

# RastreiAll: Desenvolvimento de um Sistema de Rastreamento de Máquinas via GPS

Yuri H. G. Kachio<sup>1</sup>, Fernando Sambinelli<sup>2</sup>

Instituto Federal de São Paulo – Campus Hortolândia – SP – Brazil

kachiohideki@gmail.com, sambinelli@ifsp.edu.br

**Abstract.** *The various advances in information as well as other equipment have evolved disproportionately faster than the methods of protecting the devices that contain them, as well as ways to recover any type of equipment. In view of the need to increase the chances of recovery and safety of these devices, this work concluded the development a software that will allow the management of tracking devices (GPS), as well as a physical prototype of a tracker. This article details the process of developing the WEB application, as well as the development of the physical prototype.*

**Resumo.** *Os avanços tecnológicos e a proliferação de dispositivos eletrônicos superaram o desenvolvimento dos métodos de proteção e recuperação desses equipamentos. Para abordar essa lacuna e melhorar a segurança e a chance de recuperação desses dispositivos, este trabalho apresenta um software para gerenciamento de dispositivos localizadores (GPS), juntamente com um protótipo físico de um rastreador. Este artigo descreve o processo de desenvolvimento da aplicação web e do protótipo físico, com o objetivo de aprimorar a gestão e a segurança desses dispositivos.*

## 1. Introdução

À medida que a tecnologia avança nas aplicações para facilidade e agilidade dos processos diários (comunicação, trabalho, transações financeiras, entre outros), ela também cresce no âmbito de segurança. Com a digitalização das informações, passou-se a desenvolver mais aplicativos voltados para a segurança e a integridade da informação (PEREIRA, 2013).

Em contrapartida, Pereira (*Ibidem*) postula que, embora haja muito investimento na integridade das informações, os métodos de proteção dos equipamentos físicos não acompanharam esse avanço. Havendo, desse modo, grande preocupação em proteger a integridade e a privacidade dos dados que eles contêm, mas com baixa discussão com relação às máquinas e aos equipamentos (*hardware*).

Entre os avanços da última década, uma das tecnologias que se tornou cada vez mais presente dentro do cotidiano das empresas e pessoas foi o sistema de posicionamento global, mais conhecido pela sigla GPS (em inglês, *Global Positioning System*). Como ressalta Kobs (2021), os avanços dessa tecnologia ao longo das últimas décadas possibilitaram o surgimento de sistemas de rastreamento e monitoramento de diferentes tipos de pertences, estendendo seu uso tanto para fins empresariais ou fabris, quanto para uso pessoal.

Com a necessidade crescente de formas de monitoramento dos equipamentos, esse trabalho buscou desenvolver um protótipo de um sistema de gerenciamento de equipamentos que permita aumentar a probabilidade de recuperação em caso de furto, roubo ou perda dos bens presentes nas empresas e na indústria em geral utilizando-se da tecnologia de GPS.

Para esquematização, na Seção 2, são apresentados os principais conceitos utilizados durante o desenvolvimento do trabalho. Na Seção 3, há um conjunto de trabalhos correlatos ao tema estudado, com o intuito de possibilitar uma análise para desenvolvimento desse projeto. A Seção 4 descreve a metodologia utilizada no desenvolvimento do artigo. Na Seção 5, apresenta-se como se deu o desenvolvimento do protótipo. A Seção 6 contém as considerações finais desse trabalho, bem como possíveis melhorias. Por fim, a Seção 7 possui as referências bibliográficas utilizadas durante a escrita e o desenvolvimento deste artigo.

## **2. Referencial Teórico**

O objetivo desta seção é apresentar o referencial teórico utilizado na produção desse trabalho, resumindo os principais conceitos aplicados.

### **2.1. Indústria 4.0**

A Indústria 4.0 prevê a integração entre humanos e máquinas, mesmo que em posições geográficas distantes, formando grandes redes e fornecendo produtos e serviços de forma autônoma (SILVA et al, 2015).

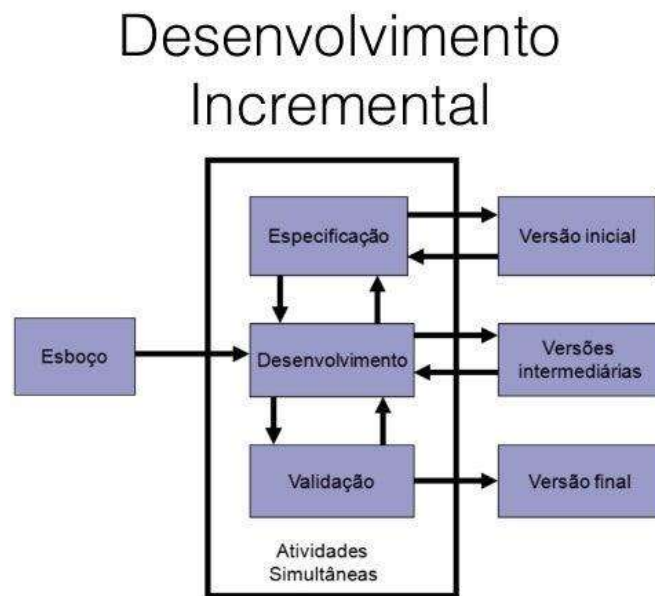
De acordo com o Portal da Indústria, a Indústria 4.0 representa a automação industrial e a integração de diferentes tecnologias, como Inteligência Artificial, Robótica, Internet das Coisas e Computação em Nuvem, com o intuito de promover a digitalização das atividades industriais, melhorando os processos e aumentando a produtividade.

Em outras palavras, como discorrem Santos et.al (2018), a Indústria 4.0 representa uma evolução natural dos sistemas industriais anteriores, desde a mecanização do trabalho, ocorrida no século XVIII, até a automação da produção, nos dias atuais. Nos últimos anos, a aplicação de automação e sistemas de informação melhoraram significativamente a produtividade nas fábricas. Dessa forma, compreendem objetos “inteligentes” (máquinas, produtos ou dispositivos) que trocam informações de forma autônoma, funcionando em colaboração com o mundo físico ao seu redor.

### **2.2. Modelo de Desenvolvimento Incremental**

No Desenvolvimento Incremental (Figura 1), os clientes, inicialmente, identificam, em um esboço, os requisitos do sistema e selecionam quais possuem maior relevância e aqueles com menor relevância. Em seguida, é definida uma série de iterações de entrega, na qual, em cada uma, é fornecido um subconjunto de funcionalidades executáveis, dependendo de suas prioridades (FAGUNDES et al, 2008).

Ademais, após a identificação dos incrementos, as funcionalidades a serem entregues na primeira iteração são detalhadas e desenvolvidas (SOMMERVILLE, 2003 *apud Ibidem*). Paralelamente a esse desenvolvimento, outras funcionalidades podem ser analisadas para fazerem parte dos outros incrementos. Uma vez que cada incremento é concluído e entregue, os clientes podem colocá-lo em operação. Assim, permitir que os clientes experimentem o sistema gradualmente facilita o esclarecimento das funcionalidades para os incrementos subsequentes e, à medida em que novos incrementos são concluídos, eles são integrados às iterações existentes, de modo que o sistema melhora a cada novo incremento entregue.



**Figura 1. Desenvolvimento Incremental**

### 2.3. Versões

De acordo com Sommerville (2003, *apud Ibidem*), ao utilizar o Modelo de Desenvolvimento Incremental, o projeto é desenvolvido através de versões. Na primeira versão, o *software* contém apenas um “núcleo”, somente com o que é essencial para seu funcionamento e, posteriormente, a cada nova versão, são adicionadas outras funcionalidades ou modificações, de acordo com o que o cliente solicitar. Tanto o que compõe o núcleo quanto o que conterà nas versões posteriores são definidos através do grau de importância referido pelo usuário, até que o sistema esteja finalizado.

### 2.4. Sistema de Informação para Gerenciamento

De acordo com Sommerville (2003, *apud ibidem*), ao utilizar o Modelo de Desenvolvimento Incremental, o projeto é desenvolvido através de versões. Na primeira versão, o *software* contém apenas um “núcleo”, somente com o que é essencial para seu

funcionamento e, posteriormente, a cada nova versão, são adicionadas outras funcionalidades ou modificações, de acordo com o que o cliente solicitar. Tanto o que compõe o núcleo quanto o que conterà nas versões posteriores são definidos através do grau de importância referido pelo usuário, até que o sistema esteja finalizado.

## **2.5. Aplicação Web**

A utilização da infraestrutura da Web introduz características específicas nas aplicações, tornando necessário um projeto que leve em consideração suas particularidades. Nesse contexto, o projeto de uma Aplicação Web deve abordar aspectos relacionados às dimensões estrutural, navegacional e de apresentação, conforme discutido por Fraternali e Paolini, 1998, conforme citado por Conte e Ribeiro, 2005.

- **Estrutural (Conceitual):** Esta dimensão diz respeito à organização das informações que a aplicação Web irá manipular e aos relacionamentos entre essas informações. É fundamental estabelecer como os dados serão estruturados e organizados no sistema para garantir uma funcionalidade eficaz e coerente.
- **Navegacional:** A dimensão navegacional envolve o modo como os usuários irão acessar as informações por meio da aplicação. Isso abrange a criação de interfaces de usuário intuitivas e a definição de fluxos de navegação que permitam aos usuários encontrar e interagir com os dados de forma eficiente.
- **Apresentação:** A dimensão de apresentação trata de como as informações e as opções de acesso serão apresentadas aos usuários da aplicação Web. Isso inclui o design da interface do usuário, a escolha de elementos visuais, esquemas de cores e a disposição geral dos elementos na página, visando proporcionar uma experiência de usuário atraente e eficaz.

Portanto, ao projetar uma aplicação Web, é essencial abordar essas três dimensões para garantir que a aplicação seja capaz de oferecer uma experiência eficiente e satisfatória aos seus usuários (Fraternali & Paolini, 1998, citados por Conte & Ribeiro, 2005).

## **2.6. Banco de Dados**

Durante o desenvolvimento, será necessário o armazenamento dos dados que serão utilizados no sistema. Para isso, na área de tecnologia da informação, existem os bancos de dados, que são um conjunto de informações e/ou dados que são armazenados de forma organizada e estruturada em um sistema de gerenciamento de banco de dados (SGBD) e que se relacionam entre si com o propósito de criar algum sentido (ORACLE, 2021 apud FARIA, 2022).

## **2.7. Sistema Embarcado**

Um sistema embarcado é qualquer dispositivo que inclui um computador programável, que é utilizado para resolver ou amenizar problemas de cunho específico. Um sistema embarcado opera, geralmente, com poucos recursos e executa algoritmos, buscando resolver problemas e obter respostas em tempo real (SILVA, 2021). Alguns exemplos de

sistemas embarcados utilizados no dia a dia são encontrados dentro de: micro-ondas, geladeiras, máquinas de lavar, dentre outros.

## **2.8. NodeMCU**

A plataforma utilizada para o desenvolvimento do projeto é o NodeMCU. Composta por um módulo ESP8266, é um microprocessador de 32 bits com suporte para conexão Wi-Fi; além de uma das portas de alimentação e programação, dez entradas digitais e uma analógica (MOLLY, 2016 apud SILVEIRA, 2016).

O uso do ESP8266 (NodeMCU) tem se tornado comum por dois fatores: o baixo custo e os recursos suficientes para diversas aplicações que envolvam a Internet das Coisas (SINGE et. al., 2014 apud SILVEIRA, 2016).

## **2.9. GPS**

O GPS é um sistema de posicionamento global que possui a característica de trabalhar em tempo real. Ele foi desenvolvido pelo departamento de defesa dos Estados Unidos nos últimos anos do século passado. Atualmente, a geração de coordenadas geográficas pode ser considerada uma tarefa simples; porém, ao longo do desenvolvimento humano, sempre se procurou uma maneira de localização terrestre precisa para substituir as maneiras de orientação pouco precisas proporcionadas pela observação de astros (SILVA, 2021).

Além do rastreamento GPS, outra funcionalidade importante é a *geofencing*. Uma *geofence* é um perímetro, delineado virtualmente para uma área geográfica real. Pode-se, com isso, determinar se o objeto saiu ou adentrou a área demarcada (KOBS, 2021).

## **2.9. Modelagem do Banco de Dados**

A modelagem de dados é o principal componente de informações conceituais do banco de dados. Dentre as técnicas, a entidade-relacionamento (ER) produz um modelo inteligível para o desenvolvedor do banco, assim como pelo usuário final. Um diagrama entidade relacionamento (DER) é um tipo de fluxograma que ilustra como "entidades", pessoas, objetos ou conceitos se relacionam dentro de um sistema. Esse diagrama é comumente utilizado para projetar ou depurar bancos de dados relacionais em diferentes áreas (Frank et al, 2021).

## **3. Trabalhos Correlatos**

Nesta seção, são expostos os trabalhos relacionados, bem como as principais características quando comparados entre si e de que forma serviram como base para o desenvolvimento do projeto. Para seleção dos trabalhos correlatos utilizados, foi utilizado Google Acadêmico, nele, foram utilizadas algumas palavras-chave, por exemplo rastreadores, sistemas de rastreadores, etc.

### **3.1. “Xereta”- Sistema de Rastreamento de Computadores**

“Xereta” é um sistema de monitoramento de locomoção de computadores que emite notificações por e-mail sempre que o dispositivo cadastrado estiver em uma coordenada em que não esteve anteriormente. Ele permite desativar as notificações, para que seja possível registrar novas coordenadas sem receber notificações e reativá-las uma vez que tenham sido automaticamente cadastradas pelo sistema.

Na interface de exibição dos computadores, é exibido o tipo do computador cadastrado, a latitude, a longitude e o horário da última leitura. Uma questão interessante é que é possível filtrar a exibição por tipo de computador, habilitando ou desabilitando os tipos através dos filtros.

### **3.2. *Pocket Guard***

O *Pocket Guard* é um sistema de monitoramento de proximidade, no qual, no aplicativo mobile, são cadastrados dispositivos rastreadores e a distância máxima do celular que a esses dispositivos podem estar antes de enviarem uma notificação para o celular. Além do alerta, é possível ainda acessar os dados de localização do rastreador cadastrado, desde que esse dispositivo esteja conectado ao celular via bluetooth.

Na interface de cadastro dos rastreadores e a exibição de suas informações o usuário pode adicionar ou selecionar um rastreador. Ao selecionar o dispositivo, é possível ver a distância do rastreador em relação ao celular, ativar ou desativar o alarme e remover o dispositivo.

### **3.3. *Pocket Guard***

O sistema de rastreamento veicular desenvolvido tem como principal característica um dispositivo que atua como rastreador de carro e um aplicativo mobile, no qual o usuário coloca o local em que está e para onde gostaria de ir. Dessa forma, o aplicativo traça uma rota ao pressionar o botão “traçar rota”. Caso o carro saia da rota pré-estabelecida, o aplicativo emitirá notificações, informando a localização atual do dispositivo rastreador.

Na interface uma vez que a rota esteja traçada, é possível visualizar informações como, por exemplo, a latitude e longitude atual do dispositivo rastreador; utilizar a centralização do mapa no dispositivo rastreador; utilizar a centralização do mapa no dispositivo rastreador e, o mapa em si que é retirado do Google Maps, através de uma API; além de traçar uma nova rota. É possível, ainda, mover o mapa, aumentar ou diminuir o zoom.

### **3.4. MTAAlert – Rastreador Veicular**

O MTAAlert é um sistema que se propõe a monitorar quedas em motocicletas e bicicletas. No caso de um acidente, ele é responsável por alertar pessoas próximas ao condutor sobre o ocorrido, através do envio da localização atual do condutor via e-mail. Para isso, além de um protótipo físico, foi desenvolvido um sistema *web*, que permite gerenciar os dados do veículo, do condutor e das pessoas que devem ser notificadas caso um acidente ocorra.

### 3.5. Comparativo com os Trabalhos Correlatos

Nesta subseção, é possível observar, na tabela 1, a comparação entre o atual projeto desenvolvido (RastreiAll) com relação aos demais projetos mencionados neste trabalho.

**Tabela 2. Comparativo sobre as Principais Funcionalidades entre os Trabalhos Correlatos**

	RastreiAll	Xereta	<i>Pocket Guard</i>	Rastramento	MTAlert
Capacidade de realizar o cadastramento de rastreadores	X	X			X
Capacidade de definir um raio de segurança para monitoramento dos equipamentos	X		X	X	
Personalização do Raio de Segurança	X		X	X	
Envio de Email	X	X		X	X
Cadastro de Email por Rastreador	X				

Como é possível observar na tabela, o RastreiAll, além de possuir as funcionalidades proporcionadas por projetos semelhantes, possui ainda o diferencial de permitir o cadastro, personalizável de para qual email será enviado o alerta quando houver a necessidade de ser disparado.

### 4. Metodologia

Na fase inicial do projeto, estabeleceu-se um núcleo essencial, determinado por meio da análise de requisitos funcionais e não funcionais. Com os requisitos definidos, procedeu-se à criação de modelos de banco de dados, que desempenharam um papel duplo: armazenar os dados da aplicação e servir de base para a prototipagem da interface planejada. Uma vez que o núcleo foi desenvolvido, foram feitas novas versões da aplicação *web* que adicionavam outras funcionalidades e refinamentos nas versões anteriores.

Para cada versão, o ciclo de desenvolvimento foi repetido; desde o levantamento de requisitos até a última etapa de testes (funcionais manuais e unitários). As tecnologias utilizadas para o desenvolvimento da aplicação *web* são: o banco de dados MySQL, para o desenvolvimento do *backend*; e para a API, utilizou-se, nesse projeto, o PHP; A API tem como intuito garantir que os responsáveis sejam alertados quando um equipamento ultrapassa os limites de segurança, para isso implementou-se uma função que envia um *e-mail* como notificação. No *frontend* da aplicação, foram utilizados HTML e CSS. Essa funcionalidade foi testada utilizando uma ferramenta chamada Mailtrap, que permite simular o envio de *e-mails* de forma segura durante a fase de desenvolvimento e teste.

Assim, podemos verificar e ajustar o envio de *e-mails* antes de enviá-los aos usuários, garantindo um processo mais eficaz e confiável.

## 5. Desenvolvimento do Trabalho

Esta seção descreve as atividades que foram realizadas durante as fases de desenvolvimento deste trabalho.

### 5.1. Visão de Negócio

Na visão de negócio representada na Figura 2 o usuário comum do sistema é capaz de cadastrar o localizador, o sistema irá então verificar as informações do rastreador e caso necessário irá disparar um alerta através de um email previamente configurado.

O rastreador configurado se refere a um dispositivo que tenha registrado seu nome e o acesso a API. Não necessariamente um usuário tem que realizar as configurações desse rastreador. Esse dispositivo vai receber as coordenadas dos satélites e posteriormente o dispositivo a enviará fazendo uma requisição GET a API.

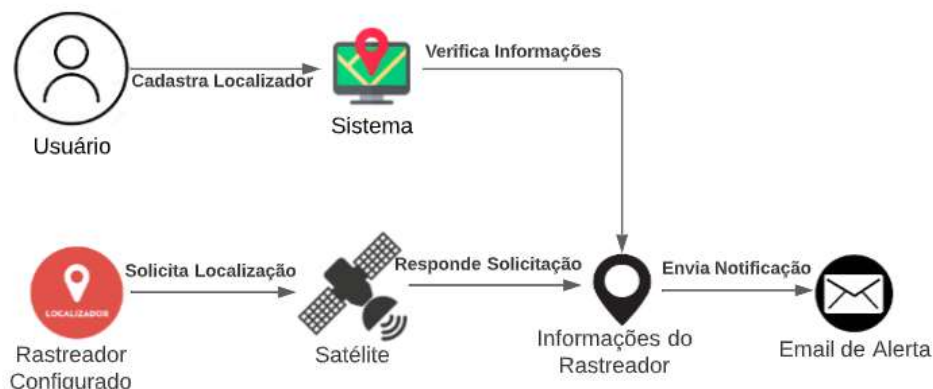


Figura 3. Visão de Negócio

### 5.2. Levantamento de Requisitos

O levantamento de requisitos foi realizado múltiplas vezes. A cada vez que essa etapa foi feita, era desenvolvida uma nova versão do software, que buscava acrescentar novas funcionalidades ao trabalho.

Os primeiros requisitos fazem referência ao núcleo considerado essencial para o funcionamento do projeto. São eles:

- A possibilidade de cadastrar, visualizar, editar ou deletar os rastreadores do sistema *WEB*;
- Envio do e-mail caso o dispositivo rastreador saia da área cadastrada através de uma API;
- Comunicação do rastreador com a API.



Nas demais versões, foram implementadas separadamente: login; cadastro de usuários; níveis de usuários e, por último, foram realizadas melhorias na interface.

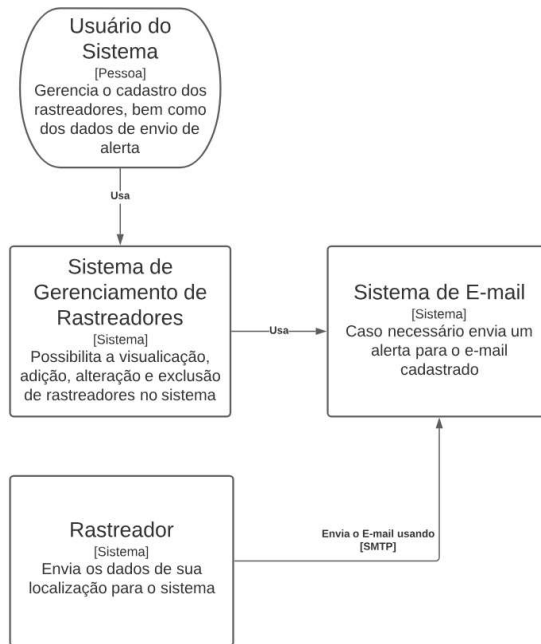
### 5.3. Contexto do Software

Dentre os últimos artefatos técnicos gerados, está um modelo C4 de nível 1 ou Nível de Contexto (melhor explicado a baixo). O intuito desse modelo era possibilitar um melhor entendimento e uma visualização de como o trabalho seria utilizado.

O modelo C4, também conhecido como *Context, Containers, Components, and Code* (Contexto, Contêineres, Componentes e Código), é uma abordagem de modelagem arquitetônica para representar a arquitetura de software de forma clara e hierárquica. Ele divide a arquitetura em quatro níveis distintos, cada um com seu próprio foco e detalhamento.

- Nível de Contexto (Context ou Nível 1): Este é o nível mais alto e fornece uma visão geral do sistema e seu ambiente externo. Ele descreve as interações de alto nível entre o sistema e os atores externos, identificando os principais casos de uso e os limites do sistema.
- Nível de Contêineres (Containers ou Nível 2): Neste nível, o sistema é dividido em contêineres, que são unidades lógicas de agrupamento para componentes relacionados. Os contêineres representam ambientes de tempo de execução nos quais os componentes são implantados e executados. Eles incluem aplicativos da *web*, servidores de aplicativos, bancos de dados, entre outros.
- Nível de Componentes (Components ou Nível 3): Aqui, os contêineres são decompostos em componentes individuais. Os componentes são unidades funcionais que realizam tarefas específicas e têm dependências claras entre si. Este nível permite detalhar a estrutura interna de cada contêiner e suas interações.
- Nível de Código (Code ou Nível 4): O nível mais baixo representa a implementação real dos componentes em código-fonte. Ele inclui classes, módulos, funções e outros artefatos de programação que compõem os componentes.

O modelo C4 ajuda a equipe de desenvolvimento a comunicar efetivamente a arquitetura do sistema em diferentes níveis de detalhamento, facilitando a compreensão e a colaboração. É uma abordagem valiosa para descrever e documentar a arquitetura de software de forma clara e concisa (Brown, Wilson, & Yu, 2018). A Figura 2 demonstra o diagrama criado.



**Figura 4. C4 nível 1**

#### 5.4. Protótipo de Tela e Histórias de Usuário

Com o intuito de idealizar as telas do sistema *web*, foram produzidos alguns protótipos da interface de gerenciamento de cadastro de usuários, *login* e de gerenciamento e cadastro dos rastreadores. Conjuntamente com o desenvolvimento dos protótipos, foram criados e definidos critérios de aceite, bem como histórias de usuário para cada versão de desenvolvimento. A proposta das histórias é auxiliar no levantamento de requisitos através da perspectiva do usuário, permitindo que a equipe de desenvolvimento visualize o que o usuário final precisa. Foi utilizada a seguinte história de usuário para embasar os critérios de aceite, bem como a prototipação: Como usuário, quero cadastrar, editar, visualizar e deletar rastreadores, para ser notificado caso saiam da área cadastrada. Já os critérios de aceite foram definidos como é possível observar na Figura 3.

# Prototipação e História de Usuário

## CRUD Rastreadores

Como usuário quero cadastrar, editar, visualizar e deletar rastreadores, para ser notificado caso saiam da área cadastrada.

Nome do Rastreador	Latitude de Segurança	Longitude de Segurança	Área	Email	Ações
					Editar Excluir

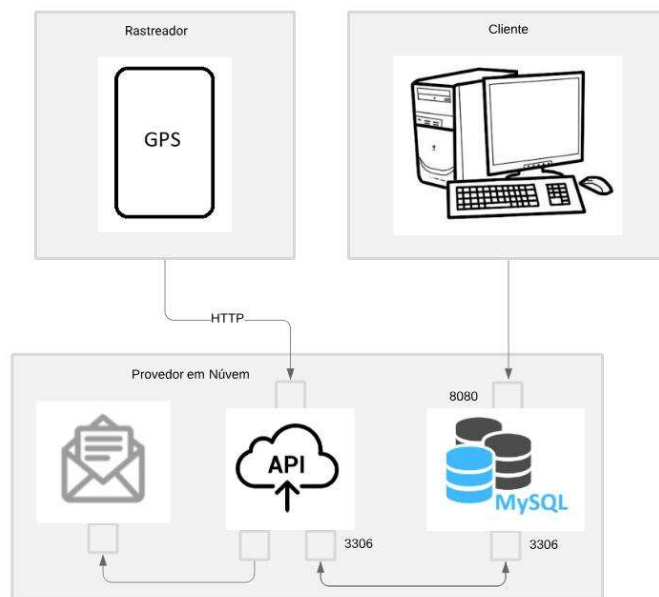
### Critérios de Aceite:

- Deve haver campos para cadastrar nome do rastreador, latitude de segurança, longitude de segurança, área de segurança e o email que receberá a notificação.
- Deve haver um botão responsável por salvar as informações inseridas nos campos de preenchimento.
- Deve haver uma tabela de exibição dos rastreadores cadastrados.
- Dentro da tabela deve haver um campo de "ações" onde haverá os botões de edição e exclusão dos dados dos rastreadores cadastrados.
- O botão de edição deve popular os campos de preenchimento com os dados cadastrados do respectivo rastreador.

Figura 5. Prototipação e História de Usuário

## 5.5. Visão da Arquitetura

Foi desenvolvido um diagrama para representar a Visão da Arquitetura. Na Figura 3, é possível observar que a arquitetura foi separada em três componentes principais: o cliente, o rastreador e o provedor em nuvem. Os dispositivos que consumirem a API do sistema *web* serão o cliente, que poderá ter acesso ao banco de dados. Os dispositivos GPS atuarão como rastreadores e consumirão a API, que poderá acessar o banco de dados e, caso necessário, fazer envio para diferentes serviços de *e-mail*. Para que o rastreador se comunique com a API, utiliza-se do protocolo HTTP, através da porta 3333; enquanto a comunicação do cliente com o banco de dados, bem como da API com o banco de dados, dá-se através da porta 3306. A comunicação da API com o serviço de *e-mail* é relativa ao serviço de *e-mail* selecionado pelo usuário. No caso do protótipo, foi utilizada a ferramenta "Mailtrap". Segundo os próprios desenvolvedores, o Mailtrap é "uma caixa de *e-mails*, que serve para realizar fluxos de testes e cenários, analisar o conteúdo de *e-mails*, permitindo pontuação de spam e validação de HTML e CSS" e utiliza a porta 2525.

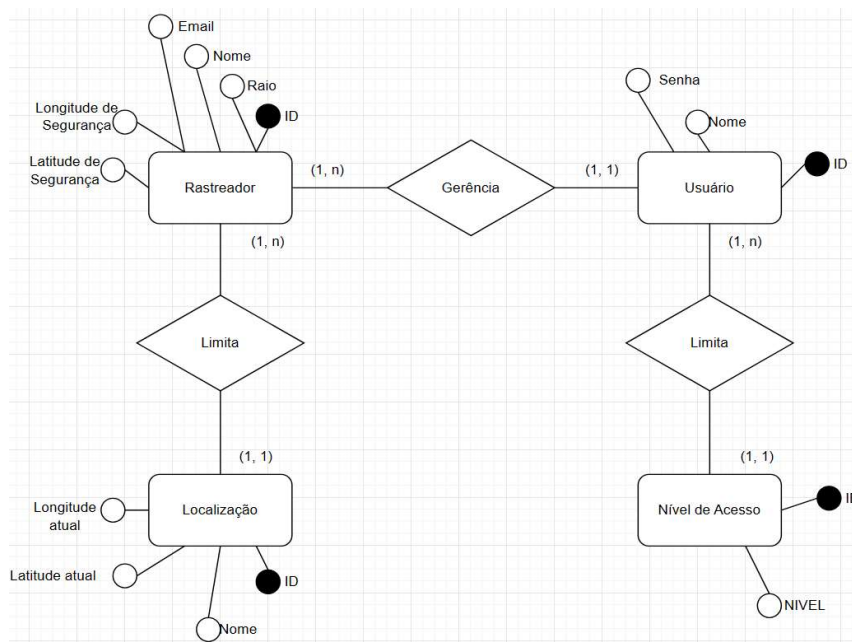


**Figura 6. Arquitetura Geral**

### 5.6. Modelagem do banco de dados

Para representar a modelagem das entidades e dos dados relacionados dentro do trabalho, foi utilizado o modelo conceitual. Para Heuser (2004, apud Gamboa, 2022), esse modelo registra a estrutura que o Banco de Dados terá, independentemente do Sistema de Gerenciamento de Banco de Dados (SGBD) usado. A Figura 4 ilustra o diagrama Entidade-Relacionamento realizada para projetar o banco de dados da aplicação.

Vale ressaltar, que para possibilitar o cálculo da área de segurança o dispositivo Rastreador possui os campos Latitude de Segurança, Longitude de Segurança, bem como o Raio, através dos quais é realizado o cálculo para formar uma área de segurança (raio). Uma vez que a API seja chamada, é possível validar se o rastreador está ou não dentro do raio formado.



**Figura 7. Diagrama Entidade-Relacionamento**

## 5.7. Desenvolvimento do Núcleo do Sistema

Neste capítulo, é descrito como foi implementado o núcleo do sistema, composto pela API, pelo sistema de gerenciamento dos rastreadores e pela comunicação do protótipo com a API.

### 5.7.1. Gerenciamento de Rastreadores

Os dados que devem ser preenchidos para que um rastreador seja cadastrado com sucesso no banco são: o nome do rastreador; a latitude da área de segurança; a longitude da área de segurança; o tamanho do raio da área de segurança; o e-mail ao qual deve ser enviado o alerta caso o dispositivo rastreador saia da respectiva área. Além disso, é exibida uma tabela que permite a visualização, a edição e a exclusão dos rastreadores cadastrados.

### 5.7.2. Desenvolvimento da API

A API desenvolvida consulta os dados persistidos através do nome do dispositivo rastreador que faz o envio de seu nome, bem como de suas coordenadas geográficas. Uma vez recuperadas as informações de segurança no banco, a API é responsável por efetuar a comparação dos dados recebidos com os dados cadastrados e, caso necessário, envia um alerta ao e-mail previamente cadastrado, que contém em seu corpo as coordenadas atuais do dispositivo. As verificações são realizadas a cada cinco minutos, com o intuito de evitar sobrecarga no servidor, tanto por excesso de requisições, quanto pela quantia de dados gravados no banco de dados.

Para realizar o cálculo e verificar se o dispositivo rastreador está dentro ou fora da área de segurança pré-cadastrada, é passada na requisição o nome e a localização atual do rastreador. Através desses dados, é feita a comparação dos dados cadastrados com o

banco, fazendo a comparação das diferenças na latitude e longitude atual, com as da área de segurança, posteriormente verificando se o resultado é maior, menor ou igual ao raio estabelecido anteriormente.

### 5.7.3. Protótipo do Rastreador

O dispositivo escolhido para prototipagem e simulação do funcionamento de um rastreador foi o NodeMCU ESP8266, juntamente com o módulo de GPS NEO-6M. Segundo Pedrosa (2021), dentre as características vantajosas do ESP8266, vale ressaltar seu baixo custo, seu suporte integrado a redes Wi-Fi, bem como a compatibilidade com a IDE do Arduino. Da mesma forma, o módulo se propõe a ter um baixo custo, mantendo uma arquitetura compacta. Nas Figuras 5 e 6, é possível observar o esquema elétrico do NodeMCU e do módulo de GPS NEO-6VM2, respectivamente. O módulo de GPS é conectado ao ESP8266 através dos pinos TX e RX, que são conexões UART0 e UART1, conexões seriais que permitem a transmissão (TX) e a recepção (RX) ao mesmo tempo dos bits de dados. Além disso, existem as conexões de alimentação positiva (VCC) e negativa (GND).

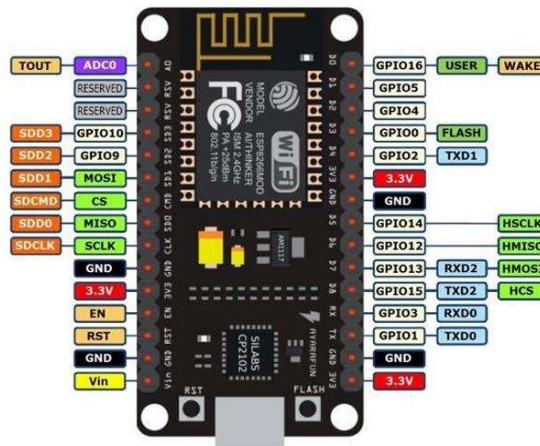


Figura 8. Esquema Elétrico ESP8266

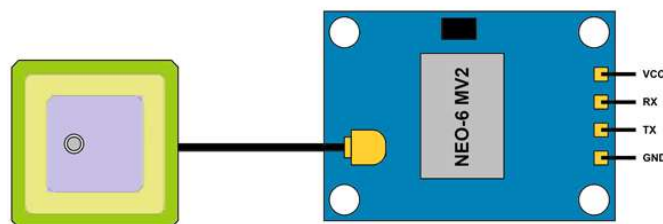


Figura 9. Esquema Elétrico NEO-6MV2

Já na Figura 7, é possível observar o resultado final do protótipo já montado. Para realizar a conexão do módulo GPS ao NodeMCU, foi realizada a soldagem de pinos em

cada uma das conexões de ambas as placas. O ESP8266 foi montado em uma protoboard e, posteriormente, foi realizada a conexão através de cabos macho-fêmea entre o módulo e o NodeMCU. O ESP8266 por questões de praticidade o *Wifi* do ESP8266 foi utilizado para se conectar a rede local e realizar requisições para a API para conseguir testar e validar o correto funcionamento do sistema.

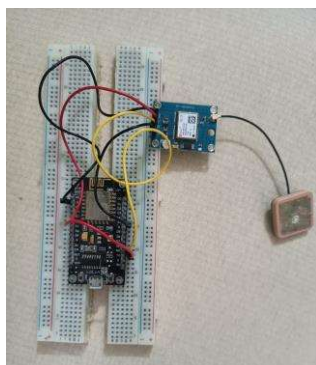


Figura 10. ESP 8266

## 5.8. Novas Iterações

Nesta subseção, serão apresentadas as principais adições e alterações que ocorreram nas iterações seguintes.

### 5.8.1. Mudança no Design da Tela Principal

Depois das primeiras avaliações à tela apresentada na Figura 8, a imagem abaixo acabou por ser a versão final da tela de rastreadores. Essa imagem é vista da perspectiva de um administrador do sistema que pode, além de gerenciar os dispositivos rastreadores e realizar o logout, acessar a aba de usuários. Para um usuário comum do sistema, o botão 'usuários' não seria visível, e o botão de 'logout' estaria verticalmente centralizado.

Nome do Rastreador	Latitude de Segurança	Longitude de Segurança	Raio	Email	Ações
r1	0.000000	0.000000	5	teste@teste.com	Editar Excluir

Figura 110. Versão atual da Tela dos Rastreadores

### 5.8.2. Login

Ao tentar acessar o site, a primeira página visualizada é a página de autenticação, na qual o usuário deverá fornecer suas credenciais; até o momento, usuário e senha. Uma vez validados os dados com relação ao banco de dados, a página será redirecionada, mantendo um token do usuário, para a página principal do sistema. Na Figura 9, é possível observar a tela de login do dispositivo.

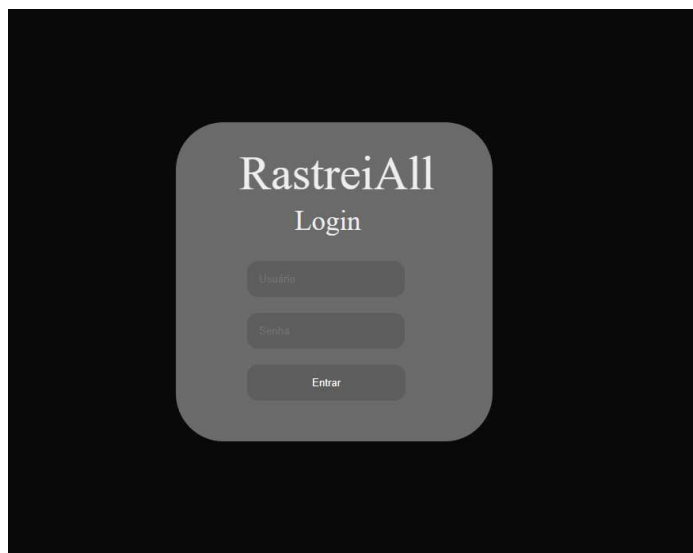


Figura 121. Tela de Login

### 5.8.3. Cadastro de Usuário

A tela de cadastro de usuários pode ser acessada assim que um administrador fizer login no sistema. Será possível visualizar um botão chamado "usuários" no canto superior direito da tela que direciona para a tela de gerenciamento de usuários. Na tela de gerenciamento, haverá três níveis diferentes de usuários que podem ser adicionados, editados, excluídos ou atualizados. O primeiro e maior nível consegue gerenciar usuários em todos os três níveis existentes no sistema; já os demais níveis não podem alterar, excluir ou adicionar suas credenciais. O segundo nível, também de administrador, consegue gerenciar usuários de segundo e terceiro nível, tendo permissão para adicioná-los, excluí-los ou alterá-los. O terceiro nível é o nível de usuário comum; esse nível apenas permite a utilização das funções principais do sistema, sequer tendo acesso à tela de gerenciamento de usuários. Na Figura 10, é possível observar a tela de gerenciamento de usuários.



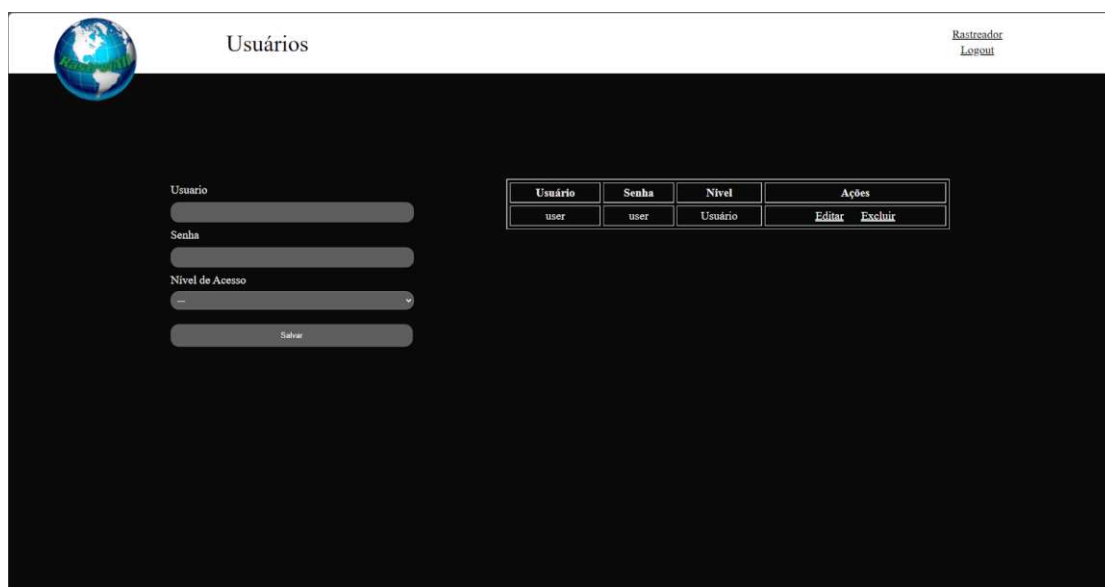


Figura 132. Tela de Gerenciamento de Usuários

## 5.9. Testes

A cada iteração, eram realizados testes funcionais manuais para todas as funções presentes. Uma vez finalizada a aplicação, foram realizados testes na API utilizando o PHPUnit, nos quais foi avaliado o correto funcionamento da parte do sistema referente ao envio do sistema, para tanto, foram realizados testes para validar se o dispositivo estaria fora da área de segurança, dentro da área de segurança, bem como do envio do email. Para isso, são passados parâmetros fictícios, simulando tanto dados cadastrados no banco, quanto dados do próprio rastreador. Da mesma forma, é enviado um *e-mail* com dados fictícios de um dispositivo que saiu da área.

## 6. Conclusões

Esse trabalho teve como intuito desenvolver um protótipo de um sistema de monitoramento da localização de diferentes equipamentos que estejam na área externa de uma empresa. Com esse projeto, busca-se aumentar as chances de recuperar ou encontrar eventuais situações que pudessem fazer com que o objeto deixasse a área de segurança. Para tanto, o sistema foi parcialmente desenvolvido, com enfoque nas principais funcionalidades para o bom uso da aplicação. Essas funcionalidades seriam a possibilidade de gerenciamento e login de usuário; gerenciamento de rastreadores; a API responsável por disparar o alerta e o protótipo físico de um dispositivo rastreador.

Dentre as futuras funcionalidades para implementação, algumas que possivelmente se destacariam seriam a criação de tabelas no banco de dados, responsável por armazenar a localização de cada rastreador individualmente, bem como a possibilidade de visualização desses dados. Também, a possibilidade de integração com sistemas de monitoramento interno, que ativariam o dispositivo rastreador ao saírem do ambiente interno, com o intuito de reduzir o consumo de energia e viabilizar seu uso em

equipamentos presentes em ambientes internos sem nenhum tipo de eventual problema. Vale ressaltar ainda, a utilização de módulos 3G (ou melhores) para ampliar a rede de cobertura e envio dos alertas. Com o desenvolvimento deste trabalho, foi possível aplicar e ampliar conhecimentos adquiridos nas disciplinas lecionadas no curso de Análise e Desenvolvimento de Sistemas, dentre elas, estão presentes: Banco de Dados, Desenvolvimento de Sistemas Web, Engenharia de Software, etc.

## 7. References

Brown, S., Wilson, S., & Yu, K. (2018). The C4 model for visualizing software architecture. Leanpub. Portal da Indústria. Indústria 4.0: Entenda seus conceitos e fundamentos. 2022. Disponível em: <<https://www.portaldaindustria.com.br/industria-de-a-z/industria-4-0/>>. Último Acesso em: 19 de Out. 2022.

Castro, D. Sistema de Informação para Gestão de Equipamentos. ISEP. São Paulo, 2016. Disponível em: <[https://recipp.ipp.pt/bitstream/10400.22/9782/1/DM\\_DiogoCastro\\_2016\\_MEEC.pdf](https://recipp.ipp.pt/bitstream/10400.22/9782/1/DM_DiogoCastro_2016_MEEC.pdf)>. Último Acesso em: 22 de Jun. 2022.

Fagundes, P. Deters, J. Santos, S. COMPARAÇÃO ENTRE OS PROCESSOS DOS MÉTODOS ÁGEIS: XP, SCRUM, FDD E ASD EM RELAÇÃO AO DESENVOLVIMENTO ITERATIVO INCREMENTAL. E-Tech: Atualidades Tecnológicas para Competitividade Industrial, Florianópolis, v. 1, n. 1, p. 37-46, 2008. Disponível em: <<https://etech.sc.senai.br/edicao01/article/view/21/18>>. Último Acesso em: 22 de Jun. 2022.

Kobs, F. Pinotkewicz, T. MONITORAMENTO DE VEÍCULOS EM ÁREAS PRIVADAS DE ORGANIZAÇÕES. Revista Eletrônica do Alto Vale do Itajaí – REAVI, v.10, no 17, p.060-067, 2021. Disponível em: <<https://periodicos.udesc.br/index.php/reavi/article/view/21355/13887>>. Último Acesso em: 22 de Jun. 2022.

Melo, A. Sistema para rastreamento veicular via GPS. UFSC. Blumenau, 2021. Disponível em:

<<https://repositorio.ufsc.br/bitstream/handle/123456789/223808/TCC.pdf?sequence=1&isAllowed=y>>. Último Acesso em: 22 de Jun. 2022.

MELO, C. M. Helene. A IMPORTÂNCIA DO SISTEMA DE INFORMAÇÃO NA GESTÃO DO SISTEMA ÚNICO DE SAÚDE (SUS). UNILAB. São Francisco do Conde, 2018. Disponível em:

<[https://repositorio.unilab.edu.br/jspui/bitstream/123456789/702/3/2018\\_arti\\_hmelo.pdf](https://repositorio.unilab.edu.br/jspui/bitstream/123456789/702/3/2018_arti_hmelo.pdf)>. Último Acesso em: 22 de Jun. 2022.

Pepino, C. Dias, G. POCKET GUARD – DISPOSITIVO PORTÁTIL DE SEGURANÇA IOT. UTFPR. Curitiba, 2018. Disponível em:

<[http://repositorio.utfpr.edu.br/jspui/bitstream/1/8467/1/CT\\_DAELN\\_2018\\_1\\_06.pdf](http://repositorio.utfpr.edu.br/jspui/bitstream/1/8467/1/CT_DAELN_2018_1_06.pdf)>. Último Acesso em: 22 de Jun. 2022.

Pereira, B. PROTÓTIPO DE UM SISTEMA DE RASTREAMENTO PARA COMPUTADORES. Blumenau, 2013. Disponível em:

<[http://dsc.inf.furb.br/arquivos/tccs/monografias/2013\\_2\\_barbara-dias\\_pereira\\_monografia.pdf](http://dsc.inf.furb.br/arquivos/tccs/monografias/2013_2_barbara-dias_pereira_monografia.pdf)>. Último Acesso em: 22 de Jun. 2022.

Santos, B. Alberto, et. al. INDÚSTRIA 4.0: DESAFIOS E OPORTUNIDADES. Revista Produção e Desenvolvimento. Brasil, 2018. Disponível em:

<<https://revistas.cefetrj.br/index.php/producaoedesenvolvimento/article/view/e316/193>>. Último Acesso em: 22 de Jun. 2022.

Schwab, K. Zahidi, S. The Future of Jobs Report 2020. World Economic Forum. 2020 Disponível em: <<https://www.weforum.org/reports/the-future-of-jobs-report-2020>>. Último Acesso em: 22 de Jun. 2022.

Silva, Eder; Oliveira, Rodolfo. MT Alert – Rastreador Veicular. IFSP, Hortolândia, 2018. Disponível em:

<[https://hto.ifsp.edu.br/portal/images/thumbnails/images/IFSP/Cursos/Coord\\_ADS/Arquivos/TCCs/2018/TCC\\_EderAntoniodaSilva\\_HT1420445.pdf](https://hto.ifsp.edu.br/portal/images/thumbnails/images/IFSP/Cursos/Coord_ADS/Arquivos/TCCs/2018/TCC_EderAntoniodaSilva_HT1420445.pdf)>. Último Acesso em: 19 de Out. 2022.

Souza, R. Osnete. Processos de apoio ao desenvolvimento de aplicações Web. USP – São Carlos, 2005. Disponível em:

<[https://www.teses.usp.br/teses/disponiveis/55/55134/tde23072005101245/publico/AplicacoesWeb\\_osnee.pdf](https://www.teses.usp.br/teses/disponiveis/55/55134/tde23072005101245/publico/AplicacoesWeb_osnee.pdf)>. Último acesso em: 22 de Jun. 2022.

# Documento Digitalizado Público

## Artigo de TCC - Versão Final

**Assunto:** Artigo de TCC - Versão Final  
**Assinado por:** Fernando Sambinelli  
**Tipo do Documento:** Anexo  
**Situação:** Finalizado  
**Nível de Acesso:** Público  
**Tipo do Conferência:** Documento Digital

Documento assinado eletronicamente por:

- **Fernando Sambinelli, PROFESSOR ENS BASICO TECN TECNOLOGICO**, em 04/12/2023 14:26:08.

Este documento foi armazenado no SUAP em 04/12/2023. Para comprovar sua integridade, faça a leitura do QRCode ao lado ou acesse <https://suap.ifsp.edu.br/verificar-documento-externo/> e forneça os dados abaixo:

**Código Verificador:** 1508852

**Código de Autenticação:** ace1b1a082

