análise e desenvolvimento de sistemas

Ronaldo bilhar

pet monitor:

Segurança e Tecnologia

Estrela

2019

ronaldo bilhar

pet monitor:

Segurança e Tecnologia

Trabalho de portfólio apresentado como requisito parcial para a obtenção de média semestral.

Orientadora: Profa. Luana Gomes de Souza

Estrela

2019

SUMÁRIO

[1 INTRODUÇÃO 3](#_Toc23076102)

[2 APLICATIVO PET MONITOR 4](#_Toc23076103)

[2.1 O ALGORÍTMO 4](#_Toc23076104)

[2.2 INTERFACE 6](#_Toc23076105)

[2.2.1 CÓDIGO DO DOCUMENTO HTML 7](#_Toc23076106)

[2.2.2 CÓDIGO CSS 8](#_Toc23076107)

[2.3 DESENVOLVIMENTO E CICLO DE VIDA DO SOFTWARE 11](#_Toc23076108)

[CONCLUSÃO 14](#_Toc23076109)

[REFERÊNCIAS 15](#_Toc23076110)

# INTRODUÇÃO

Após a concepção de um problema e uma ideia de resolução deste problema relatado por um cliente, onde houve uma dificuldade de achar seu pet que se perdeu durante um passeio, desenvolvemos uma solução em forma de aplicativo web.

Criamos, juntamente com o cliente, um programa que indica a posição do pet em relação ao seu tutor além de criar um perímetro seguro previamente configurado, emitindo um alerta em caso de transposição desse perímetro.

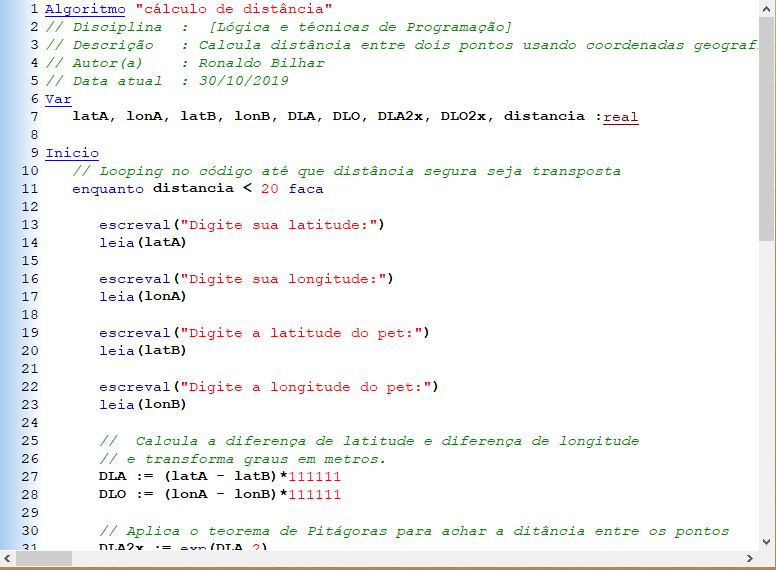
Nesse sentido, apresentaremos o desenvolvimento do software e suas funcionalidades.

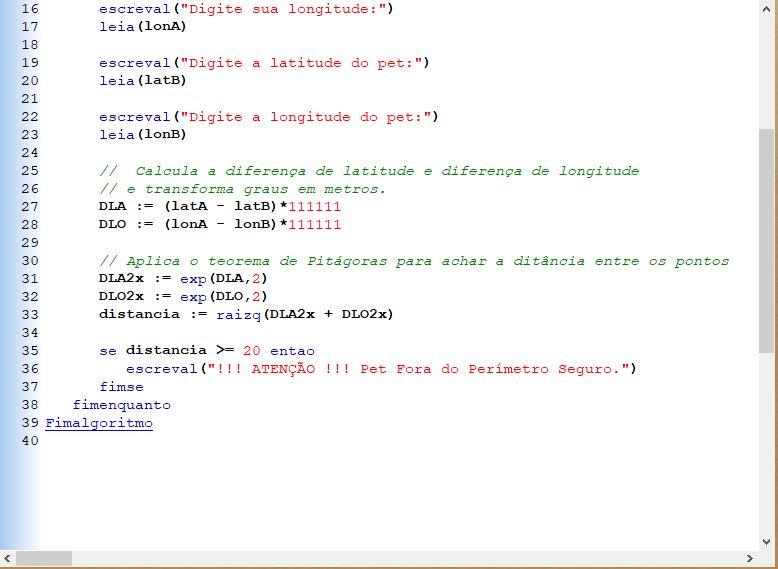
# APLICATIVO PET MONITOR

O aplicativo desenvolvido tem como principal função criar um perímetro de segurança configurado pelo usuário. Esse perímetro tem o objetivo de auxiliar e alertar o tutor caso o animal transponha o perímetro. Funções como histórico de passeios, mapa da região e interação com outros usuários também se faz presente.

## O ALGORÍTMO

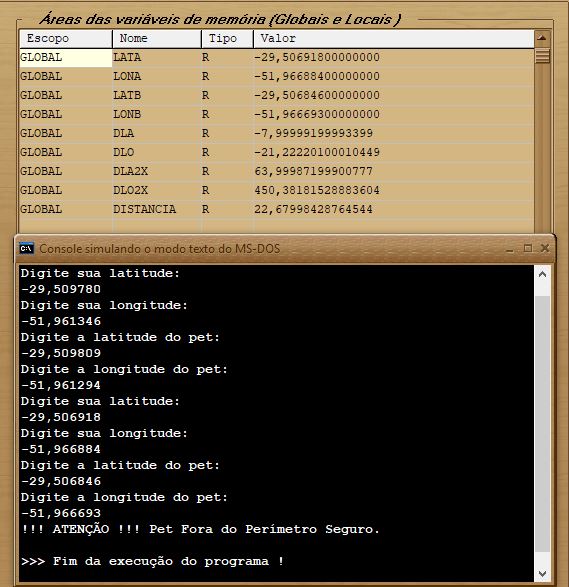
**Figura 2.1.1 | Algoritmo que calcula a distância**





**Fonte: o autor**

**Figura 2.1.2 | Console simulando entradas do usuário**

****

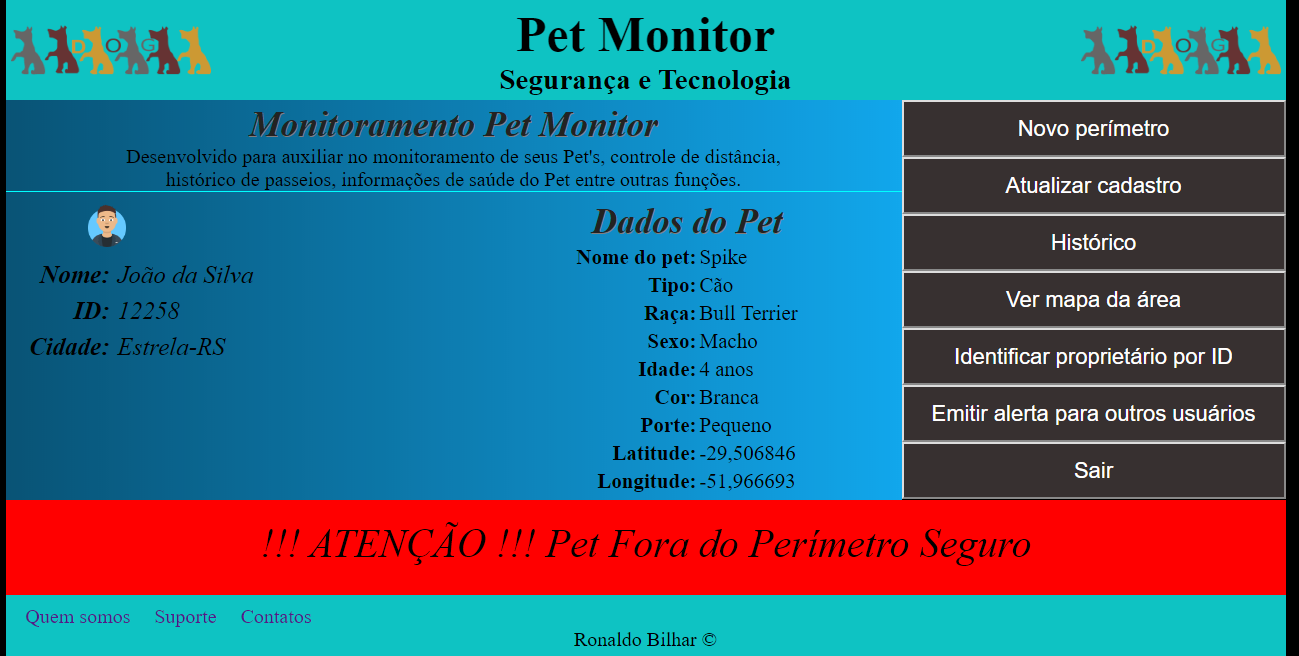
Para o uso do aplicativo devemos alertar que este serve apenas cálculos para pequenas distâncias, pois o algoritmo não leva em conta distância como altitude e apresenta uma porcentagem de erro devido a dificuldade de se calcular a longitude, pois a mesma se difere quando se afastamos da linha do Equador.

As figuras mostram o algoritmo e também a simulação do mesmo no console, primeiro foram inseridas coordenadas que estariam em uma distância menor que 20 metros. Podemos perceber que ele entrou em loop, e após a inserção de coordenadas que apresentariam uma distância maior que 20 metros, o algoritmo disparou o alerta, atendendo assim a necessidade do usuário.

## interface

Aqui apresentamos como seria o protótipo de interface do aplicativo web, desenvolvido em HTML5 e CSS.

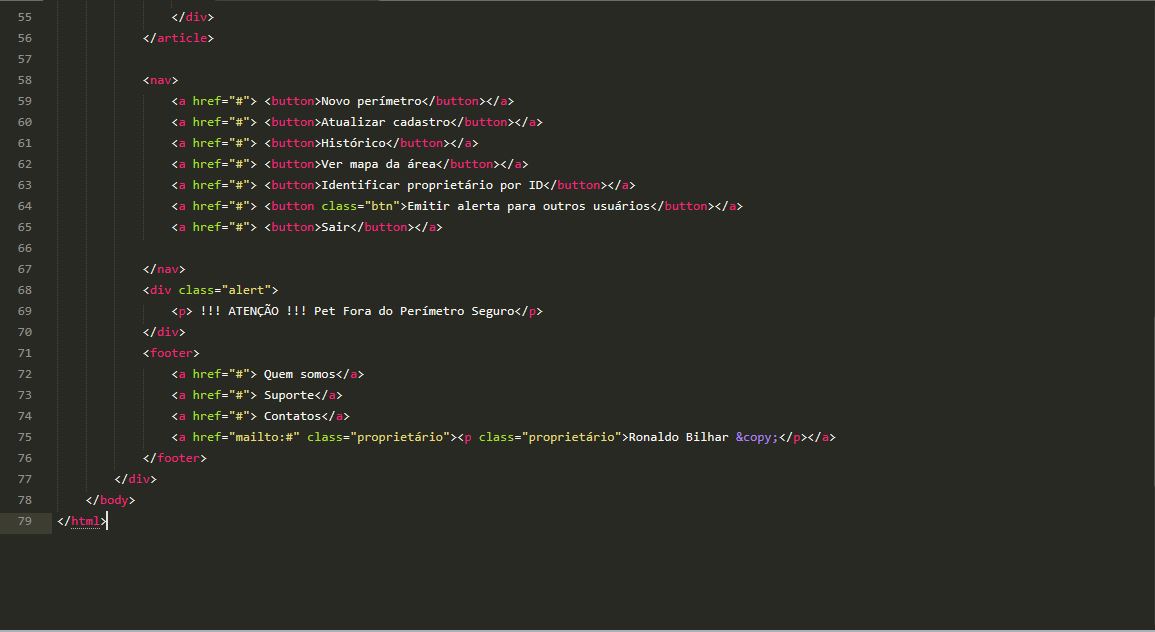
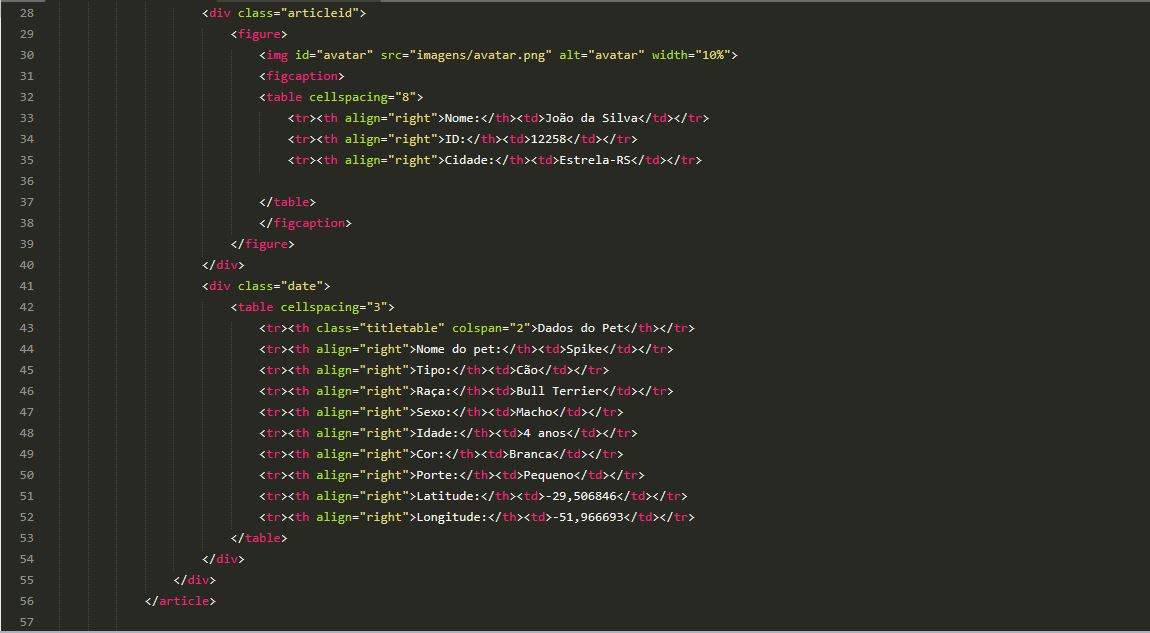
**Figura 2.2 | Protótipo de interface HTML**



**Fonte: o autor**

### CÓDIGO DO DOCUMENTO HTML

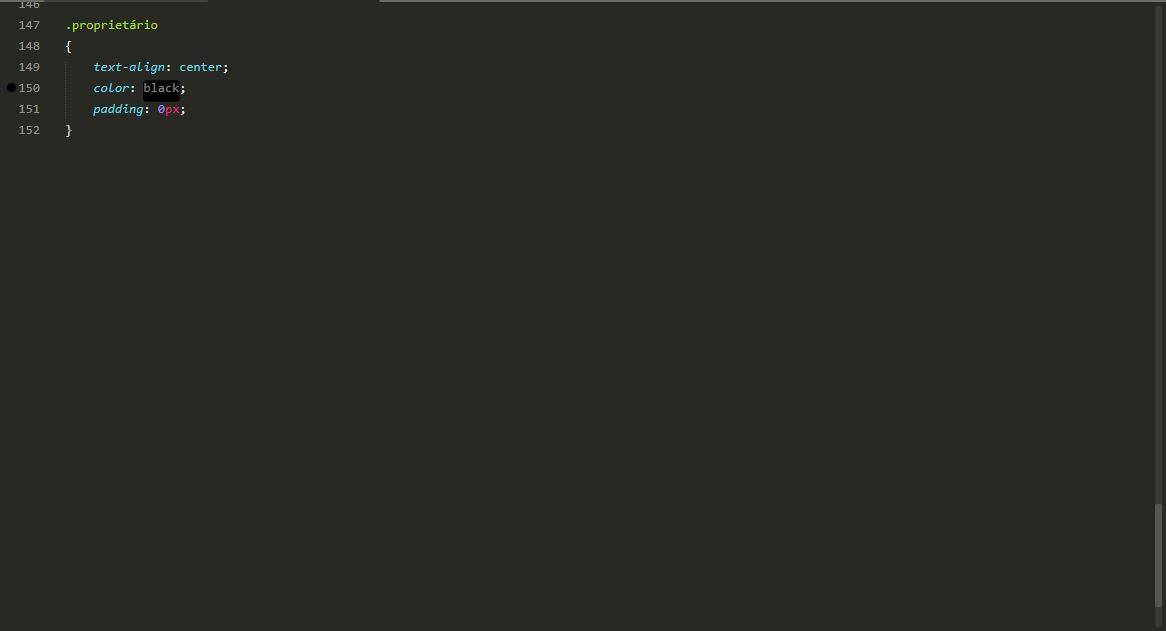
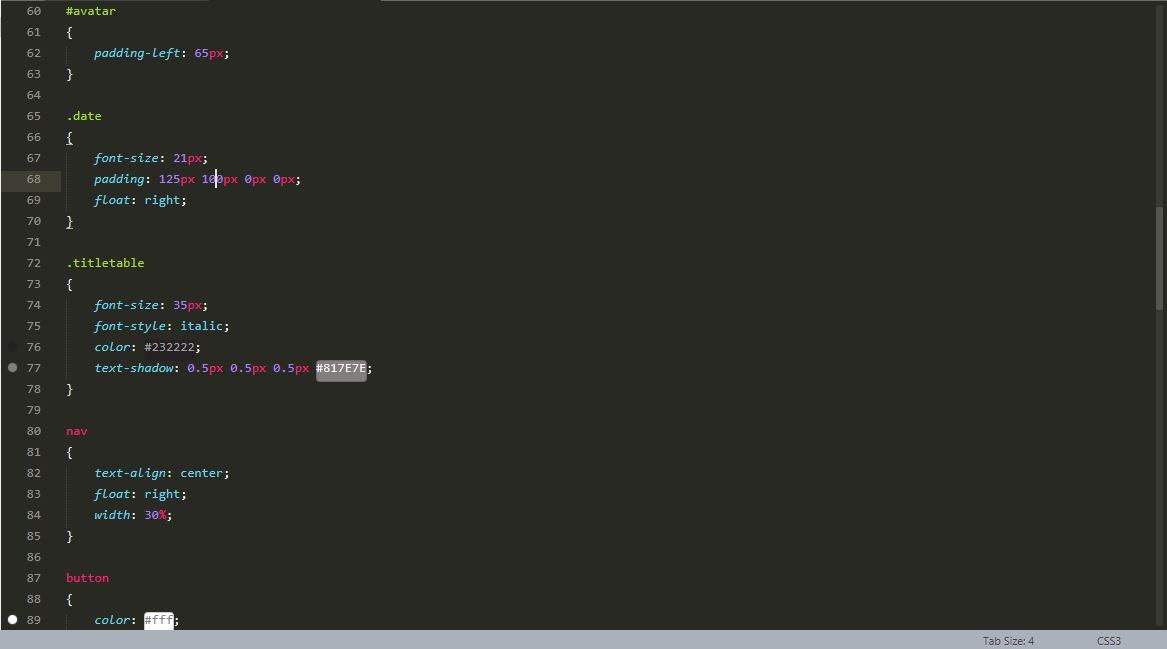
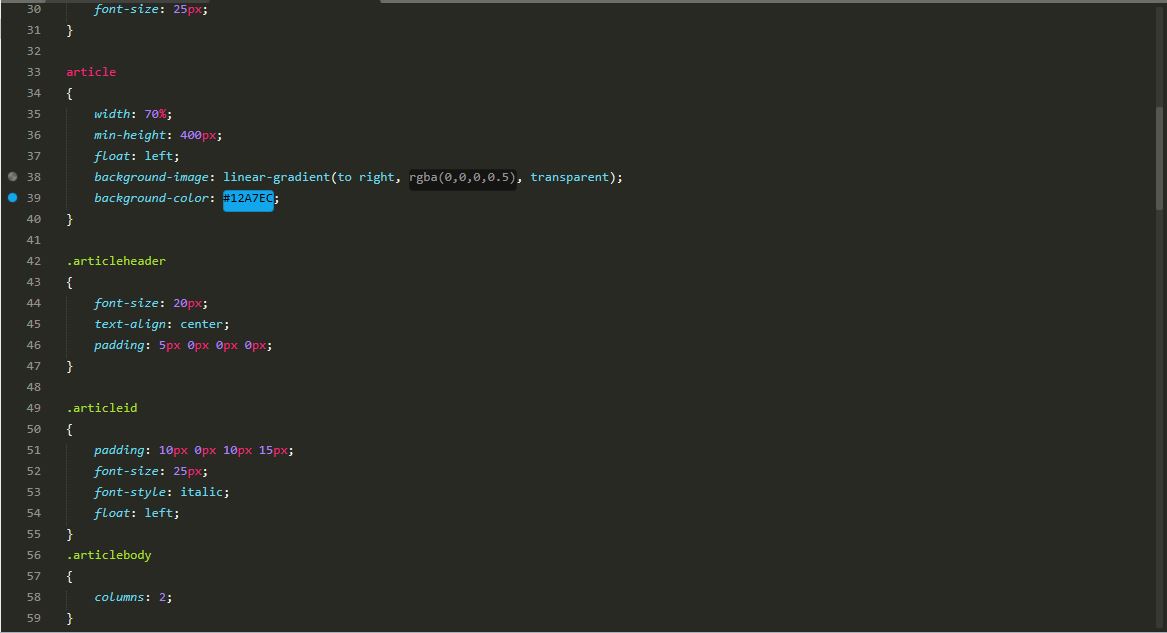
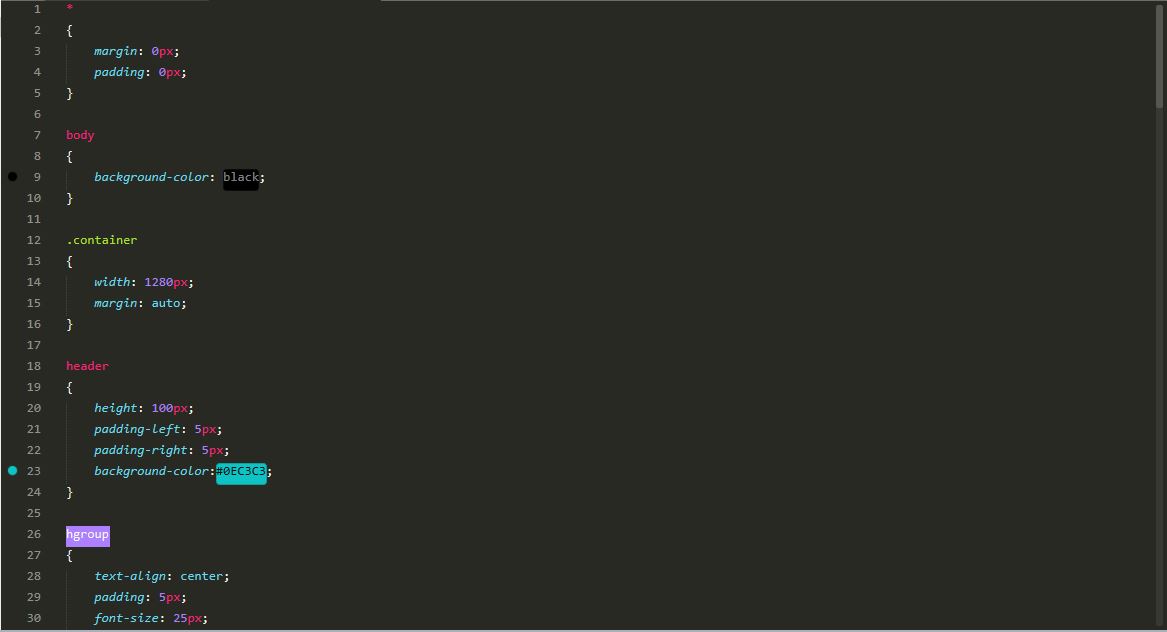
**Figuras 2.2.1 | Código HTML da interface**



**Fonte: o autor**

### CÓDIGO CSS

**Figuras 2.2.2 – Código CSS**



**Fonte: o autor**

## DESENVOLVIMENTO E CICLO DE VIDA DO SOFTWARE

Para a construção do software, levamos em conta os requisitos descritos por nosso cliente, entre eles estão:

O aplicativo deve monitorar / medir a distância entre tutor e pet;

Ser configurável pelo usuário, para determinar qual distância será segura ou não;

Manter um histórico de passeios para que possa ser configurada como distância segura, uma distância anteriormente utilizada;

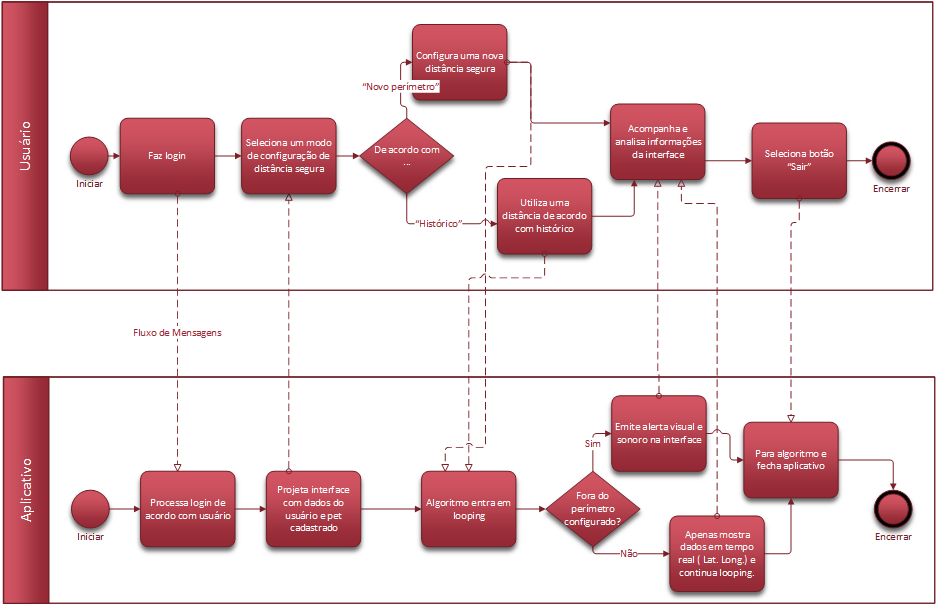
O histórico também apresentará data, hora, local e informações relacionadas à saúde do pet;

Emitir um alerta sonoro / visual caso o pet ultrapasse a distância segura pré-configurada;

Tendo em vista que, o projeto será desenvolvido em forma de aplicativo web e que além de cumprir sua principal funcionalidade deve satisfazer o usuário com suas funcionalidades secundárias e ter uma constante evolução e o mais importante, ser simples, adotamos o método XP (eXtremmeProgramming) para o desenvolvimento do aplicativo. Este método nos proporciona a criação do aplicativo em uma forma simples e também nos possibilita a preparação para uma evolução e mudanças constante.

O requisito de negócio compreende a principal função do aplicativo, para melhor expressar, criamos um diagrama usando o BPMN.

**Figura 2.3 – Diagrama BPMN de funcionalidade e interação usuário/aplicativo**



**Fonte: o autor**

Após a descrição e validação dos requisitos, criamos um backlog das funcionalidades do aplicativo, que são:

* Apresentar na tela principal os dados do usuário e do pet, bem como a descrição em tempo real da latitude e longitude do pet;
* Novo perímetro: será utilizado para configurar uma distância (em metros) manualmente pelo usuário;
* Atualizar cadastro: o usuário poderá modificar informações pessoais e também do pet;
* Histórico: apresentará dados de saúde do pet e informações do último passeio (calorias perdidas, distância percorrida, temperatura no local, entre outros), e também terá a possibilidade de o usuário configurar uma distância segura de acordo com uma gravada no histórico;
* Ver mapa da área: abre o Google Maps para uma inspeção na área em que se encontra o usuário;
* Identificar proprietário por ID: caso se encontre um pet perdido, pode se identificar o dono pelo ID que estará gravado na coleira;
* Emitir alerta para outros usuários: possibilita um usuário a emitir um alerta de pet perdido, para que outros usuários do aplicativo ajudem na recuperação do pet;

Com o entendimento do processo de negócio, verificação e validação de requisitos, criação do backlog das funcionalidades, designamos as funções dos desenvolvedores a fim de implementar o projeto o mais rápido possível, com sua forma simplificada, mas realmente funcional.

Em um próximo ciclo, seria usado Bootstrap para uma melhor interface tanto na questão estética quanto na questão responsiva. E a cada ciclo colocaríamos uma função ou incrementaríamos uma já implementada. Um exemplo é a questão de o algoritmo medir a distância apenas em um modo plano, não levando em conta medições como diferença de altitude, onde faríamos um novo algoritmo para que isso fosse corrigido. Pensando na questão evolutiva, em um ciclo poderíamos avaliar a ideia de usar a tecnologia de realidade aumentada para conduzir o usuário ao pet no momento que este ultrapassasse o perímetro de seguro.

# CONCLUSÃO

O trabalho desenvolvido originou-se da necessidade e ideia de um cliente, usamos nossos conhecimentos afim de construir um programa que atendesse suas necessidades. Para isso buscamos compreender as regras de negócio e requisitos de software juntamente com o cliente, usando o método XP para sua construção. O protótipo de interface e o algoritmo construídos, atendem a necessidade inicial do cliente, e a metodologia usada, concede-nos possibilidades de melhorar e evoluir o software para que este se adapte à constante evolução que a tecnologia vive nos dias atuais.

###### REFERÊNCIAS

ARTERO, Marcio Aparecido; SCHEFFER, Vanessa Cadan. **Algoritmos e lógica de programação.** Londrina: Editora e Distribuidora Educacional S.A., 2018

COORDENADAS GEOGRÁFICAS. Disponível em: <https://pt.wikipedia.org/wiki/Coordenadas_geográficas>. Acesso em: 10 out. 2019

FABRIS, Polyanna Pacheco Gomes; PERNI, Luis Cláudio. **Processos de software.** Londrina: Editora e Distribuidora Educacional S.A., 2014.

FALCONI, Carlos Eduardo**. Calculando distâncias e direções utilizando Coordenadas Geográficas.** Disponível em: <https://www.pilotopolicial.com.br/calculando-distancias-e-direcoes-utilizando-coordenadas-geograficas/>. Acesso em: 18 out. 2019

FALCONI, Carlos Eduardo. **Como calcular a precisão de uma Coordenada Geográfica.** Disponível em: <https://www.pilotopolicial.com.br/como-calcular-a-precisao-de-uma-coordenada-geografica/>. Acesso em: 15 out. 2019

FALCONI, Carlos Eduardo. **Navegação Aérea:** como converter coordenadas geográficas. Disponível em: [https://www.pilotopolicial.com.br/navegacao-aerea-como-converter-coordenadas-geograficas/](https://www.pilotopolicial.com.br/navegacao-aerea-como-converter-coordenadas-geograficas/%20) . Acesso em: 15 out. 2019