

Mestrado Profissional
Uso Sustentável de Recursos Naturais em Regiões Tropicais

KARIEN BATISTA DA SILVA BARBOSA

**PROPOSTA DE UM SISTEMA, EM APOIO AO TRABALHO DE
MONITORAMENTO DE ABELHAS COM ETIQUETAS ELETRÔNICAS,
UTILIZANDO OS CONCEITOS DE QA/QC**

Belém / PA

2019

KARIEN BATISTA DA SILVA BARBOSA

**PROPOSTA DE UM SISTEMA, EM APOIO AO TRABALHO DE
MONITORAMENTO DE ABELHAS COM ETIQUETAS ELETRÔNICAS,
UTILIZANDO OS CONCEITOS DE QA/QC**

Dissertação apresentada como requisito parcial para obtenção do título de Mestre em Uso Sustentável de Recursos Naturais em Regiões Tropicais, área de Ciências Ambientais, do Instituto Tecnológico Vale Desenvolvimento Sustentável (ITV DS).

Orientador: Paulo Antônio de Souza Júnior, Dr.

Belém / PA

2019

Dados Internacionais de Catalogação na Publicação (CIP)

B328 Barbosa, Karien Batista da Silva

Proposta de um sistema, em apoio ao trabalho de monitoramento de abelhas com etiquetas eletrônicas, utilizando os conceitos de QA/QC. / Karien Batista da Silva Barbosa – Belém, PA, 2019.

39 f.: il.

Dissertação (mestrado) – Instituto Tecnológico Vale, 2019.

Orientador(a): Paulo Antônio de Souza Júnior, Dr.

1. Sistema - Monitoramento. 2. Sistema - Etiqueta eletrônica. 3. Sistema - Abelhas. 4. Sistema - Dados. I. Título.

CDD 23. ed. 621.392

Bibliotecário (a) responsável: Nisa Gonçalves. CRB 2: 525

KARIEN BATISTA DA SILVA BARBOSA

**PROPOSTA DE UM SISTEMA, EM APOIO AO TRABALHO DE
MONITORAMENTO DE ABELHAS COM ETIQUETAS ELETRÔNICAS,
UTILIZANDO OS CONCEITOS DE QA/QC**

Dissertação apresentada como requisito parcial para obtenção do título de Mestre em Uso Sustentável de Recursos Naturais em Regiões Tropicais, área de Ciências Ambientais, do Instituto Tecnológico Vale Desenvolvimento Sustentável (ITV DS).

Data da aprovação:

Banca examinadora:

Dr. Paulo Antônio de Souza Júnior
Orientador – Instituto Tecnológico da Vale (ITV)

Dr. Sergio Ivan Viademonte da Rosa
Membro interno - Instituto Tecnológico da Vale (ITV)

Dr. Vijayendra Kumar Garg
Membro externo – Universidade de Brasília (UnB)

Dedico este trabalho primeiramente a Deus, pois sem Ele nada seria e aos meus pais por todo incentivo, força e dedicação neste momento da minha vida.

AGRADECIMENTOS

Primeiramente a Deus, por ter confiado em mim, por ter atendido as minhas orações e ter me dado forças e coragem para lutar e alcançar esse grande objetivo da minha vida.

Aos meus pais, Paulo e Vera, por sempre acreditarem em minha capacidade e me motivando a acreditar que posso mais do que imagino. Meu infinito agradecimento.

Ao Instituto Tecnológico da Vale (ITV), que através do Programa de Mestrado Profissional em Uso Sustentável de Recursos Naturais em Regiões Tropicais, me proporcionou um aperfeiçoamento gratuito através da Bolsa de Ensino ofertada e a oportunidade de realizar e concretizar mais uma etapa da minha vida profissional.

Ao Prof. Paulo de Souza, meu orientador, pela disposição, orientação excelente e por acreditar no meu potencial de uma forma que eu não acreditava ser capaz de corresponder. Sempre disponível e disposto a ajudar, querendo que eu aproveitasse cada segundo do mestrado para absorver o máximo de conhecimento. Meu infinito agradecimento.

Ao meu coorientador Helder Arruda, pelo apoio e por estar disponível respondendo minhas dúvidas sempre que necessitava.

Aos meus amigos do mestrado, por dividirem comigo as angústias, as alegrias, sempre incentivando uns aos outros e acreditando que tudo daria certo. Foi bom poder contar com vocês.

Aos professores e funcionários do ITV que com ensinamentos, orientações e amizades me ajudaram ativa ou passivamente neste projeto. Vocês todos são referenciais.

RESUMO

Atualmente as organizações preocupam-se não somente com a entrega dos resultados, mas com a confiabilidade deles. A Gestão e o Controle da qualidade devem ser considerados para análise e identificação da fonte dos problemas e possíveis causas, assim como os meios para sua solução. A presente dissertação apresenta uma proposta de automatizar os dados coletados do Projeto *Swarm Sensing*, e implementar um sistema, em Python, com base na utilização dos conceitos de QA/QC, tendo como enfoque a aplicação prática das atividades garantindo o controle e a integridade das informações em tempo real para sua validação. Para apresentação da base de dados de apoio foram levantadas informações quantitativas da *Commonwealth Scientific and Industrial Research Organisation* (CSIRO) e da Empresa Brasileira de Pesquisa Agropecuária (EMBRAPA). Os resultados mostraram que a proposta do sistema servirá para priorização das ideias levantadas e a viabilidade do projeto.

Palavras-chave: Qualidade. Garantia da qualidade. Confiabilidade. Dados.

ABSTRACT

Organizations today are concerned not only with the delivery of the results, but with the reliability of the results. Management and Quality Control should be considered for analysis and identification of the source of the problems and possible causes, as well as the means for their solution. The present dissertation presents a proposal to automate the data collected from the Swarm Sensing Project, and to implement a system based on the use of a QA / QC, focusing on the application of activities, ensuring control and integrity of information in real time for validation purposes. For the presentation of the support database, quantitative information was collected from the Commonwealth Scientific and Industrial Research Organisation (CSIRO) and the Brazilian Agricultural Research Corporation (EMBRAPA). The results served to prioritize the ideas raised and the viability of the project.

Keywords: Quality. Quality assurance. Reliability. Data.

LISTA DE ILUSTRAÇÕES

Figura 1 -	Arquitetura RFID	21
Figura 2 -	Modelo Diagrama de Entidade e Relacionamento	22
Figura 3 -	Alimentador com uma solução de água com açúcar e mel	24
Figura 4 -	O experimento das abelhas realizado em Geeveston Tasmânia	25
Figura 5 -	Diagrama de Entidade e Relacionamento	28
Tabela 1 -	Usuário	29
Tabela 2 -	Login	30
Tabela 3 -	Monitoramento	30
Tabela 4 -	Status	31
Tabela 5 -	Abelhas	31
Figura 6 -	Tela de login do sistema	32
Figura 7 -	Upload dos arquivos dos sensores das abelhas	33
Figura 8 -	Relatório dados sensores das abelhas	33

LISTA DE SIGLAS E ABREVIATURAS

CSIRO – Commonwealth Scientific and Industrial Research Organisation

CSV – Comma Separated Values

DER – Diagrama Entidade Relacionamento

DS – Desenvolvimento Sustentável

ITV – Instituto Tecnológico Vale

QA – Quality Assurance

QC – Quality Control

RFID – Radio Frequency IDentification

SUMÁRIO

1	INTRODUÇÃO	12
1.1	CONTRIBUIÇÕES DESTE TRABALHO	15
1.2	PROBLEMA E HIPÓTESE	15
2	REFERENCIAL TEÓRICO E FUNDAMENTAÇÃO CIENTÍFICA	17
2.1	GARANTIA DA QUALIDADE (QA)	17
2.1.1	Medição da qualidade	17
2.2	CONTROLE DA QUALIDADE (QC)	18
2.3	TECNOLOGIA RFID	20
2.3.1	Arquitetura RFID	21
2.4	DIAGRAMAS ENTIDADE-RELACIONAMENTO	22
3	MATERIAL E MÉTODOS	23
3.1	CARACTERIZAÇÃO DA ÁREA DE ESTUDO	23
3.2	DADOS	23
3.2.1	Componentes Embrapa	23
3.2.2	Componentes Tasmânia	24
4	RESULTADOS	25
4.1	DER: DIAGRAMA DE ENTIDADE E RELACIONAMENTO	27
4.2	DD: DICIONÁRIO DE DADOS DO BANCO DE DADOS	29
5	DISCUSSÃO	32
6	CONCLUSÃO	37
	REFERÊNCIAS	38
	APÊNDICES	40

1 INTRODUÇÃO

As espécies de abelhas têm sido estudadas em diversas pesquisas, relacionadas principalmente pelo seu processo de polinização exercendo, portanto contribuição na definição do padrão de distribuição espacial das espécies, assim como na sua riqueza e abundância, além da promoção do fluxo de genes, estruturação trófica e na dinâmica fenológica (JANZEN, 1970; SMITH, 1973; BAWA, 1985 apud CASTRO, 2008).

As abelhas estão entre os principais polinizadores de culturas agrícolas economicamente importantes em todo o mundo (WINFREE, 2010 apud ARENA, 2018, p. 1). No entanto, o número de colônias de abelhas europeias (*Apis mellifera*) na Europa e na América do Norte, especialmente, tem diminuído no último século (SOUZA, 2018, p. 18). Esse declínio tem um impacto negativo na cadeia alimentar nacional e internacional, sendo os valores econômicos uma pequena parcela comparada a função polinizadora para a diversidade biológica, visto que as abelhas são usadas como um organismo modelo para uma série de estudos de neurobiologia, incluindo cognição, percepção, visão, genética e comportamento. (SRINIVASAN, 2010, p. 267-284; ZAYED, 2012, p. 591-615; SEELEY, 1995 apud SOUZA, 2018, p. 1).

A afirmação feita por Albert Einstein, no livro “Não deixem morrer as abelhas”, página 25, relaciona o grau de importância delas para a vida do planeta terra.

Se as abelhas desaparecerem da face da Terra, a humanidade terá apenas mais quatro anos de existência. Sem abelhas não há polinização, não há reprodução da flora, sem flora não há animais, sem animais, não haverá raça humana. (FARIA; NATURE, 2016)

Inicialmente, compreender a reprodução, o crescimento, manutenção das colônias e principalmente o comportamento das abelhas em relação ao ambiente (visitas mais e menos frequentes das abelhas) tornaram-se um obstáculo para os estudos. A coleta de dados era realizada de diversos métodos, como marcação de insetos com microdots, padrões de cores e códigos QR são comuns em pesquisa entomológica, porém os procedimentos são demorados e trabalhosos (WHITEHEAD, 2012, p. 171-175; DEGEN, 2015, p. 45-57; MERSCH, 2013, p. 1090-1093 apud SOUZA, 2018).

Pesquisas começaram a serem realizadas em todo o mundo para preservação de colmeias. Em 2011, especialistas em Salamanca (Espanha), realizaram um projeto que envolve colocar microchips em abelhas para investigar as causas por trás dos elevados índices de mortalidade entre elas. Em 2013, na Austrália (realizado pela CSIRO) e em 2014 no Brasil (realizado pelo ITV), o Projeto *Swarm Sensing* teve início tendo como meta a utilização de uma tecnologia de microssores para especificar informações detalhadas do comportamento das abelhas e sobre o ambiente em que se encontram inseridas.

Atualmente, a tecnologia RFID tem sido utilizada para identificação de objetos, auxiliando no rastreamento, transmissão e armazenamento dos dados remotamente através de dispositivos chamados de *tags*. Os dados são coletados quando as abelhas passam nas proximidades dos vários pontos de checagem distribuídos. Dos pontos de coleta, as informações são enviadas a uma central permitindo a visualização das rotas dos insetos.

Segundo Souza (2014),

As abelhas são insetos sociais que retornam ao mesmo ponto e operam segundo uma programação muito previsível. Qualquer mudança no seu comportamento indica uma mudança em seu ambiente. Se pudermos modelar seus movimentos, poderemos reconhecer muito rapidamente quando sua atividade apresentar variação, e poderemos identificar a causa. Isso nos ajudará a entender como maximizar sua produtividade, bem como monitorar qualquer risco de biossegurança.

Porém, a aplicação da tecnologia RFID pode falhar se a etiqueta, integrada as abelhas, armazenar uma quantidade muito grande de informações, mesmo as que tem um raio de ação de poucos metros, podem sofrer interceptação e extravio de suas informações. Dentro dessa realidade, podemos considerar que os principais aspectos que contribuem para esses erros são:

- Leituras do sensor: As leituras de sensor da mesma situação não podem ser repetidas exatamente quando as gravamos em momentos diferentes ou usamos sensores diferentes. As trajetórias de diferentes objetos em movimento são diferentes, mesmo que os objetos se movam pela mesma rota. (SOUZA, 2018)

- Distância do leitor: A leitura também pode ser dificultada em ambientes com muitos obstáculos, principalmente aqueles com objetos metálicos de grande volume;
- Capacidades limitadas do equipamento de detecção e do servidor de banco de dados: Existe uma grande quantidade de pontos (dados) em excesso e, portanto, os recursos de armazenamento são desperdiçados, o que também reduz a eficiência da consulta do usuário. Por outro lado, a falta de pontos críticos torna a consulta dos usuários imprecisa. (SOUZA, 2018)

Com o aumento dos experimentos e com a maior complexidade da leitura dos dados, tornou-se necessário garantir e controlar a qualidade das informações obtidas através das abelhas. Dados de má qualidade podem ser os motivos de falhas nos processos de negócio e custos relacionados às áreas organizacionais, podendo até haver perda parcial ou total destes (INMON, 1997 apud ARAÚJO; BATISTA, 2007).

Dessa forma, as utilizações de métodos de qualidade auxiliam na análise do problema e suas possíveis causas, assim como as ações para sua solução. Um conjunto de dados pode estar correto, mas se a sua compreensão for difícil, então todos esses tópicos diminuem a qualidade da informação (WANG, 1994 apud OLIVEIRA; AMARAL, 1999).

O presente trabalho apresenta uma proposta de automatizar os dados coletados do Projeto *Swarm Sensing*, e implementar um sistema, em Python, utilizando os conceitos de Garantia da Qualidade (Quality Assurance) e Controle da Qualidade (Quality Control) para detecção de erros (inconsistências das entradas e saídas das abelhas), gerando um relatório e emitindo avisos com o objetivo de propor ações para correção de problemas.

Conseqüentemente, permitirá coleta dos dados de forma adequada buscando eliminar incertezas nos processos de decisão e confiabilidade do projeto. Realizou-se uma pesquisa exploratória e qualitativa, para retirada de possíveis problemas e causas relacionados acerca do projeto e assuntos que se encontram pendentes de solução.

1.1 CONTRIBUIÇÕES DESTE TRABALHO

- Desenvolvimento de um Diagrama de ER no MySQL, que possibilite a visualização da proposta do sistema, juntamente com as suas funcionalidades.
- Estudos de Casos de um software de metodologia para identificação e classificação dos tipos de erros com objetivo de contribuir na análise causal das situações de falha, auxiliando na definição das medidas preventivas e suas possíveis soluções.
- Definição do formato de mensagem que será enviada pelo sistema, contendo os dados referentes às coordenadas da localização, data e hora dos chips ao "passar" pelos leitores;

O objetivo geral da pesquisa é uma proposta de automatização da coleta de dados do Projeto *Swarm Sensing*, e implementar um sistema em Python, com base na utilização dos conceitos de QA/QC, com a finalidade de detectar e emitir avisos de erros para um melhor Controle e Monitoramento da Qualidade dos Dados em busca de uma Previsão e Redução de Problemas na coleta de Informações.

Nesse sentido, os objetivos específicos desse projeto são:

- a) Propor um método simples e fácil de aplicar, sendo, portanto, um instrumento prático e não apenas uma discussão teórica sobre a solução para previsão e redução de problemas associados à obtenção dos dados.
- b) Servir de fonte de informação para o usuário sobre as etiquetas eletrônicas, o que pode ser relevante para aqueles que possuem pouco conhecimento sobre a tecnologia.

1.2 PROBLEMA E HIPÓTESE

Para produzir resultados confiáveis, corretos e úteis que subsidiem uma diminuição de custos e, conseqüentemente, um aumento da qualidade, as técnicas utilizadas de manipulação de dados devem ter uma base para garantir que eles sejam precisos e exatos.

A sistemática de rastreamento atual (tecnologia RFID) apresenta alguns problemas e falhas que podem prejudicar a rastreabilidade e o controle das

informações, entre elas: falhas ao coletar informações sobre abelhas passando através de colmeias ou alimentadores; orientação e o posicionamento espacial das abelhas ao entrarem nas colmeias devido à polarização de algumas etiquetas RFID. Como consequência, torna-se difícil interpretar em que comportamento uma abelha individual estava envolvida e quanto tempo essa abelha permaneceu dentro e / ou fora da colmeia. (SOUZA, 2018)

A proposta de implantação de um sistema com base na utilização dos conceitos de QA/QC torna-se uma ferramenta necessária em apoio ao trabalho de monitoramento de abelhas utilizando microchips, trazendo melhorias na manipulação das grandes quantidades de dados gerados por redes de sensores. Além disso, reduz a quantidade de inspeção humana necessária, uma vez que a maioria dos insetos sociais têm uma vida adulta curta, de alguns dias a alguns meses, estudando-os com a ajuda de *tags* requer visitas regulares às colônias. Como consequência, as colmeias são abertas regularmente, alterando o ambiente interno, enfatizando a colônia como um todo e matando alguns dos insetos no processo. Alguns insetos são extremamente sensíveis aos odores emitidos por colas e, portanto, os insetos equipados com *tags* podem ser atacados por outros indivíduos. (SOUZA, 2018).

O foco na prevenção de erros, na possibilidade de previsão e identificação de problemas fez com o conceito de QA/QC fosse amplamente utilizado de maneira corriqueira e segura para se verificar a exatidão dos resultados, com o objetivo de proporcionar um maior controle e acompanhamento dos dados instantaneamente, sendo-os sinalizados e corrigidos mais rapidamente do que pode ser feito manualmente.

Procura-se com este trabalho confirmar a existência de uma relação de causa dos problemas de dados com as suas consequências ou efeitos.

2 REFERENCIAL TEÓRICO E FUNDAMENTAÇÃO CIENTÍFICA

2.1 GARANTIA DA QUALIDADE (QA)

Sensores ambientais podem produzir dados de baixa qualidade ou falhar completamente por muitas razões (GANESAN, 2004 *apud* CAMPBELL, 2013). As leituras dos sensores podem ser danificadas ou destruídas por: falhas no *wifi*, como conexão lenta; falhas no fornecimento de energia; sinistros ambientais, como inundações, incêndios. Os sensores também podem funcionar mal quando não são mantidos adequadamente ou quando são inseridos em ambientes limitados, com muitos obstáculos.

Nesse sentido, surge o conceito de Garantia da Qualidade como um conjunto sistemático e planejado de ações para determinar se as atividades estão de acordo com as políticas e processos planejados. Se o planejamento for seguido, o cliente adquire confiança nos produtos e serviços, os quais provavelmente estarão de acordo com suas necessidades (requisitos). (SOTILLE, 2014)

O processo de Garantia da Qualidade serve para garantir que os produtos e processos de software, no ciclo de vida dos projetos, estejam de acordo com os requisitos especificados e referenciados aos planos estabelecidos (LAZZARI, 2009 *apud* ROCHA, 2001, p. 13). Sommerville (2007) descreve Garantia da Qualidade como sendo o processo de definição de como a qualidade de software pode ser atingida e como a organização de desenvolvimento sabe que o software possui o nível de qualidade desejado.

A melhoria contínua dos processos contribui para reduzir desperdícios, eliminar atividades desnecessárias que implicam em custos e, o mais importante, prevenir alguma falha na qualidade do software. Um conjunto completo de atividades deve ser seguido, principalmente, pela equipe Engenharia de Software.

2.1.1 Medição da Qualidade

A qualidade, enquanto conceito, evoluiu da adequação ao padrão para a adequação às necessidades latentes dos clientes (MARTINS; NETO, 1998 *apud* SHIBA, 1993). As métricas de medição da qualidade atribuídas a este trabalho foram:

- Frequência de defeitos: Taxa de defeitos, causa do defeito. Qual é a incidência de defeitos em uma tarefa e quais são as causas?

- Taxa de erros: com que frequência são permitidos erros do sistema e o que constitui um erro do sistema?

- Confiabilidade: o sistema poderá continuar em funcionamento se ocorrerem defeitos?

2.2 CONTROLE DA QUALIDADE (QC)

Os conceitos de QA/QC são normalmente utilizados juntos, mas tem significados diferentes. Para que a melhoria contínua dos processos dê certo, torna-se essencial o controle destes. Definimos então o controle de qualidade como um conjunto de processos ou etapas para garantir que a rede e os protocolos do sensor sejam desenvolvidos e respeitados de forma a minimizar as imprecisões nos dados produzidos e sendo capaz de identificar possíveis falhas que possam vir a ocorrer ou as que já ocorreram. (CAMPBELL, 2013).

Através do conceito de Controle da Qualidade é possível garantir a qualidade. De acordo com a NBR ISO 9000-2005 “controle da qualidade é parte da gestão da qualidade, focada no atendimento dos requisitos da qualidade”, garantindo que as informações prestadas estão de acordo com as exigências seja do cliente, do mercado e eventualmente, da legislação aplicável a ele.

Seguindo a trilogia de JURAN (2008), os objetivos do Controle da Qualidade aplicados ao projeto *Swarm Sensing* são:

- Planejar, manter e melhorar a qualidade.
- Monitorar resultados, a fim de determinar se eles estão de acordo com os padrões relevantes de qualidade e mantê-los sob controle.
- Identificar e eliminar causas de problemas e resultados insatisfatórios.

Seguindo a trilogia de DEMING (1990), os objetivos do Controle da Qualidade aplicados ao projeto *Swarm Sensing* são:

- Eliminar a necessidade de inspeção em massa, priorizando a internalização da qualidade do produto. A qualidade não se origina da inspeção, mas do melhoramento do processo.

- Melhorar constantemente o sistema de planejamento, produção e serviços.
- Implantar métodos de treinamento no local de trabalho.
- Reformular métodos de supervisão; instituir a liderança com o objetivo de ajudar as pessoas a realizar um trabalho melhor.

2.3 TECNOLOGIA RFID

RFID (*Radio Frequency Identification*) é uma tecnologia utilizada na identificação de objetos ou pessoas através de ondas eletromagnéticas de radiofrequência, sem a necessidade de campo visual direto e até através de objetos feitos dos mais diversos materiais (ex: madeira, plástico, papel) para capturar as informações contidas em um dispositivo eletrônico conhecido como “etiqueta RFID”. (MOTA, 2012 *apud* PREDIGER, 2014).

Segundo Bhatt e Glover (2007 *apud* JUNIOR, 2007), RFID é a abreviação de *Radio Frequency Identification* - Identificação por Radiofrequência. Tecnologia de identificação que utiliza frequência de rádio ou variações de campo magnético para comunicação entre componentes. O objetivo dessa tecnologia é identificar, rastrear e gerenciar documentos e produtos e até mesmo animais ou indivíduos, sem contato nem a necessidade de um campo visual (FLORES, 2014 *apud* RFID-COE, 2007), além de oferecer benefícios para quem tenha necessidade de registrar bens físicos.

O uso de etiquetas RFID no projeto *Swarm Sensing* permite que cada abelha seja monitorada, possibilitando o monitoramento em tempo-real das entradas e saídas, com informações sendo atualizadas automaticamente conforme a movimentação e permitindo realizar remotamente o armazenamento e recuperação de informações.

Algumas *tags* RFID podem ser lidas à grandes distâncias, sem a necessidade de um campo visual, desde que estejam dentro do alcance da antena de um leitor, o que possibilita múltiplas leituras simultaneamente e com as abelhas em movimento. Além disso, as etiquetas de RFID são resistentes e possuem alta durabilidade, podendo ser utilizadas em ambientes hostis, e até na parte interna dos objetos, pois as ondas de rádio conseguem ultrapassar vários tipos de materiais.

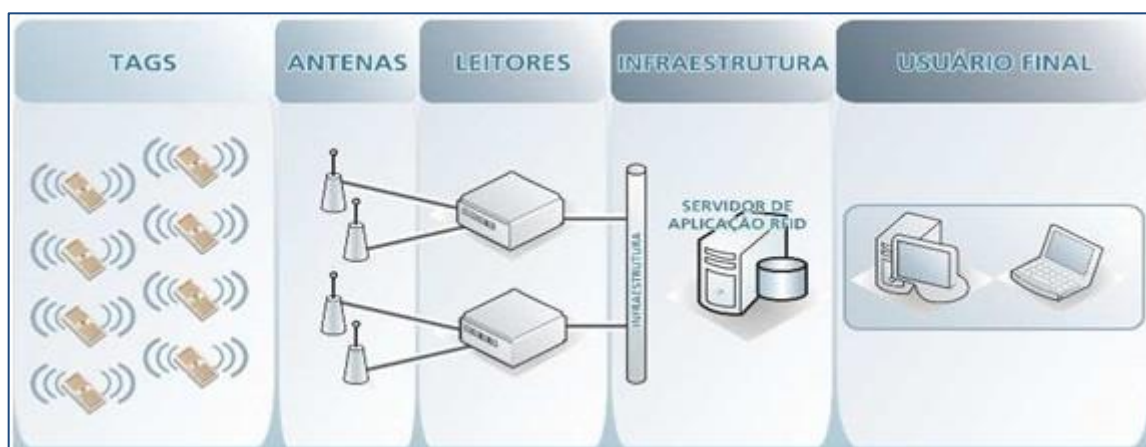
Há a possibilidade de leitura e escrita, o que permite gravar, apagar e editar constantemente as informações, também a sua alta capacidade de memória, o que propicia o armazenamento de todas as informações pertinentes. Por ser regravável e por sua alta durabilidade, a etiqueta RFID pode ser reutilizada diversas vezes dependendo da aplicação (HOLANDA, 2014 *apud* BERNARDO, 2004; FIGUEREIDO, 2004; STANTON, 2004; SOUZA, 2009).

2.3.1 Arquitetura RFID

O sistema RFID possui três módulos:

- Identificadores ou transponders ou TAGs são etiquetas fixadas no objeto (abelha) para rastrear ou controlar;
- Leitor (transceiver) responsável pelo envio da frequência portadora do comando de leitura e também, pela recepção e decodificação do sinal recebido, enviando-o diretamente ao computador ou microprocessador, que utilizará essa informação;
- Antena, presente nos dois módulos anteriores, serve para a eficiente transmissão e recepção dos sinais nos dois sentidos.

Figura 1 – Arquitetura RFID



Fonte: Repositório Digital da FATEC, (20??)

As *tags* são sempre anexadas ao objeto que se quer monitorar ou controlar. Os leitores e antenas podem ser portáteis ou fixos às estruturas que fazem parte do sistema. A *tag* capta o sinal e responde ao leitor também por radiofrequência, enviando todas as informações nela contida. O leitor envia tais informações recebidas para o sistema computacional no qual tem instalado um software específico para reconhecer e identificar essas informações.

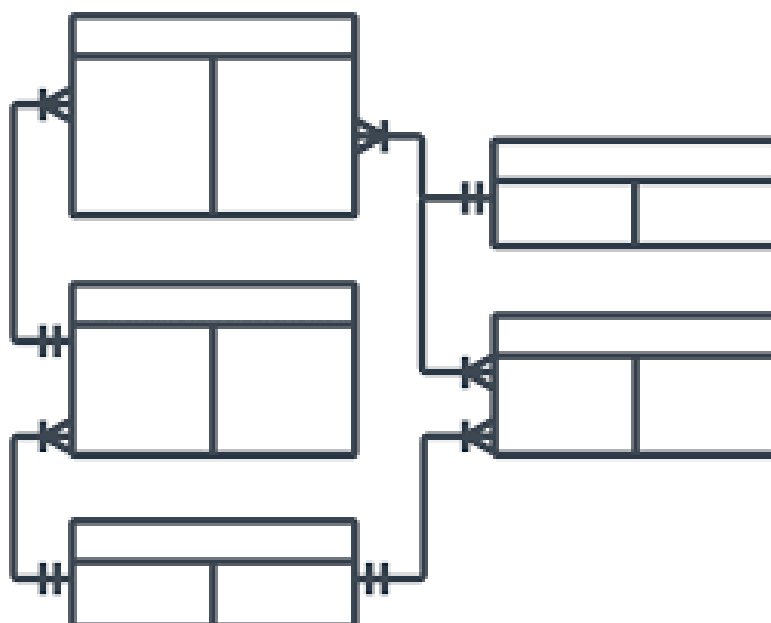
2.4 DIAGRAMA DE ENTIDADES E RELACIONAMENTOS (DER)

Devido a necessidade de armazenar e manipular dados, sendo armazenados de forma organizada e que permitam um posterior acesso eficiente, torna-se necessário projetar um banco de dados.

O Diagrama de Entidades e Relacionamentos (DER) utiliza elementos gráficos para descrever o modelo de dados de um sistema com alto nível de abstração. O DER parte do princípio que o mundo real é formado por um conjunto de objetos chamados entidades e por relacionamentos entre esses objetos (NETO, 2010 *apud* SILBERSCHATZ, 1999).

“O objeto básico que um modelo ER representa é uma entidade” (SILVA; FIGUEIREDO, 2009 *apud* ELMASRI; NAVATHE, 2005). Uma entidade é algo que existe no mundo real (como uma pessoa ou empresa), e que possui atributos, ou seja, características que definem essa entidade. Para criar as relações entre as entidades, usam-se relacionamentos, para associar as entidades. Por fim, é necessário definir as limitações de cada relacionamento. Essas limitações são denominadas cardinalidades, e identificam o número de entidades às quais outra entidade pode ser associada. A figura abaixo, exemplifica o contexto.

Figura 2 - Modelo Diagrama-Entidade-Relacionamento.



Fonte: Repositório Digital da Lucichart, (20??)

O objetivo do DER é facilitar o projeto de Banco de Dados, possibilitando especificar a estrutura lógica geral do Banco de Dados.

3 MATERIAL E MÉTODOS

Neste capítulo serão apresentadas as metodologias usadas no desenvolvimento do projeto e a modelagem do mesmo.

3.1 CARACTERIZAÇÃO DA ÁREA DE ESTUDO

O experimento de campo foi realizado em dois lugares:

- Instalado na Embrapa - Empresa Brasileira de Pesquisa Agropecuária, na região de Belém, Pará. O clima da região de acordo com a classificação climática de Köppen-Geiger, é do tipo equatorial, com verão quente e chuvoso.
- Experimento de campo realizado na Tasmânia, Austrália. De acordo com a Köppen e Geiger o clima é classificado como clima oceânico temperado. 12,4 °C é a temperatura média. Pluviosidade média anual de 592 mm.

3.2 DADOS

3.2.1 Componentes Embrapa

O Sistema de detecção do movimento de abelhas é obtido a partir de três componentes:

1. Etiquetas de RFID que são coladas ao tórax das abelhas;
2. Unidades de leitura das etiquetas que são estrategicamente posicionadas para detectar a presença das abelhas
3. Software de controle do Sistema.

As unidades têm uma série de recursos de comunicação como wi-fi, Bluetooth, a porta micro USB. Há recursos para sincronização de relógio do microcontrolador interno (Edison Intel™) via sinais de GPS, que também servem para localização geográfica do kit e de garantia do funcionamento da banda autorizada pela autoridade de telecomunicações na região onde ele opera (no Brasil a ANATEL, na Austrália a ACMA). A fonte de energia do Sistema é flexível (painéis solares ou fontes retificadas de energia convencionais entre 12V e 60V). O sistema ainda tem sensores internos (eg, temperatura) e dispõe de quatro antenas que podem ser sincronizadas (duty cycle) conforme a necessidade do experimento.

3.2.2 Componentes Tasmânia

Quatro colmeias com leitor de identificação por radiofrequência (RFID) instalado em cada entrada da colmeia. As visitas as colmeias são realizadas regularmente pelos cientistas (por exemplo, uma ou duas vezes por semana) para etiquetar as abelhas com tags RFID. Usando esta configuração, a abelha passa pela entrada da colmeia e é detectada em um leitor particular; o indivíduo detectado, e a data e hora da detecção são registradas. Os dados são organizados em arquivos CSV diários individuais, com base no horário UTC. (SOUZA, 2013).

Figura 3 - Alimentador com uma solução de água com açúcar e mel.



Fonte: Souza, (2017).

Figura 4 – O experimento das abelhas em Geeveston, Tasmânia, Austrália: (a) entrada da colmeia; (b) leitor de RFID instalado embaixo da entrada que detecta a passagem de abelhas marcadas.



Fonte: Souza, (2017).

4 RESULTADOS

O sistema foi desenvolvido na linguagem de programação Python, para emissão de relatórios nos quais constará o status do sistema podendo realizar filtros por espécie de abelha, serviço, status e data das atualizações. Outro relatório disponível será para verificar os alertas de erros no sistema. As informações de erros do sistema serão enviadas como alertas via e-mail, sendo feitas diretamente nos scripts responsáveis pelo monitoramento.

Primeiramente, o usuário fará o requerimento de acesso ao sistema, gerando *Login* e *Senha*. Antes de realizar a validação dos status do sistema, o script de monitoramento irá verificar se os serviços instalados estão disponíveis. Caso esteja, será verificado o status do serviço. Caso não esteja, será enviado um e-mail para o responsável pelo monitoramento, informando que o servidor onde os serviços estão instalados está indisponível.

Ao acessar o sistema, o usuário realizará o upload dos arquivos no formato CSV, descrevendo os dados das abelhas (id, temperatura, origem) e esses arquivos serão salvos em um Banco de Dados PostgreSQL. O sistema vai ler as inconsistências das entradas e saídas dos sensores das abelhas e gerar um relatório de inconsistências e suas possíveis causas.

Se algum serviço configurado estiver parado, o script de monitoramento deverá alertar o responsável via e-mail. Se o erro estiver configurado no sistema, ok. Caso o erro não esteja inserido, também será enviado um alerta informando que não foi possível a solução e conseqüentemente a inserção de um novo erro.

Para a construção da aplicação, foi necessária a utilização das seguintes ferramentas:

- a) MySQL Workbench, para criação das tabelas e consulta de dados;
- b) WampServer 2.0

WAMP é o acrônimo para Windows, Apache, MySQL e PHP. Funciona como um servidor virtual na máquina permitindo testes de todas as funções, tudo armazenado no computador e não exatamente hospedado na internet (LONGEN, 2019).

A escolha do *wamp* deve-se por agilizar o processo de trabalho tanto para desenvolvedores quanto para designers.

- c) Django

Django é um framework open source desenvolvido em Python, ele é voltado para criação de aplicações web, seu padrão MVT (model - template - view) facilita o desenvolvimento do protótipo (ANDRADE, 2019).

- d) Java Script

JavaScript é uma linguagem de programação foi desenvolvido por Brendan Eich, que permite a criação de conteúdos dinâmicos, controle de mídias e animações para deixar as aplicações mais interativo e interessante (ZAMPIERI, 2019).

- e) Python: linguagem de programação desenvolvida no ano de 1989 por Guido van Rossum.

Python é uma linguagem de programação orientada a objetos, de código aberto sendo constantemente usada para o desenvolvimento rápido de aplicações Kay (2005).

A escolha por Python deve-se por causa de sua facilidade de uso, uma linguagem de alto nível, multiplataforma e por ter diversas bibliotecas para realização de qualquer tarefa que seria necessária no desenvolvimento do protótipo.

4.1 DER: DIAGRAMA DE ENTIDADE E RELACIONAMENTO

Mediante a modelagem da base de dados, foi decidida a realização de um modelo de entidade-relacionamento com o propósito de visualizar os relacionamentos entre as entidades do banco de dados e a determinação de quais seriam os formatos destas entidades.

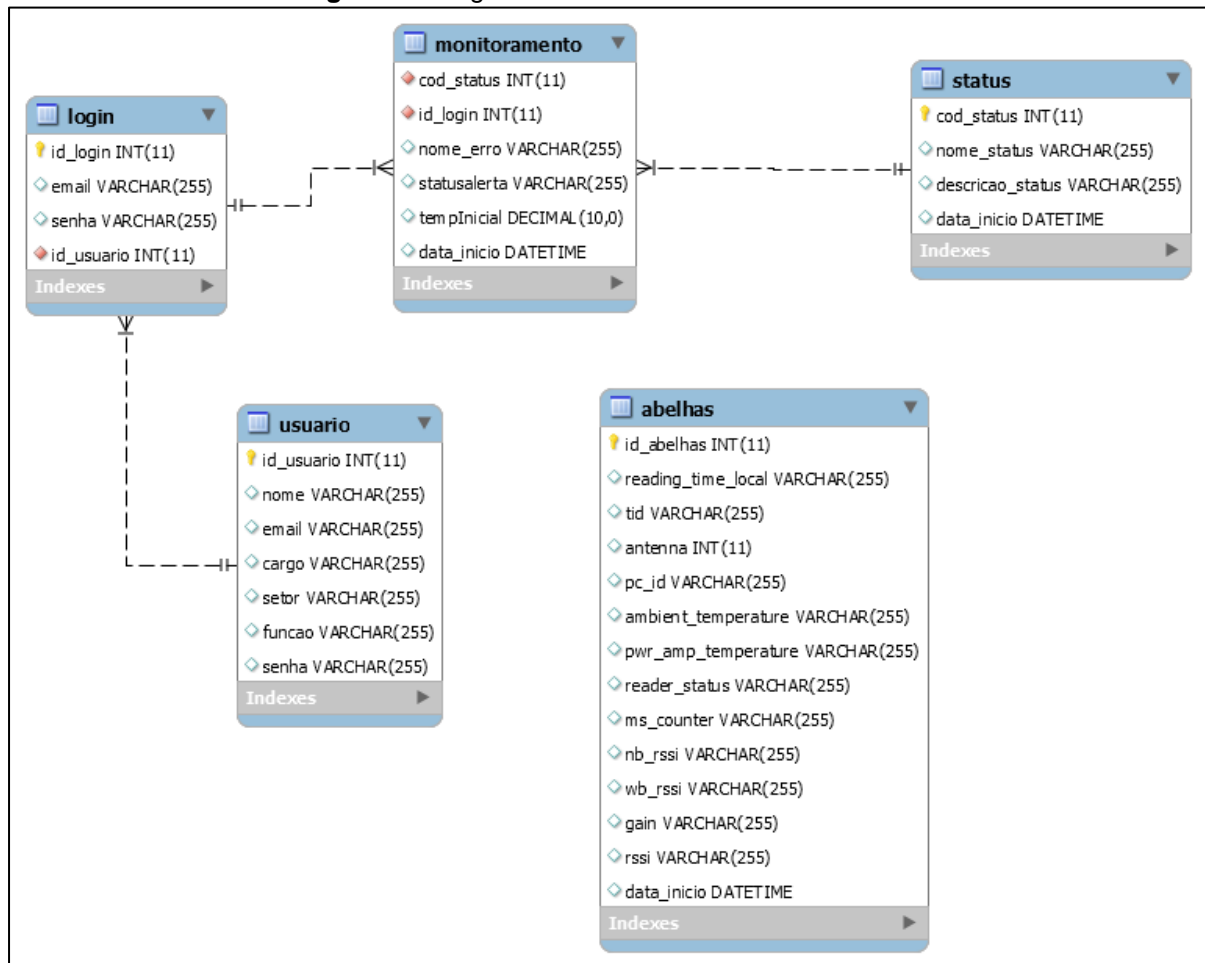
As tabelas Login e Usuário (Figura 5) foram desenvolvidas para permitir o cadastramento e armazenamento das informações do usuário no sistema. Esses registros também são utilizados no controle de permissão de acessos.

A tabela *Monitoramento* (Figura 5) responsável pelo armazenamento dos dados básicos de todos os nós do ambiente que está sendo monitorado.

A tabela *Status* (Figura 5) responsável pelo armazenamento de informações dos status do sistema. Utilizada para manter a base das mensagens a serem enviadas aos contatos, onde ocorre o envio de alertas.

A tabela *Abelhas* (Figura 5) responsável por todas informações referentes as abelhas.

Figura 5 - Diagrama de Entidade-Relacionamento.



Fonte: Elaborado pelo autor, (2018).

4.2 DD: DICIONÁRIO DE DADOS DO BANCO DE DADOS

Tabela 1 - Usuários.

entidade: Usuário - cadastramento do usuário no sistema		
Atributos	Descrição	Tipo
Id usuário	ID único por usuário - chave de identificação do usuário	ID
nome	Nome atribuído ao usuário	Texto
e-mail	E-mail atribuído ao usuário para obter informações e acesso ao sistema	Texto
cargo	Classificação da carreira de um profissional como: estagiário, júnior, pleno, sênior, <i>master</i> e especialista por intermédio de códigos. Nem todos terão acesso aos dados.	Numérico
setor	Classificação do setor de trabalho: Administrativo, Suporte, Ciência, Analistas. Nem todos terão acesso aos dados.	Numérico
função	Classificação da função no trabalho: Suporte - TI, Cientista Ambiental, Analista de Sistemas.	Numérico
senha	Senha de acesso ao sistema com no mínimo 6 caracteres	Texto

Fonte: elaborado pelo autor, (2018).

Tabela 2 - Login.

entidade: Login - responsável pelo armazenamento das credenciais para acessar o sistema. Esses registros são utilizados no controle de acesso. Caso o login e senha não constarem na tabela, o acesso à página será negado.		
Atributos	Descrição	Tipo
id_login	ID único por usuário - chave de identificação do usuário	ID
e-mail	E-mail atribuído ao usuário para obter informações e acesso ao sistema	Texto
senha	Senha de acesso ao sistema com no mínimo 6 caracteres	Texto

Fonte: elaborado pelo autor, (2018).

Tabela 3 - Monitoramento.

entidade: Monitoramento - responsável pelo armazenamento dos dados básicos de todos os nós do ambiente que está sendo monitorado(abelhas)		
Atributos	Descrição	Tipo
cod_status (chave secundária)	Código do status do sistema (ligado ou desligado).	ID
id_login	ID único por usuário - chave de identificação do usuário. Quem fez aquele serviço?	Texto
statusalerta	Se o envio de alertas esta ativo ou inativo	Texto
nome_erro	registro das informações dos erros e avisos	Texto
tempInicial	Timestamp de quando os dados foram cadastrados	Date Time
data_inicio	Data de inclusão do último status dos dados cadastrados.	Date Time

Fonte: elaborado pelo autor, (2018).

Tabela 4 - Status.

entidade: status - armazena informações dos status dos sistema. utilizada para manter a base das mensagens a serem enviadas aos contatos, ondem ocorrem envios de alerta.		
Atributos	Descrição	Tipo
cod_status (chave primária)	Código do status do serviço	Texto
nome_status	como está o status do sistema: normal, parado, com avisos.	Texto
descricao_status	Descrição do status: erro.	Texto
data_inicio	data de inclusão da verificação	Date Time

Fonte: elaborado pelo autor, (2018).

Tabela 5 - Abelhas

entidade: Abelhas - dados referentes as abelhas		
Atributos	Descrição	Tipo
id_abelhas	ID informações da abelha, número de identificação das abelhas.	Texto
reading_time_local	data, hora e local da movimentação das abelhas.	Numérico
antenna	qual antena a abelha moveu-se	Numérico
pc_id	qual pc a movimentação foi registrada	Texto
ambient_temperature	temperatura ambiente	Numérico

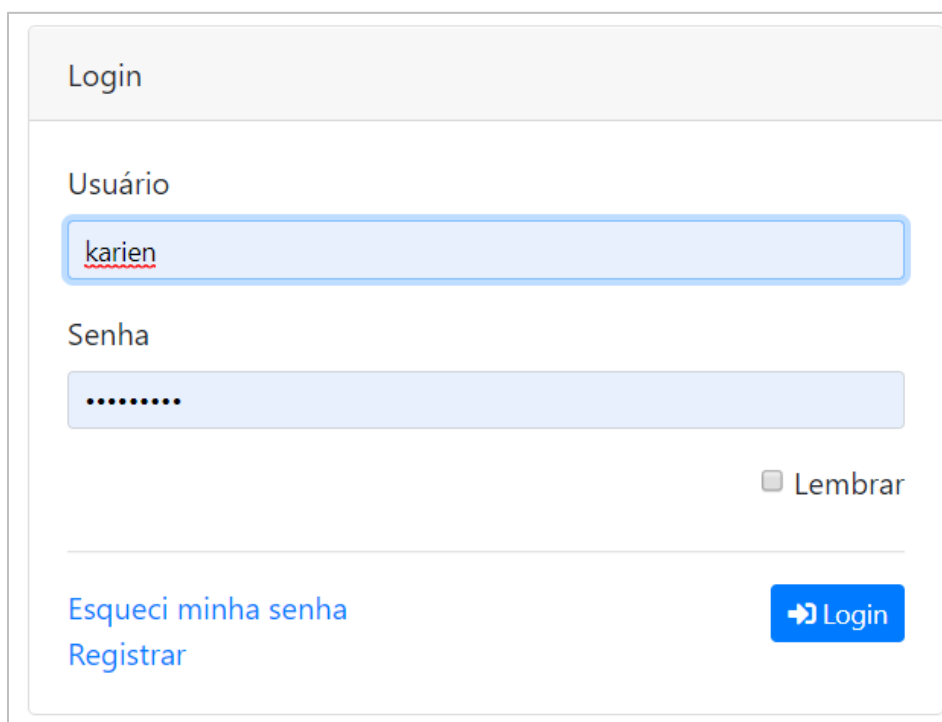
Fonte: elaborado pelo autor, (2018).

5 DISCUSSÃO

Um dos objetivos do trabalho foi a proposta de um sistema para identificar as possíveis falhas do equipamento de leitura de sensores dos chips das abelhas e emissão de avisos.

Primeiramente, o usuário efetuará seu login para acessar o sistema.

Figura 6 – Tela de login do sistema.



A imagem mostra a interface de login de um sistema. No topo, há um cabeçalho com o texto "Login". Abaixo dele, há dois campos de entrada: "Usuário" com o texto "karien" e "Senha" com pontos para ocultar o conteúdo. À direita do campo de senha, há uma caixa de seleção desativada com o rótulo "Lembrar". Na base da interface, há dois links azuis: "Esqueci minha senha" e "Registrar". Um botão azul com o texto "Login" e um ícone de seta para a direita está posicionado à direita dos links.

Fonte: elaborado pelo autor, (2018).

Ao acessar o sistema, o usuário realizará o upload dos arquivos no formato CSV, por meio do Botão 'Browse', que mostrará uma tela do sistema operacional onde será possível escolher os arquivos desejados. Em seguida, ao utilizar o botão 'Processar', o sistema realizará a leitura dos dados dos sensores das abelhas e esses arquivos serão salvos em um Banco de Dados PostgreSQL. Na interface desenvolvida, é possível visualizar o upload dos arquivos, a emissão de relatórios de histórico de leitura diário, alteração e visualização das informações.

Figura 7 - Upload dos arquivos dos sensores das abelhas.



Fonte: elaborado pelo autor, (2018).

Figura 8 - Relatório dados sensores das abelhas

ID	Tempo de leitura	ID da abelha	Antena	PC ID	Tempera
3422	2017-04-12T06:35:41.109480Z	0d0110200100101300000114	4	Edison-FZED437D00CBW501	38
3423	2017-04-12T06:35:41.369621Z	0d0110200100101300000114	4	Edison-FZED437D00CBW501	39
3424	2017-04-12T06:35:41.625701Z	0d0110200100101300000114	4	Edison-FZED437D00CBW501	37
3425	2017-04-12T06:35:41.889322Z	0d0110200100101300000114	4	Edison-FZED437D00CBW501	37
3426	2017-04-12T06:35:42.150658Z	0d0110200100101300000114	3	Edison-FZED437D00CBW501	39

Fonte: elaborado pelo autor, (2018).

O sistema vai ler as inconsistências das entradas e saídas dos sensores das abelhas e gerar um relatório de inconsistências e suas possíveis causas.

Os benefícios gerados com essa nova sistemática será:

- a) visualização: facilidade em visualizar o status dos serviços.
- b) emissão de relatórios: geração de relatórios de inconsistências e suas possíveis causas.

- c) histórico: produção de histórico diário dos arquivos em CSV possibilitando análise de problemas recorrentes.
- d) alerta por e-mail: encaminhamento de e-mail quando há indisponibilidade ou intervenção, agilizando o processo de comunicação à equipe sobre a existência de inconformidades no sensor.

O segundo objetivo foi a proposta de um método simples e fácil de aplicar para redução de problemas associados à obtenção dos dados. O sistema vai ler as inconsistências das entradas e saídas dos sensores das abelhas e gerar um relatório de inconsistências e suas possíveis causas. Foram utilizados critérios para análise das possíveis instabilidades no sistema:

- 1 Probabilidade: possibilidade desse evento ocorrer;
- 2 Tempo estimado para Solução: análise da solução;
- 3 Problema: quais os erros que podem ocorrer;
- 4 Consequência: os possíveis resultados ocasionados pelos problemas encontrados.
- 5 Prioridade: método GUT (gravidade, urgência e tendência) aplicado;
- 6 G.U.T: resultado.

- 1 Probabilidade foram destacados 3 subcritérios:
 - a) Baixa: improvável que ocorra durante o próximo ano, sem interrupções e investimento baixo para solucionar.
 - b) Média: pode ocorrer durante o próximo ano, ocasionando interrupções e prejuízos financeiros limitados.
 - c) Alta: muito provável que possa ocorrer no próximo ano, resultando em perdas significativas de dados e sendo consideradas medidas preventivas para evitar que o risco se agrave.
- 2 Tempo Estimado para Solução:
 - a) Curto: aumento de 5 – 10% do tempo.

- b) Médio: aumento de 10 – 20% do tempo.
- c) Alto: mais de 20% do aumento do tempo.

3 Prioridade: foi aplicado o método GUT (gravidade, urgência e tendência), onde,

G: gravidade, impacto que o problema pode causar se não for solucionado. A pontuação da gravidade varia de 1 a 5 seguindo o seguinte subcritério:

- 1: mínima gravidade;
- 2: pouco grave;
- 3: grave;
- 4: muito grave;
- 5: extremamente grave

U: urgência no prazo para se resolver um determinado problema. A pontuação da urgência varia de 1 a 5 seguindo o seguinte critério:

- 1: pode esperar;
- 2: pouco urgente;
- 3: urgente, merece atenção no curto prazo;
- 4: muito urgente;
- 5: necessidade de ação imediata.

T: tendência, representa o potencial de crescimento do problema, a probabilidade do problema se tornar maior com o passar do tempo. A pontuação da tendência varia de 1 a 5 seguindo o seguinte critério:

- 1: não irá mudar;
- 2: irá piorar a longo prazo;
- 3: irá piorar a médio prazo;
- 4: irá piorar a curto prazo;
- 5: irá piorar rapidamente.

A partir da proposta do sistema e da utilização dos critérios, adotando-se o método QA/QC será possível estabelecer a relação de causa e efeito entre possibilidade de prever e identificar problemas no controle e monitoramento dos dados com um tempo de solução rápido alcançando os objetivos apresentados na introdução.

O terceiro objetivo refere-se às Fontes de Informações para o usuário sobre as etiquetas eletrônicas, o que pode ser relevante para aqueles que possuem pouco conhecimento sobre a tecnologia. A implantação do sistema, a tarefa de monitorar e acompanhar as entradas e saídas dos sensores das abelhas será viável para os demais integrantes das equipes onde podem ficar focados em suas tarefas, não precisando interrompê-las para tratar situações de inconformidade nos dados dos sensores.

As principais limitações do Sistema QA/QC:

- Consumo de energia do Sistema aumenta porque o programa de QA/QC em tempo real requer mais trabalho. Se um problema for identificado, deverá haver mais comunicação ao usuário o que também acarreta um maior consumo de energia. Isso pode ser crítico em regiões temperadas (com curtos dias de inverno) ou com grande cobertura de nuvens (como na Amazônia) onde os painéis solares não recebem muita radiação solar para recarregar as baterias.
- Um QA/QC deve ser revisado frequentemente e a relevância dos critérios anteriormente estabelecidos deve ser questionada a fim de se utilizar um Sistema eficiente.
- Um novo sensor (p.ex., pH do mel, humidade, vibração) deverá ser acompanhado de um novo critério.
- As relações entre sensores pode também ser investigadas para se verificar se há falha de um sensor.

6 CONCLUSÃO

Esta dissertação propôs o desenvolvimento de um sistema, em Python, com base na utilização dos conceitos de QA/QC, de monitoramento que possibilitasse analisar os dados das abelhas e a leitura das inconsistências das entradas e saídas dos sensores proporcionando a geração de relatórios para acompanhamento deles e suas possíveis causas.

O sistema possibilitará a centralização das informações facilitando o diagnóstico, a visualização, a emissão dos relatórios, a geração de histórico diário dos dados dos sensores e o envio de alertas por e-mail.

A análise de causas mais detalhadas é possível. A causa da falha de um wi-fi pode ser muito diversa e a investigação detalhada de falhas deve ser tratada com o máximo rigor caso a caso. Por exemplo, a simples proximidade de uma fonte de microondas pode afetar roteadores de wi-fi. Isso não é inerente ao Sistema, mas ao ambiente em que ele está instalado. Portanto, embora um QA/QC forneça um bom nível de supervisão de falhas, trabalhos básicos de investigação de causas (p.ex. considerando o ambiente, erro humano ou eletrônico e mecânico) deve ser conduzido. Como consequência, o QA/QC poderá ser melhorado.

Conclui-se através das validações descritas acima, que a implantação do sistema desenvolvido, apresentará benefícios e bons resultados, além de atingir os seguintes resultados e metas:

- Elaboração de relatórios com qualidade e precisão nas decisões.
- Produzir um levantamento de dados conciso e com confiabilidade.
- Produzir contribuições técnicas sobre o tema para publicação em revistas e congressos nacionais e internacionais.
- Avaliação contínua dos dados.

REFERÊNCIAS

- CASTRO, C. C. de; OLIVEIRA, P. E. A. M. de; PIMENTEL, R. M. de M. Reproductive biology of the herkogamous vine *Chiococca alba* (L.) Hitchc. (Rubiaceae) in the Atlantic Rain Forest, SE Brazil. **Rev. Brasil. Bot.**, v. 31, n. 2, p. 317-321, abr./jun. 2008. Disponível em: [Revista Brasil. Bot., V.31, n.2, p.317-321, abr.-jun. 2008](http://www.scielo.br/rbrb/v31n2/317-321)
- SOUZA, P.; HUANG, G.; ZHANG, Y.; HE, J. CIRCE: Correcting Imprecise Readings and Compressing Excrescent points for querying common patterns in uncertain sensor streams. **Information Systems**, v. 38, p. 1234–1251, 2013. <https://doi.org/10.1016/j.is.2012.01.003>
- SUSANTO, F.; GILLARD, T.; SOUZA, P.; VINCENT, B.; BUDI, S.; ALMEIDA, A.; PESSIN, G.; ARRUDA, H.; WILLIAMS, R. N.; ENGELKE, U.; MARENDY, P.; HIRSCH, P.; AND HE, J. Addressing RFID Misreadings to Better Infer Bee Hive Activity. **IEEE Access**, v. 6, p. 2169-3536, 2018. DOI: 10.1109/ACCESS.2018.2844181.
- JUNIOR, J. A. **RFID** - Identificação por Radiofrequência. p. 2-3, 2007.
- FLORES, A. P. P.; JUNIOR, E. R. **A Tecnologia de Identificação por Radiofrequência Aplicada ao Gerenciamento Logístico**: Panorama Atual de Casos e Aplicações. 2014.
- WANDERLEY, M. N. D.; HOLANDA, L. M. C.; OLIVEIRA, J. B. **A Implantação da Tecnologia Radio Frequency Identification (rfid) em Processos Logísticos de uma Indústria de Baterias**. UNIFAVIP/DEVRY, 2014.
- NETO, M. M. F.; MAGALHÃES, R. L. **AprendER**: Ferramenta de Apoio à Construção de Diagrama Entidade Relacionamento para Deficientes Visuais, 2010.
- SILVA, J. P. F.; FIGUEIREDO, I. P. A. **Modelo Relacional de um Sistema de Gerenciamento de Vagas de Estágio**, 2009.
- POTTS, S. G.; BIESMEIJER, J. C.; KREMEN, C.; NEUMANN, P.; SCHWEIGER, O.; KUNIN, W. E. Global pollinator declines: trends, impacts and drivers. **Trends in Ecology and Evolution**, v. 25, p. 345-353, 2010. DOI: 10.1016/j.tree.2010.01.007
- ARAÚJO E. M. T.; BATISTA M. L. S. 2007. Artigo: Como ter uma boa visão sobre as Qualidades dos Dados. <https://www.devmedia.com.br/uma-visao-sobre-a-qualidade-dos-dados/6973>
- ARENA, M. V. N.; MARTINES, M. R.; SILVA, T. N.; DESTÉFANI, F. C.; MASCOTTI, J. C. S.; TOPPA, R. Multiple-scale approach for evaluating the occupation of stingless bees in Atlantic forest patches. **Forest Ecology and Management**, v. 430, p. 509-516, 15 dez. 2018. DOI [10.1016/j.foreco.2018.08.038](https://doi.org/10.1016/j.foreco.2018.08.038)

CARRIZO, D.; ALFARO, A. Quality assurance method in a software development methodology: a practice approach. Ingeniare. **Revista Chilena de Ingeniería**, v. 26, n. 1, p. 114-129, 2018.

BBC (14 de setembro de 2011). Disponível em: <http://g1.globo.com/ciencia-e-saude/noticia/2011/09/cientistas-usam-microchips-em-abelhas-para-investigar-esvaziamento-de-colmeias.html>. Acesso em: 01 mar. 2019

FARIA, A.; NATURE, W. T. Abelhas desaparecendo: Einstein estaria certo? 31 de maio de 2016. Disponível em: <https://ecoa.org.br/abelhas-desaparecendo-einstein-estaria-certo/>. Acesso em: 03 mar. 2019.

CASTELHANO, K. B.; SANTOS, R.D. e SANTOS, R. Aplicação da tecnologia rfid na logística aeroportuária. Disponível em: <http://www.fatecguaratingueta.edu.br/revista/index.php/RCOTGH/article/view/91/115>. Acesso em: 20 jun. 2019.

LONGEN, A. (21 de janeiro de 2019). *O que é WAMP? Um Guia Amigável para Iniciantes*. Fonte: Hostinger Blog: <https://www.hostinger.com.br/tutoriais/o-que-e-wamp> Acesso em: 23 jun. 2019

ZAMPIERI, G. (22 de janeiro de 2019). *O que é JavaScript?* Fonte: Hostinger Blog: <https://www.hostinger.com.br/tutoriais/o-que-e-javascript/> Acesso em: 23 jun. 2019.

ANDRADE, A. P. (25 de março de 2019). *O que é Django?* Fonte: TreinaWeb: <https://www.treinaweb.com.br/blog/o-que-e-django/>. Acesso em: 23 jun. 2019.

KAY, Russell. Python. Computerworld. (09 de maio de 2005). Disponível em: https://www.computerworld.com.au/article/131562/here_comes_python/. Acesso em: 23 jun. 2019.

APÊNDICES

Apêndice A - Esboço da elaboração do algoritmo de criação de avisos.

Tempos obtidos para a abelha 0d0110200100101300000114 no dia 12/04/2017, referentes a antena 1:

```
from leituras.models import Leitura
tempos_antena1 = Leitura.objects.filter(reading_time_local__year=2017,
reading_time_local__month=4, reading_time_local__day=12,
antenna=1).values_list('reading_time_local')
import pprint
pprint.pprint(list(tempos_antena1))
[(datetime.datetime(2017, 4, 12, 6, 44, 46, 300777, tzinfo=<UTC>),),
 (datetime.datetime(2017, 4, 12, 22, 20, 46, 142701, tzinfo=<UTC>),),
 (datetime.datetime(2017, 4, 12, 22, 52, 2, 489299, tzinfo=<UTC>),),
 (datetime.datetime(2017, 4, 12, 23, 6, 46, 196445, tzinfo=<UTC>),),
 (datetime.datetime(2017, 4, 12, 23, 9, 59, 22847, tzinfo=<UTC>),),
 (datetime.datetime(2017, 4, 12, 23, 9, 59, 529199, tzinfo=<UTC>),),
 (datetime.datetime(2017, 4, 12, 23, 44, 5, 225151, tzinfo=<UTC>),),
 (datetime.datetime(2017, 4, 12, 23, 46, 1, 507576, tzinfo=<UTC>),),
 (datetime.datetime(2017, 4, 12, 23, 47, 4, 456864, tzinfo=<UTC>),)]
```

Tempos obtidos para a abelha 0d0110200100101300000114 no dia 12/04/2017, referentes a antena 2:

```
from leituras.models import Leitura
tempos_antena2 = Leitura.objects.filter(reading_time_local__year=2017,
reading_time_local__month=4, reading_time_local__day=12,
antenna=2).values_list('reading_time_local')
import pprint
pprint.pprint(list(tempos_antena2))
[(datetime.datetime(2017, 4, 12, 6, 35, 51, 612834, tzinfo=<UTC>),),
 (datetime.datetime(2017, 4, 12, 6, 35, 51, 898882, tzinfo=<UTC>),),
 (datetime.datetime(2017, 4, 12, 6, 35, 52, 162655, tzinfo=<UTC>),),
 (datetime.datetime(2017, 4, 12, 6, 35, 52, 415896, tzinfo=<UTC>),),
 (datetime.datetime(2017, 4, 12, 6, 44, 45, 783261, tzinfo=<UTC>),),
 (datetime.datetime(2017, 4, 12, 6, 44, 46, 47224, tzinfo=<UTC>),),
 (datetime.datetime(2017, 4, 12, 22, 20, 45, 888702, tzinfo=<UTC>),),
 (datetime.datetime(2017, 4, 12, 22, 20, 46, 391576, tzinfo=<UTC>),),
 (datetime.datetime(2017, 4, 12, 22, 20, 46, 649477, tzinfo=<UTC>),),
 (datetime.datetime(2017, 4, 12, 22, 26, 21, 54787, tzinfo=<UTC>),),
 (datetime.datetime(2017, 4, 12, 22, 26, 21, 570859, tzinfo=<UTC>),),
```

(datetime.datetime(2017, 4, 12, 22, 31, 31, 803462, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 22, 31, 32, 282855, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 22, 35, 22, 229106, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 22, 35, 22, 454478, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 22, 35, 22, 782185, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 22, 35, 22, 999485, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 22, 52, 1, 692268, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 22, 52, 1, 999950, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 22, 52, 2, 212840, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 22, 55, 48, 723228, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 22, 55, 48, 976817, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 22, 55, 49, 235437, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 22, 55, 49, 502450, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 6, 44, 647853, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 6, 44, 901340, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 6, 45, 159936, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 6, 45, 424143, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 6, 45, 676061, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 6, 45, 938595, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 9, 58, 505805, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 9, 58, 763335, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 9, 59, 265390, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 32, 12, 429169, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 32, 12, 675907, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 32, 12, 917237, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 32, 14, 194899, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 32, 14, 433895, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 32, 14, 729659, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 32, 14, 984411, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 32, 15, 248732, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 32, 15, 494382, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 32, 15, 735808, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 38, 1, 963840, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 38, 2, 229890, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 38, 2, 476778, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 38, 2, 733523, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 38, 2, 987699, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 38, 3, 226299, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 38, 3, 469716, tzinfo=<UTC>)),

```
(datetime.datetime(2017, 4, 12, 23, 44, 4, 248222, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 44, 4, 445259, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 44, 4, 700876, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 44, 4, 966743, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 45, 59, 433084, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 45, 59, 699122, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 45, 59, 951104, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 46, 0, 215939, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 46, 0, 475156, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 46, 0, 733886, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 46, 0, 992066, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 46, 1, 245878, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 47, 3, 399198, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 47, 3, 666290, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 47, 3, 938014, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 47, 4, 191572, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 47, 5, 604692, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 47, 5, 907679, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 58, 15, 146108, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 58, 15, 355134, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 58, 15, 659714, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 58, 15, 858076, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 58, 16, 111613, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 58, 16, 397201, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 58, 16, 661358, tzinfo=<UTC>)),
(datetime.datetime(2017, 4, 12, 23, 58, 16, 966810, tzinfo=<UTC>))] ]
```

Depois de obter os tempos para uma dada abelha, num dado dia, referente as antenas, vai ser preciso comparar esses valores. Como as listas estão organizadas por ordem crescente de tempo, é possível tomar o primeiro valor de uma lista e compará-lo com o primeiro valor da outra lista:

Valor da primeira lista (antena 1): `datetime.datetime(2017, 4, 12, 6, 44, 46, 300777, tzinfo=<UTC>)`

Valor da segunda lista (antena 2): `datetime.datetime(2017, 4, 12, 6, 35, 51, 612834, tzinfo=<UTC>)`

Nas listagens acima a quantidade de leituras da antena 2 é bem maior que a da antena 1 para aquela abelha, naquele dia. Este registro significa que a abelha saiu mais vezes que entrou ou o sensor não detectou a entrada dela naquele dia.

Apêndice B – [...]

```
"""Views da
aplicação
leituras."""
"
```

```
from datetime import datetime, timedelta
from django.views import generic
from django.core.mail import BadHeaderError, send_mail
from django.http import HttpResponseRedirect
from django.core.urlresolvers import reverse
from .models import Leitura
```

```
class ProcessarArquivoView(generic.TemplateView):
    """Partial contendo o formulário de envio de arquivo."""
```

```
    template_name = 'leituras/processar_arquivo.html'
```

```
processar_arquivo = ProcessarArquivoView.as_view()
```

```
class LeiturasListView(generic.TemplateView):
    """Partial contendo a lista de leituras."""
```

```
    template_name = 'leituras/leituras_lista.html'
```

```
leituras_lista = LeiturasListView.as_view()
```

```

def send_email(request):
    leit = Leitura.objects.values_list('bee_id',
'reading_time_local').order_by('bee_id')
    for codigo, data_hora in leit:
        for data in data_hora:
            data_temp = datetime.strptime(data, '%Y-%m-%d %H:%M:%S')
            dado =
Leitura.objects.filter(bee_id=codigo).filter(reading_time_local__gt=F(data_te
mp) - timedelta(hours=12))
            if not dado and (dado.len() % 2) == 0:
                subject = 'alerta_bee'
                message = 'Não foram encontrados datas pares respectivos a
entrada e saída.'
                from_email = 'kr_barbosa@yahoo.com.br'
                if subject and message and from_email:
                    try:
                        send_mail(subject, message, from_email,
['admin@bee.com'])
                    except BadHeaderError:
                        return HttpResponse('Cabeçalho do email é inválido.')
                return HttpResponseRedirect(reverse('leituras:email'))

```

"""Módulo de configuração de
urls do projeto."""

```

from django import urls
from django.conf import settings
from django.conf.urls import static
from django.contrib import admin
from rest_framework import documentation

```

```

from . import api, views

```

```

urlpatterns = [
    urls.re_path(r'^$', views.index, name='index'),
    urls.re_path(r'^contas/',
urls.include('allauth.urls')),
    urls.re_path(r'^avisos/',
urls.include('avisos.urls')),
    urls.re_path(r'^leituras/',

```

```

urls.include('leituras.urls'),
    urls.re_path(r'^api/v1/', api.urls),
    urls.re_path(r'^api-auth/',
urls.include('rest_framework.urls',
namespace='rest_framework')),
    urls.re_path(r'^admin/', admin.site.urls),
    urls.re_path(r'^docs/',
documentation.include_docs_urls(title='Documentação
funcional da API')),
]

# media files in development
urlpatterns += static.static(settings.MEDIA_URL,
document_root=settings.MEDIA_ROOT)

if settings.DEBUG:
    import debug_toolbar
    urlpatterns += [
        urls.re_path(r'^__debug__/',
urls.include(debug_toolbar.urls))
    ]

<!doctype html>{%
load pipeline static
%}

<html lang="en">
<head>
    {% block meta %}
        <meta charset="UTF-8">
        <meta http-equiv="X-UA-Compatible"
content="IE=edge">
        <meta content="width=device-width,
initial-scale=1, maximum-scale=1, user-
scalable=no" name="viewport">
    {% endblock meta %}

    <title>{% block title %}Bem vindo{%
endblock title %}</title>

    <link rel="stylesheet" href="{% static
'bootstrap/dist/css/bootstrap.css' %}">

```

```

<link rel="stylesheet" href="{% static
'bootstrap-vue/dist/bootstrap-vue.min.css'
%}">
<link rel="stylesheet" href="{% static
'@fontawesome/fontawesome-
free/css/all.min.css' %}">
<link rel="stylesheet" href="{% static
'vue-loading-overlay/dist/vue-loading.css'
%}">

```

```
{% stylesheet 'base' %}
```

```

{% block styles %}{% endblock styles %}
</head>
<body>
  {% block body %}
    <div id="app" v-cloak>
      <b-container fluid>
        <b-row class="mt-1">
          <b-col></b-col>
          <b-col cols="6" md="10">
            <b-navbar toggleable="lg"
type="dark" variant="dark" style="border-
radius: .25em">
              <b-navbar-toggle target="nav-
collapse"></b-navbar-toggle>
              <b-collapse id="nav-collapse"
is-nav>
                <b-navbar-brand href="/#/"
class="mr-auto">BEE</b-navbar-brand>
                <b-navbar-nav class="ml-
auto">
                  <b-navbar-nav>
                    <b-nav-item
href="/#/processar/arquivo/"><span
class="text-white">Enviar
Arquivo</span></b-nav-item>
                  </b-navbar-nav>

```



```

        <b-nav-item-dropdown>
          <!-- Using 'button-
content' slot -->
          <template slot="button-
content">
            <span class="text-
white">Bem vindo(a) <em>{{
user.get_full_name|default:user.username
}}</em></span>
            </template>
            <b-dropdown-item
href="/contas/logout" onclick="return
confirm('Deseja sair da aplicação?')">
              <i class="fas fa-
sign-out-alt"></i> Logout
            </b-dropdown-item>
          </b-nav-item-dropdown>
        </b-navbar-nav>
      </b-collapse>
    </b-navbar>
  </b-col>
  <b-col></b-col>
</b-row>

  <router-view></router-view>
</b-container>
</div>
{% endblock body %}

{% block scripts %}{% endblock scripts %}

<script src="{% static
'jquery/dist/jquery.min.js' %}"></script>
<script src="{% static
'axios/dist/axios.min.js' %}"></script>
<script src="{% static
'vue/dist/vue.min.js' %}"></script>
<script src="{% static 'vue-
router/dist/vue-router.min.js'
%}"></script>

```

```

    <script src="{% static 'vee-
validate/dist/vee-validate.min.js'
%}"></script>
    <script src="{% static 'vee-
validate/dist/locale/pt_BR.js'
%}"></script>
    <script src="{% static 'vue-loading-
overlay/dist/vue-loading.min.js'
%}"></script>
    <script src="{% static 'bootstrap-
vue/dist/bootstrap-vue.min.js'
%}"></script>

    {% javascript 'avisos' %}
    {% javascript 'leituras' %}
    {% javascript 'app' %}
</body>
</html>

```

Upload do arquivo

```

<b-row
class="mt-
1">

```

```

<loading :active.sync="loading" :is-full-page="false"></loading>
<b-col></b-col>
<b-col cols="6" md="10">
    <b-alert variant="success" :show="sucessos.length > 0" dismissible>
        [[ sucessos[0] ]]
    </b-alert>
    <b-alert variant="danger" :show="erros.length > 0" dismissible>
        [[ erros[0] ]]
    </b-alert>
    <b-card class="text-center">
        <b-form-file name="file" v-model="file" placeholder="Clique para
selecionar o arquivo a ser enviado"
            drop-placeholder="Arraste o arquivo e largue aqui para selecioná-lo"
            accept=".csv" v-validate.initial="'required'"></b-form-file>
        <b-button class="mt-1" variant="primary" @click="enviarArquivo"
:disabled="errors.any()" block>
            <i class="fas fa-file-csv"></i> Processar
        </b-button>
    </b-card>
</b-col>

```

```
<b-col></b-col>  
</b-row>
```