plataforma, foram feitas perguntas aos usuários para coletar a opinião a respeito da experiência geral no uso da interface e das melhorias que podem ser implementadas. Foi discutido também se o modelo atual do sistema contempla as atividades realizadas pelos agentes no trabalho de campo.

O próximo tópico desta seção detalha os principais resultados obtidos durante a avaliação do Mosquito Zero pelos usuários.

6.2 Resultados

Os usuários de maneira geral se mostraram satisfeitos com a proposta da aplicação e consideraram intuitiva a forma como são feitas as principais atividades no sistema. O Mosquito Zero foi julgado, por ambos, como extremamente útil para o registro e a visualização dos dados coletados e para o acompanhamento do trabalho de campo.

Na opinião do agente, utilizar uma ferramenta digital como o Mosquito Zero, para registrar as informações da pesquisa seria muito importante, principalmente pela possibilidade de padronizar a coleta dos dados, que muitas vezes são realizadas de maneiras diferentes por outros colegas de trabalho. O usuário também comentou a respeito da segurança proporcionada pelo registro dos dados no sistema, destacando a prevenção do retrabalho na recuperação das informações, caso o formulário físico seja perdido ou danificado após a atividade de campo.

¹² <https://aws.amazon.com/pt/ec2/>

¹³ <https://pm2.io/>

¹⁴ <https://vercel.com/>

Com relação a interface e recursos oferecidos, o agente levantou algumas questões importantes a serem consideradas. Apesar das informações estarem claras e bem distribuídas nas páginas da aplicação, o usuário comentou que seria interessante o agrupamento da lista dos imóveis pelos quarteirões ao qual fazem parte, garantindo uma maior organização na listagem das visitas e facilitando a navegação e o planejamento das atividades de campo. Outro ponto importante, foi a sugestão da adição de um novo recurso na aplicação, para o registro de observações ou justificativas do não cumprimento da atividade de pesquisa em uma determinada área, por motivos diversos como a ocorrência de chuvas fortes na região ou de reformas que impossibilitem o acesso aos imóveis, por exemplo.

Por fim, o teste com o supervisor forneceu informações importantes para a adaptação do sistema a algumas necessidades específicas não consideradas na primeira versão. O usuário julgou a possibilidade de acompanhamento das atividades bastante útil e necessária, pois atualmente a atualização a respeito do andamento no trabalho de campo pelos agentes é informada com um certo atraso, dificultando a visualização do progresso.

O supervisor também comentou que, apesar da visualização das áreas e do progresso geral do agente serem bastante claras e informativas, seria interessante a possibilidade de visualizar valores quantitativos referente ao número de imóveis e quarteirões visitados pelos agentes, tanto de modo geral quanto em cada dia de atividades de campo realizadas. Esses dados, portanto, serviram para fornecer maiores detalhes do trabalho feito pelos agentes e também para facilitar a geração posterior de relatórios com o resumo das pesquisas realizadas.

6.3 Considerações

O desenvolvimento inicial do Mosquito Zero como uma plataforma web, mostrou-se como uma boa solução para o problema, uma vez que os usuários expressaram preocupação na utilização de um aplicativo *mobile* durante as atividades de campo em regiões mais perigosas da cidade. Apesar da aplicação ser responsiva e poder ser utilizada através dos navegadores *web* em smartphones ou tablets, a proposta não dispensa completamente a utilização do formulário físico, os usuários afirmaram preferir não utilizar esses dispositivos durante as atividades de campo. No entanto, o registro das informações coletadas poderá ser realizado ao final da atividade de pesquisa, facilitando a análise posterior das informações coletadas e garantindo que os dados não sejam perdidos.

Outro ponto importante discutido durante a avaliação do projeto, foi o de garantir que a interface seja vista de forma agradável e convidativa aos agentes que possuem uma maior resistência na utilização de soluções tecnológicas como a do Mosquito Zero, de modo que estes possam também aderir a ideia de utilizar a plataforma para realizar as atividades de registro das pesquisas sem grandes dificuldades. Felizmente, na opinião dos usuários que utilizaram a plataforma, a forma simples e direta como as informações e os recursos são dispostos na interface, podem ser úteis para reduzir o atrito entre os agentes menos familiarizados com o uso de aplicativos e o sistema do Mosquito Zero.

7. TRABALHOS FUTUROS

Inicialmente, para garantir que o Mosquito Zero seja acessível o suficiente para utilização pelos agentes de vigilância que não estão acostumados com o uso de tecnologias desse tipo, será feita uma avaliação com esses usuários específicos. O intuito da avaliação será de coletar feedbacks para possíveis melhorias da

interface, a fim de amenizar ou eliminar os problemas que possam existir na utilização do sistema por esses usuários.

Além disso, serão feitos novos testes com uma quantidade maior de potenciais usuários a fim de obter e realizar a implementação das sugestões a serem oferecidas durante a avaliação da interface, juntamente com as considerações feitas durante o primeiro teste, sobretudo referentes às melhorias do registro, organização e visualização dos dados das atividades realizadas. Desse modo, tanto o agente quanto o supervisor poderão acompanhar o andamento das pesquisas de forma mais completa e eficiente.

A fim de maximizar a utilidade do sistema, também serão implementadas soluções que atendam as necessidades em outras atividades realizadas pelos agentes, além do trabalho de campo, e também pelos supervisores, proporcionando uma plataforma abrangente que ofereça recursos para as diferentes demandas operacionais. Também serão feitos estudos sobre formas de automatizar o processo de atribuição das áreas e dos imóveis a serem visitados pelos agentes de vigilância, para que esta atividade possa ser feita de forma mais rápida pelo supervisor. Por fim, a possibilidade de gerar relatórios automáticos com base nas informações registradas tanto pelos agentes como pelos supervisores, pode ser útil e bastante relevante para auxiliar no trabalho desses profissionais.

8. AGRADECIMENTOS

Primeiramente, meus agradecimentos vão para o meu orientador, o professor Tiago Lima Massoni, pela paciência e pelo apoio prestado ao longo de todo o desenvolvimento deste trabalho. Além disso, não poderia deixar de agradecer à minha família, pelo incentivo e suporte constantes; e à minha esposa, pelo amor e compreensão durante este período. Quero também agradecer os meus amigos, que estiveram comigo em toda a minha jornada acadêmica, me motivando a superar os maiores desafios. Ainda, aos agentes que se disponibilizaram para utilizar a plataforma, meus sinceros agradecimentos pela colaboração e contribuição para este estudo. Por fim, mas não menos importante, parableno-me pela insistência e dedicação não só para este trabalho, como em toda a minha trajetória na graduação.

9. REFERÊNCIAS

- [1] DATASUS Tecnologia da Informação a Serviço do SUS, TabNet. “Informações de Saúde (TABNET)” Disponível em: <https://datasus.saude.gov.br/informacoes-de-saude-tabnet/>.
- [2] DATASUS Tecnologia da Informação a Serviço do SUS, TabNet. “Dengue - Notificações Registradas no Sistema de Informação de Agravos de Notificação - Brasil”. Disponível em: <http://tabnet.datasus.gov.br/cgi/tabcgi.exe?sinanet/cnv/denguebr.def>.
- [3] Governo da Paraíba. Secretaria da Saúde. “Monitoramento das arboviroses”. Disponível em: <https://paraiba.pb.gov.br/diretas/saude/consultas/vigilancia-em-saude-1/paineis-de-monitoramento-01>.
- [4] Secretária de Saúde do Distrito Federal. “Agentes de Vigilância Ambiental em Saúde se destacam no combate à dengue e outras doenças”. Disponível em: [https://www.saude.df.gov.br/web/guest/w/agentes-de-vigilancia-ambiental-em-saude-se-destacam-no-combate-a-dengue-e-outras-doenhas#:~:text=E2%80%9C%3%89%20um%20profissional%](https://www.saude.df.gov.br/web/guest/w/agentes-de-vigilancia-ambiental-em-saude-se-destacam-no-combate-a-dengue-e-outras-doenhas#:~:text=E2%80%9C%3%89%20um%20profissional%20)

20que%20desempenha,de%20import%C3%A2ncia%20m
%C3%A9dica%E2%80%9D%2C%20assinalou.

- [5] Brasil. Ministério da Saúde. “Vigilância e controle do vetor”. Disponível em: <https://www.gov.br/saude/pt-br/assuntos/saude-de-a-a-z/a/aedes-aegypti/vigilancia-entomologica>.
- [6] Aldosery, A. et al. MEWAR: Development of a Cross-Platform Mobile Application and Web Dashboard System for Real-Time Mosquito Surveillance in Northeast Brazil. *Frontiers in Public Health*, 9, 754072. Publicado online em 27 de outubro de 2021. Disponível em: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8578800/>.
- [7] UCL Institute for Risk and Disaster Reduction. Mosquito population modelling for early warning system and rapid health authority response (MEWAR). Disponível em: <https://www.ucl.ac.uk/risk-disaster-reduction/research-projects/2024/jan/mosquito-population-modelling-early-warning-system-and-rapid-health>.
- [8] Indeed. “O que faz um agente de vigilância ambiental em saúde”. Disponível em: <https://br.indeed.com/conselho-de-carreira/encontrando-emprego/que-faz-agente-vigilancia-ambiental#:~:text=Agentes%20de%20vigil%C3%A2ncia%20ambiental%20em%20sa%C3%BAde%20%28AVAS%29%20s%C3%A3o,de%20risco%2C%20visitas%20t%C3%A9cnicas%2C%20pesquisa%20e%20a%C3%A7%C3%B5es%20educacionais>.
- [9] Brasil. Ministério da Saúde. “Programa Nacional de Controle da Dengue - Instituído em 24 de julho de 2002”. Disponível em: https://www.gov.br/saude/pt-br/centrais-de-conteudo/publicacoes/svsa/dengue/pncd_2002.pdf/view.
- [10] Awari by Fluency. “Arquitetura de Software Cliente-Servidor: Estrutura e Interação entre os Componentes”. Disponível em: <https://awari.com.br/arquitetura-de-software-cliente-servidor-estrutura-e-interacao-entre-os-componentes/>.
- [11] Devmedia. “Conhecendo o modelo arquitetural REST”. Disponível em: <https://www.devmedia.com.br/conhecendo-o-modelo-arquitetural-rest/28052>.
- [12] C4Model. “The C4 model for visualising software architecture”. Disponível em: <https://c4model.com/>.
- [13] JSON. “Introducing JSON”. Disponível em: <https://www.json.org/json-en.html>.