



**INSTITUTO FEDERAL**

Rio Grande do Sul

Campus Osório

Vitor Colombo Nunes

**NOME APP: UMA FERRAMENTA  
COLABORATIVA PARA ESTUDO DA FAUNA  
MARINHA NO LITORAL NORTE DO RIO  
GRANDE DO SUL**

Osório

2025

Vitor Colombo Nunes

**NOME APP: UMA FERRAMENTA COLABORATIVA  
PARA ESTUDO DA FAUNA MARINHA NO LITORAL  
NORTE DO RIO GRANDE DO SUL**

Trabalho de Conclusão de Curso apresentado  
como requisito parcial para obtenção do título  
de Tecnólogo em Análise e Desenvolvimento de  
Sistemas.

Orientador: Marcelo Paravisi

Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Sul – IFRS

*Campus Osório*

Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas

Osório

2025

Vitor Colombo Nunes

**NOME APP: UMA FERRAMENTA COLABORATIVA  
PARA ESTUDO DA FAUNA MARINHA NO LITORAL  
NORTE DO RIO GRANDE DO SUL**

Trabalho de Conclusão de Curso apresentado  
como requisito parcial para obtenção do título  
de Tecnólogo em Análise e Desenvolvimento de  
Sistemas.

---

**Marcelo Paravisi**  
Orientador

---

**Professor**  
Convidado 1

---

**Professor**  
Convidado 2

Osório  
2025

## AGRADECIMENTOS

Os agradecimentos principais são direcionados à Gerald Weber, Miguel Frasson, Leslie H. Watter, Bruno Parente Lima, Flávio de Vasconcellos Corrêa, Otavio Real Salvador, Renato Machnievscz<sup>1</sup> e todos aqueles que contribuíram para que a produção de trabalhos acadêmicos conforme as normas ABNT com L<sup>A</sup>T<sub>E</sub>X fosse possível.

Agradecimentos especiais são direcionados ao Centro de Pesquisa em Arquitetura da Informação<sup>2</sup> da Universidade de Brasília (CPAI), ao grupo de usuários *latex-br*<sup>3</sup> e aos novos voluntários do grupo *abnT<sub>E</sub>X2*<sup>4</sup> que contribuíram e que ainda contribuirão para a evolução do abnT<sub>E</sub>X2.

---

<sup>1</sup> Os nomes dos integrantes do primeiro projeto abnT<sub>E</sub>X foram extraídos de <<http://codigolivre.org.br/projects/abntex/>>

<sup>2</sup> <<http://www.cpai.unb.br/>>

<sup>3</sup> <<http://groups.google.com/group/latex-br>>

<sup>4</sup> <<http://groups.google.com/group/abntex2>> e <<http://www.abntex.net.br/>>

## RESUMO

Segundo a [ABNT \(2003, 3.1-3.2\)](#), o resumo deve ressaltar o objetivo, o método, os resultados e as conclusões do documento. A ordem e a extensão destes itens dependem do tipo de resumo (informativo ou indicativo) e do tratamento que cada item recebe no documento original. O resumo deve ser precedido da referência do documento, com exceção do resumo inserido no próprio documento. (...) As palavras-chave devem figurar logo abaixo do resumo, antecedidas da expressão Palavras-chave:, separadas entre si por ponto e finalizadas também por ponto.

**Palavras-chave:** latex. abntex. editoração de texto.

## ABSTRACT

This is the english abstract.

**Keywords:** latex. abntex. text editoration.

## LISTA DE ILUSTRAÇÕES

Figura 1 – Fluxo de trabalho do Jira gerado para o desenvolvimento do projeto. . . . .	15
Figura 2 – Ciclo de desenvolvimento incremental. . . . .	21
Figura 3 – Interface do sistema SISS-Geo. . . . .	24
Figura 4 – Recorte da região costeira do litoral norte do Rio Grande do Sul. . . . .	24
Figura 5 – Interface do Sistema Urubu. . . . .	25
Figura 6 – Captura de tela de uma página <i>web</i> de registro de visualização de ocorrências cadastradas no SIMBA. . . . .	27
Figura 7 – Captura de tela de um mapa georreferenciado gerado a partir dos dados de jornada de monitoramento cadastrado no SIMBA. . . . .	27
Figura 8 – Captura de tela da página <i>web</i> de monitoramento cadastrada no SIMBA. . .	28

## LISTA DE TABELAS

Tabela 1 – Cronograma de desenvolvimento — Cinza: Planejamento inicial, Azul: Re-planejamento, Vermelho: Prazos adiados. . . . .	17
Tabela 2 – Dependências do Projeto <i>Flutter</i> . . . . .	33
Tabela 3 – Requisitos funcionais do sistema . . . . .	37
Tabela 4 – Requisitos não funcionais do sistema . . . . .	38
Tabela 5 – Estrutura de collections do banco de dados . . . . .	38



## **LISTA DE ABREVIATURAS E SIGLAS**

ABNT	Associação Brasileira de Normas Técnicas
abnTeX	ABsurdas Normas para TeX

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>11</b>
<b>2</b>	<b>METODOLOGIA</b>	<b>13</b>
2.1	Metodologia de desenvolvimento de <i>software</i>	14
<b>3</b>	<b>CRONOGRAMA</b>	<b>16</b>
<b>4</b>	<b>REFERENCIAL TEÓRICO</b>	<b>18</b>
4.1	Ciência Cidadã	18
4.2	Agenda 2030	18
4.3	Objetivos de Desenvolvimento Sustentável (ODS)	19
4.4	Metodologia Iterativa Incremental	20
4.5	Kanban	21
4.6	Jira	21
<b>5</b>	<b>TRABALHOS CORRELATOS</b>	<b>23</b>
5.1	SISS-Geo	23
5.2	Sistema Urubu	25
5.3	SIMBA	26
5.4	Análise Comparativa e Direcionamentos do Projeto	28
<b>6</b>	<b>TECNOLOGIAS E FRAMEWORKS</b>	<b>30</b>
6.1	Flutter	30
6.2	Dart	30
6.3	Firebase	31
6.4	Pacotes e Plugins do Flutter	32
6.4.1	Hive	34
6.4.2	Flutter Map	34
6.4.3	Excel	34
6.4.4	Graphic	34
<b>7</b>	<b>RESULTADOS OBTIDOS</b>	<b>35</b>
7.1	Engenharia de Requisitos	35
7.1.1	Requisitos Funcionais	37
7.1.2	Requisitos Não Funcionais	38
7.2	Projeto Comportamental e Estrutural	38
7.2.1	Fluxo de Autenticação	38

7.2.2	Estrutura de Dados . . . . .	38
7.3	<b>Projeto Arquitetural . . . . .</b>	<b>39</b>
7.4	<b>Projeto e Implementação da Interface com o Usuário . . . . .</b>	<b>39</b>
7.5	<b>Projeto Gerencial do Protótipo . . . . .</b>	<b>39</b>
7.6	<b>Implementação do Protótipo . . . . .</b>	<b>39</b>
7.7	<b>Projeto e Execução de Testes e Verificação de Qualidade do Protótipo . .</b>	<b>39</b>
8	<b>CONCLUSÃO . . . . .</b>	<b>40</b>
	<b>REFERÊNCIAS . . . . .</b>	<b>41</b>
	<b>APÊNDICE A – QUISQUE LIBERO JUSTO . . . . .</b>	<b>43</b>
	<b>APÊNDICE B – NULLAM ELEMENTUM URNA VEL IMPERDIET SODALES ELIT IPSUM PHARETRA LIGULA AC PRETIUM ANTE JUSTO A NULLA CURABITUR TRISTIQUE ARCU EU METUS . . . . .</b>	<b>44</b>
	<b>APÊNDICE A – MORBI ULTRICES RUTRUM LOREM. . . . .</b>	<b>45</b>
	<b>ANEXO B – CRAS NON URNA SED FEUGIAT CUM SOCIIS NA- TOQUE PENATIBUS ET MAGNIS DIS PARTURIENT MONTES NASCETUR RIDICULUS MUS . . . . .</b>	<b>46</b>
	<b>ANEXO C – FUSCE FACILISIS LACINIA DUI . . . . .</b>	<b>47</b>

## 1 INTRODUÇÃO

Este projeto de conclusão de curso é fruto de uma colaboração com o CECLIMAR (Centro de Estudos Costeiros Limnológicos e Marinhos), onde se identificou a necessidade de aprimorar o processo de coleta e monitoramento de dados da fauna costeira. Diante dos desafios atuais, propõe-se o desenvolvimento de um aplicativo de Ciência Cidadã (MARTINS; CABRAL, 2021). Este tipo de aplicativo permite que o público geral contribua com dados científicos, aumentando o alcance e a eficiência da pesquisa. Atualmente, no CECLIMAR, a coleta dos dados de monitoramento é realizada manualmente: as pessoas enviam informações via WhatsApp (2024), e um pesquisador do órgão encaminha os dados principais para uma bolsista, que os classifica e registra em uma planilha eletrônica, armazenando as fotos em uma pasta do Google (2024b). Este método manual tem levado a inconsistências nos dados e exigido revisões periódicas pelo gestor do projeto.

A solução proposta é a criação de uma aplicação que facilite o registro de observações pela população. Posteriormente, um pesquisador do CECLIMAR realizará a classificação da espécie do animal por meio do sistema, indicando características como a espécie e o estado de decomposição. A automação desse processo visa reduzir as inconsistências e otimizar a gestão dos dados coletados. Este projeto tem como objetivo geral desenvolver um aplicativo de Ciência Cidadã para otimizar o processo de coleta, classificação e gestão de dados da fauna costeira voltado para atender as demandas de profissionais do CECLIMAR.

A aplicação possui como objetivos específicos automatizar o processo de coleta e armazenamento das ocorrências para garantir precisão dos dados e facilitar os registros de observações da fauna costeira com a ajuda da população a partir de uma interface amigável e intuitiva. Padronizar e reduzir as inconsistências nos registros, para proporcionar um banco de dados robusto e conciso pensando em minimizar a necessidade de revisões periódicas dos dados, e facilitar a realização de pesquisas e a análise de dados recebidos, liberando recursos para outras atividades de pesquisa. Além de promover a participação ativa da comunidade na conservação da biodiversidade costeira e no monitoramento ambiental.

Com este contexto, podemos afirmar que este trabalho possui seu desenvolvimento alinhado com a Agenda 2030 da ONU (NAÇÕES UNIDAS, 2015a), usando a integração e aplicação de tecnologias no desenvolvimento sustentável, visando abranger os itens 14 (Vida na água), 15 (Vida terrestre) e 9 (Indústria, inovação e infraestrutura). Além disso, o projeto busca promover a difusão e aplicação dos princípios da ciência cidadã, ao facilitar a colaboração entre a comunidade e cientistas. A ciência cidadã amplia a participação pública na pesquisa científica, proporcionando uma abordagem colaborativa e inclusiva na gestão ambiental. Portanto, este trabalho busca não apenas oferecer soluções práticas para o monitoramento da fauna na região costeira do Rio Grande do Sul, mas também promover uma mudança de paradigma na

forma como a ciência é realizada, enfatizando a importância da participação e colaboração da comunidade na construção de um futuro sustentável.

## 2 METODOLOGIA

Para a realização do embasamento deste trabalho, foram utilizadas tanto a metodologia de pesquisa bibliográfica quanto a de pesquisa documental. A primeira foi essencial para o levantamento de metodologias já consolidadas e amplamente estudadas, como as que serão abordadas no referencial teórico e, a seguir, nesta seção. Já a segunda foi empregada para identificar diferentes aplicações correlatas e para a elaboração do referencial teórico, além de ter sido utilizada na análise de dados internos do CECLIMAR, conforme comentado na introdução deste trabalho.

Segundo Gil et al. (), a pesquisa bibliográfica é desenvolvida com base em materiais como livros e artigos científicos, ou seja, materiais já consolidados. Trata-se de uma pesquisa de grande importância, pois permite que os pesquisadores acessem diversos dados e informações dispersos que, individualmente, seriam muito trabalhosos e custosos de se coletar. Nesse tipo de pesquisa, entretanto, é necessário ter cuidado com citações de terceiros, que podem interpretar de forma equivocada algum dado ou informação originalmente levantados.

Ainda segundo o autor, a pesquisa documental se diferencia pela natureza das fontes de informação. Enquanto as pesquisas bibliográficas consistem essencialmente em um apanhado de contribuições de diversos autores sobre determinado assunto, a pesquisa documental ocorre por meio de materiais que ainda não receberam tratamento analítico ou que podem ser reelaborados, a depender dos objetos de pesquisa. As fontes da pesquisa documental são mais diversas e podem incluir conversas pessoais, entrevistas, documentos ou sites.

A metodologia de pesquisa bibliográfica foi realizada através da plataforma Google Scholar, publicações presentes no portal do Sistema de Informação sobre a Biodiversidade Brasileira (SiBBR), além de livros disponibilizados na biblioteca do Instituto Federal Campus Osório. Já a metodologia de pesquisa documental foi levantada a partir de arquivos internos, relatórios das Nações Unidas e conteúdos disponibilizados por desenvolvedores ou organizações que participaram do desenvolvimento das aplicações correlatas.

Após a fase de revisão bibliográfica se iniciará o desenvolvimento do sistema. Esta fase será realizada a partir do levantamento de requisitos junto de profissionais do CECLIMAR e, os requisitos levantados serão cadastrados e refinados para desenvolvimento cíclico do sistema. Cada ciclo visará a entrega de um produto com incrementos de requisitos pré estabelecidos. Ao final de cada um dos ciclos de desenvolvimento o sistema será disponibilizado para testes com servidores do CECLIMAR, os *feedbacks* recebidos serão analisados, refinados e postos para desenvolvimento no ciclo seguinte. Um ponto de atenção no desenvolvimento desse sistema é que, por se tratar de um sistema de ciência cidadã, deve estar adequado com a LGPD para que possa ser publicado na Play Store para o uso da sociedade.

## 2.1 METODOLOGIA DE DESENVOLVIMENTO DE SOFTWARE

Para o desenvolvimento deste projeto foi escolhida uma abordagem de metodologia ágil relacionada ao ciclo iterativo incremental. Utilizando o Kanban como método de gestão de fluxo de trabalho, a fim de melhorar a eficiência e qualidade do produto final a partir da visualização das tarefas. A ferramenta escolhida para realizar este gerenciamento foi o Jira.

Segundo [Pressman \(2011\)](#), os ciclos de desenvolvimento incremental podem ser divididos em 5 principais etapas: comunicação, planejamento, modelagem (análise e projeto), construção (codificação e testes) e emprego (entrega, *feedback*). As etapas por ele descritas serão aplicadas neste projeto.

A comunicação será marcada por reuniões agendadas com os profissionais do CECLIMAR para definição de escopo e levantamento de requisitos do sistema. O planejamento será a análise, o refinamento e as definições de quais funcionalidades serão desenvolvidas em cada ciclo. Durante a modelagem será realizada a prototipação e análise dos pontos levantados na etapa anterior. Na construção será o momento onde se dará a codificação e os testes da aplicação. E, por fim, durante o emprego o sistema será disponibilizado para os profissionais do CECLIMAR e uma porcentagem de usuários que realizarão testes e retornarão *feedback* que serão analisados, catalogados e inseridos no *Backlog* para serem puxados em um ciclo posterior.

O uso desta metodologia tem o objetivo de realizar uma primeira entrega que possa ser considerada um Mínimo Produto Viável (MPV) atendendo aos requisitos básicos propostos inicialmente, mesmo que ainda se note a ausência de funcionalidades complementares. Esse MVP se tornará então a base de avaliação que permitirá identificar necessidades adicionais e ajustes necessários. Com base nessa análise, é planejado o próximo incremento, ajustando a primeira entrega e adicionando novas funcionalidades conforme as necessidades.

Neste projeto foi montado um fluxo de trabalho no Jira para desenvolvimento de *software* visando auxiliar no processo e manter a visibilidade das tarefas de ponta a ponta. Para o workflow principal, foi montado um esquema com *Backlog*, Refinamento, Em desenvolvimento, Aguardando teste, Em teste, Correção de *bugs* e *Done* (Figura 1).

O *Backlog* é a coluna onde todas as tarefas, user stories e *bugs* serão inicialmente posicionados. Nesta etapa, as tarefas serão priorizadas antes de andarem para o próximo estágio.

No Refinamento, as tarefas puxadas do *Backlog* são detalhadas para um desenvolvimento mais assertivo. São refinados critérios de aceitação, estimativas de tempo e algum débito técnico.

As tarefas que estiverem em desenvolvimento são as que tiveram, efetivamente, o seu desenvolvimento iniciado. Assim que o desenvolvimento estiver finalizado, as tarefas serão transferidas para aguardando teste, onde ficarão até serem puxadas para testes mais detalhados.

No estágio de teste, os critérios de aceite e a presença de *bugs* serão testados com o intuito de manter a qualidade do produto final. As tarefas que tiverem *bugs* ou divergências de regras de

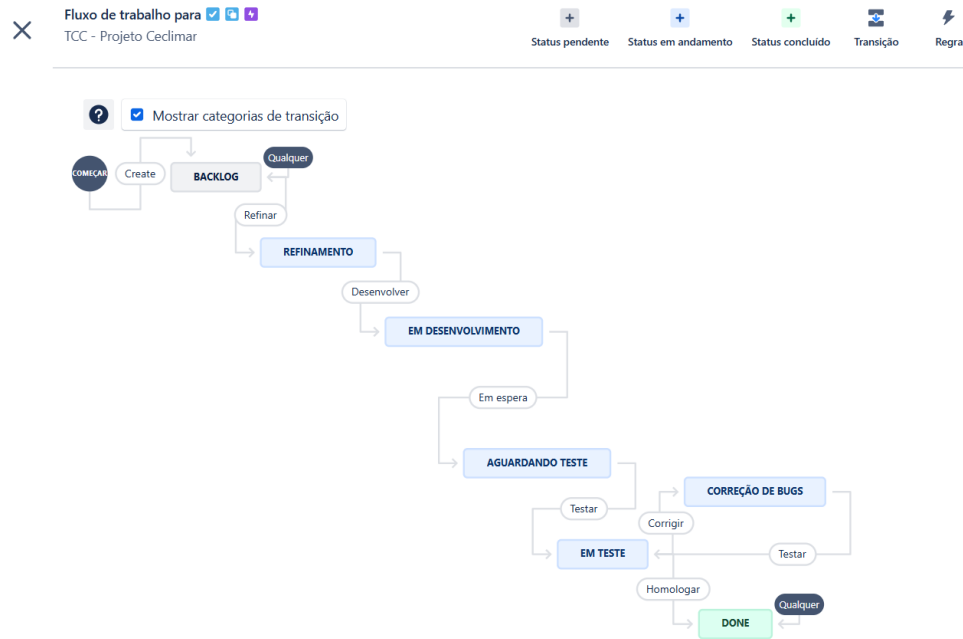


Figura 1 – Fluxo de trabalho do Jira gerado para o desenvolvimento do projeto.

Fonte: Autor

negócio identificadas serão movidas para a coluna Correção de *bugs* para que sejam corrigidas.

E, por último, após a homologação das tarefas nas etapas anteriores as tarefas são movidas para *Done* que indicará sua finalização.



### 3 CRONOGRAMA

O cronograma ilustrado na Tabela 1 representa a trajetória de desenvolvimento do projeto ao longo de 16 meses, bem como as divergências e replanejamentos que foram realizados no decorrer do tempo. O trabalho teve seu início no dia 28 de março de 2024, com a realização de uma reunião de definição de escopo junto do orientador Marcelo Paravisi. No dia 1º de abril, deu-se continuidade com a definição do tema, em uma reunião externa com um membro do CECLIMAR. Essa etapa foi fundamental para delimitar o foco do trabalho e aprofundar a compreensão das dificuldades, necessidades, oportunidades e pontos críticos do projeto

Os meses seguintes, de maio a julho de 2024, foram dedicados à revisão bibliográfica e à definição metodológica. A definição metodológica se deu durante o mês de maio tendo início no dia 06, enquanto a parte de revisão bibliográfica se iniciou no dia 05 do mesmo mês e foi finalizada no final de julho.

O levantamento de requisitos e regras de negócio se deu a partir do dia 10 de maio e se estendeu até o final de agosto para que os alinhamentos de definições com a equipe do CECLIMAR fossem mais abertos e constantes. É importante ressaltar que durante o desenvolvimento houve mudanças de funcionalidades e ajustes de regras de negócio durante o período de testes que fizeram com que fosse necessário revisar esse tópico.

Esse período inicial de estruturação se mostrou essencial para assegurar uma base sólida para iniciar o desenvolvimento. A parte de desenvolvimento técnica se iniciou com a organização da implementação do sistema no dia 20 de maio de 2024 e, com base nisso, se iniciou a codificação do aplicativo no dia 1º de junho do mesmo ano.

Inicialmente, o desenvolvimento tinha um planejamento de conclusão para o final de outubro, porém, embora as atividades tenham progredido conforme o cronograma, em uma reunião com o orientador do projeto foi decidido realizar uma alteração no cronograma adicionando uma etapa de testes disponibilizando a aplicação com testadores tanto do projeto parceiro, como terceiros que contribuíram para que a aplicação adquirisse uma maturidade maior. Com isso, o prazo final de codificação foi replanejado para o início de maio de 2025 (Tabela 1).

A parte de redação da parte escrita do trabalho de conclusão foi iniciada paralelamente com a codificação para obter uma documentação mais precisa do processo e estava planejada para ser concluída até o final de novembro de 2024, porém, também foi afetada pelo replanejamento. Em vermelho na Tabela 1 podemos ver que esta etapa foi realizada dentro do prazo previsto até o mês de outubro, porém em novembro foi adiada para maio e junho para priorizar a conclusão da codificação e o aprimoramento da qualidade da aplicação. O desenvolvimento continuou sendo a atividade central até maio de 2025, acompanhado de execuções constantes de testes. No mês de maio de 2025 a redação do trabalho retornou e se estendeu, junto da revisão textual, até o final

de junho para ser entregue dentro da data máxima de 26 de junho. A apresentação para a banca até o momento havia sido definida, porém tem prazo máximo de 11 de julho de 2025.

Esse cronograma evidencia o fluxo de trabalho realizado desde o início do planejamento do projeto, bem como as alterações ocorridas durante seu desenvolvimento. Para a elaboração desse material, foi fundamental que os períodos de escrita e codificação estivessem bem alinhados, permitindo traçar e documentar, com maior precisão, a linha do tempo apresentada, desde o início até a entrega final planejada.

Meses	Reunião def. de escopo	Definição de tema	Revisão bibliográfica	Def. metodológica	Levantamento de requisitos	Organização de implementação	Desenvolvimento	Redação de TCC	Revisão textual	Apresentação	Testes
Mar/24											
Abr/24											
Mai/24											
Jun/24											
Jul/24											
Ago/24											
Set/24											
Out/24											
Nov/24											
Dez/24											
Jan/25											
Fev/25											
Mar/25											
Abr/25											
Mai/25											
Jun/25											

Tabela 1 – Cronograma de desenvolvimento — Cinza: Planejamento inicial, Azul: Replanejamento, Vermelho: Prazos adiados.

Fonte: Autor

## 4 REFERENCIAL TEÓRICO

### 4.1 CIÊNCIA CIDADÃ

A Ciência Cidadã representa uma ponte entre a comunidade científica e o público geral, permitindo que pessoas sem formação científica formal possam contribuir em pesquisas científicas. Essa metodologia colaborativa tem se destacado em diversas áreas de pesquisa como na conservação da biodiversidade e na preservação ambiental. Através da Ciência Cidadã é possível que voluntários colem e analisem dados, fornecendo informações que agregam no conhecimento acadêmico e auxiliam na resolução de questões sociais.

Segundo [Palma \(2016\)](#), trata-se de uma metodologia de pesquisa promissora na produção de conhecimento para ser aplicada em diversos campos científicos. Essa abordagem se destaca, em especial, com seu potencial de geração de dados e análises, temporal e espacial, quando comparada com os métodos tradicionais de pesquisa. Para [Wildschut \(2017\)](#), a metodologia da ciência cidadã tem potencial para ampliar o escopo de pesquisas e aumentar e aprimorar a capacidade na coleta de dados e que os cidadãos que participam podem contribuir com informações importantes enquanto aprendem sobre as mais diversas áreas científicas.

A construção de conhecimento colaborativo realizado entre cidadãos e cientistas se mostra uma maneira poderosa de construção de conhecimentos, que agrega tanto no meio científico quanto social. Estes projetos instigam que as pessoas participem de maneira voluntária e ativa na resolução de situações do dia-a-dia da nossa sociedade, disseminando conhecimento de diversas áreas e fazendo com que diversos conteúdos saiam de suas bolhas científicas e obtenham um alcance maior.

### 4.2 AGENDA 2030

A Agenda 2030 para o Desenvolvimento Sustentável é um plano de ação global adotado pelas Nações Unidas em 2015, com o objetivo de promover a prosperidade enquanto protege o planeta. Ela estabelece 17 Objetivos de Desenvolvimento Sustentável (ODS), que são subdivididos em 169 metas específicas. Esses objetivos englobam uma ampla gama de questões sociais, econômicas e ambientais, como erradicação da pobreza, igualdade de gênero, educação de qualidade, água limpa e saneamento, energia acessível e não poluente, trabalho decente e crescimento econômico ([NAÇÕES UNIDAS, 2015b](#)).

Segundo a organização, os ODS são projetados com os três pilares do desenvolvimento sustentável: econômico, social e ambiental. A Agenda enfatiza a importância de garantir que os direitos humanos de todos sejam realizados e que haja igualdade de gênero e empoderamento de mulheres e meninas. Além disso, reconhece que a erradicação da pobreza em todas as suas

formas é o maior desafio global e, um requisito indispensável para o desenvolvimento sustentável.

A implementação da Agenda 2030 requer a mobilização de recursos e uma Parceria Global para o Desenvolvimento Sustentável, envolvendo todos os países, partes interessadas e pessoas. A Agenda é mundial, e se aplica a todos os países levando em conta diferentes realidades nacionais, capacidades e níveis de desenvolvimento. Ela promove a paz, justiça e instituições eficazes, e destaca a necessidade de ações urgentes sobre a mudança climática para proteger o planeta para as gerações presentes e futuras (NAÇÕES UNIDAS, 2015a).

#### 4.3 OBJETIVOS DE DESENVOLVIMENTO SUSTENTÁVEL (ODS)

Os ODS são um conjunto de 17 metas estabelecidas pelas Nações Unidas para abordar os principais desafios de desenvolvimento no Brasil e em todo o mundo. Esses objetivos visam criar um futuro mais justo, equitativo e sustentável para todos (NAÇÕES UNIDAS, 2015b). São eles:

1. **ODS 1:** Erradicação da Pobreza: Acabar com a pobreza em todas as suas formas, em todos os lugares.
2. **ODS 2:** Fome Zero e Agricultura Sustentável: Garantir a segurança alimentar, melhorar a nutrição e promover a agricultura sustentável.
3. **ODS 3:** Saúde e Bem-Estar: Assegurar uma vida saudável e promover o bem-estar para todas as idades.
4. **ODS 4:** Educação de Qualidade: Garantir uma educação inclusiva, equitativa e de qualidade.
5. **ODS 5:** Igualdade de Gênero: Alcançar a igualdade de gênero e empoderar todas as mulheres e meninas.
6. **ODS 6:** Água Limpa e Saneamento: Garantir a disponibilidade e gestão sustentável da água e saneamento para todos.
7. **ODS 7:** Energia Limpa e Acessível: Assegurar o acesso a fontes de energia acessíveis, confiáveis, sustentáveis e modernas.
8. **ODS 8:** Trabalho Decente e Crescimento Econômico: Promover o crescimento econômico inclusivo e sustentável, emprego pleno e produtivo e trabalho decente para todos.
9. **ODS 9:** Indústria, Inovação e Infraestrutura: Construir infraestruturas resilientes, promover a industrialização inclusiva e sustentável e fomentar a inovação.
10. **ODS 10:** Redução das Desigualdades: Reduzir as desigualdades dentro e entre países.
11. **ODS 11:** Cidades e Comunidades Sustentáveis: Tornar as cidades e os assentamentos humanos inclusivos, seguros, resilientes e sustentáveis.

12. **ODS 12:** Consumo e Produção Responsáveis: Assegurar padrões de consumo e produção sustentáveis.
13. **ODS 13:** Ação Contra a Mudança Global do Clima: Tomar medidas urgentes para combater a mudança climática e seus impactos.
14. **ODS 14:** Vida na Água: Conservar e usar de forma sustentável os oceanos, mares e recursos marinhos.
15. **ODS 15:** Vida Terrestre: Proteger, restaurar e promover o uso sustentável dos ecossistemas terrestres, gerir florestas de forma sustentável, combater a desertificação e deter a perda de biodiversidade.
16. **ODS 16:** Paz, Justiça e Instituições Eficazes: Promover sociedades pacíficas e inclusivas para o desenvolvimento sustentável, proporcionar acesso à justiça para todos e construir instituições eficazes, responsáveis e inclusivas em todos os níveis.
17. **ODS 17:** Parcerias e Meios de Implementação: Fortalecer os meios de implementação e revitalizar a parceria global para o desenvolvimento sustentável.

Os itens acima são um apelo global visando acabar com a pobreza, proteger o meio ambiente e o clima e garantir que as pessoas, em todos os lugares, possam desfrutar de paz e de prosperidade. São objetivos nos quais as Nações Unidas estão contribuindo para atingir a Agenda 2030 no Brasil ([NAÇÕES UNIDAS, 2015b](#)).

#### 4.4 METODOLOGIA ITERATIVA INCREMENTAL

A metodologia iterativa incremental é uma abordagem de desenvolvimento de *software* que divide o processo em ciclos repetitivos. Cada iteração resulta em uma versão incrementada do *software*, que é construída sobre a versão anterior com adições e melhorias. Segundo [Pressman \(2011\)](#), essa metodologia permite que as equipes avaliem e integrem feedbacks mais rapidamente, adaptando-se às mudanças e refinando o produto ao longo do tempo. Isso contrasta com o modelo tradicional em cascata, onde cada fase deve ser concluída antes da próxima começar, sem retorno para fases anteriores.

De acordo com [Pressman \(2011\)](#), o modelo incremental é uma abordagem de desenvolvimento de *software* que foca na entrega gradual de funcionalidades em sucessivas versões incrementais. Nesse modelo, o desenvolvimento do *software* é dividido em várias partes ou incrementos, cada um entregando uma versão operante do produto, que é aprimorada e expandida em lançamentos subsequentes.

[Pressman \(2011\)](#) afirma que o modelo incremental integra elementos de fluxos de processos lineares e paralelos. Na Figura 2 é possível observar essa relação a partir das sequências

lineares de forma escalonada que demonstram que ao longo do tempo cada incremento adiciona funcionalidades ou aprimoramentos ao sistema, permitindo uma evolução constante e contínua do produto final.

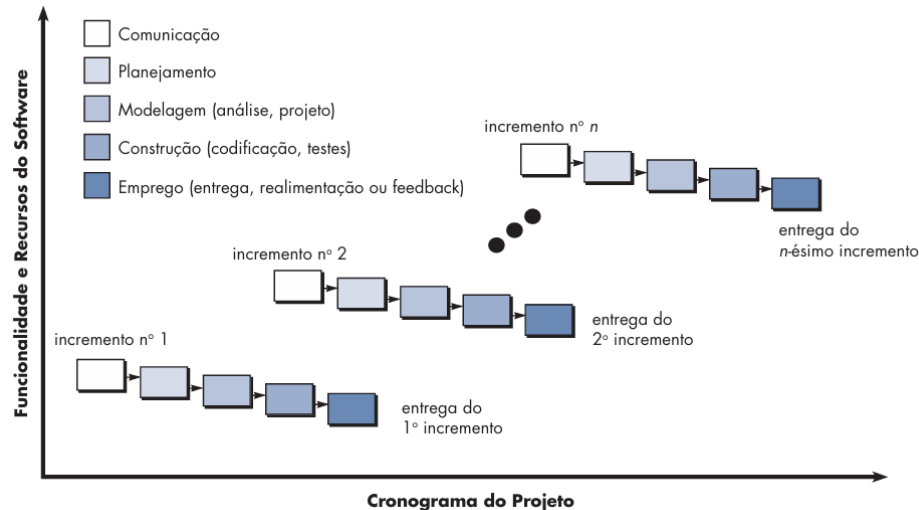


Figura 2 – Ciclo de desenvolvimento incremental.

Fonte: [Pressman \(2011\)](#).

#### 4.5 KANBAN

O Kanban é um método de gestão de fluxo de trabalho que fornece uma visualização das tarefas que devem ser realizadas, quando entregá-las e quanto ainda é necessário para finalizá-las com o objetivo de aumentar a eficiência e o entendimento do desenvolvimento ([OHNO, 1988](#)). Originou-se no sistema da Toyota no Japão dos anos 1940 e foi adaptado para o desenvolvimento de *software* e sistemas de TI por David J. Anderson em 2004. Para isso, o Kanban utiliza de um quadro dividido em colunas para representar os diferentes estágios do trabalho, onde os cartões que representam as tarefas se movem de uma coluna para outra, refletindo o progresso real ([ZAYAT; SENVAR, 2020](#)).

#### 4.6 JIRA

O Jira é um *software* comercial de gerenciamento de projetos desenvolvido pela empresa Atlassian. Ele permite a criação, acompanhamento e gerência de tarefas e projetos a partir de uma interface personalizável que suporta diversas metodologias ágeis como Scrum e Kanban, facilitando a colaboração e comunicação durante o desenvolvimento.

A ferramenta pode ser usada para planejar *sprints*, atribuir tarefas, acompanhar *bugs*, gerar relatórios e analisar o desempenho da equipe. Ele possui integração com uma variedade de ferramentas de desenvolvimento e oferece funcionalidades para personalizar fluxos de tra-

balho, campos e painéis, tornando-o adaptável às necessidades específicas de cada projeto ou organização.

## 5 TRABALHOS CORRELATOS

Com o intuito de contextualizar e ter uma melhor visão de onde posicionar o projeto dentro do campo de pesquisa escolhido, foi realizada uma seleção de trabalhos correlatos, cujos principais critérios de aceitação incluíram a abordagem de temas como: monitoramento de fauna, desenvolvimento sustentável, gestão ambiental e ciência cidadã. Garantindo assim, relevância e alinhamento do projeto com as necessidades e avanços da área.

O ponto de partida para as pesquisas foi o [SIBBr \(2024\)](#). A plataforma online, pertencente ao sistema gov.br ([Governo Federal do Brasil, 2024](#)), que integra dados e informações sobre a biodiversidade e os ecossistemas de diversas fontes, os tornando acessíveis e livres para usos diversos. Nele, também é possível ter acesso ao sistema Ciência Cidadã, que consiste em uma colaboração entre a comunidade e cientistas na coleta de dados para pesquisa científica.

A partir do levantamento bibliográfico, pesquisas na plataforma Play Store ([Google, 2024c](#)), presente nos dispositivos Android ([Google, 2024a](#)), foram realizadas baseadas em alguns projetos e palavras-chave pré-selecionadas. As principais palavras-chave usadas para realizar as buscas foram: animais costeiros, coleta de dados, monitoramento ambiental, ciência cidadã.

Nesta seção, serão apresentadas três aplicações separadas no levantamento documental que demonstraram possuir características e funcionalidades que podem auxiliar na definição e desenvolvimento deste sistema. Estas aplicações possuem algumas características em comum, porém cada uma delas também traz características únicas que serão importantes balizadoras nas tomadas de decisão deste projeto.

### 5.1 SISS-GEO

O Sistema de Informação em Saúde Silvestre (SISS-Geo) da FIOCRUZ é desenvolvido pela Plataforma Institucional Biodiversidade e Saúde Silvestre, com apoio do Laboratório Nacional de Computação Científica. É gratuito, disponível em *smartphones* e na *web*, para o monitoramento da saúde dos animais silvestres em ambientes naturais, rurais e urbanos. Apoia a investigação da ocorrência de agentes causadores de doenças, como agentes infecciosos, que podem acometer pessoas e animais. Como instrumento de ciência cidadã torna possível, a partir de registros realizados por cidadãos comuns, profissionais de saúde, meio ambiente, pesquisadores e especialistas em vida silvestre, agir para a prevenção e controle de zoonoses e a conservação da biodiversidade brasileira ([CHAME et al., 2015](#)).

A aplicação *mobile* foi lançada em 2014 e está disponível tanto para IOS quanto para Android (Figura 3) e possui mais de 10.000 downloads, com avaliação média de 4,7/5 baseada em 116 avaliações de usuários. Foram registrados, até 26 de março de 2024, 33 mil registros e quase 13 mil usuários colaboradores.



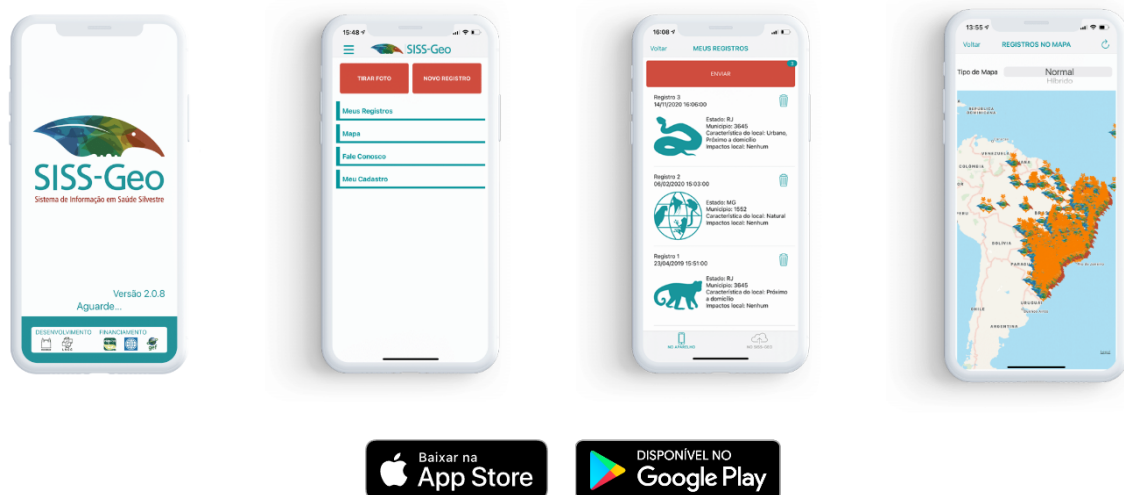


Figura 3 – Interface do sistema SISS-Geo.

Fonte: SISS-Geo (2024).

O sistema é bem consolidado e amplamente utilizado no Brasil, possuindo uma ótima aceitação entre especialistas e cidadãos em diversas regiões do país. Apesar de possuir opções para identificação de dados da fauna costeira, estes se mostram limitados quando comparados com os demais biomas. Ao alterar a escala de análise, é possível perceber, com um recorte mais detalhado do cordão litorâneo, que este não é o principal foco da aplicação e, atualmente, ela possui uma participação muito maior nas regiões continentais (Figura 4).

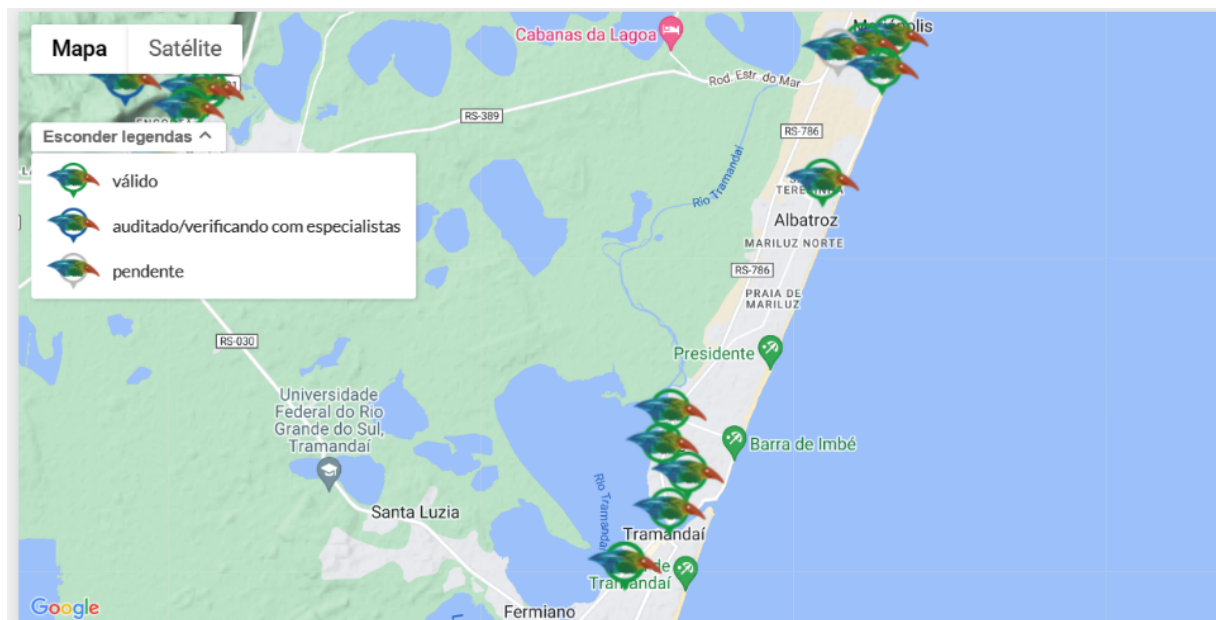


Figura 4 – Recorte da região costeira do litoral norte do Rio Grande do Sul.

Fonte: SISS-Geo (2024).

Principais pontos fortes: funcionalidade *offline*, iniciativa ambiental de grande contribuição, aplicação leve, ideia colaborativa de preservação, georreferenciamento de dados com boa visualização.

Entre as principais reclamações dos usuários estão: sistema pouco intuitivo, poucas opções de animais pré-cadastrados, interface confusa com formulário por vezes muito técnico.

## 5.2 SISTEMA URUBU

O Sistema Urubu é uma iniciativa do Centro Brasileiro de Ecologia de Estradas da UFLA, sob a coordenação do professor Alex Bager. Criado em 2014, o aplicativo de ciência cidadã é a maior rede para conservação da biodiversidade brasileira, destinada à coleta e gestão de informações de fauna selvagem ao longo de rodovias e ferrovias no Brasil.

A aplicação permite que voluntários enviem registros de animais atropelados e infraestruturas viárias por meio de aplicativo móvel, enquanto especialistas validam e caracterizam esses registros para torná-los confiáveis. Os dados coletados são centralizados em um banco de dados e disponibilizados em um Sistema de Informações Geográficas, facilitando a visualização e análise pelos usuários. O Sistema Urubu também oferece ferramentas como o Urubu *web*, para gestão e validação dos dados, e o Urubu Map, para visualização geográfica dos registros (CASTRO; BAGER, 2019). Ao longo de seus anos de existência, o sistema reuniu mais de 25 mil usuários e 150 mil registros de animais atropelados em todo o território brasileiro, demonstrando seu impacto e relevância na conservação da biodiversidade.

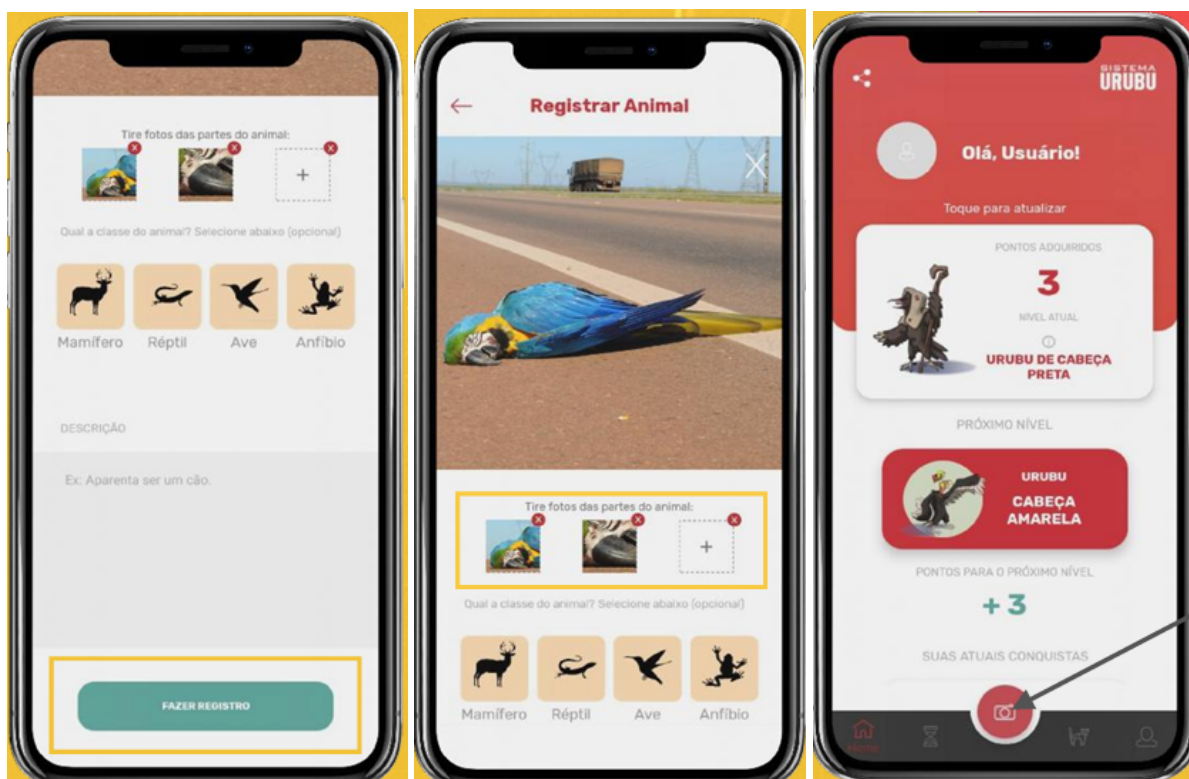


Figura 5 – Interface do Sistema Urubu.

Fonte: Capturas de tela do sistema urubu disponíveis no manual de uso do aplicativo.

O Sistema Urubu apresenta uma interface amigável e um fluxo de obtenção e envio de dados de registros intuitivo (Figura 5.a e 5.b), além de um fluxo gamificado com progressão

recompensada para os usuários que contribuem com o projeto (Figura 5.c).

O fluxo de funcionamento da aplicação se inicia com a morte de algum animal em uma rodovia ou ferrovia, quando esse animal é encontrado por usuário a entrada de registro pode ser realizada via *mobile*, *web* ou por importação via planilha de Excel. Os dados de cada registro são então passados por uma análise de profissionais que decidirão se serão inseridos nos dados finais.

Apesar de ter alcançado importantes números de aceitação e uso entre 2014 e 2023, atualmente o sistema se encontra desativado e indisponível para download em todas as plataformas e seu site se encontra fora do ar.

Em contato com o coordenador do projeto Alex Brag por e-mail, foi informado de que o Sistema Urubu foi desativado devido a falta de recursos. Segundo seu relato, além dos custos para manter a aplicação no ar com investimentos contínuos, os aplicativos de ciência cidadã requerem muita comunicação e relacionamento com os participantes, o que torna a estrutura mais complexa e custosa.

### 5.3 SIMBA

O Sistema de Monitoramento da Biota Aquática (SIMBA), é um sistema *web* de gerenciamento de dados criado pelo Laboratório de Oceanografia Biológica da UNIVALI com a finalidade de armazenar dados coletados por instituições executoras dos projetos de monitoramentos de praias. O desenvolvimento do SIMBA se iniciou para auxiliar nos fluxos de dados entre os atores sociais envolvidos nos Projetos de Monitoramento de Praias (PMPs) e fazer com que os dados obtidos sejam disponibilizados e divulgados para a população.

Os PMPs são desenvolvidos para o atendimento de condicionantes de licenciamento ambiental federal, conduzido pelo IBAMA, de atividades de exploração e produção de petróleo e gás natural de bacias *offshore* sob atuação da Petrobras. Estes projetos tem o objetivo de avaliar as possíveis interferências na área de abrangência dos projetos, analisando tanto tetrápodes marinhos (aves, tartarugas e mamíferos) por meio do monitoramento das praias, atendimento veterinário aos animais debilitados e da coleta de dados de animais mortos, quanto resíduos sólidos encontrados (Petrobras, 2024).

Atualmente os projetos estão presentes nas bacias de Santos, Campos, Espírito Santo, Sergipe-Alagoas e Potiguar.

O SIMBA conta com funcionalidades de cadastro de ocorrências de fauna, resíduos sólidos (Figura 6), exames, jornadas de campo com os caminhamentos realizados. Os pontos de ocorrência de cada cadastro são georreferenciados, assim como as jornadas de monitoramento, gerando mapas para melhor visualização e análise dos dados (Figura 7).

O sistema possui uma área de acesso liberada ao público onde é possível visualizar dados já levantados e validados e uma área de acesso restrito liberada para pesquisadores. Possui uma

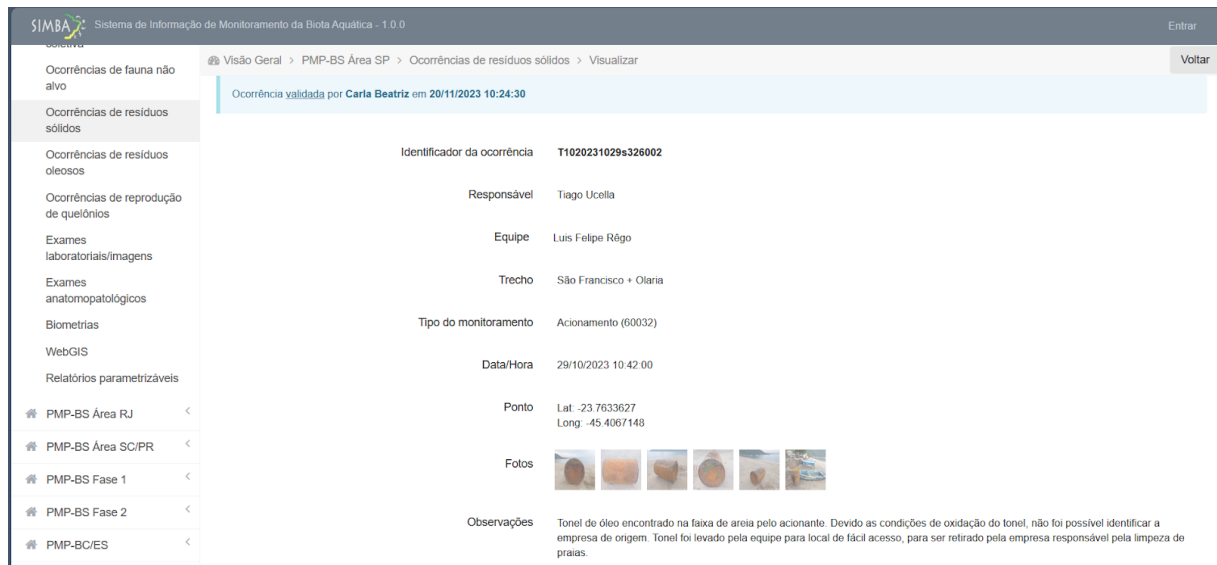


Figura 6 – Captura de tela de uma página *web* de registro de visualização de ocorrências cadastradas no SIMBA.

Fonte: (Petrobras, 2024).

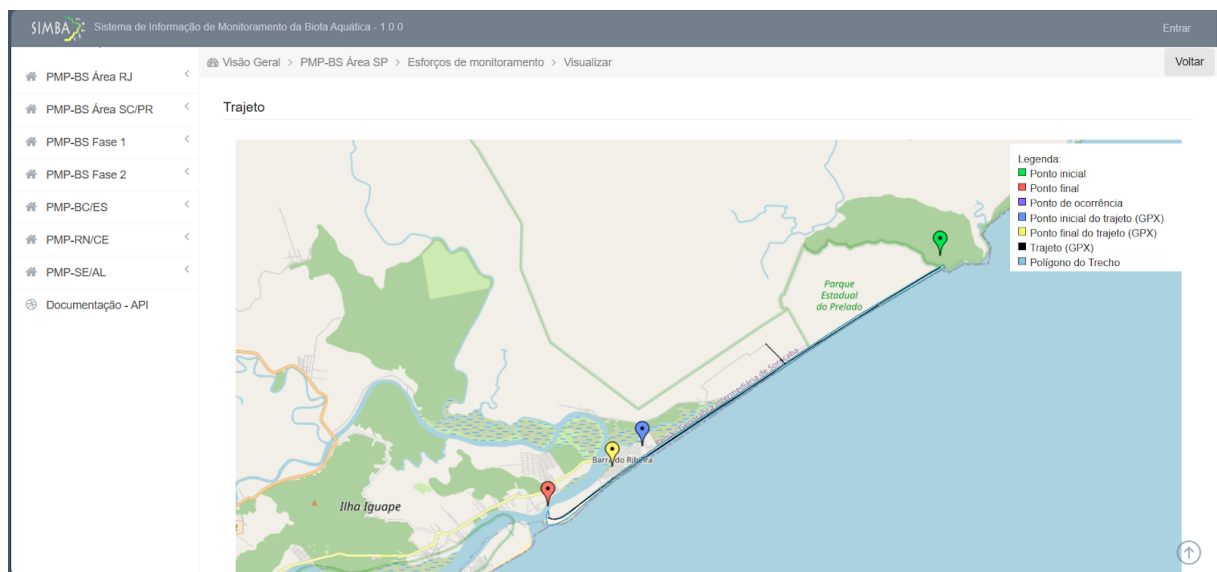


Figura 7 – Captura de tela de um mapa georreferenciado gerado a partir dos dados de jornada de monitoramento cadastrado no SIMBA.

Fonte: (Petrobras, 2024).

interface mais limpa e direta, atendendo a um estilo de aplicação profissional apenas com o conteúdo necessário (Figura 8). Os dados são bem organizados e acessíveis para quem estiver interessado em acompanhar os estudos e evidências coletadas. Porém, para leigos essa interface pode, inicialmente, trazer um pouco de estranheza devido ao visual menos apelativo e aos termos mais científicos apresentados.

A geração de mapas georreferenciados e a possibilidade de uso *offline* são aspectos importantes a serem destacados, pois permitem uma melhor análise e acompanhamento gráfico

Registro informado pela Instituição Executora **Trecho 07**

Registro criado por **Pedro Sardinha do Prado** em 13/12/2023 12:11:00 e alterado por **Shany Nagaoka** em 19/12/2023 15:53:38

**Informações do monitoramento**

Código	Tipo	Tipo de veículo	Trecho	Equipe	Responsável	Data/Hora início	Data/Hora término	Ponto inicial	Ponto final	Completo	Arquivo gpx
1633906	Terrestre	Quadriciclo	Iguape - Praia da Jureia	Não informado	Pedro Sardinha do Prado	13/12/2023 12:11	13/12/2023 12:57	Lat: -24.5735929 Long: -47.2452564	Lat: -24.6714038 Long: -47.4142526	Sim	IPeC_13-DEZ-23_Jureia.gpx

**Condições do ambiente durante o monitoramento**

	Condição do céu	Condição do mar	Maré	Vento	Direção do vento	Observações climáticas
Início do monitoramento	Parcialmente nublado	2	Vazia	3 - Brisa suave - 12 até 19 km/h	Leste	Não informado
Fim do monitoramento	Aberto	3	Enchente	4 - Brisa fraca - 20 até 29 km/h	Leste	Não informado

Ocorrências registradas

Figura 8 – Captura de tela da página web de monitoramento cadastrada no SIMBA.

Fonte: (Petrobras, 2024).

dos dados coletados e possibilitam que estes dados sejam obtidos até mesmo em locais mais isolados sem conexão com a internet. Outra função interessante é a liberação de dados para o público geral que se mostrar interessado em acompanhar os resultados e andamentos dos estudos nas praias onde os PMPs estão sendo realizados.

#### 5.4 ANÁLISE COMPARATIVA E DIRECIONAMENTOS DO PROJETO

Com a análise dos principais trabalhos correlatos existentes, foi possível levantar os principais pontos de sucesso e alguns aspectos que precisam de melhorias, os quais influenciam diretamente na capacidade de sistemas similares alcançarem seus objetivos e obterem a aceitação do público-alvo.

A receptividade do Sistema Urubu e do SISS-Geo indicou um interesse significativo da população em participar de projetos de ciência cidadã, evidenciado pelos números expressivos de *downloads* e registros de dados apresentados anteriormente.

Cada um dos três sistemas possui suas peculiaridades e objetivos distintos: enquanto o SISS-Geo está mais voltado ao monitoramento da saúde de animais silvestres e à investigação de agentes causadores de doenças, o Sistema Urubu foca principalmente no monitoramento de animais atropelados em rodovias e ferrovias. Por sua vez, o SIMBA busca fornecer visibilidade ao monitoramento costeiro realizado em regiões de exploração e produção de petróleo.

Sendo assim, a partir da análise prévia, foi possível constatar que existe uma oportunidade de posicionamento para o sistema proposto neste projeto, que se enquadra como uma plataforma com interface amigável e intuitiva, visando otimizar o processo de coleta, classificação e gestão de dados da fauna costeira. O sistema poderá se apoiar em pontos fortes identificados nos sistemas correlatos, como a funcionalidade offline do SISS-Geo, a interface amigável do Sistema Urubu e

a organização dos dados do SIMBA.

O sistema proposto buscará oferecer uma experiência de usuário agradável e descomplicada para o registro de observações da fauna costeira, com opções abrangentes de espécies para classificação e ferramentas de visualização de dados claras e acessíveis. Ao adotar uma abordagem centrada no usuário e priorizar a simplicidade e eficiência, pretende-se ampliar o engajamento da comunidade na coleta de dados científicos e contribuir significativamente para o monitoramento e conservação da biodiversidade costeira.

## 6 TECNOLOGIAS E FRAMEWORKS

Neste capítulo, serão apresentadas as tecnologias escolhidas para o desenvolvimento do aplicativo, destacando os aspectos que influenciaram nas escolhas e como elas contribuíram para o desenvolvimento do projeto.

### 6.1 FLUTTER

O *Flutter* é um conjunto de ferramentas de UI (Interface do Usuário) criado pelo *Google* em 2017. Se diferencia por possuir um suporte multiplataforma de código aberto projetado para permitir a reutilização de código em diversos sistemas operacionais, como *iOS*, *Android*, *web* e *desktop*, enquanto possibilita que as aplicações interajam diretamente com os serviços de cada plataforma. Como publicado por seus desenvolvedores, o *Flutter* oferece uma solução centralizada para o desenvolvimento multiplataforma que facilita a criação de componentes visuais ao mesmo tempo em que realiza a integração com recursos nativos de cada tipo de dispositivo ([Flutter, 2025](#)).

Com esse *framework*, os aplicativos são compilados a partir de um único código-base para as plataformas *mobile*, *web* e *desktop*. O desenvolvimento no *Flutter* é baseado em *Widgets*, componentes escritos em *Dart* que são responsáveis pela visualização das telas e pela interação com o usuário. Esses componentes podem ou não estarem associados a um estado, quando estão, são chamados de *Stateful Widgets* e, quando não, de *Stateless Widgets* ([Flutter, 2025](#)).

Além disso, o *Flutter* possui a funcionalidade de *hot reload*. Uma ferramenta que permite que alterações sejam feitas no código enquanto aplicação está em execução, gerando uma atualização em tempo real da interface do usuário e da lógica da aplicação, mantendo inalterado seu estado de execução.

A escolha pelo *Flutter* se deu principalmente pela sua capacidade multiplataforma a partir de uma única base de código, otimizando tempo e recursos. Outro fator decisivo foi a integração com os diversos serviços do *Firebase*, que facilita a implementação de funcionalidades essenciais como autenticação, banco de dados em tempo real e armazenamento. Além disso, o *Flutter* é uma tecnologia emergente que vem ganhando ampla adoção na indústria de desenvolvimento *mobile*, oferecendo uma comunidade ativa, atualizações constantes e um ecossistema em rápida evolução, o que garante maior longevidade e suporte para o projeto.

### 6.2 DART

*Dart* é uma linguagem de programação lançada pelo *Google* em 2013, criada com foco em produtividade e performance para aplicações multiplataforma. A linguagem é orientada a



objetos e oferece tipagem estática forte, porém também é possível optar pelo uso de tipagem dinâmica, o que pode ser útil em algumas situações. (Dart, 2025).

Um aspecto importante do *Dart* é sua flexibilidade de compilação. É uma linguagem que utiliza tanto a compilação *just-in-time* (JIT) quanto *ahead-of-time* (AOT). Em tempo de desenvolvimento, o *Dart* roda no *Dart VM* usando JIT, o que permite recompilar apenas partes do código em tempo real sem reiniciar toda a aplicação. Essa funcionalidade viabiliza o recurso de *hot reload* do *Flutter*, permitindo que alterações sejam refletidas imediatamente na aplicação sem alterar o estado da interface. Após realizar o lançamento de uma versão, o *Dart* usa AOT para compilar o código diretamente para código nativo, o que garante um desempenho maior com relação ao JIT (Dart, 2025).

### 6.3 FIREBASE

O *Firebase* é uma plataforma de desenvolvimento de aplicativos móveis e *web* fornecida pelo *Google*, que oferece uma ampla gama de serviços *backend* para facilitar a criação de aplicações robustas e escaláveis. Tem como característica permitir o desenvolvimento centralizado sem a necessidade de infraestrutura própria, fornecendo soluções completas para autenticação de usuários, banco de dados em tempo real, armazenamento de arquivos, envio de notificações, análises de uso e monitoramento de desempenho. Além disso, a plataforma integra ferramentas para teste, distribuição e acompanhamento do ciclo de vida do aplicativo com suporte multiplataforma (Google Firebase, 2025).

Neste projeto, o *Firebase* gerencia a autenticação e o armazenamento de dados, utilizando o *Firebase Authentication* para autenticar usuários com diferentes métodos de login e o *Firebase Storage* para armazenar imagens, além do banco de dados NoSQL *Cloud Firestore Database* para salvar os dados gerais do sistema. A escolha do *Firebase* se deu pela facilidade de integração com *Flutter* e pela variedade e robustez dos serviços oferecidos. A aplicação foi construída com a utilização dos benefícios do plano *Spark* que, apesar de ser um plano gratuito, traz diversas ferramentas com limites de uso suficientemente altos para projetos iniciais.

Dentre os principais serviços disponibilizados no plano utilizado que foram utilizados no projeto, estão o *Firebase Authentication*, que possibilita a autenticação de usuários via e-mail/senha e *Google*, com suporte para até 50 mil usuários ativos mensais sem custo. O *Firebase Analytics* que fornece relatórios detalhados sobre o comportamento e uso do aplicativo, permitindo a análise de métricas essenciais. O *App Distribution* que possibilita a distribuição de versões de teste do aplicativo para usuários selecionados durante as fases de desenvolvimento e teste. Além do *Cloud Firestore Database* que oferece até 1 GB de armazenamento, com limites diários de 50 mil leituras, 20 mil gravações e 20 mil exclusões de documentos. E do *Cloud Storage* que disponibiliza 5 GB de armazenamento, 1 GB de download diário, além de 20 mil operações de upload e 50 mil operações de download por dia (Google Firebase, 2025).



## 6.4 PACOTES E PLUGINS DO FLUTTER

Como o *Flutter* possui uma arquitetura baseada em pacotes e *plugins*, para o desenvolvimento de algumas funcionalidades específicas da aplicação proposta neste trabalho foram selecionados algumas dependências externas para facilitar e aumentar as possibilidades de desenvolvimento.

As dependências do projeto podem ser classificadas em três categorias. Bibliotecas puramente em Dart, que não interagem com *APIs* nativas da plataforma, como *excel* (para manipulação de planilhas), *intl* (internacionalização e formatação de dados) e *diacritic* (remoção de acentos e caracteres especiais).

Plugins, que integram o código *Dart* com *APIs* nativas dos sistemas operacionais *Android* e *iOS*. Exemplos incluem *geolocator* (geolocalização), *image\_picker* (acesso à câmera e galeria) e *permission\_handler* (gerenciamento de permissões).

E as ferramentas auxiliares, que contribuem com a geração de código e automação de processos durante o desenvolvimento, como *hive\_generator* (geração de adaptadores para o *Hive*) e *build\_runner* (execução de processos automáticos de geração de código).

Essas dependências permitem incorporar funcionalidades reutilizáveis no projeto como renderização de mapas, autenticação com Firebase e diversos formatos de gráficos sem a necessidade de implementações manuais que seriam custosas e extensas. Além disso, possibilitam o acesso a recursos nativos dos dispositivo, como a geolocalização, gerenciamento de permissões, câmera e o armazenamento local no dispositivo.

A Tabela 2 apresenta todas as dependências utilizadas no projeto, especificando a versão de cada uma, bem como uma breve descrição de sua finalidade. Essa relação visa demonstrar a amplitude de recursos externos incorporados ao aplicativo e documentar as versões exatas para fins de controle de compatibilidade.

Tabela 2 – Dependências do Projeto *Flutter*

<b>Dependência</b>	<b>Versão</b>	<b>Descrição breve</b>
flutter	sdk: flutter	Framework principal para desenvolvimento de apps móveis.
flutter_localizations	sdk: flutter	Suporte a localização (i18n) para Flutter.
cupertino_icons	^1.0.2	Ícones estilo iOS para Flutter.
firebase_core	^3.8.1	Inicialização e integração do Firebase.
firebase_auth	^5.3.4	Autenticação com Firebase.
cloud_firestore	^5.4.4	Integração com o banco Firestore.
firebase_analytics	^11.3.3	Monitoramento e análise de uso (analytics).
firebase_app_check	^0.3.2+5	Segurança para apps usando Firebase.
google_sign_in	^6.2.2	Autenticação via Google.
persistent_bottom_nav_bar	^6.2.1	Barra de navegação inferior persistente.
image_picker	^1.1.2	Seleção de imagens da galeria ou câmera.
firebase_storage	^12.3.4	Armazenamento de arquivos no Firebase.
diacritic	^0.1.3	Remoção de acentos (diacríticos) de strings.
provider	^6.1.2	Gerenciamento de estado simplificado.
photo_view	^0.15.0	Visualização de imagens com zoom e rotação.
geolocator	^13.0.2	Geolocalização e rastreamento de posição.
geocoding	^3.0.0	Geocodificação e reversa.
phosphor_flutter	^2.1.0	Ícones customizados da biblioteca Phosphor.
url_launcher	^6.3.1	Abertura de URLs externas (navegadores, apps, etc.).
accordion	^2.6.0	Componente de interface tipo acordeão.
cached_network_image	^3.4.1	Cache para imagens carregadas da web.
hive	^2.2.3	Banco de dados local rápido e leve.
hive_generator	^2.0.1	Gerador de código para Hive.
path_provider	^2.1.1	Acesso a diretórios do sistema de arquivos.
connectivity_plus	^5.0.2	Deteção de status de conectividade (online/offline).
build_runner	^2.4.8	Utilitário para geração de código automático.
skeletonizer	^1.4.2	Animação skeleton loader (placeholders enquanto carrega).
badges	^3.1.2	Criação de badges/insígnias visuais.
pie_chart	^5.4.0	Gráficos de pizza.
graphic	^2.5.0	Biblioteca de gráficos avançados.
share_plus	^10.1.4	Compartilhamento nativo de conteúdo.
exif	^3.0.1	Leitura de dados EXIF de imagens.
latlong2	^0.9.0	Tipos de dados para latitude e longitude.
open_filex	^4.3.4	Abertura de arquivos em apps externos.
excel	^4.0.6	Manipulação de arquivos Excel (.xlsx).
flutter_map	^8.1.1	Renderização de mapas (OpenStreetMap, etc.).
intl	^0.19.0	Internacionalização e formatação de datas/números.
permission_handler	^12.0.0+1	Gerenciamento de permissões do sistema operacional.
device_info_plus	^11.3.3	Informações detalhadas sobre o dispositivo.

Fonte: Autor

### 6.4.1 Hive

O Hive é um banco de dados leve e extremamente rápido, escrito inteiramente em *Dart*, projetado para ser usado em aplicações Flutter e *Dart* nativas. Ele segue o modelo de banco de dados chave-valor, oferecendo uma alternativa eficiente ao SQLite e a outros bancos de dados mais complexos, especialmente em contextos mobile (LEIER; CONTRIBUTORS, 2025).

No projeto o Hive foi escolhido como opção de banco de dados local para armazenamento dos registros em momentos em que o dispositivo está offline, permitindo que até 40 registros sejam armazenados localmente e sincronizados assim que a conexão com a internet retornar. A principal vantagem do Hive é a simplicidade e velocidade em dispositivos móveis.

### 6.4.2 Flutter Map

O *Flutter Map* é um plugin gratuito, multiplataforma e de código aberto que permite a renderização de mapas dentro de aplicativos *Flutter*. Oferece uma interface fácil de usar para adicionar camadas, marcadores personalizados e interações em mapas. Possui suporte de diversos provedores de mapas, como *OpenStreetMap*, *Mapbox* e *Google Maps*, permitindo que os desenvolvedores escolham a fonte de dados que melhor atende às suas necessidades. Além disso, ele oferece suporte a recursos avançados, como geolocalização, desenho de rotas e manipulação de gestos do usuário, adição de *layers* (FLUTTER\_MAP, 2025).

Neste projeto, o *Flutter Map* foi utilizado para renderizar o mapa com marcadores personalizados das localizações dos registros realizados pelos usuários, permitindo uma visão geográfica dos dados coletados.

### 6.4.3 Excel

O *excel* é uma biblioteca de código aberto para o *Dart* que permite a leitura e escrita de arquivos do Microsoft Excel no formato *XLSX* com customizações avançadas de template (SINGH, 2021). Neste projeto foi utilizada para construir a planilha de registros personalizada para download e compartilhamento.

### 6.4.4 Graphic

*Graphic* é uma biblioteca de visualização de dados para *Flutter*, baseada na Gramática de Gráficos de Leland Wilkinson. Com ela é possível criar gráficos personalizados de diversos tipos. Além disso, oferece suporte a interações avançadas como: seleção, legenda, tooltips, zoom, e animações para transições e carregamento dos gráficos (CHEN, 2024).

A aplicação desta biblioteca no projeto foi pensada para gerar uma visualização gráfica dos dados de registros coletados para maior síntese de algumas informações chaves. Para o resultado final, foram gerados gráficos na forma de pizza e barras.

## 7 RESULTADOS OBTIDOS

Este capítulo demonstra os artefatos produzidos para o *software*, incluindo todas as informações associadas desde a análise até os testes, de acordo com o escopo da disciplina Projeto I.

### 7.1 ENGENHARIA DE REQUISITOS

Os requisitos foram organizados em duas categorias: funcionais e não funcionais. A seguir são apresentados os artefatos e resultados desta etapa. TODO REFERENCIAR EXPLIOCAAO DE REQUISITOS



### 7.1.1 Requisitos Funcionais

Tabela 3 – Requisitos funcionais do sistema

Código	Descrição
RF01	O sistema deve permitir o login via Firebase Authentication com e-mail e senha.
RF02	O sistema deve permitir o login via conta Google integrada ao Firebase Authentication.
RF03	O sistema deve unificar dados quando uma conta possui ambos os métodos de login.
RF04	O sistema deve permitir atualização do perfil de usuário (nome, foto, e-mail).
RF05	O sistema deve permitir redefinição de senha através do fluxo do Firebase Authentication (envio de e-mail de redefinição).
RF06	O sistema deve permitir a exclusão da conta do usuário e remoção de todos os seus dados (incluindo imagem de perfil armazenada no Firebase Storage).
RF07	O sistema deve manter os registros do usuário (subcollection) mesmo após a exclusão da conta, para preservação de dados científicos.
RF08	O sistema deve permitir envio de registros simples e técnicos com campos específicos.
RF09	O sistema deve permitir envio de até 2 imagens por registro, sendo uma obrigatória.
RF10	O sistema deve permitir envio de registros utilizando GPS do dispositivo em tempo real para obter as coordenadas geográficas.
RF11	O sistema deve permitir envio de registros utilizando a opção número da guarita para basear as coordenadas geográficas.
RF12	O sistema deve permitir envio de registros utilizando coordenadas da cidade para definir a localização.
RF13	O sistema deve permitir envio de registros utilizando ponto de referência como campo aberto.
RF14	O sistema deve funcionar offline, salvando localmente até 40 registros utilizando o Hive para posterior sincronização.
RF15	O sistema deve detectar automaticamente a reconexão de rede e enviar os registros pendentes.
RF16	O sistema deve validar os formulários de envio de registros com restrições específicas (ex.: limite de caracteres).
RF17	O sistema deve implementar um sistema de conquistas para monitoramento de número de registros de aves, répteis e mamíferos.
RF18	O sistema deve diferenciar dois tipos de usuários: cientista cidadão (user) e pesquisador (admin).
RF19	O sistema deve permitir que um pesquisador cadastre outro pesquisador, com geração automática de senha segura (8 dígitos alfanuméricos e caracteres especiais).
RF20	O sistema deve exibir no perfil do usuário o total de registros, conquistas, últimos 10 registros, opções de editar perfil e logout.
RF21	O sistema deve apresentar menu inferior com opções: início, registros, novo registro, perfil e sobre.
RF22	O sistema deve disponibilizar um painel de registros com estatísticas, mapas e filtros detalhados para acompanhamento.
RF23	O sistema deve permitir avaliação de registros por pesquisadores.

### 7.1.2 Requisitos Não Funcionais

Tabela 4 – Requisitos não funcionais do sistema

Código	Descrição
RNF01	O sistema deve ser desenvolvido utilizando Flutter e Dart, garantindo compatibilidade multiplataforma.
RNF02	O sistema deve utilizar autenticação segura via Firebase, com criptografia adequada para senhas e tokens.
RNF03	O sistema deve apresentar alta disponibilidade e ser resiliente a falhas de rede (offline-first).
RNF04	O sistema deve ser capaz de armazenar localmente até 40 registros offline utilizando Hive e sincronizar automaticamente após reconexão.
RNF05	O sistema deve seguir as melhores práticas de UX, com validação clara de formulários e feedbacks visuais para os usuários.
RNF06	O sistema deve garantir a privacidade dos dados do usuário, omitindo dados sensíveis em exportações para cientistas cidadãos.
RNF07	O sistema deve apresentar desempenho aceitável mesmo em dispositivos móveis de média capacidade.
RNF08	O sistema deve fornecer acessibilidade básica, permitindo navegação simples e intuitiva.

## 7.2 PROJETO COMPORTAMENTAL E ESTRUTURAL

O sistema implementa o fluxo de autenticação e as estruturas de dados conforme o planejamento estabelecido.

### 7.2.1 Fluxo de Autenticação

O sistema utiliza o Firebase Authentication como provedor principal para autenticação e gerenciamento de usuários. (descreve como já feito anteriormente)

### 7.2.2 Estrutura de Dados

Tabela 5 – Estrutura de collections do banco de dados

Collection	Descrição
counters	Armazena contadores gerais (ex.: número total de animais cadastrados e registros realizados).
animals	Armazena todas as espécies já registradas no sistema, com dados taxonômicos completos.
users	Armazena informações dos usuários. Cada usuário contém uma subcollection registers com seus registros individuais.

### 7.3 PROJETO ARQUITETURAL

(O conteúdo descritivo será detalhado com diagramas e justificativas da arquitetura, conforme discutido.)

### 7.4 PROJETO E IMPLEMENTAÇÃO DA INTERFACE COM O USUÁRIO

(Apresenta telas, wireframes e decisões visuais adotadas.)

### 7.5 PROJETO GERENCIAL DO PROTÓTIPO

(Detalhamento sobre cronogramas, milestones e gestão de atividades do Projeto I.)

### 7.6 IMPLEMENTAÇÃO DO PROTÓTIPO

(Apresentação da implementação já realizada, destacando módulos ou telas entregues.)

### 7.7 PROJETO E EXECUÇÃO DE TESTES E VERIFICAÇÃO DE QUALIDADE DO PROTÓTIPO

(Descrever testes aplicados, resultados parciais e estratégias futuras.)

## CONSIDERAÇÕES FINAIS

O capítulo consolidou os resultados até aqui obtidos, alinhando-os ao planejamento inicial do projeto, garantindo rastreabilidade entre requisitos, projeto e implementação.



## 8 CONCLUSÃO

Sed consequat tellus et tortor. Ut tempor laoreet quam. Nullam id wisi a libero tristique semper. Nullam nisl massa, rutrum ut, egestas semper, mollis id, leo. Nulla ac massa eu risus blandit mattis. Mauris ut nunc. In hac habitasse platea dictumst. Aliquam eget tortor. Quisque dapibus pede in erat. Nunc enim. In dui nulla, commodo at, consectetur nec, malesuada nec, elit. Aliquam ornare tellus eu urna. Sed nec metus. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas.

Phasellus id magna. Duis malesuada interdum arcu. Integer metus. Morbi pulvinar pellentesque mi. Suspendisse sed est eu magna molestie egestas. Quisque mi lorem, pulvinar eget, egestas quis, luctus at, ante. Proin auctor vehicula purus. Fusce ac nisl aliquam ante hendrerit pellentesque. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Morbi wisi. Etiam arcu mauris, facilisis sed, eleifend non, nonummy ut, pede. Cras ut lacus tempor metus mollis placerat. Vivamus eu tortor vel metus interdum malesuada.

Sed eleifend, eros sit amet faucibus elementum, urna sapien consectetur mauris, quis egestas leo justo non risus. Morbi non felis ac libero vulputate fringilla. Mauris libero eros, lacinia non, sodales quis, dapibus porttitor, pede. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Morbi dapibus mauris condimentum nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Etiam sit amet erat. Nulla varius. Etiam tincidunt dui vitae turpis. Donec leo. Morbi vulputate convallis est. Integer aliquet. Pellentesque aliquet sodales urna.

## REFERÊNCIAS

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. *NBR 6028*: Resumo - apresentação. Rio de Janeiro, 2003. 2 p. Citado na página 4.

CASTRO Érika P.; BAGER, A. Sistema urubu: a ciência cidadã em prol da conservação da biodiversidade. *Revista Brasileira de Tecnologias Sociais*, Editora UNIVALI, v. 6, n. 2, p. 111–130, 2019. Acesso em: 08/04/2024. Disponível em: <<https://periodicos.univali.br/index.php/rbts/article/view/15264>>. Citado na página 25.

CHAME, M. et al. Sistema de informação em saúde silvestre - “siss-geo”. *Grandes Desafios da Computação no Brasil*, Sociedade Brasileira de Computação, v. 3, 2015. Citado na página 23.

CHEN, L. *graphic: A Flutter data visualization library*. 2024. <<https://pub.dev/packages/graphic>>. MIT License. Accessed: 10 May 2025. Citado na página 34.

Dart. *Dart Documentation*. 2025. Accessed: 2025-05-08. Disponível em: <<https://dart.dev/docs>>. Citado na página 31.

Flutter. *Flutter Documentation*. 2025. Accessed: 2025-05-08. Disponível em: <<https://docs.flutter.dev/>>. Citado na página 30.

FLUTTER\_MAP. *flutter\_map: Interactive maps for Flutter based on Leaflet*. 2025. <[https://pub.dev/packages/flutter\\_map](https://pub.dev/packages/flutter_map)>. BSD 3-Clause License. Accessed: 10 May 2025. Citado na página 34.

GIL, A. C. et al. *Como elaborar projetos de pesquisa*. [S.l.: s.n.], v. 4. Citado na página 13.

Google. *Android*. 2024. Acesso em: 15 abr. 2024. Disponível em: <<https://www.android.com>>. Citado na página 23.

Google. *Google Drive*. 2024. <<https://www.google.com/drive/>>. Acesso em: 07/05/2025. Citado na página 11.

Google. *Google Play Store*. 2024. Acesso em: 10 abr. 2024. Disponível em: <<https://play.google.com/store>>. Citado na página 23.

Google Firebase. *Firebase Documentation*. 2025. Acesso em: 10 maio 2025. Disponível em: <<https://firebase.google.com/docs?authuser=0&hl=pt-br>>. Citado na página 31.

Governo Federal do Brasil. *Gov.br*. 2024. Acesso em: 18 abr. 2024. Disponível em: <<https://www.gov.br>>. Citado na página 23.

LEIER, S.; CONTRIBUTORS. *Hive - Lightweight and blazing fast key-value database written in pure Dart*. 2025. <<https://pub.dev/documentation/hive/latest/>>. Acessado em: maio 2025. Citado na página 34.

MARTINS, D. G. d. M.; CABRAL, E. H. d. S. Panorama dos principais estudos sobre ciência cidadã. *ForScience*, v. 9, n. 2, p. e01030, out. 2021. Disponível em: <<https://forscience.ifmg.edu.br/index.php/forscience/article/view/1030>>. Citado na página 11.

NAÇÕES UNIDAS. *Agenda 2030 para o Desenvolvimento Sustentável*. 2015. <<https://brasil.un.org/pt-br/91863-agenda-2030-para-o-desenvolvimento-sustentavel>>. Acesso em: 10/05/2024. Citado 2 vezes nas páginas 11 e 19.

NAÇÕES UNIDAS. *Sobre o nosso trabalho para alcançar os Objetivos de Desenvolvimento Sustentável no Brasil*. 2015. <<https://brasil.un.org/pt-br/sdgs>>. Acesso em: 10/05/2024. Citado 3 vezes nas páginas 18, 19 e 20.

OHNO, T. *Toyota Production System: Beyond Large-Scale Production*. [S.l.]: Productivity Press, 1988. 176 p. Citado na página 21.

PALMA, D. A. *Monitoramento de qualidade da água com o enfoque ciência cidadã: estudo de cem Brasíliaândia*. 76 p. Monografia — Universidade de Brasília, Brasília, 2016. Citado na página 18.

Petrobras. *SIMBA - Sistema de Monitoramento da Biodiversidade Aquática*. 2024. Acesso em: 12/04/2024. Disponível em: <<https://simba.petrobras.com.br/simba/web/>>. Citado 3 vezes nas páginas 26, 27 e 28.

PRESSMAN, R. S. *Engenharia de Software – uma Abordagem Profissional*. 7. ed. [S.l.]: AMGH Editora LTDA, 2011. Citado 3 vezes nas páginas 14, 20 e 21.

SIBBR. *SIBBr: Sistema de Informação sobre a Biodiversidade Brasileira*. 2024. <<https://sibbr.gov.br/>>. Acesso em: 08/04/2024. Citado na página 23.

SINGH, K. *excel: A Flutter and Dart library for reading and writing Excel files*. 2021. <<https://pub.dev/packages/excel>>. MIT License. Accessed: 10 May 2025. Citado na página 34.

SISS-Geo. *Plataforma Institucional Biodiversidade e Saúde Silvestre. Sistema de Informação em Saúde Silvestre*. 2024. <<https://sisgeo.incc.br/apresentacao.xhtml>>. Acesso em: 12/04/2024. Citado na página 24.

WhatsApp. *WhatsApp*. 2024. <<https://www.whatsapp.com/>>. Acesso em: 07/05/2025. Citado na página 11.

WILDSCHUT, D. The need for citizen science in the transition to a sustainable peer-to-peer-society. *Futures*, v. 91, p. 46–52, 2017. Citado na página 18.

ZAYAT, W.; SENVAR, O. Framework study for agile software development via scrum and kanban. *International Journal Of Innovation And Technology Management*, World Scientific Pub Co Pte Ltd, v. 17, n. 04, p. 203002–1–203002–24, 2020. Acesso em: 16/06/2024. Disponível em: <<https://www.worldscientific.com/doi/10.1142/S0219877020300025>>. Citado na página 21.

## APÊNDICE A – QUISQUE LIBERO JUSTO

Quisque facilisis auctor sapien. Pellentesque gravida hendrerit lectus. Mauris rutrum sodales sapien. Fusce hendrerit sem vel lorem. Integer pellentesque massa vel augue. Integer elit tortor, feugiat quis, sagittis et, ornare non, lacus. Vestibulum posuere pellentesque eros. Quisque venenatis ipsum dictum nulla. Aliquam quis quam non metus eleifend interdum. Nam eget sapien ac mauris malesuada adipiscing. Etiam eleifend neque sed quam. Nulla facilisi. Proin a ligula. Sed id dui eu nibh egestas tincidunt. Suspendisse arcu.

**APÊNDICE B – NULLAM ELEMENTUM URNA VEL IMPERDIET SODALES  
ELIT IPSUM PHARETRA LIGULA AC PRETIUM ANTE JUSTO A NULLA  
CURABITUR TRISTIQUE ARCU EU METUS**

Nunc velit. Nullam elit sapien, eleifend eu, commodo nec, semper sit amet, elit. Nulla lectus risus, condimentum ut, laoreet eget, viverra nec, odio. Proin lobortis. Curabitur dictum arcu vel wisi. Cras id nulla venenatis tortor congue ultrices. Pellentesque eget pede. Sed eleifend sagittis elit. Nam sed tellus sit amet lectus ullamcorper tristique. Mauris enim sem, tristique eu, accumsan at, scelerisque vulputate, neque. Quisque lacus. Donec et ipsum sit amet elit nonummy aliquet. Sed viverra nisl at sem. Nam diam. Mauris ut dolor. Curabitur ornare tortor cursus velit.

Morbi tincidunt posuere arcu. Cras venenatis est vitae dolor. Vivamus scelerisque semper mi. Donec ipsum arcu, consequat scelerisque, viverra id, dictum at, metus. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut pede sem, tempus ut, porttitor bibendum, molestie eu, elit. Suspendisse potenti. Sed id lectus sit amet purus faucibus vehicula. Praesent sed sem non dui pharetra interdum. Nam viverra ultrices magna.

Aenean laoreet aliquam orci. Nunc interdum elementum urna. Quisque erat. Nullam tempor neque. Maecenas velit nibh, scelerisque a, consequat ut, viverra in, enim. Duis magna. Donec odio neque, tristique et, tincidunt eu, rhoncus ac, nunc. Mauris malesuada malesuada elit. Etiam lacus mauris, pretium vel, blandit in, ultricies id, libero. Phasellus bibendum erat ut diam. In congue imperdiet lectus.

## **ANEXO A – MORBI ULTRICES RUTRUM LOREM.**

Sed mattis, erat sit amet gravida malesuada, elit augue egestas diam, tempus scelerisque nunc nisl vitae libero. Sed consequat feugiat massa. Nunc porta, eros in eleifend varius, erat leo rutrum dui, non convallis lectus orci ut nibh. Sed lorem massa, nonummy quis, egestas id, condimentum at, nisl. Maecenas at nibh. Aliquam et augue at nunc pellentesque ullamcorper. Duis nisl nibh, laoreet suscipit, convallis ut, rutrum id, enim. Phasellus odio. Nulla nulla elit, molestie non, scelerisque at, vestibulum eu, nulla. Ut odio nisl, facilisis id, mollis et, scelerisque nec, enim. Aenean sem leo, pellentesque sit amet, scelerisque sit amet, vehicula pellentesque, sapien.

**ANEXO B – CRAS NON URNA SED FEUGIAT CUM SOCIIS NATOQUE  
PENATIBUS ET MAGNIS DIS PARTURIENT MONTES NASCETUR RIDICULUS  
MUS**

Sed consequat tellus et tortor. Ut tempor laoreet quam. Nullam id wisi a libero tristique semper. Nullam nisl massa, rutrum ut, egestas semper, mollis id, leo. Nulla ac massa eu risus blandit mattis. Mauris ut nunc. In hac habitasse platea dictumst. Aliquam eget tortor. Quisque dapibus pede in erat. Nunc enim. In dui nulla, commodo at, consectetur nec, malesuada nec, elit. Aliquam ornare tellus eu urna. Sed nec metus. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas.

## ANEXO C – FUSCE FACILISIS LACINIA DUI

Phasellus id magna. Duis malesuada interdum arcu. Integer metus. Morbi pulvinar pellentesque mi. Suspendisse sed est eu magna molestie egestas. Quisque mi lorem, pulvinar eget, egestas quis, luctus at, ante. Proin auctor vehicula purus. Fusce ac nisl aliquam ante hendrerit pellentesque. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Morbi wisi. Etiam arcu mauris, facilisis sed, eleifend non, nonummy ut, pede. Cras ut lacus tempor metus mollis placerat. Vivamus eu tortor vel metus interdum malesuada.