



INSTITUTO FEDERAL DE CIÊNCIA E TECNOLOGIA DE PERNAMBUCO

Campus Recife

Departamento Acadêmico de Sistemas Eletrônicos
Tecnologia em Análise e Desenvolvimento de Sistemas

ISABELLA CHIARA DE SIQUEIRA

**SOFTWARE DE GERENCIAMENTO, ARMAZENAMENTO, PROCESSAMENTO E
VISUALIZAÇÃO DE DADOS EM TEMPO REAL PARA APLICAÇÃO NO SISTEMA
TELEMÉTRICO RAILBEE**

Recife
2021

ISABELLA CHIARA DE SIQUEIRA

**SOFTWARE DE GERENCIAMENTO, ARMAZENAMENTO, PROCESSAMENTO E
VISUALIZAÇÃO DE DADOS EM TEMPO REAL PARA APLICAÇÃO NO SISTEMA
TELEMÉTRICO RAILBEE**

Monografia apresentada ao Curso Tecnólogo em
Análise e Desenvolvimento de Sistemas do
Instituto Federal de Pernambuco, como requisito
parcial para obtenção do título de Tecnólogo em
Análise e Desenvolvimento de Sistemas.

Orientador: Prof. Henrique Correia Torres
Santos.

Coorientador: Prof. Dr. Rômulo César Carvalho
de Araújo.

Recife
2021

FICHA CATALOGRÁFICA

S618s

Siqueira, Isabella Chiara de.

Software de gerenciamento, armazenamento, processamento e visualização de dados em tempo real para aplicação no sistema telemétrico Railbee / Isabella Chiara de Siqueira; orientador Prof. Henrique Correia dos Santos; coorientador Prof. Dr. Rômulo César Carvalho de Araújo. Recife: IFPE, 2021.

46f.; il.

Trabalho de Conclusão de Curso (Tecnólogo em Análise e Desenvolvimento de Sistemas) – IFPE - Recife

Inclui Referências.

1. Gerenciamento de configurações de software 2. Sistema de transmissão de dados. 3. Railbee. 4. Gerenciamento de dados I. Siqueira, Isabella Chiara de. II. IFPE. III. Título.

CDD 004.21

SOFTWARE DE GERENCIAMENTO, ARMAZENAMENTO, PROCESSAMENTO E
VISUALIZAÇÃO DE DADOS EM TEMPO REAL PARA APLICAÇÃO NO SISTEMA
TELEMÉTRICO RAILBEE.

Trabalho aprovado. Recife, 24 de fevereiro de 2021.

Professor Orientador: Henrique Correia Torres Santos

Rômulo César Carvalho de Araújo

Marcos Antônio Eugênio Araújo

Recife

2021

AGRADECIMENTOS

Gostaria de agradecer e dedicar esta dissertação às seguintes pessoas:

Minha mãe, meu pai e meus irmãos que sempre acreditaram em mim, me apoiando em todos os meus trabalhos.

Meu orientador Henrique Correia Torres Santos por ter me guiado com seu conhecimento e principalmente por ter compartilhado todo seu entusiasmo com a área de programação comigo.

Meu coorientador Prof. Dr. Rômulo César Carvalho de Araújo por ter me dado a oportunidade de participar do grupo de pesquisa RailBee e de desenvolver este trabalho.

O grupo de pesquisa e extensão RailBee por ter me acolhido e dado todo o suporte com os equipamentos e no desenvolvimento deste projeto.

Ao CNPq-SETEC/MEC por financiar a bolsa o projeto RailBee.

Ao IFPE por financiar a bolsa concedida através do programa institucional de bolsas de iniciação - PIBIC.

Todo o corpo docente do curso de Tecnólogo em Análise e Desenvolvimento de Sistemas.

RESUMO

Este projeto tem como objetivo o desenvolvimento do software para o sistema telemétrico RailBee[®] que tem como base o monitoramento dos trens urbanos, usando mecanismos de rede de sensores sem fio e sistemas embarcados. Software este que recebe e exibe, em tempo real, os dados das variáveis referentes a velocidade real do trem, corrente do motor de tração, aplicação de freio, pressão das bolsas de ar da suspensão do trem, condição de cabine (frente ou ré) e indicação de portas (abertas ou fechadas). O sistema telemétrico possui uma natureza inovadora e é composto por quatro subsistemas: um dispositivo embarcado no veículo captando as informações do trem (estação móvel), uma repetidora (estação roteadora) de sinais de radiofrequência (RF), um conversor de sinais de RF para Ethernet (estação base) e um software para o tratamento, armazenamento e exibição dos dados (estação central). As atividades principais do software da estação central são de tratar os dados dos trens, exibi-los para seus usuários do Centro de Controle Operacional (CCO) e os armazenar em uma base de dados para futuras pesquisas. Para alcançar seus objetivos, o projeto é moldado por uma arquitetura de microsserviços e sistema de *message brokers*, além de permitir execução concorrente através de *multithread* e uma interface com o usuário reativa através do uso de *websockets*. O sistema mantém múltiplas conexões simultâneas com dispositivos embarcados, para realizar a recepção dos pacotes de dados dos trens. Esses pacotes são tratados, armazenados no banco de dados e encaminhados para filas de armazenamento, onde o destino é uma aplicação *web* que exibe as informações dos trens para os usuários. Esse monitoramento em tempo real e o arquivamento dessas informações proporcionam a realização de análises e relatórios, melhorando a qualidade na prestação do serviço realizado pela empresa de transporte público sobre trilhos.

Palavras-chaves: RailBee. Estação Central. Software. Pacote de dados.

ABSTRACT

This project aims to develop software for the RailBee® telemetric system based on the monitoring of city trains using wireless sensor network mechanisms and embedded systems. This software receives and displays, in real time, variable data for actual train speed, traction motor current, brake application, train suspension airbag pressure, cab condition (forward or reverse) and indication of doors (open or closed). The telemetric system has an innovative nature and is composed of four subsystems: a device embedded in the vehicle capturing the train's information (mobile station), a repeater (router station) of radio frequency (RF), an RF signal converter Ethernet (base station) and a software for processing, storing and displaying data (central station). The main activities of the central station software are to process train data, display it to its Operational Control Center (CCO) users and store it in a database for future research. To achieve its objectives, the project is modeled by a microservice architecture and messaging broker system, in addition to allowing the competitor to run through multithreaded and a reactive user interface using websockets. The system maintains multiple simultaneous connections to embedded devices to receive train data packets. These packages are handled, stored in the database, and forwarded to storage queues, where the destination is a web application that displays train information to users. This real-time monitoring and archiving of this information enables analysis and reporting, improving the quality of service provided by the public transit company.

Keywords: RailBee. Central Station. Software. Data package.

LISTA DAS FIGURAS

Figura 1 - Representação do Sistema RailBee®

14

Figura 2 - Diagrama de Casos de Uso do Usuário 20

Figura 3 - Diagrama Entidade-Relacionamento 33

Figura 4 - Fluxograma Representante da Arquitetura do Sistema 38

Figura 5 - Tela inicial do software da Estação Central 45

Figura 6 - Tela de monitoramento 45

Figura 7 - Menu das linhas ferroviárias 46

Figura 8 - Menu para filtrar os trens 46

Figura 9 - Menu para ocultar variáveis 47

Figura 10 - Painel das variáveis digitais 47

SUMÁRIO DAS TABELAS

Tabela 1 - Lista de requisitos	16
Tabela 2 - Fazer login	21
Tabela 3 - Fazer logout	21
Tabela 4 - Exibir informações dos trens	21
Tabela 5 - Selecionar linha ferroviária	22
Tabela 6 - Ocultar Trem	22
Tabela 7 - Retornar Trem Oculto	22
Tabela 8 - Ocultar Variável	23
Tabela 9 - Retornar Variável Oculta	23
Tabela 10 - Cadastrar Usuário ou Administrador	24
Tabela 11 - Editar Informações de Usuário	24
Tabela 12 - Apagar Usuário	24
Tabela 13 - Cadastrar Linha Ferroviária	25
Tabela 14 - Editar Linha Ferroviária	25
Tabela 15 - Apagar Linha Ferroviária	26
Tabela 16 - Cadastrar Estação de Passageiros	26
Tabela 17 - Editar Estação de Passageiros	27
Tabela 18 - Apagar Estação de Passageiros	27
Tabela 19 - Cadastrar Trem	28
Tabela 20 - Editar Trem	28
Tabela 21 - Apagar Trem	28
Tabela 22 - Cadastrar Dispositivo Estação Móvel	29
Tabela 23 - Editar Dispositivo Estação Móvel	29
Tabela 24 - Apagar Dispositivo Estação Móvel	30
Tabela 25 - Cadastrar Dispositivo Estação Base	30
Tabela 26 - Editar Dispositivo Estação Base	31
Tabela 27 - Apagar Dispositivo Estação Base	31
Tabela 28 - Cadastrar Dispositivo Estação Roteadora	31
Tabela 29 - Editar Dispositivo Estação Roteadora	32
Tabela 30 - Apagar Dispositivo Estação Roteadora	32

LISTA DE SIGLAS

AMQP	<i>Advanced Message Queuing Protocol</i>
API	<i>Application Programming Interface</i> (Interface de Programação de Aplicativos)
CBTU/METROREC	Companhia Brasileira de Trens Urbanos/Metrô do Recife
CCO	Centro de Controle Operacional
EB	Estação Base
EM	Estação Móvel
ER	Estação Roteadora
EC	Estação Central
IDE	<i>Integrated Development Environment</i> (Ambiente de Desenvolvimento Integrado)
MAC	<i>Media Access Control</i> (Controle de Acesso ao Meio)
TCP/IP	<i>Transmission Control Protocol</i> (Protocolo de Controle de Transmissão) <i>/ Internet Protocol</i> (Protocolo de Internet)
RF	<i>Radio Frequency</i> (Radiofrequência)
SQL	<i>Structured Query Language</i>

SUMÁRIO

1 INTRODUÇÃO.....	11
1.1 Objetivos.....	12
2 FUNDAMENTAÇÃO TEÓRICA.....	13
2.1 Sistema Telemétrico RailBee®.....	13
2.2 Microsserviços.....	14
3.3 Sistema de troca de mensagem (Filas)	15
3 PROJETO DE UM SOFTWARE PARA MONITORAMENTO EM TEMPO REAL DO SISTEMA TELEMÉTRICO RAILBEE.....	16
3.1 Metodologia do Desenvolvimento.....	16
3.2 Fase de Requisitos.....	17
3.2.1 <i>Requisitos.....</i>	17
3.2.2 <i>Casos de Uso.....</i>	20
3.3 Diagrama Entidade-Relacionamento.....	33
3.4 Tecnologias.....	35
3.4.1 <i>Java.....</i>	35
3.4.2 <i>Django.....</i>	36
3.4.3 <i>RabbitMQ.....</i>	36
3.4.4 <i>WebComponents.....</i>	36
3.4.5 <i>WebSocket.....</i>	37
3.4.6 <i>PostgreSQL.....</i>	37
3.4.7 <i>Visual Studio Code.....</i>	37
3.4.8 <i>Eclipse.....</i>	37
3.5 Arquitetura.....	38
3.5.1 <i>Receptor.....</i>	38
3.5.2 <i>Conversor.....</i>	38
3.5.3 <i>Aplicação Web.....</i>	39
4 CONCLUSÃO.....	40
REFERÊNCIAS.....	41
APÊNDICE A.....	43
APÊNDICE B.....	44

1 INTRODUÇÃO

A evolução da urbanização trouxe como consequência a modernidade dos transportes públicos. No cotidiano de uma metrópole, como a Região Metropolitana do Recife, pessoas estão se locomovendo, constantemente, de um ponto a outro por meios desses transportes. Um deles é o Sistema Metroferroviário do Recife, que transporta cerca de 400 mil passageiros por dia (CBTU, 2018). Por esse motivo, A Companhia Brasileira de Trens Urbanos - Metrô do Recife (CBTU/Recife) deve operar de forma que imprevistos e acidentes sejam evitados, e caso ocorram devem ser tratados adequadamente para não afetar a operação comercial dos trens e dos serviços prestados aos usuários do sistema de transporte sob trilhos.

Para que se faça a manutenção e monitoramento dos trens utiliza-se o oscilógrafo, um aparelho que observa e registra as variações da corrente elétrica do trem. Porém o seu uso requer um conjunto de processos que demandam tempo para sua execução. Por esse motivo, esse modelo se trata de um monitoramento periódico. Tendo isto em vista, vários estudos para viabilizar o monitoramento em tempo real dos trens vêm sendo feito com o passar dos anos. Logo, a implantação do sistema telemétrico dinâmico sem fio em veículos rodoferroviários em malhas metroferroviárias denominado RailBee[®] oferecerá um conjunto de informações detalhadas sobre o desempenho desses veículos (ARAÚJO, 2009).

O RailBee[®] é um sistema de telemetria, desenvolvido por um grupo de ensino, pesquisa e extensão do IFPE - Campus Recife, que realiza a transmissão através de uma rede fixa ou sem fio de dados provindos de dispositivos remotos distribuídos sobre uma área geográfica, para uma central de monitoramento. Tratando-se de uma tecnologia de telemetria metroviária eficiente, de fácil implementação e de baixo custo (SANTOS, 2010, pg.17), o sistema promove o monitoramento dos trens em tempo real para que decisões sejam tomadas de forma mais rápida e segura. Ele é formado por quatro estações: Estação Móvel, um dispositivo instalado no trem captando suas variáveis e as transmite por radiofrequência; Estação Roteadora, dispositivos que repetem os sinais enviados da Estação Móvel até a Estação Base; Estação Base, um dispositivo coordenador que recebe e processa os sinais vindos da Estação Móvel e os envia para a Estação Central; Estação Central, um software de monitoramento e armazenamento das

informações dos trens localizada na Centro de Controle de Operações (CCO) da CBTU-Recife.

Este trabalho tem como finalidade apresentar o desenvolvimento do software de monitoramento da Estação Central que, através do protocolo *Ethernet* com uma rede de fibra óptica, recebe em tempo real os dados enviados pelas Estações Bases. Esses dados chegam até a Estação Central em pacotes que passam por processos de recepção, verificação e armazenamento de dados. Neste pacote estão contidos os valores das variáveis do trem. São elas: velocidade real, pressão das bolsas de ar primária e secundárias do trem, corrente do motor de tração, aplicação dos freios, condição de cabine (frente ou ré) e indicação das portas (abertas ou fechadas). Além de realizar o tratamento dos pacotes para o monitoramento, o software também tem a responsabilidade de armazenar esses valores para que sejam efetuados análises e relatórios posteriores pelos usuários do CCO.

Uma das etapas do cronograma de atividades do grupo de ensino, pesquisa e extensão RailBee® é a de realizar a transferência de tecnologia para a CBTU/Recife. Tendo isto em vista, um sistema que permita um controle de acesso às informações coletadas, sua administração e gerenciamento por parte da equipe técnica do metrô precisa ser desenvolvido.

1.1 Objetivos

Este projeto tem como objetivo o desenvolvimento do Software de Gerenciamento e Armazenamento de Dados em Tempo Real do Sistema Telemétrico RailBee.

Os objetivos específicos são:

- Desenvolver um software para que colete e gerencie os dados enviados das Estações Base;
- Desenvolver uma interface de monitoramento das seguintes variáveis dos trens: velocidade real, pressão das bolsas de ar, aceleração do veículo, corrente do motor de tração, condição de cabine (frente ou ré) e as portas (abertas ou fechadas) do trem;
- Criar uma base de dados relacional para armazenar e gerenciar a grande quantidade de dados recebidos das estações.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 Sistema Telemétrico RailBee®

O sistema telemétrico RailBee® foi definido e desenvolvido na tese de doutorado do professor doutor Rômulo César Carvalho de Araújo, com o objetivo de monitorar, em tempo real, os trens em circulação nas vias permanentes do metrô do Recife. O sistema utiliza um método original que está de acordo com o protocolo de comunicação padrão internacional ZigBee/IEEE 802.15.4. O sistema é composto por quatro estações:

- A. Estação Móvel (EM), uma Rede de Sensores Sem Fio (RSSF) localizada no computador de bordo do trem que, com a tecnologia ZigBee, capta os dados do veículo e os envia via radiofrequência para uma Estação Base diretamente ou através de uma Estação Roteadora;
- B. Estação Roteadora (ER), um dispositivo localizado ao longo das vias permanentes entre as estações de passageiros, servindo como uma repetidora de sinais;
- C. Estação Base (EB), um sistema localizado nas estações de passageiros que recebe os sinais da EM e converte-os para o padrão Ethernet e os envia para a Estação Central;
- D. Estação Central (EC), um sistema de supervisão localizado no Centro de Controle de Operações (CCO), exibindo os dados recebidos assim que chegam no CCO.

Através da figura 1 pode-se ter uma visão gráfica de como são organizados os subsistemas.

Figura 1: Representação do Sistema RailBee® para uma rede de estação de trem urbano.



Fonte: ARAÚJO, R. C. C. (2009, p. 27)

O sistema telemétrico RailBee® tem como objetivo auxiliar o monitoramento dos trens nas vias permanentes trazendo benefícios aos diagnósticos e ações preventivas de operação e manutenção do sistema metroferroviário. Também favorece a tomada de decisões preditivas, com base em prognósticos auferidos das medidas em tempo real, de forma a possibilitar estratégias de intervenção que possam evitar a degradação parcial ou total dos serviços de transporte prestados à população. (SANTOS, 2010)

2.2 Microserviços

Com a evolução das técnicas de desenvolvimento de software, a etapa de elaborar a arquitetura do sistema também obteve grandes avanços. Dentre os diferentes tipos de arquiteturas existentes, a de microserviços surgiu, de acordo com (LEWIS and FOWLER, 2014), com grande relevância no passar dos anos, com o intuito de descrever uma maneira mais específica de projetar aplicativos de software como conjuntos de serviços executáveis de forma independente.

Os microserviços, segundo (NEWMAN, 2015), são pequenos e têm como foco realizar uma ação específica, sendo ela de alto ou baixo grau de dificuldade. Ou seja, cada parte do código tem sua função clara e definida, realizando tarefas únicas. Esta delegação de atividades faz com que seja evitado o acúmulo de código que não faz parte da lógica aplicada em um determinado componente.

A integração dos microsserviços é feita de forma eficiente, pois eles são autônomos. Cada microsserviço funciona como um entidade/componente para a arquitetura no geral. Logo é possível realizar sua execução em máquinas diferentes se for preciso. A comunicação entre os serviços é realizada através de APIs, *Application Programming Interface* (Interface de Programação de Aplicativos). Além disso, os microsserviços possuem como característica a diversidade de tecnologias que comunicando entre si trabalham em conjunto. Utilizar diferentes tipos de linguagem entrega a possibilidade de escolher ferramentas que sejam adequadas para diferentes tipos de situações, melhorando assim o desempenho do projeto.

2.3 Sistema de troca de mensagem (Filas)

De acordo com Thadeu 2012, os sistemas de *middleware* orientado a mensagem (MOMs) realizam o trabalho com uma troca assíncrona de mensagens. Nestes sistemas existem dois tipos de entidades: uma que desempenha o papel de um produtor, que seria o remetente, e a outra que desempenha o papel do consumidor, que seria o destinatário. Uma de suas características importantes é o de garantir que as mensagens sejam entregues de um ponto ao outro. Este sistema utiliza filas transacionais para salvar as mensagens recebidas pelo MOM. Assim, a remoção de uma mensagem enviada pelo remetente da fila só é realizada com a confirmação do recebimento do destinatário.

Os sistemas baseados em *message brokers* descendem dos MOMs. Porém ao contrário destes, eles agem de forma que promovem uma maior flexibilidade para o roteamento e filtragem de mensagens, além de permitir a adição de regras de negócio ao processamento das mensagens. Um *message broker* oferece também ferramentas administrativas para o usuário, dando a liberdade para o gerenciamento das regras para o roteamento e filtragem de mensagens.

Existem vários tipos de protocolos usados por *message brokers*. O padrão *Advanced Message Queuing Protocol* (AMQP) é uma proposta recentemente criada por um conjunto de empresas, tais como JPMorgan Chase, Microsoft Corporations, Cisco Systems, Red Hat, entre outras. Foi projetado para oferecer o suporte para uma ampla variedade de aplicativos de mensagens, se tornando até um protocolo padrão para este tipo de sistema.

3 SOFTWARE DE GERENCIAMENTO, ARMAZENAMENTO, PROCESSAMENTO E VISUALIZAÇÃO DE DADOS EM TEMPO REAL PARA APLICAÇÃO NO SISTEMA TELEMÉTRICO RAILBEE

3.1 Metodologia do Desenvolvimento

Por se tratar de um desenvolvimento de software, a metodologia deste projeto é baseada nos processos do ciclo de vida de desenvolvimento de software (SOMMERVILLE, 2003), sendo eles as seguintes fases:

- Fase de requisitos: levantamento dos requisitos e elaboração dos diagramas de caso de uso para fornecer uma descrição clara e consistente do que o sistema deve fazer;
- Fase de projeto: concepção da arquitetura do sistema visando atender os requisitos definidos e definição das tecnologias a serem utilizadas;
- Fase de implementação: elaboração do projeto a nível de linguagem de programação;
- Fase de testes: realização de testes, de acordo com o que foi estabelecido nos requisitos, no que foi implementado e
- Fase de produção: implantação do produto final, versionamento, atualizações no software, etc.

Para o desenvolvimento do software da EC a fase de produção não será englobado nesse projeto, sendo apenas realizadas as de requisitos, projeto, implementação e testes.

3.2 Fase de Requisitos

3.2.1 Requisitos

Tabela 1 - Lista de requisitos

Número	Nome	Descrição
RF001	Receber Dados da Estação Base	O sistema deve se conectar e receber os dados enviados pelos dispositivos embarcados das Estações Base.

RF002	Tratar os Pacotes Recebidos	O sistema deve realizar o tratamento dos pacotes recebidos. O tratamento consiste em dividir o pacote de acordo com as seguintes informações requisitadas: endereço do dispositivo, as variáveis analógicas (velocidade, pressão das bolsas de ar, corrente e freio) e as variáveis digitais (portas (abertas ou fechadas) e condição de cabine). Exceto pelo endereço do dispositivo, as outras variáveis devem passar por cálculos que resultarão no no valor real para análise.
RF003	Apresentar as Informações dos Trens	O sistema deve apresentar as informações processadas em tempo real de execução para o usuário. Essas informações são: velocidade, pressão das bolsas de ar, corrente, freio, portar (abertas ou fechadas) e condição de cabine. A disposição desses dados devem ser em tabela e cada conjunto dessas informações está relacionada a um trem.
RF004	Armazenar no Banco de Dados as Informações Processadas	O sistema deve armazenar no seu banco de dados as informações processadas.
RF005	Permitir a Customização da Interface Principal da Visualização dos Dados	O usuário deve poder organizar os elementos da interface principal que apresenta os dados dos trens de forma que lhe seja mais favorável. O usuário deve poder escolher: qual linha se deseja observar, quais trens que terão suas informações exibidas e quais variáveis deverão ser monitoradas.
RF006	Visualizar Informações Expandidas do Trens	O usuário deve poder escolher um ou mais trens apresentados na tela para visualizar suas informações de forma mais detalhada e em gráficos.
RF007	Criar Usuário	O administrador deve poder cadastrar um novo usuário comum ou um outro administrador. O usuário possui e-mail, senha, nome, sobrenome e o cargo de usuário normal. Já o administrador deve ter nome, sobrenome, e-mail, senha e cargo de usuário administrador. O e-mail é único para cada usuário e administrador. A senha deve ter no mínimo de 8 caracteres e máximo de 64 caracteres.
RF008	Fazer Login	O usuário e administrador deve poder iniciar uma sessão no sistema usando o e-mail e senha

		cadastrados.
RF009	Fazer Logout	O usuário e administrador deve poder encerrar uma sessão no sistema.
RF010	Editar Informações	O administrador deve poder alterar as informações da sua conta e/ou de outras cadastradas .
RF011	Apagar Usuário	O administrador deve poder apagar sua conta e/ou de outras cadastradas.
RF012	Cadastrar Linha Ferroviária	O administrador deve poder cadastrar uma nova linha ferroviária no sistema. A linha possui o nome. O nome deve ter no máximo 15 caracteres e ser única.
RF013	Cadastrar Estação de Passageiros	O administrador deve poder cadastrar uma nova estação de passageiros no sistema. A estação de passageiros possui o nome completo, a sigla que corresponda ao nome, o tipo de estação (se é integrada ou não) e a(s) linha(s) na qual a estação faz parte. O nome deve conter no máximo 25 caracteres e a sigla deve ter 3 caracteres e ser única.
RF014	Cadastrar Trem	O administrador deve poder cadastrar um novo trem no sistema. O trem possui o seu número e a linha na qual ele percorre.
RF015	Cadastrar Dispositivo Estação Móvel	O administrador deve poder cadastrar um novo dispositivo estação móvel no sistema. A estação móvel possui o <i>Media Access Control</i> (MAC) do dispositivo, o tipo de dispositivo (se é analógica ou digital), o trem no qual está instalado e o tipo de pacote que ele envia.
RF016	Cadastrar Dispositivo Estação Roteadora	O administrador deve poder cadastrar um novo dispositivo estação roteadora no sistema. A estação roteadora possui o MAC do dispositivo, as duas estações base nas quais retransmite o pacote e a sua localização na via permanente.
RF017	Cadastrar Dispositivo Estação Base	O administrador deve poder cadastrar um novo dispositivo estação base no sistema. A estação base possui o IP, a porta pela qual se comunica, se está ativa ou não e a estação de passageiros na qual está instalada.

RF018	Apagar Linha Ferroviária	O administrador deve poder apagar uma linha ferroviária cadastrada no sistema.
RF019	Apagar Estação de Passageiros	O administrador deve poder apagar uma estação de passageiros cadastrada no sistema.
RF020	Apagar Trem	O administrador deve poder apagar um trem no sistema cadastrado no sistema.
RF021	Apagar Dispositivo Estação Móvel	O administrador deve poder apagar um dispositivo estação móvel cadastrado no sistema.
RF022	Apagar Dispositivo Estação Roteadora	O administrador deve poder apagar um dispositivo estação roteadora cadastrado no sistema.
RF023	Apagar Dispositivo Estação Base	O administrador deve poder apagar um dispositivo estação base cadastrado no sistema.
RF024	Editar Linha Ferroviária	O administrador deve poder editar os dados de uma linha ferroviária cadastrada no sistema.
RF025	Editar Estação de Passageiros	O administrador deve poder editar os dados de uma estação de passageiros cadastrada no sistema.
RF026	Editar Trem	O administrador deve poder editar os dados de um trem no sistema cadastrado no sistema.
RF027	Editar Dispositivo Estação Móvel	O administrador deve poder editar os dados de um dispositivo estação móvel cadastrado no sistema.
RF028	Editar Dispositivo Estação Roteadora	O administrador deve poder editar os dados de um dispositivo estação roteadora cadastrado no sistema.
RF029	Editar Dispositivo Estação Base	O administrador deve poder editar os dados de um dispositivo estação base cadastrado no sistema.

Fonte: Autor (2019)

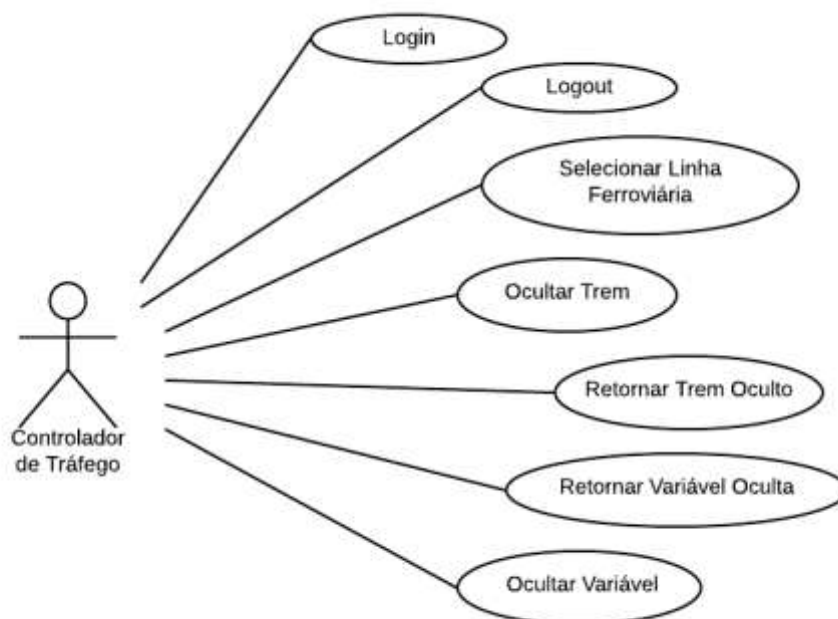
3.2.2 Casos de Uso

O diagrama apresentado na figura abaixo contém as funcionalidades do usuário Controlador de Tráfego, o qual está no CCO monitorando os trens nas linhas ferroviárias. Ele é responsável por:

- Efetuar o *login* e *logout* no sistema;
- Selecionar a linha ferroviária que deseja monitorar no momento;
- Ocultar um ou mais trens que não deseja visualizar;
- Retornar um trem que foi oculto anteriormente;

- Ocultar uma ou mais variáveis que não deseja visualizar;
- Retornar uma variável que foi ocultada anteriormente.

Figura 2: Diagrama de Casos de Uso do Usuário



Fonte: Autor (2019)

Neste sistema também há um usuário Administrador, que irá realizar gerenciamento das informações armazenadas no banco de dados.

Tabela 2 - Fazer login

Caso de Uso	Login
Descrição	O usuário e/ou administrador inicia uma sessão no sistema usando o e-mail e senha cadastrados.
Atores	Usuário; Administrador
Pré-condições:	O e-mail e senha usados pelo usuário e/ou administrador devem estar cadastrados no sistema.
Sequência de Eventos (Principal)	1. Na tela inicial o usuário insere os dados nos campos de e-mail e senha; 2. O usuário clica em confirmar; 3. O sistema valida os campos preenchidos pelo usuário; 4. O sistema cria uma nova sessão para o usuário; 5. O usuário é direcionado para a tela de visualização das informações
Pós-condições	Um nova sessão deve ser iniciada e o usuário ou administrador terá acesso às funcionalidades do sistema.

Fonte: Autor (2019)

Tabela 3 - Fazer logout

Caso de Uso	Logout
Descrição	O usuário ou administrador encerra a sessão no sistema.
Atores	Usuário; Administrador
Pré-condições:	O usuário ou administrador deve estar logado.
Sequência de Eventos (Principal)	1. O usuário ou administrador na tela principal clica no botão de "Sair"; 2. O sistema encerra a sessão do usuário; 3. O usuário ou administrador é redirecionado para a tela de login.
Pós-condições	O usuário ou administrador encerrará a sessão e voltará para a tela de login.

Fonte: Autor (2019)

Tabela 4 - Exibir informações dos trens

Caso de Uso	Exibição das Informações dos Trens em Tempo Real
Descrição	O sistema exibe as informações dos trens em tempo real na tela principal.
Atores	Sistema
Pré-condições:	O usuário deve estar logado; O sistema deve estar conectado a pelo menos uma Estação Base; O sistema deve estar conectado a fila e recebendo pacotes da mesma.
Sequência de Eventos (Principal)	1. O usuário faz o login no sistema; 2. O sistema realiza a conexão com a fila; 3. O sistema consome o pacote de dados recém recebido da fila; 4. O sistema realiza o tratamento dos dados do pacote; 5. O sistema armazena os dados tratados no banco de dados; 6. O sistema exibe as informações extraídas do tratamento na tela principal para o usuário.
Pós-condições	As informações dos trens serão exibidas na tela.

Fonte: Autor (2019)

Tabela 5 - Selecionar linha ferroviária

Caso de Uso	Selecionar a Linha Ferroviária
Descrição	O usuário seleciona linha ferroviária que deseja monitorar os seus trens.
Atores	Usuário
Pré-condições:	O usuário deve estar logado.
Sequência de Eventos (Principal)	1. O usuário clica na lista de opções de linhas de ferroviária; 2. O usuário clica na opção de linha desejada; 3. O sistema exibe a lista de trens daquela linha assim como suas variáveis;
Pós-condições	Será exibido as informações dos trens pertencentes da linha escolhida.

Fonte: Autor (2019)

Tabela 6 - Ocultar Trem

Caso de Uso	Ocultar Trem
Descrição	O usuário escolhe um ou mais trens que deseja ocultar a sua visualização.
Atores	Usuário
Pré-condições:	O usuário deve estar logado.
Sequência de Eventos (Principal)	1. O usuário seleciona um ou mais trens na tela principal; 2. O usuário clica em “Ocultar trens” no menu; 3. O sistema desabilita os trens selecionados deixando apenas os não selecionados amostra.
Pós-condições	Será ocultado apenas o(s) trem(ns) escolhidos pelo usuário.

Fonte: Autor (2019)

Tabela 7 - Retornar Trem Oculto

Caso de Uso	Retornar Trem Oculto
Descrição	O usuário escolhe retornar o(s) trem(ns) ocultado(s) anteriormente.
Atores	Usuário
Pré-condições:	O usuário deve estar logado; O sistema deve possuir pelo menos um trem ocultado anteriormente

Sequência de Eventos (Principal)	1. O usuário clica na opção de “Mostrar Trens” no menu. 2. O sistema exibe os trens ocultos na tela principal.
Pós-condições	O sistema exibirá todos os trens ocultos.

Fonte: Autor (2019)

Tabela 8 - Ocultar Variável

Caso de Uso	Ocultar Variável
Descrição	O usuário oculta a variável que não deseja visualizar.
Atores	Usuário
Pré-condições:	O usuário deve estar logado;
Sequência de Eventos (Principal)	1. O usuário clica no menu onde para cada variável cadastrada terá a opções de oculta-la; 2. O usuário clica na opção desejada; 3. O sistema oculta a variável.
Pós-condições	O sistema ocultará a variável escolhida pelo usuário.

Fonte: Autor (2019)

Tabela 9 - Retornar Variável Oculta

Caso de Uso	Retornar Variável Oculta
Descrição	O usuário escolhe retornar a(s) variável(eis) oculta(s) anteriormente.
Atores	Usuário
Pré-condições:	O usuário deve estar logado; O sistema deve possuir pelo menos uma variável oculta.
Sequência de Eventos (Principal)	1. O usuário clica no menu onde para cada variável oculta terá a opção de desocultar a mesma; 2. O usuário clica na opção desejada; 3. O sistema exibe a variável escolhida;
Pós-condições	O sistema exibirá a variável oculta.

Fonte: Autor (2019)

Tabela 10 - Cadastrar Usuário ou Administrador

Caso de Uso	Cadastrar Usuário ou Administrador
Descrição	O administrador cadastra um novo usuário ou administrador no sistema.
Atores	Administrador
Pré-condições:	O administrador deve estar logado.
Sequência de Eventos (Principal)	1. O administrador clica na opção de Usuários; 2. O administrador clica em cadastrar uma nova conta; 3. O administrador preenche os campos; 4. O administrador clica em Salvar; 5. O administrador registra no banco o novo cadastro.
Pós-condições	Será registrado no banco um novo usuário.

Fonte: Autor (2019)

Tabela 11 - Editar Informações de Usuário

Caso de Uso	Editar Informações de Usuário
Descrição	O administrador edita as informações cadastradas anteriormente de um usuário.
Atores	Administrador
Pré-condições:	O administrador deve estar logado.
Sequência de Eventos (Principal)	1. O administrador, na tela de administrador, clica na opção "Usuários"; 2. O administrador seleciona a conta que deseja editar; 3. O administrador altera o campo de desejado; 4. O administrador clica em salvar; 5. O administrador registra a alteração.
Pós-condições	O registro do usuário será modificado.

Fonte: Autor (2019)

Tabela 12 - Apagar Usuário

Caso de Uso	Apagar Usuário
Descrição	O administrador deleta a conta de um usuário cadastrado no sistema.
Atores	Administrador

Pré-condições:	O administrador deve estar logado.
Sequência de Eventos (Principal)	1. O administrador, na tela de administrador, clica na opção “Usuários”; 2. O administrador seleciona o cadastro desejado; 3. O administrador clica em “Deletar”; 4. O administrador apaga o registro da conta; 5. O administrador é redirecionado para a tela de administrador.
Pós-condições	A conta do usuário será deletada do sistema.

Fonte: Autor (2019)

Tabela 13 - Cadastrar Linha Ferroviária

Caso de Uso	Cadastrar Linha Ferroviária
Descrição	O administrador cadastra uma nova linha ferroviária no sistema.
Atores	Administrador
Pré-condições:	O administrador deve estar logado.
Sequência de Eventos (Principal)	1. O administrador clica na opção de Linhas; 2. O administrador clica em cadastrar uma nova linha; 3. O administrador preenche os campos; 4. O administrador clica em Salvar; 5. O administrador registra no banco a nova linha.
Pós-condições	O sistema terá registrado no banco uma nova linha.

Fonte: Autor (2019)

Tabela 14 - Editar Linha Ferroviária

Caso de Uso	Editar Linha Ferroviária
Descrição	O administrador edita as informações cadastradas anteriormente de uma linha.
Atores	Administrador
Pré-condições:	O administrador deve estar logado.
Sequência de Eventos (Principal)	1. O administrador , na tela de administrador, clica na opção “Linhas”; 2. O administrador seleciona a linha que deseja editar; 3. O administrador altera o campo de desejado; 4. O administrador clica em salvar;

	5. O administrador registra a alteração.
Pós-condições	O registro da linha será modificado.

Fonte: Autor (2019)

Tabela 15 - Apagar Linha Ferroviária

Caso de Uso	Apagar Linha Ferroviária
Descrição	O administrador deleta a linha ferroviária cadastrada no sistema.
Atores	Administrador
Pré-condições:	O administrador deve estar logado.
Sequência de Eventos (Principal)	1. O administrador, na tela de administrador, clica na opção “Linhas”; 2. O administrador seleciona a linha desejada; 3. O administrador clica em “Deletar”; 4. O administrador apaga o registro da linha; 5. O administrador é redirecionado para a tela de administrador.
Pós-condições	A linha será deletada do sistema.

Fonte: Autor (2019)

Tabela 16 - Cadastrar Estação de Passageiros

Caso de Uso	Cadastrar Estação de Passageiros
Descrição	O administrador cadastra uma nova estação de passageiros no sistema.
Atores	Administrador
Pré-condições:	O administrador deve estar logado.
Sequência de Eventos (Principal)	1. O administrador clica na opção de Estação de Passageiros; 2. O administrador clica em cadastrar uma nova estação de passageiros; 3. O administrador preenche os campos; 4. O administrador clica em Salvar; 5. O administrador registra no banco a nova estação de passageiros.
Pós-condições	O sistema terá registrado no banco uma nova estação de passageiros.

Fonte: Autor (2019)

Tabela 17 - Editar Estação de Passageiros

Caso de Uso	Editar Estação de Passageiros
Descrição	O administrador edita as informações cadastradas anteriormente de uma estação de passageiros.
Atores	Administrador
Pré-condições:	O administrador deve estar logado.
Sequência de Eventos (Principal)	1. O administrador , na tela de administrador, clica na opção “Estações de Passageiros”; 2. O administrador seleciona a estação que deseja editar; 3. O administrador altera o campo de desejado; 4. O administrador clica em salvar; 5. O administrador registra a alteração.*
Pós-condições	O registro da estação de passageiros será modificado.

Fonte: Autor (2019)

Tabela 18 - Apagar Estação de Passageiros

Caso de Uso	Apagar Estação de Passageiros
Descrição	O administrador deleta a estação de passageiros cadastrada no sistema.
Atores	Administrador
Pré-condições:	O administrador deve estar logado.
Sequência de Eventos (Principal)	1. O administrador, na tela de administrador, clica na opção “Estações de Passageiros”; 2. O administrador seleciona a estação de passageiros desejada; 3. O administrador clica em “Deletar”; 4. O administrador apaga o registro da estação de passageiros; 5. O administrador é redirecionado para a tela de administrador.
Pós-condições	A estação de passageiros será deletada do sistema.

Fonte: Autor (2019)

Tabela 19 - Cadastrar Trem

Caso de Uso	Cadastrar Trem
Descrição	O administrador cadastra um novo trem no sistema.
Atores	Administrador
Pré-condições:	O administrador deve estar logado.
Sequência de Eventos (Principal)	1. O administrador clica na opção de Trem; 2. O administrador clica em cadastrar um novo trem; 3. O administrador preenche os campos; 4. O administrador clica em Salvar; 5. O administrador registra no banco o novo trem.
Pós-condições	O sistema terá registrado no banco um novo trem.

Fonte: Autor (2019)

Tabela 20 - Editar Trem

Caso de Uso	Editar Trem
Descrição	O administrador edita as informações cadastradas anteriormente de um trem.
Atores	Administrador
Pré-condições:	O administrador deve estar logado.
Sequência de Eventos (Principal)	1. O administrador , na tela de administrador, clica na opção "Trens"; 2. O administrador seleciona a trem que deseja editar; 3. O administrador altera o campo de desejado; 4. O administrador clica em salvar; 5. O administrador registra a alteração.
Pós-condições	O registro do trem será modificado.

Fonte: Autor (2019)

Tabela 21 - Apagar Trem

Caso de Uso	Apagar Trem
Descrição	O administrador deleta o trem cadastrado no sistema.
Atores	Administrador

Pré-condições:	O administrador deve estar logado.
Sequência de Eventos (Principal)	1. O administrador, na tela de administrador, clica na opção “Trens”; 2. O administrador seleciona o trem desejado; 3. O administrador clica em “Deletar”; 4. O administrador apaga o registro do trem; 5. O administrador é redirecionado para a tela de administrador.
Pós-condições	O trem será deletado do sistema.

Fonte: Autor (2019)

Tabela 22 - Cadastrar Dispositivo Estação Móvel

Caso de Uso	Cadastrar Dispositivo Estação Móvel
Descrição	O administrador cadastra um novo dispositivo estação móvel.
Atores	Administrador
Pré-condições:	O administrador deve estar logado.
Sequência de Eventos (Principal)	1. O administrador clica na opção de Estações Móveis; 2. O administrador clica em cadastrar uma nova estação móvel; 3. O administrador preenche os campos; 4. O administrador clica em Salvar; 5. O administrador registra no banco a nova estação móvel.
Pós-condições	O sistema terá registrado no banco um novo dispositivo estação móvel.

Fonte: Autor (2019)

Tabela 23 - Editar Dispositivo Estação Móvel

Caso de Uso	Editar Dispositivo Estação Móvel
Descrição	O administrador edita as informações cadastradas anteriormente de uma estação móvel.
Atores	Administrador
Pré-condições:	O administrador deve estar logado.
Sequência de Eventos (Principal)	1. O administrador , na tela de administrador, clica na opção “Estações Móveis”; 2. O administrador seleciona a estação móvel que deseja editar; 3. O administrador altera o campo de desejado;

	4. O administrador clica em salvar; 5. O administrador registra a alteração.
Pós-condições	O registro da estação móvel será modificado.

Fonte: Autor (2019)

Tabela 24 - Apagar Dispositivo Estação Móvel

Caso de Uso	Apagar Dispositivo Estação Móvel
Descrição	O administrador deleta o dispositivo estação móvel cadastrado no sistema.
Atores	Administrador
Pré-condições:	O administrador deve estar logado.
Sequência de Eventos (Principal)	1. O administrador, na tela de administrador, clica na opção “Estações Móveis”; 2. O administrador seleciona a estação móvel desejada; 3. O administrador clica em “Deletar”; 4. O administrador apaga o registro da estação móvel; 5. O administrador é redirecionado para a tela de administrador.
Pós-condições	A estação móvel será deletada do sistema.

Fonte: Autor (2019)

Tabela 25 - Cadastrar Dispositivo Estação Base

Caso de Uso	Cadastrar Dispositivo Estação Base
Descrição	O administrador cadastra um novo dispositivo estação base.
Atores	Administrador
Pré-condições:	O administrador deve estar logado.
Sequência de Eventos (Principal)	1. O administrador clica na opção de Estações Base; 2. O administrador clica em cadastrar uma nova estação base; 3. O administrador preenche os campos; 4. O administrador clica em Salvar; 5. O administrador registra no banco a nova estação base.
Pós-condições	O sistema terá registrado no banco um novo dispositivo estação base.

Fonte: Autor (2019)

Tabela 26 - Editar Dispositivo Estação Base

Caso de Uso	Editar Dispositivo Estação Base
Descrição	O administrador edita as informações cadastradas anteriormente de uma estação base.
Atores	Administrador
Pré-condições:	O administrador deve estar logado.
Sequência de Eventos (Principal)	1. O administrador, na tela de administrador, clica na opção “Estações Base”; 2. O administrador seleciona a estação base que deseja editar; 3. O administrador altera o campo de desejado; 4. O administrador clica em salvar; 5. O administrador registra a alteração.
Pós-condições	O registro da estação base será modificado.

Fonte: Autor (2019)

Tabela 27 - Apagar Dispositivo Estação Base

Caso de Uso	Apagar Dispositivo Estação Base
Descrição	O administrador deleta o dispositivo estação base cadastrado no sistema.
Atores	Administrador
Pré-condições:	O administrador deve estar logado.
Sequência de Eventos (Principal)	1. O administrador, na tela de administrador, clica na opção “Estações Bases”; 2. O administrador seleciona a estação base desejada; 3. O administrador clica em “Deletar”; 4. O administrador apaga o registro da estação base; 5. O administrador é redirecionado para a tela de administrador.
Pós-condições	A estação base será deletada do sistema.

Fonte: Autor (2019)

Tabela 28 - Cadastrar Dispositivo Estação Roteadora

Caso de Uso	Cadastrar Dispositivo Estação Roteadora
Descrição	O administrador cadastra um novo dispositivo estação roteadora.

Atores	Administrador
Pré-condições:	O administrador deve estar logado.
Sequência de Eventos (Principal)	1. O administrador clica na opção de Estações Roteadoras; 2. O administrador clica em cadastrar uma nova estação roteadora; 3. O administrador preenche os campos; 4. O administrador clica em Salvar; 5. O administrador registra no banco a nova estação roteadora.
Pós-condições	O sistema terá registrado no banco um novo dispositivo estação roteadora.

Fonte: Autor (2019)

Tabela 29 - Editar Dispositivo Estação Roteadora

Caso de Uso	Editar Dispositivo Estação Roteadora
Descrição	O administrador edita as informações cadastradas anteriormente de uma estação roteadora.
Atores	Administrador
Pré-condições:	O administrador deve estar logado.
Sequência de Eventos (Principal)	1. O administrador , na tela de administrador, clica na opção “Estações Roteadoras”; 2. O administrador seleciona a estação roteadora que deseja editar; 3. O administrador altera o campo de desejado; 4. O administrador clica em salvar; 5. O administrador registra a alteração.
Pós-condições	O registro da estação roteadora será modificado.

Fonte: Autor (2019)

Tabela 30 - Apagar Dispositivo Estação Roteadora

Caso de Uso	Apagar Dispositivo Estação Roteadora
Descrição	O administrador deleta o dispositivo estação roteadora cadastrado no sistema.
Atores	Administrador
Pré-condições:	O administrador deve estar logado.

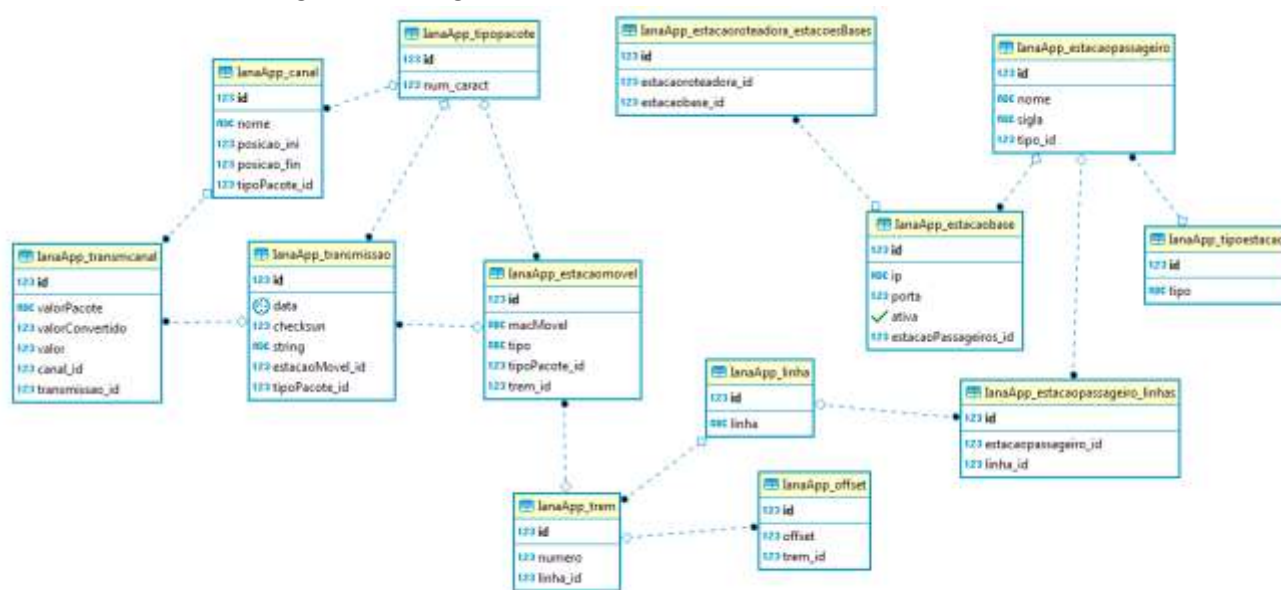
Sequência de Eventos (Principal)	1. O administrador, na tela de administrador, clica na opção “Estações Roteadoras”; 2. O administrador seleciona a estação roteadora desejada; 3. O administrador clica em “Deletar”; 4. O administrador apaga o registro da estação roteadora; 5. O administrador é redirecionado para a tela de administrador.
Pós-condições	A estação roteadora será deletada do sistema.

Fonte: Autor (2019)

3.3 Diagrama Entidade-Relacionamento

A figura abaixo representa o diagrama Entidade-Relacionamento do banco de dados do sistema.

Figura 3: Diagrama Entidade-Relacionamento



Fonte: Autor (2019)

• lanaApp_trem

A tabela Trem armazena a lista de todos os trens do sistema metroferroviário. Ela possui os seguintes campos: “ID”, “numero” do veículo e “linha_id” que informa em qual linha aquele trem circula.

• lanaApp_estacaopassageiro

A tabela EstacaoPassageiro armazena a lista de todas as estações de passageiros do sistema metroferroviário. Ela possui os seguintes campos: “ID”,

“nome”, “sigla” e “tipo_id” que informa que tipo de estação é, sendo as opções “Integradora” se for integrada a um terminal de ônibus ou “Não-Integradora”.

- **lanaApp_tipoestacao**

A tabela TipoEstacao armazena os tipos de estação de passageiros existentes. Ela possui os seguintes campos: “ID” e “tipo”.

- **lanaApp_linha**

A tabela Linha armazena a lista de todas as linhas ferroviárias. Ela possui os seguintes campos: “ID” e “linha”.

- **lanaApp_offset**

A tabela Offset armazena o valor da variável utilizada no cálculo da pressão das bolsas de ar. Este valor é gerado do peso do trem quando está “vazio”, mas é relativo de veículo para veículo. A tabela possui os seguintes campos: “ID”, “offset” e “trem_id”.

- **lanaApp_estacaomovel**

A tabela EstacaoMovel armazena a lista de todos os dispositivos EM instalados. Ela possui os seguintes campos: “ID”, “macMovel” que é o endereço Mac do dispositivo, “tipo” que informa se a EM transmite dados Digitais ou Analógicos, “tipoPacote_id” e “trem_id” que informa em qual trem ele está instalado.

- **lanaApp_tipopacote**

A tabela TipoPacote armazena os tipos de pacote que são transmitidos pelas EM. Ela possui os seguintes campos: “ID” e “num_caract”, este último campo sendo a informação que define a diferença de um tipo de pacote para outro.

- **lanaApp_estacaobase**

A tabela EstacaoBase armazena a lista de todos os dispositivos EB instalados. Ela possui os seguintes campos: “ID”, “ip” e “porta” para realizar a conexão socket, “ativa” e “estacaopassageiro_id” que informa em qual estação de passageiros ele está instalado.

- **lanaApp_canal**

A tabela Canal armazena os tipos de variáveis do trem. Ela possui os seguintes campos: “ID”, “nome”, “posicao_ini” e “posicao_fin” que informam a posição do dado dentro do pacote hexadecimal e “tipoPacote_id”.

- **lanaApp_transmissao**

A tabela Transmissao armazena os pacotes na sua forma hexadecimal. Ela possui os seguintes campos: “ID”, “data” que foi recebido o pacote, “checksum”, “string” que é o próprio pacote hexadecimal, “estacaoMovel_id” e “tipoPacote_id”.

- **lanaApp_transmcanal**

A tabela TransmCanal armazena os valores das variáveis captadas. Ela possui os seguintes campos: “ID”, “valorPacote” e “valorConvertido” que são respectivamente o valor em hexadecimal e decimal da variável no pacote, “valor” que é o valor real da variável após passar pelos cálculos matemáticos, “canal_id” e “transmissao_id”.

3.4 Tecnologias

3.4.1 Java:

É uma linguagem de programação orientada a objeto que pertence a empresa multinacional de tecnologia e informática, *Oracle Corporation*. Essa linguagem é compilada em *bytecode* e interpretada pela sua máquina virtual a *Java Virtual Machine*. Como ela é orientada a objeto, sua base consiste na composição e interação entre diversas unidades que são chamadas de “objetos”. O Java é a base

para praticamente todos os tipos de aplicações em rede, além de ser executado em 97% dos desktops corporativos (JAVA, 2016). Neste projeto é utilizada para a implementação de dois microsserviços da arquitetura do sistema, juntamente com as tecnologias de *multithreading* e conexões *sockets* proporcionadas pela linguagem.

3.4.2 Django:

Um framework Web Python de alto nível com o propósito de incentivar um rápido desenvolvimento junto com um design limpo e prático. A escolha dessa ferramenta para a implementação do módulo Web da arquitetura veio pelas suas características que se adequam às necessidades do projeto. Algumas dessas características são: auxiliar os programadores no desenvolvimento rápido de projetos; possui ferramentas que auxiliam nas tarefas comuns de um desenvolvimento Web, como autenticação do usuário e administração de conteúdo; tem a capacidade de ser escalável e flexível conforme for preciso fazer grandes modificações ou para atender maiores demandas (DJANGO, 2019).

3.4.3 RabbitMQ:

Consiste em um servidor de mensagens *open source* que através do protocolo AMQP possibilita o tráfego de mensagens de forma rápida e confiável (RABBITMQ, 2019). A utilização de uma fila na arquitetura do sistema fará com que os dados contínuos das Estações Bases não sejam perdidos, sendo cada pacote de dado armazenado em uma fila respectiva ao trem que o enviou.

3.4.4 WebComponents:

É um conjunto de APIs com o objetivo de tornar reutilizáveis os elementos utilizados na programação Web (WEBCOMPONENTS, 2019). Elementos personalizados oferecem a possibilidade de criá-los baseados na sua finalidade. Com elementos bem estruturados as telas do projeto que conterão informações dinâmicas serão padronizadas, agilizando o seu processo de desenvolvimento

3.4.5 WebSocket:

Tecnologia que permite a comunicação bidirecional por canais *full-duplex* sobre um único soquete *TCP/IP* (MOZILLA, 2019). Ou seja, mantém uma conexão persistente entre o cliente e o servidor, onde ambas as partes podem começar a enviar dados a qualquer momento. Essa tecnologia serve neste projeto na atualização, em tempo real de execução, dos dados exibidos na interface do usuário.

3.4.6 PostgreSQL:

É o sistema de banco de dados objeto-relacional escolhido para este projeto. É *open source*, estende a linguagem *SQL* e pode ser utilizado com várias linguagens de programação, dentre elas a utilizada neste projeto: Python. Essa ferramenta vem com muitos recursos que auxiliam no desenvolvimento, integração e gerenciamento do banco (POSTGRESQL, 2019), o que é uma exigência para um sistema que estará constantemente escrevendo e consultando informações.

3.4.7 Visual Studio Code:

É um editor de código-fonte da Microsoft para as plataformas Windows, Linux e macOS. Ele possui vários recursos, como: suporte para trabalhar com diversas linguagens; suporte para depuração; controle Git incluso; software livre e de código aberto; ferramenta para refatoração de código. Por ser compatível para trabalhar com a linguagem Python, ele foi escolhido para o desenvolvimento do serviço Aplicação Web da arquitetura.

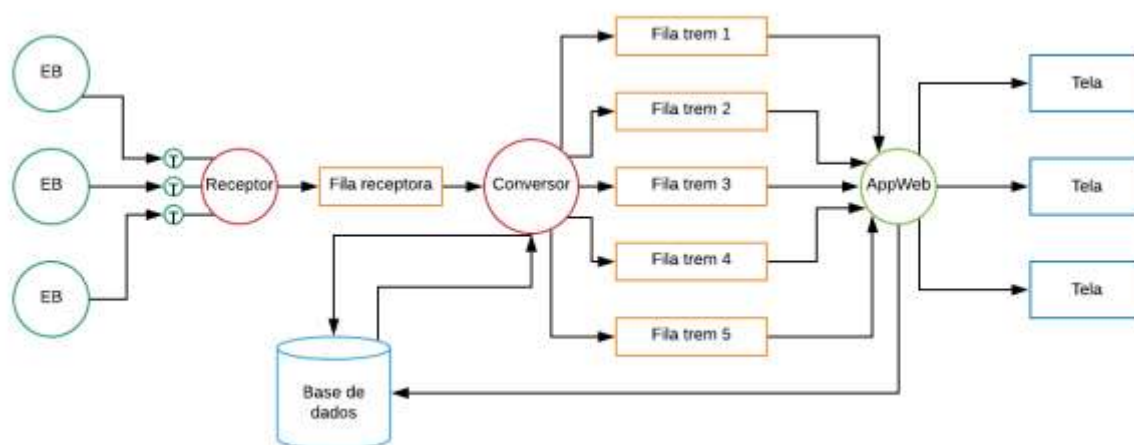
3.4.8 Eclipse:

O Eclipse consiste em uma *IDE*, um ambiente de desenvolvimento integrado, voltado para desenvolvimento em Java, porém possui também o suporte para outras linguagens e *plugins*. É a *IDE* escolhida desenvolver neste projeto os módulos da arquitetura que são em Java.

3.5 Arquitetura

Depois de realizadas as análises dos requisitos do sistema, a definição da arquitetura do software foi elaborada de acordo com a figura 4.

Figura 4: Fluxograma Representante da Arquitetura do Sistema.



Fonte: Autor (2019)

O sistema é organizado em três micro serviços que trabalham em conjunto utilizando as filas para o encaminhamento de dados entre eles. A seguir, uma explicação de cada um e suas funcionalidades:

3.5.1 Receptor

O fluxo inicia no estabelecimento da conexão do serviço “Receptor” com as EBs. Esse módulo utiliza a tecnologia de *multithreading* para abrir simultaneamente um canal de comunicação via *socket* para cada dispositivo embarcado que esteja ativo. Esse canal possibilita a recepção dos dados.

Os pacotes de dados recebidos estão em formato de bits. Então, além de estabelecer a conexão com as EBs, o Receptor também executa um processo de conversão dos pacotes, transformando-os de bits para um pacote hexadecimal. Este pacote é encaminhado para uma fila RabbitMQ representada na imagem acima como “Fila receptora”.

3.5.2 Conversor

O próximo serviço denominado “Conversor” na arquitetura, consome os pacotes hexadecimais da fila receptora. Dentro deste pacote estão presentes o endereço da EM que o enviou e as variáveis do trem. Cada EM envia atualmente apenas os valores de quatro variáveis de uma determinada natureza. Esta natureza pode ser analógica ou digital. As variáveis analógicas são as de valores contínuos e são elas: velocidade, pressão das bolsas de ar, corrente do motor de tração e o freio. As variáveis digitais são as de valores discretos, neste caso variando apenas entre 0 ou 1. São elas: a condição de cabine, se o trem está indo para frente ou para trás, e as portas, se estão abertas ou fechadas. Sabendo disso, este módulo realiza um processo de conversão das variáveis de acordo com o tipo de dado transmitido pela EM.

Para cada tipo de pacote (analógico ou digital) há nele quatro canais. Esses canais são identificados pela sua posição dentro do pacote. E, dentro desse canal está o valor da variável do trem. Logo, o módulo Conversor localiza o canal correspondente da variável dentro do pacote e converte o valor contido nela de hexadecimal para decimal. Além dessa conversão, o dado passa por mais um processo matemático que resulta no valor real da variável.

No final desse procedimento os dados são armazenados no banco de dados e o resultado gerado por esse módulo será convertido para a notação *json* com os nomes das variáveis e seus valores. Então, o Conversor, com o endereço da EM, irá identificar o trem de origem e encaminhar o *json* para a fila correspondente ao número do veículo. Para essa identificação, o Conversor utilizará as informações de endereço das EMs registradas previamente no banco de dados.

3.5.3 Aplicação Web

A aplicação web, ou “*AppWeb*” representada na arquitetura é um serviço *web* local que consome os pacotes *jsons* das filas dos trens e exibe na tela de interface do usuário. Podemos dividir sua composição em duas partes: o seu *front-end* e o seu *back-end*.

O *front-end* da aplicação é a interface que o usuário tem contato. Nela é utilizado um *WebComponent* para definir o padrão do elemento “trem” e como deve ser exibido na tela para o monitoramento. Cada elemento trem exibe o número do veículo e outros oito campos para a exibição das variáveis, como visto no apêndice

B figura 6. Para cada elemento é estabelecido uma conexão *WebSocket*. Essa conexão entre o *back-end* e o *front-end* da aplicação serve para que quando um pacote é recebido no *back* os dados são encaminhados diretamente para o *front* atualizando a todo instante as informações dos trens exibidas nos elementos de *WebComponent*.

4 CONCLUSÃO

O sistema RailBee[®] monitora os dados dos trens em tempo real, e por isso, tem grande utilidade quando se tratando da melhoria na qualidade do transporte metroferroviário. É possível, com esse sistema, detectar o estado das variáveis do trem, fazendo com que seja mais fácil e ágil a tomada de decisão para cada situação, sendo ela emergencial ou não. Com essa identificação prévia é permitido que medidas sejam tomadas de forma rápida e segura.

Apesar do sistema ainda não estar em produção no Centro de Controle Operacional da CBTU/Recife esse projeto foi apresentado em vários eventos, fez parte de projetos institucionais do IFPE e já está registrado no INPC (apêndice A).

O software da Estação Central é essencial para a complementação deste sistema, pois é a etapa final do percurso das variáveis dos trens. É nela onde é feita a transformação dos dados em informações e as armazena no seu banco de dados local. A visualização dessas informações em tempo real e os registros do banco de dados fornecidos pelo projeto oferecem a realização de relatórios e análises feitos pelos controladores de tráfego, técnicos e engenheiros. Logo, com este mecanismo é possível auxiliar e melhorar a qualidade dos serviços prestados pela empresa de transporte público ferroviário. Desta forma, o sistema procura estar em constante aprimoramento e evolução fazendo uso de novas tecnologias para atender a novas funcionalidades de acordo com as necessidades dos controladores de tráfego, assim como oferecer o máximo de conforto aos usuários dos trens.

REFERÊNCIAS

ARAÚJO, R. C. C. **Sistema Telemétrico Dinâmico Sem Fio Aplicado Aos Veículos Rodoferroviários em Malhas Metroferroviárias**. 2009. Tese (Doutorado em Engenharia Mecânica) - Programa de Pós-Graduação em Engenharia Mecânica, Paraíba, 2009. Disponível em: <http://tede.biblioteca.ufpb.br/bitstream/tede/5364/1/parte1.pdf>. Acesso em: 5 mar. 2019.

CARMO, T. R. **Uso do padrão AMQP Para Transporte de Mensagens Entre Atores Remotos**. 2012. Tese (Mestrado em Ciências), Instituto de Matemática e Estatística da Universidade de São Paulo, São Paulo, 2012. Disponível em: https://www.ime.usp.br/~reverbel/students/master_theses/thadeu_de_russo_e_carmo.pdf. Acesso em: 14 abr. 2019.

CBTU. **CBTU: Companhia Brasileira de Trens Urbanos**. Portal CBTU – Recife, 2018. Disponível em: <https://cbtu.gov.br/index.php/pt/sistemas-cbtu/recife>. Acesso em: 11 abr. 2019.

DJANGO. **Djangoproject: The web framework for perfectionists with deadlines**. Página inicial, 2019. Disponível em: <https://www.djangoproject.com>. Acesso em: 10 abr. 2019.

JAVA. **Java: Comece a trabalhar com o Java hoje mesmo**. Obtenha Informações sobre a Tecnologia Java, 2016. Disponível em: https://www.java.com/pt_BR/about/. Acesso em: 11 abr. 2019.

LEWIS, J.; FOWLER, M. **Microservices**. Estados Unidos da América, 2014. Disponível em: <https://martinfowler.com/articles/microservices.html>. Acesso em: 24 mai. 2019.

MOZILLA. **WebSockets | MDN**, 2019. Disponível em: <https://developer.mozilla.org/pt-BR/docs/WebSockets>. Acesso em: 11 abr. 2019.

NEWMAN, S. **Building Microservices: Designing Fine-Grained Systems**. 2015. O'Reilly Media, Inc , Estados Unidos da América, 2015.

PEREIRA, M.; BACK, G.; JÚNIOR, N. **Arquitetura Baseada em Microserviço**. 2018. Artigo - Católica de Santa Catarina, Santa Catarina, 2018. Disponível em: https://www.researchgate.net/profile/Murillo_De_Miranda_Pereira/publication/32952

1832_Arquitetura_baseada_em_microservico/links/5c0d2ac8299bf139c74d4639/Arquitetura-baseada-em-microservico.pdf. Acesso em: 24 mai. 2019.

POSTGRESQL. **PostgreSQL: The World's Most Advanced Open Source Relational Database**. About PostgreSQL, 2019. Disponível em: <https://www.postgresql.org/about/>. Acesso em: 10 abr. 2019.

RABBITMQ. **Messaging that just works — RabbitMQ**. Página inicial, 2019. Disponível em: <https://www.rabbitmq.com>. Acesso em: 11 abr. 2019.

SANTOS, J. L. A. **Sistema Telemétrico Para Monitoramento De Trens Através De Redes De Sensores Sem Fio E Processamento Em Sistema Embarcado**. 2010. Tese (Mestrado em Informática), Programa de Pós-Graduação em Informática - Universidade Federal da Paraíba, Paraíba, 2010.

SOMMERVILLE, I. **Engenharia de Software**. 3. ed. Pearson Addison Wesley, 2013.

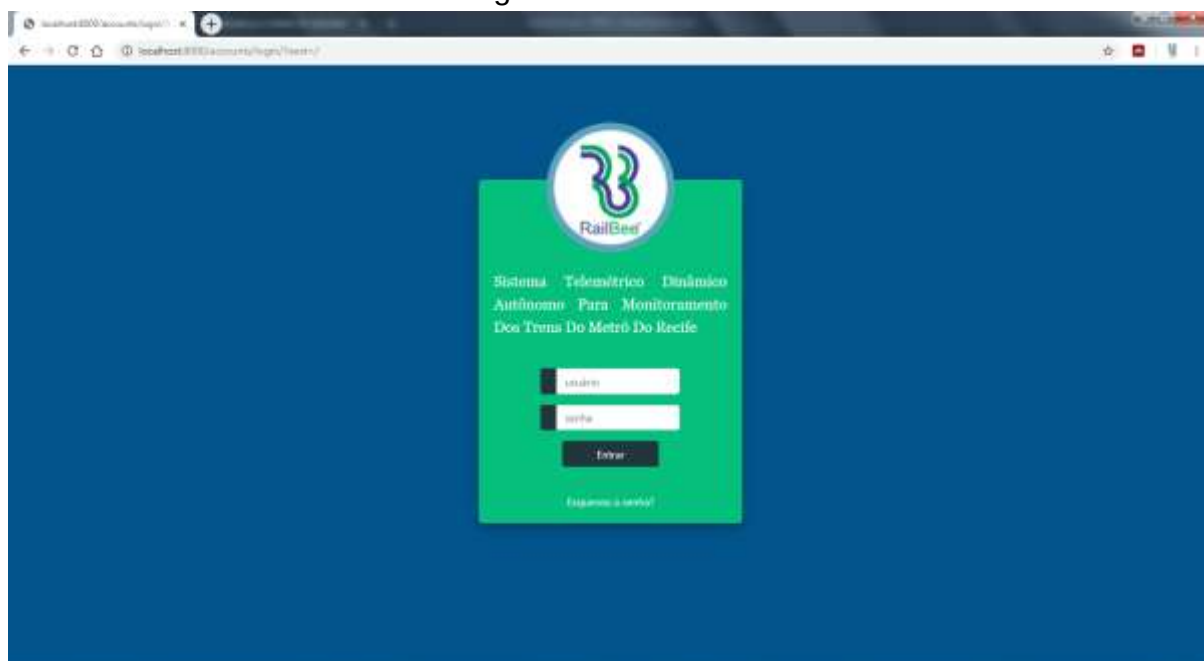
WEBCOMPONENTS. **WebComponents: There's an element for that**. Introduction – webcomponents, 2019. Disponível em: <https://www.webcomponents.org/introduction>. Acesso em: 11 abr. 2019.

APÊNDICE A: RESULTADOS OBTIDOS

- PROPRIEDADE INDUSTRIAL IANA SANTOS, H. C. T.; ARAÚJO, R. C. C.; SIQUEIRA, I. C.; PEREIRA, S. X. . IANA. 2020 Patente: Programa de Computador. Número do registro: 512020002413-0, data de registro: 17/07/2020, título: "IANA" ,Instituição de registro: INPI - Instituto Nacional da Propriedade Industrial.
- BOLSA PIBIC - GRADUAÇÃO - IFPE Isabella Chiara de Siqueira. DESENVOLVIMENTO DE SOFTWARE DE GERENCIAMENTO, ARMAZENAMENTO, PROCESSAMENTO E VISUALIZAÇÃO DE DADOS EM TEMPO REAL PARA APLICAÇÃO NO SISTEMA TELEMÉTRICO RailBee. PIBIC 2018/2019. Iniciação Científica. (Graduando em Análise e Desenvolvimento de Sistemas) - Instituto Federal de Pernambuco. Orientador: Rômulo César Carvalho de Araújo.
- Publicação de Resumo Expandido Anais do CONIC 2019 Desenvolvimento de Software de Gerenciamento, Armazenamento, Processamento e Visualização de Dados em Tempo Real para Aplicação no Sistema Telemétrico RailBee, Isabella Chiara de Siqueira, Rômulo César Carvalho de Araújo; Estudante do Curso de Análise e Desenvolvimento de Sistemas - IFPE, campus Recife; email: ics@a.recife.ifpe.edu.br Docente/Pesquisador do Departamento Acadêmico de Sistemas, Processos e Controles Eletroeletrônicos – IFPE, campus Recife; email: romuloaraujo@recife.ifpe.edu.br
- Prêmio 1º Lugar na Mostra Tecnológica do CONNEPI 2018

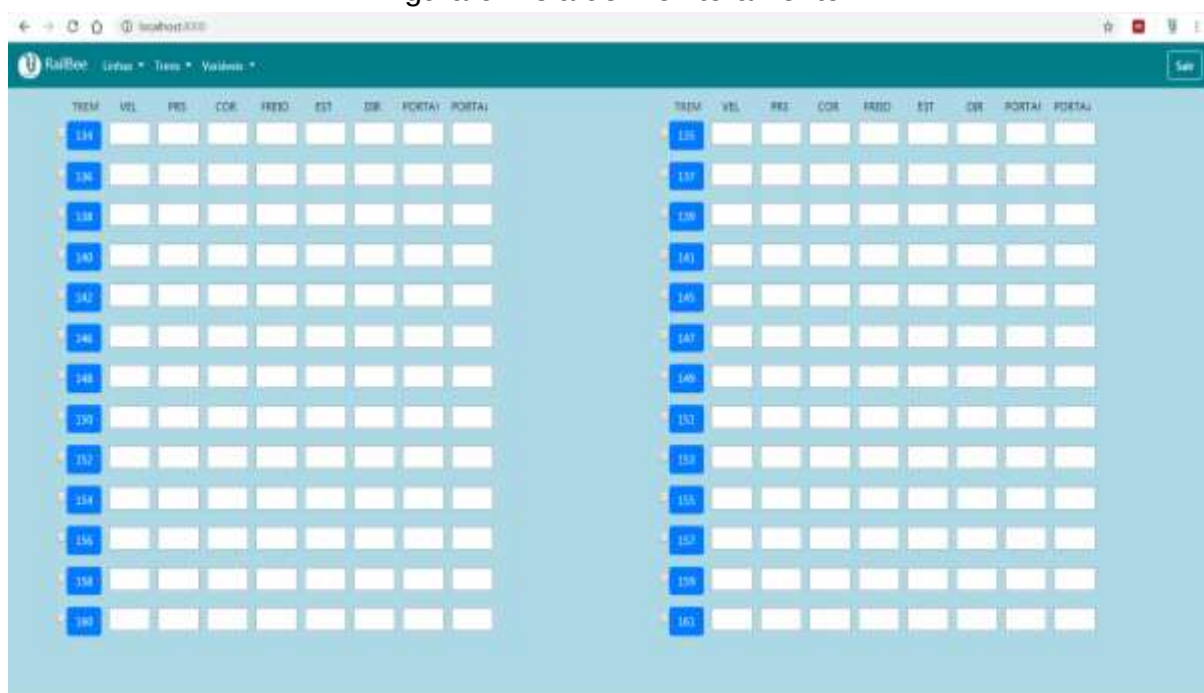
APÊNDICE B: TELAS SOFTWARE DA ESTAÇÃO CENTRAL

Figura 5: Tela inicial



Fonte: Autor (2019)

Figura 6: Tela de monitoramento



Fonte: Autor (2019)

Figura 7: Menu das linhas ferroviárias

The screenshot shows the RailBee web application interface. The top navigation bar includes the RailBee logo, a 'Sair' button, and dropdown menus for 'Linhas', 'Trens', and 'Variáveis'. The 'Linhas' dropdown menu is open, showing options: 'Centro', 'Sul', 'Diesel', and 'Todos'. The main content area displays a table with columns: TREN, COR, FREIO, EST, DIR, PORTA1, and PORTA2. The rows correspond to train numbers 134, 136, 138, 140, 142, 146, 148, and 150. Each row has a small square icon to the left of the train number and a series of input fields for the other columns.

TREN	COR	FREIO	EST	DIR	PORTA1	PORTA2
134						
136						
138						
140						
142						
146						
148						
150						

Fonte: Autor (2019)

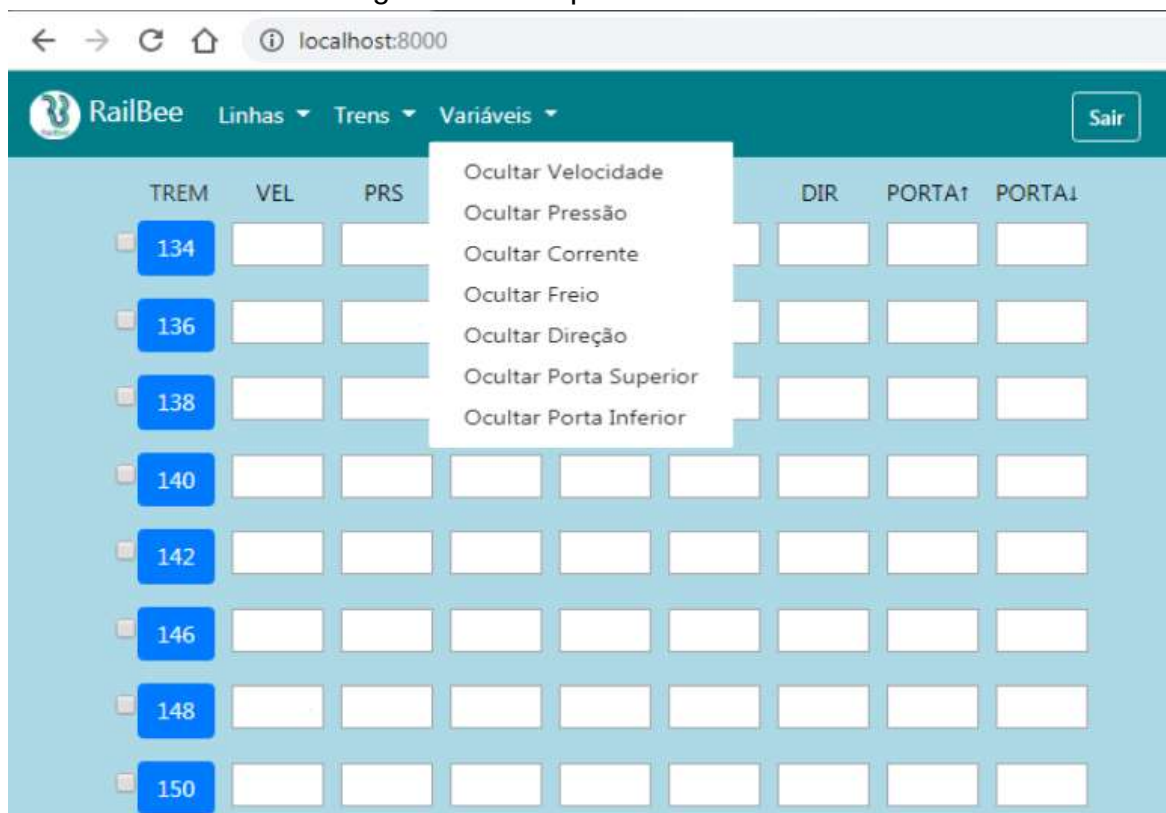
Figura 8: Menu para filtrar os trens

The screenshot shows the RailBee web application interface. The top navigation bar includes the RailBee logo, a 'Sair' button, and dropdown menus for 'Linhas', 'Trens', and 'Variáveis'. The 'Trens' dropdown menu is open, showing options: 'Filtrar Trens Seleccionados' and 'Exibir Trens Ocultos'. The main content area displays a table with columns: TREN, VEL, EST, DIR, PORTA1, and PORTA2. The rows correspond to train numbers 134, 136, 138, 140, 142, 146, 148, and 150. Each row has a small square icon to the left of the train number and a series of input fields for the other columns.

TREN	VEL	EST	DIR	PORTA1	PORTA2
134					
136					
138					
140					
142					
146					
148					
150					

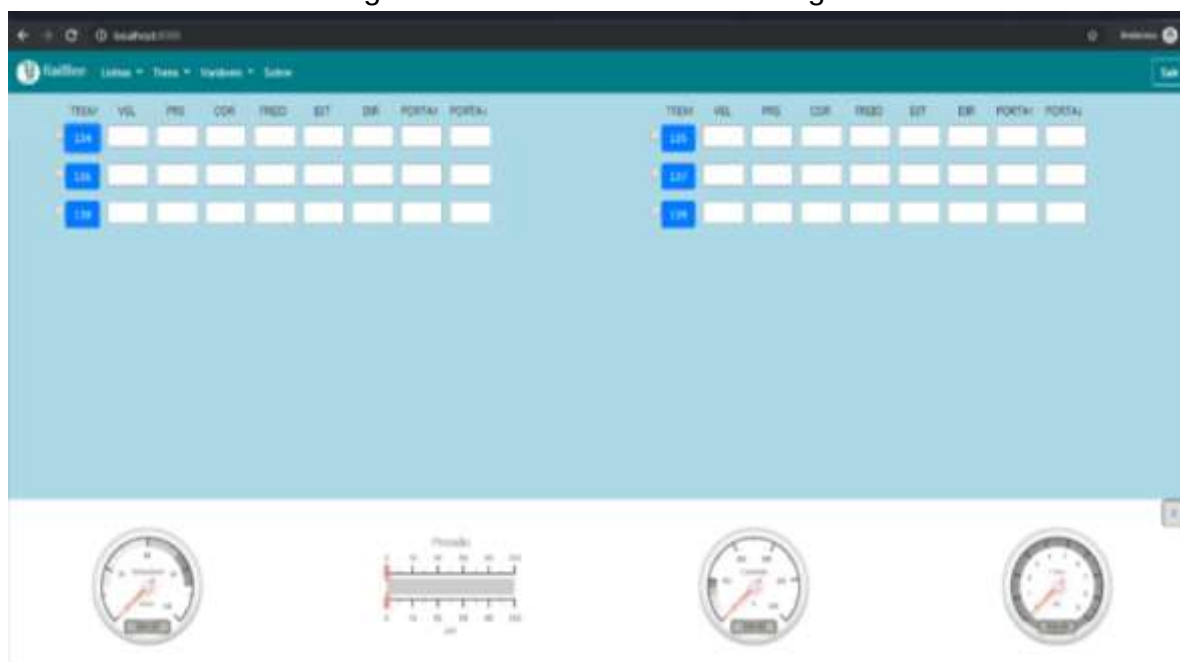
Fonte: Autor (2019)

Figura 9: Menu para ocultar variáveis



Fonte: Autor (2019)

Figura 10: Painel das variáveis digitais



Fonte: Autor (2019)