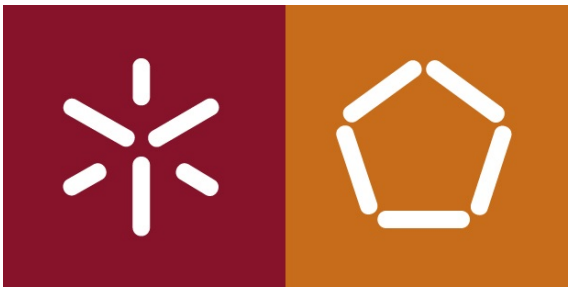

Universidade do Minho

Mestrado em Engenharia Informática

Engenharia Web

Engenharia de Sistemas de Software

Sidewalk Monitoring System

Francisco Freitas (A81580)
Inês Alves (A81368)
Pedro Freitas (A80975)

20 de Abril de 2020

Resumo

No âmbito da Unidade Curricular de Engenharia Web, enquadrada no perfil de especialização de Engenharia de Sistemas de Software, este projeto foi desenvolvido com o objetivo de abordar e compreender melhor os conceitos abordados nas aulas desta UC, sendo que se pretende implementar eventuais progressos no seu desenvolvimento à medida que vão sendo introduzidos novos conhecimentos.

Conteúdo

1 Propósito do Projeto

Nos dias de hoje é necessário um constante acompanhamento das evoluções tecnológicas e uma constante criatividade de forma a incentivarmos uma frequente inovação de forma a solucionar problemas que nos surgem no nosso dia-a-dia.

Assim no âmbito da Unidade Curricular de Engenharia Web, enquandrada no 2º Semestre do 4º Ano do curso de Engenharia Informática que pertence ao perfil de especialização de Engenharia de Sistemas de Sfotware, foi-nos apresentado um problema para solucionarmos.

Este problema foi-nos apresentado com o objetivo de realizarmos um planeamento e a conceção de uma aplicação web para monitorizar passadeiras. Assim com este trabalho, além de exigir conhecimentos de várias Unidades Curriculares previamente abordadas, exige um maior esforço e foco no desenvolver das capacidades e conhecimentos de algumas ferramentas e tecnologias de desenvolvimento de aplicações web.

2 SideWalk Monitoring System

De forma a atingir os objetivos enumerados na secção anterior foi-nos apresentado um enunciado de um problema que temos de solucionar.

Assim temos de desenvolver um sistema que se encaixa sobre um sistema *SPWS - SideWalk Proximity Warning System*. Com isto, deve ser possível a veículos autónomos perto de uma dada passadeira receber notificações sobre o estado da mesma e uma decisão sobre se é seguro e possível continuar o seu caminho e passar por uma passadeira.

Dito isto um dos objetivos deste sistema é ser um suplemento às informações sobre os estados de um semáforo, adicionando assim as informações sobre a presença de peões perto da passadeira, tornando assim a circulação mais segura.

Outro aspeto em ter em conta neste problema é garantir o registo do fluxo de peões e veículos na área dessa passadeira.

Para a notificação chegar a um veículo autónomo , o sistema de ter consciência da presença de um peão nas proximidades. Assim o dispositivo móvel do peão tem de alertar o sistema quando este se aproximar de uma passadeira. Assim existem dois cenários possíveis para as notificações:

- *Existem peões* - o veículo recebe uma notificação a alertar a presença do peão e que não é seguro atravessar.
- *Não existem peões* - o veículo recebe uma notificação com a confirmação de que é seguro passar.

Assim o sistema será formado por três componentes diferentes:

- **Vehicle app** - responsável por enviar coordenadas do veículo ao sistema principal assim como receber as notificações dele.
- **Pedestrian app** - responsável por enviar as coordenadas do peão ao sistema.
- **SPWS** - responsável por monitorizar o estado e os eventos ocorridos numa passadeira. Esta será também responsável por enviar as notificações corretas aos veículos.

3 Modelo de Domínio

Após refletirmos sobre o problema apresentado e as funcionalidades que queremos construir para poder responder a tudo que é pedido obtivemos as principais entidades do sistema e as suas relações. Na figura seguinte vemos o resultado dessa análise:

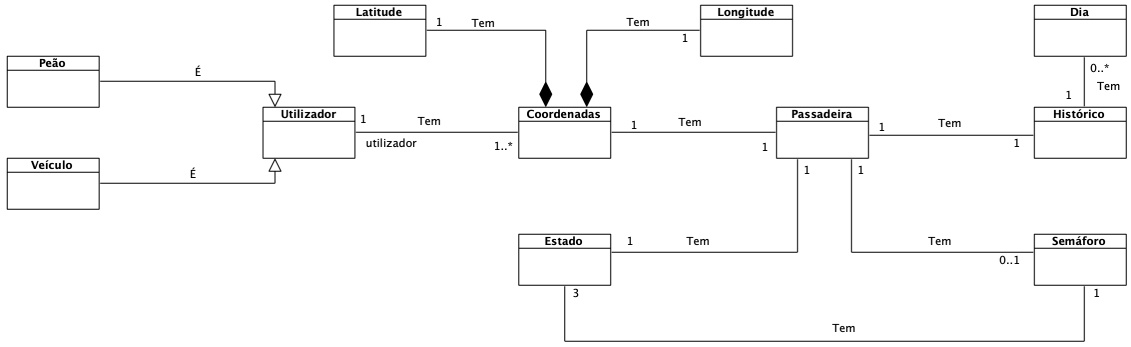

Figura 1: Modelo de Domínio

3.1 Dicionário

Na figura acima (Figura 1) vemos as entidades que estão presentes no nosso sistema. Assim para percebermos melhor o que cada uma delas é vamos fazer uma breve descrição delas.

Utilizador

- Entidade que representa um ator do sistema. Este é aquele que vai usufruir das funcionalidades que a aplicação oferece.

Peão

- É um tipo de utilizador.

Veículo

- É um tipo de utilizador.

Coordenadas

- Entidade que representa a localização de um utilização. Esta é constituído por um par de valores.

Latitude

- Refere-se à coordenada geográfica. Esta diz respeito ao primeiro valor do par da entidade *Coordenadas*.

Longitude

- Refere-se à coordenada geográfica. Esta diz respeito ao segundo valor do par da entidade *Coordenadas*.

Passadeira

- Entidade que representa uma passadeira.

Estado

- Diz respeito ao estado. Diz-nos o estado da mesma (se é possível passar ou não).

Histórico

- Diz respeito a uma passadeira. É o histórico de todos os utilizadores que passaram por ela e está organizada por dias.

Dia

- Diz respeito a uma divisão do histórico de uma passadeira.

Semáforo

- Representa os semáforos existentes numa passadeira.

3.2 Entidades Relevantes

Peão

É um tipo de utilizador da nossa aplicação. Este será responsável por enviar as suas coordenadas ao sistema quando se aproxima de uma passadeira.

Veículo

É outro tipo de utilizador. Este além de enviar as suas coordenadas quando se aproxima de uma passadeira, vai estar preparado para receber notificação do estado das passadeiras.

Passadeira

É a entidade base de todo o sistema. Toda a aplicação funciona em torno desta entidade que regulará o fluxo de dados do sistema.

4 Diagrama de Use Cases

A Figura 2 diz respeito ao diagrama de use cases desenvolvido, pensando no sistema e suas funcionalidades.

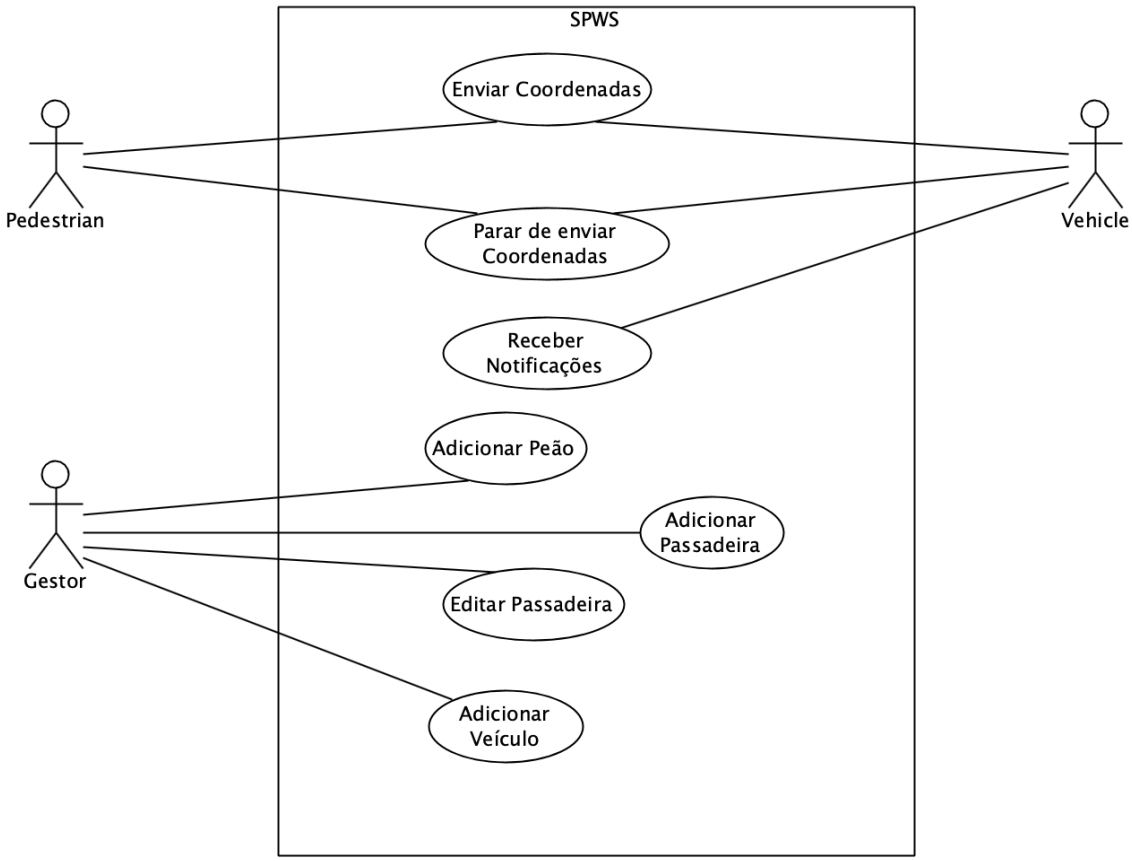

Figura 2: Diagrama de Use Cases

4.1 Atores

Pedestrian

Representa um peão cujas funcionalidades permiter enviar coordenadas, que correspondem á sua localização, e que também permitem que deixem de enviar as suas coordenadas.

Vehicle

Representa um veículo cujas funcionalidades permiter enviar coordenadas, que correspondem á sua localização, e que também permitem que deixem de enviar as suas coordenadas. É também possível ao veículo receber notificações do sistema sobre o estado da passadeira.

Gestor

Representa o gestor do sistema. Este é responsável por gerir as entidades registadas no sistema, podendo adicionar peões, veículos e passadeiras e editar as informações desta última.

4.2 Use Cases

Enviar coordenadas

Este use case reflete a situação em que um utilizador da aplicação se aproxima de uma passadeira e começa a enviar as suas coordenadas ao sistema.

Parar de Enviar coordenadas

Este use case reflete a situação em que um utilizador deixa de estar nas proximidades de uma passadeira, deixando assim de partilhar a sua localização.

Receber Notificações

Este use case reflete a situação em que um veículo se aproxima de uma determinada passadeira e começa a receber notificações, vindas do sistema, sobre o estado da passadeira.

Adicionar Peão

Este use case reflete a situação em que se adiciona um peão ao sistema e este começar a poder usufruir das funcionalidades do sistema.

Adicionar Veículo

Este use case reflete a situação em que se adiciona um peão ao sistema e este começar a poder usufruir das funcionalidades do sistema.

Adicionar Passadeira

Este use case reflete a situação em que se adiciona uma passadeira e esta é começada a ter em conta no sistema.

Editar Passadeira

Este use case reflete a situação em que se edita os dados de uma passadeira.

5 Funcionalidades e Requisitos

Neste capítulo vamos abordar as funcionalidades necessárias e básicas do sistema e transformar essas funcionalidades em requisitos funcionais.

5.1 Funcionalidades

Depois de uma análise ao problema apresentado e um debate entre a equipa de desenvolvimento e o cliente chegou-se a um consenso das funcionalidades necessárias do sistema.

Como foi referido na secção **2** existem três entidades relevantes no problema : utilizador veículo, utilizador peão e o próprio sistema. As funcionalidades são também divididas por estas entidades.

Começando pelos utilizadores veículo:

- *Um utilizador veículo tem de ser capaz de receber notificações do sistema sobre o estado da passadeira a que se aproxima. Esta notificação terá de ser instantaneamente apresentada ao utilizador, na medida que terá as informações do estado do semáforo e se existe ou não a presença de qualquer peão.*

Vamos agora apresentar a funcionalidade do utilizador peão. De realçar que esta funcionalidade também se aplica ao utilizador veículo.

- *Um utilizador envia as suas coordenadas ao sistema sempre que se aproxima de uma passadeira. Após este ter atravessado (no caso dos peões) ou passado (no caso dos veículos) a passadeira este deixa de enviar as suas coordenadas.*

Por fim temos o sistema responsável por gerir todo o fluxo de dados e o comportamento da aplicação.

- *O sistema recebe as coordenadas dos utilizadores quando estes se aproximam de uma passadeira.*
- *O sistema pode calcular a distância entre qualquer utilizador e uma passadeira.*
- *O sistema atualiza o estado de uma passadeira sempre que existe qualquer alteração. Esta alteração pode ser a mudança de estado de um semáforo ou a aproximação de um veículo ou peão.*
- *O sistema envia notificações aos utilizadores veículo do estado da passadeira a que ele se aproxima.*
- *O sistema apresentará o número de utilizadores que estão perto de uma passadeira e apresenta a respetiva localização dos mesmos.*
- *O sistema tem um histórico guardado com o histórico de quantos utilizadores passaram por uma determinada passadeira.*

5.2 Requisitos Funcionais

Vamos agora transformar as funcionalidades descritas na secção 5.1 em requisitos, aplicando o formato de *cartão de Volere*.

Req. #: 1 | **Tipo de Requisito:** Requisito Funcional | **Entidade:** Utilizador

Descrição: Qualquer utilizador deve ser capaz de enviar as coordenadas ao sistema quando se encontra a menos de 10m de uma passadeira.

Razão: Desta forma o sistema reconhece que um utilizador (veículo ou peão) se aproxima de uma passadeira.

Critério de ajuste: Depois de o utilizador comunicar as coordenadas ao sistema, este deverá adaptar o seu comportamento consoante esta nova informação.

Satisfação do Consumidor: ⋆⋆⋆⋆⋆ **Insatisfação do Consumidor:** ⋆⋆⋆⋆⋆

Prioridade: Obrigatório **Conflitos:** Nenhum

Documentação de Suporte: Nenhuma **Histórico:** Criado em 2 de abril de 2020

Req. #: 2 | **Tipo de Requisito:** Requisito Funcional | **Entidade:** Utilizador Veículo

Descrição: O utilizador veículo deverá ser capaz de receber notificações do sistema sobre o estado da passadeira a que se aproxima. Este estado inclui a informação da (in)existência de peões e o sinal de semáforo atual.
Razão: Desta forma o utilizador veículo sabe se é seguro ou não passar pela passadeira.

Critério de ajuste: Depois de o utilizador veículo receber a notificação do sistema, esta abrir-se-á e o utilizador saberá o estado atual da passadeira.

Satisfação do Consumidor: ⋆⋆⋆⋆⋆ **Insatisfação do Consumidor:** ⋆⋆⋆⋆⋆

Prioridade: Obrigatório **Conflitos:** Nenhum

Documentação de Suporte: Nenhuma **Histórico:** Criado em 2 de abril de 2020

Req. #: 3 | **Tipo de Requisito:** Requisito Funcional | **Entidade:** Sistema

Descrição: O sistema deve ser capaz de receber as coordenadas dos utilizadores e adaptar o seu comportamento consoante os dados recebidos.

Razão: O sistema deverá ser capaz de analisar o estado de uma passadeira consoante a informação(coordenadas) que recebe e tomar a decisão de que tipo de notificação enviar ao utilizador-veículo.

Critério de ajuste: O sistema recebe as coordenadas e envia à camada de back-end para ser tomada uma decisão.

Satisfação do Consumidor: ⋆⋆⋆⋆⋆ **Insatisfação do Consumidor:** ⋆⋆⋆⋆⋆

Prioridade: Obrigatório **Conflitos:** Nenhum

Documentação de Suporte: Nenhuma **Histórico:** Criado em 2 de abril de 2020

Req. #: 4 | **Tipo de Requisito:** Requisito Funcional | **Entidade:** Sistema

Descrição: O sistema pode calcular a distância entre qualquer utilizador e uma passadeira.

Razão: O sistema deverá ser capaz de calcular a distância entre qualquer utilizador e uma dada passadeira para poder decidir que notificação enviar ao utilizador-veículo.

Critério de ajuste: O sistema recebe as coordenadas e calcula a distância até uma determinada passadeira.

Satisfação do Consumidor: ⋆⋆⋆⋆⋆ **Insatisfação do Consumidor:** ⋆⋆⋆⋆⋆

Prioridade: Obrigatório **Conflitos:** Nenhum

Documentação de Suporte: Nenhuma **Histórico:** Criado em 2 de abril de 2020

Req. #: 5 | **Tipo de Requisito:** Requisito Funcional | **Entidade:** Sistema

Descrição: O sistema atualiza o estado de uma passadeira sempre que existe qualquer alteração. Esta alteração pode ser a mudança de estado de um semáforo ou a aproximação de um veículo ou peão.

Razão: O sistema deverá ser capaz de se adaptar consoante as informações que recebe. Para tal ele atualiza o estado de uma passadeira sempre que recebe dados que a incluam.

Critério de ajuste: O sistema atualiza o estado da passadeira e envia ntoifcação aos utilizadores-veículo que se aproximam dela.

Satisfação do Consumidor: ⋆⋆⋆⋆⋆ **Insatisfação do Consumidor:** ⋆⋆⋆⋆⋆

Prioridade: Obrigatório **Conflitos:** Nenhum

Documentação de Suporte: Nenhuma **Histórico:** Criado em 2 de abril de 2020

Req. #: 6 | **Tipo de Requisito:** Requisito Funcional | **Entidade:** Sistema

Descrição: O sistema envia notificações aos utilizadores veículo do estado da passadeira a que ele se aproxima.

Razão: O sistema é responsável por atualizar os utilizadores-veículo com o estado da passadeira.

Critério de ajuste: O utilizador veículo recebe a notificação do estado da passadeira sabendo se é seguro continuar ou não.

Satisfação do Consumidor: ⋆⋆⋆⋆⋆ **Insatisfação do Consumidor:** ⋆⋆⋆⋆⋆

Prioridade: Obrigatório **Conflitos:** Nenhum

Documentação de Suporte: Nenhuma **Histórico:** Criado em 2 de abril de 2020

Req. #: 7 | **Tipo de Requisito:** Requisito Funcional | **Entidade:** Sistema

Descrição: O sistema apresentará o número de utilizadores que estão perto de uma passadeira e apresenta a respetiva localização dos mesmos.

Razão: Assim os utilizadores podem ver a quantidade de outros utilizadores que estão perto da passadeira.

Critério de ajuste: O sistema mostra aos utilizadores quantos utilizadores estão à volta de uma determinada passadeira e as respetivas localizações.

Satisfação do Consumidor: ⋆⋆⋆⋆⋆ **Insatisfação do Consumidor:** ⋆⋆⋆⋆⋆

Prioridade: Obrigatório **Conflitos:** Nenhum

Documentação de Suporte: Nenhuma **Histórico:** Criado em 2 de abril de 2020

Req. #: 8 | **Tipo de Requisito:** Requisito Funcional | **Entidade:** Sistema

Descrição: O sistema tem um histórico guardado com o histórico de quantos utilizadores passaram por uma determinada passadeira.

Razão: O sistema tem um histórico relacionado com cada passadeira para fins estatísticos.

Critério de ajuste: O sistema terá associado um conjunto de dados estatísticos para cada passadeira registada no sistema.

Satisfação do Consumidor: ⋆⋆⋆⋆⋆ **Insatisfação do Consumidor:** ⋆⋆⋆⋆⋆

Prioridade: Obrigatório **Conflitos:** Nenhum

Documentação de Suporte: Nenhuma **Histórico:** Criado em 2 de abril de 2020

6 WebRatio

Vamos agora expôr a nossa abordagem ao problema na ferramenta WebRatio.
No que diz respeito a esta ferramenta temos um conjunto de três tipos de vistas:

- ***Domain Model*** **-** modelo de dados que será a base da aplicação que vamos implementar. De realçar que este modelo é um modelo *Entity-Relationship*, que ,tal como o nome indica, é um modelo baseado nas entidades e nas relações entre elas. Este modelo também estará conectado à base de dados.
- ***Site Views*** **-** representam partes autónomas da aplicação, suportando as funcionalidades de um tipo de utilizador específico.
- ***Module Definitions*** **-** representam uma parte da lógica de negócio da aplicação: processamento de dados.

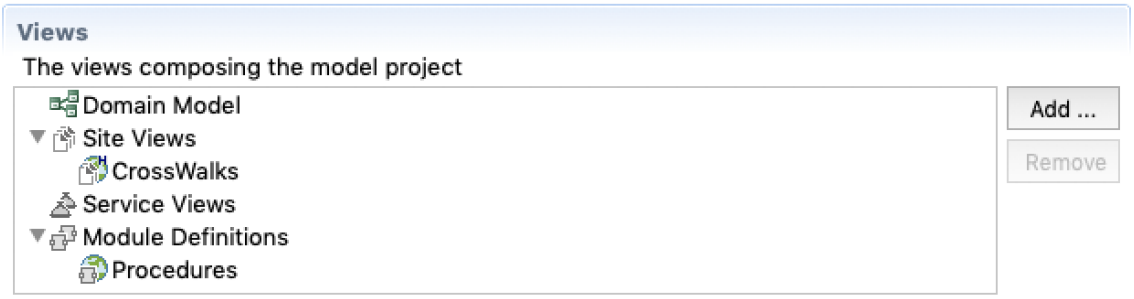

Figura 3: WebRatio - *Overview*

6.1 Domain Model

Tal como foi referido anteriormente, o modelo de domínio representa as entidades e os relacionamentos que eles têm entre as demais.

Este modelo foi construído de forma a poder responder a todas funcionalidades do problema apresentado. Por essa razão e por que também este modelo servirá como ligação à base de dados, tivemos de pensar além das entidades e procuramos representar todas as relações que estas poderiam tomar. Assim foram necessárias criar novas entidades que representam a ponte entre outras. Esta abordagem foi utilizada devido ao esquema da base de dados e à cardinalidade que as entidades têm nos seus relacionamentos, que achamos que melhor responderia a todos os pedidos, tendo sido obtido o seguinte modelo:

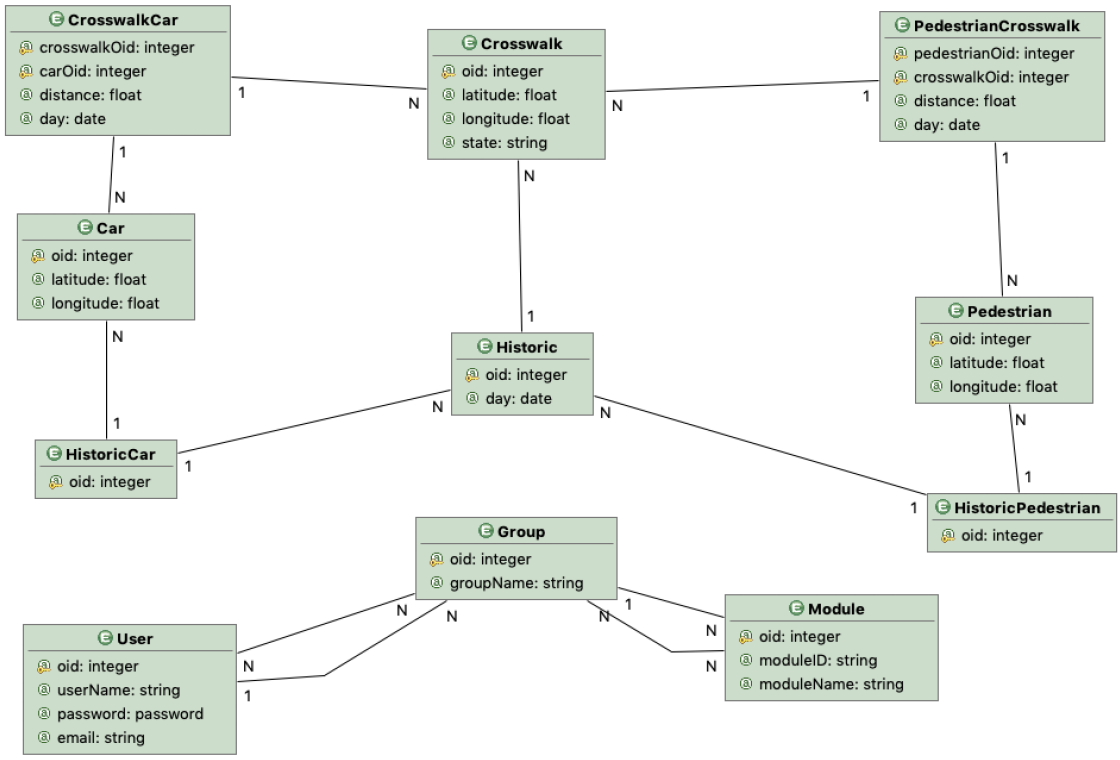

Figura 4: WebRatio - *Domain Model*

Tendo em conta a Figura 4 temos que explicar algumas decisões e expor alguns detalhes.

Comecemos por realçar as tabelas *PedestrianCrosswalk* e *CrosswalkCar*. Estas tabelas representam o tempo real na medida que a informação presente nessas tabelas corresponde ao estado atual das passadeiras. Por outras palavras se essas tabelas tiverem informação quer dizer que neste preciso momento há carros ou peões perto da passadeira.

Quanto ao histórico decidimos que este será divido por dias. Dentro de cada dia terá uma lista de peões e outra lista de veículos que passaram por aquela passadeira naquele dia.

Temos que referir também que depois de analisarmos o problema proposto, decidimos focar-nos apenas no fundamental do problema e soluciona-lo e por isso optamos por manter o mínimo de informação possível, reduzindo-a apenas ao necessário. Podíamos facilmente adicionar atributos a algumas tabelas, como por exemplo, adicionar a matricula, marca e modelo de um veículo. Porém isto apenas ia aumentar a informação a ser armazenada e como não ajuda em nada na solução, decidimos omitir esse tipo de dados.

Por fim temos de anotar que a tabela *Car* corresponde a qualquer veículo registado no sistema, e não apenas aos carros.

6.2 Site Views

Mais uma vez, as *Site Views* representam partes autónomas da nossa aplicação. Como vemos pela Figura 20 apenas temos uma única *Site View*, porém, esta é composta por três áreas diferentes:

- ***Crosswalks***
- ***Pedestrians***
- ***Vehicles***

6.2.1 Crosswalks

Área principal da nossa aplicação. Esta área é responsável por apresentar todas as passadeiras registadas na aplicação, assim como as suas coordenadas e o seu estado: se é possível passar ou não. É a partir desta página que será possível também adicionar,editar, eliminar ou consultar os dados de qualquer passadeira. Na consulta dos dados além das coordenadas e estado, é nos apresentado os peões e os veículos que estão perto da passadeira e diz-nos a distância a que eles estão dela. Podemos também consultar o histórico de peões e veículos que já passaram por ela. No projeto WebRatio vemos que está área é composta por:

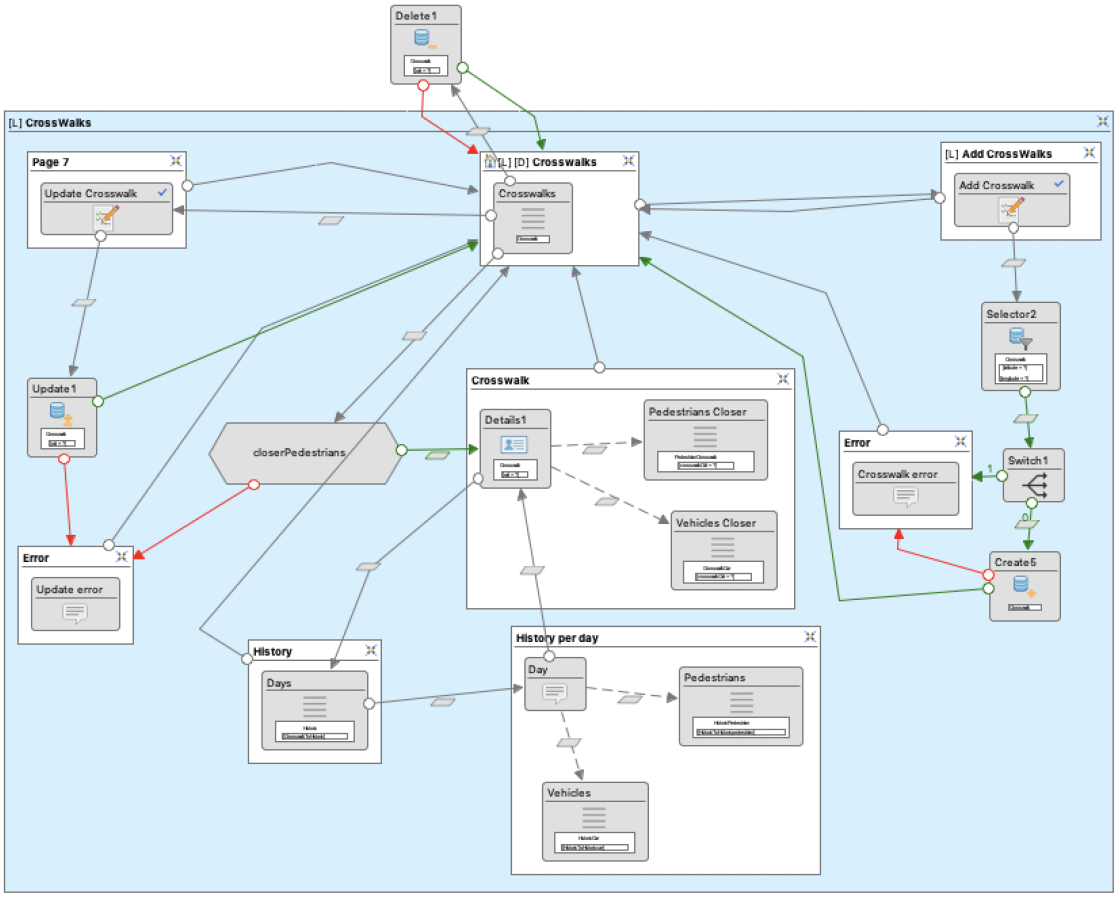

Figura 5: WebRatio - *Site View - Crosswalks*

Vamos agora mostrar como esta área ficou dividida após correrm o projeto.

Começemos então pela página principal que apresentará todas as passadeiras registadas com as suas coordenadas e respetivo estado:

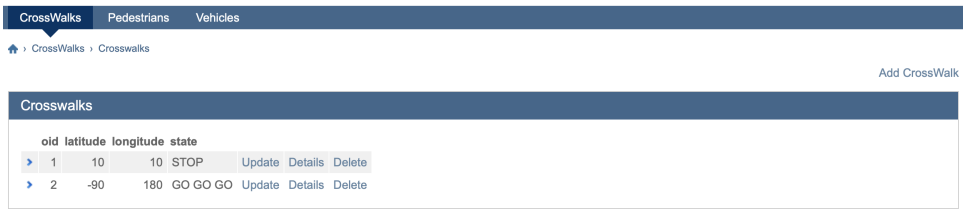

Figura 6: *Site View - Lista de passadeiras*

A partir da view representada na Figura 6 podemos ver os detalhes de uma passadeira, editar uma passadeira já existente ou adicionar uma nova.

Começando por adicionar uma passadeira:

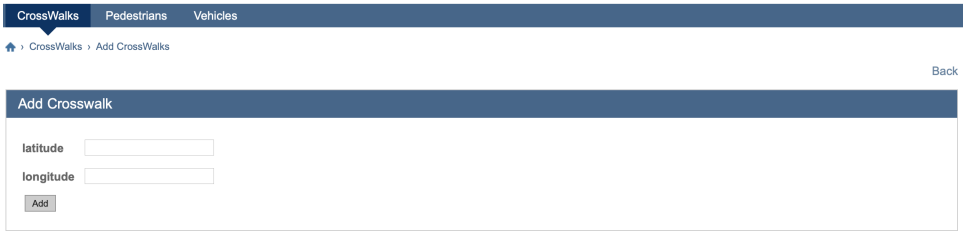

Figura 7: *Site View - Adicionar passadeira*

Quanto ao editar temos:

Figura 8: *Site View - Editar passadeira*

Por fim temos os detalhes de uma passadeira:

Figura 9: *Site View - Detalhes da passadeira*

A partir desta Figura 9 vemos que podemos consultar o histórico através do link *History* que está na primeira caixa. Este link leva-nos a uma página que mostra os dias todos que a passadeira foi atravessada, quer por veículos, quer por peões:

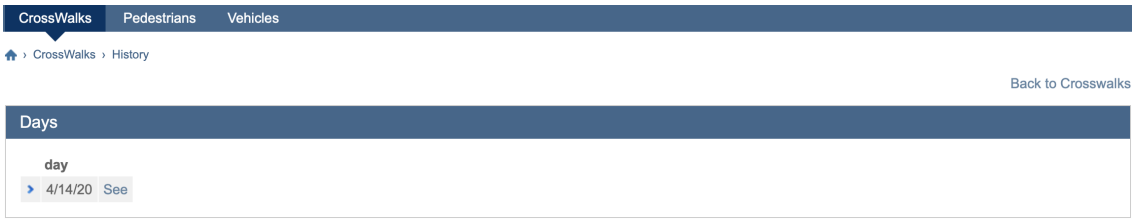

Figura 10: *Site View - Histórico dos dias da passadeira*

Após selecionarmos um dia teremos então a lista de todos os veículos e peões que passaram por ela nesse dia:

Figura 11: *Site View - Histórico da passadeira num dia*

6.2.2 Pedestrians

Área dirigida aos peões que utilizam a aplicação. Nesta vamos encontrar uma lista de peões e as suas coordenadas, sendo possível adicionar peões e atualizar os seus dados (coordenadas).

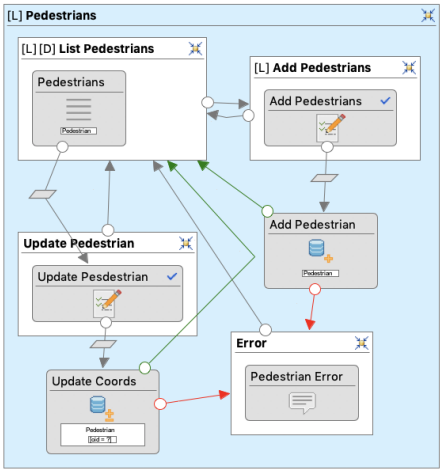

Figura 12: WebRatio - *Site View: Pedestrians*

Após corremos o projeto e dirigirmo-nos à área dos peões deparamo-nos com uma lista de todos os peões registados e suas respetivas coordenadas:

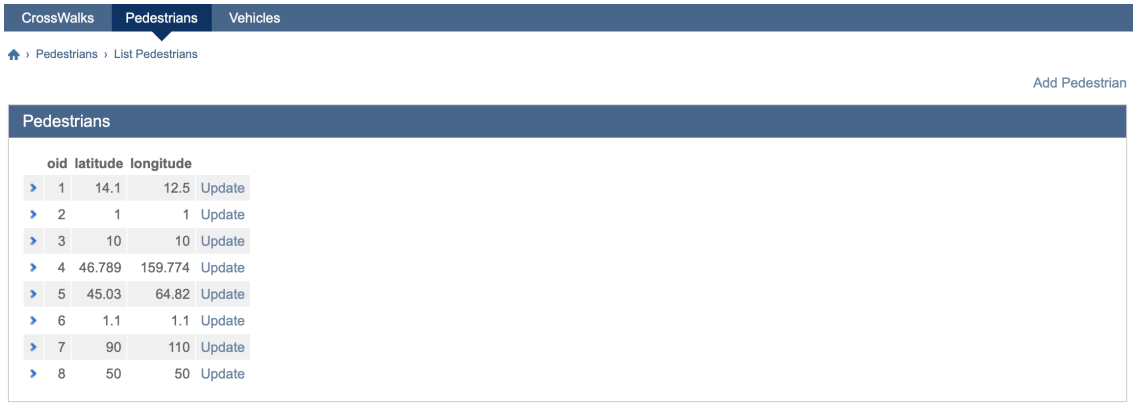

Figura 13: *Site View - Lista de peões*

Da view representada pela Figura 13 podemos optar por adicionar um novo peão ou editar um peão já existente.

Se optarmos por adicionar:

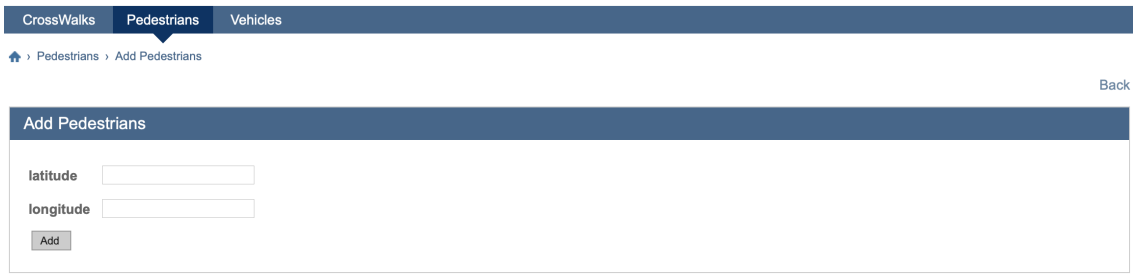

Figura 14: *Site View - Adicionar peão*

Enquanto se optarmos por editar:

Figura 15: *Site View - Editar peão*

6.2.3 Vehicles

Área dirigida aos veículos que tem um funcionamento idêntico à área dirigida aos peões.

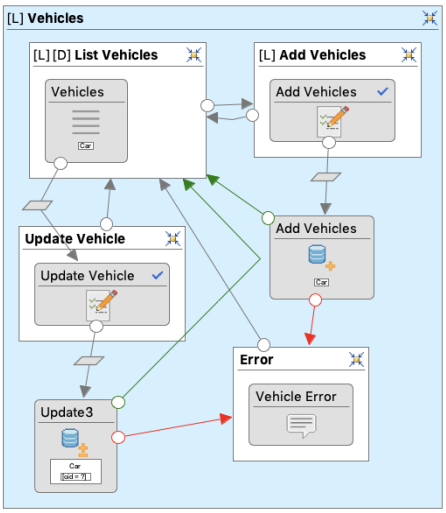

Figura 16: WebRatio - *Site View: Vehicles*

Após o projeto estar a correr e dirigirmo-nos também à área dos veículos temos a mesma situação da área dos peões, mas desta vez dirigido aos veículos:

Figura 17: *Site View - Lista de veículos*

Tal como anteriormente, a partir da Figura 17 podemos adicionar ou editar veículos.
Se adicionarmos:

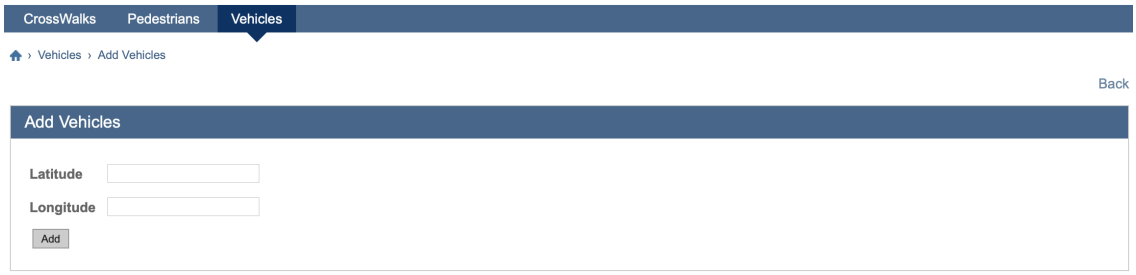

Figura 18: *Site View - Adicionar veículo*

Se editarmos:

Figura 19: *Site View - Editar veículo*

6.3 Module Definitions

Quanto às *module definitions* foi criada um *view*, de nome *Procedures*, que contém duas ações distintas:

- **closerPedestrians -** Esta view é responsável por calcular se existem peões perto de uma determinada passadeira. Se existir, este dir-nos-á quais os peões e as suas respetivas coordenadas e a sua distância à passadeira.
- **closerCars -** Esta view é responsável por calcular se existem veículos perto de uma determinada passadeira. Se existir, este dir-nos-á quais os veículos e as suas respetivas coordenadas e a sua distância à passadeira.

Tal está definido no projeto *WebRatio*:

Figura 20: WebRatio - *Procedures*

6.4 Outros aspetos relevantes

Além de toda a definição de vistas e modelo de domínio na ferramenta *WebRatio* foi necessário algum trabalho a nível de *MySQL* para a base de dados e a aplicação poderem suportar todas as funcionalidades.

Assim é importante realçar as *procedures* e as funções criadas neste nível.

6.4.1 Functions

Começando pelas funções, foi criada uma função que nos permite calcular a distância entre dois objetos, que será a base dos cálculos das distâncias entre as passadeiras e utilizadores. Após a leitura do enunciado podemos facilmente compreender a importância desta função. Para tal usamos uma função já existente do MySQL que recebe dois pointos e calcula a distância entre eles. Essa função é chamada de *ST_Distance_Sphere*.

```
CREATE DEFINER=‘root‘@‘localhost‘ FUNCTION ‘distance‘(latA float,latB float,longA float,
longB float) RETURNS float
BEGIN
Declare distance float;
   select ST_Distance_Sphere(
    point(longA,latA),
    point(longB,latB)
) into distance;
RETURN distance;
END
```

6.4.2 Procedures

Quanto aos procedures temos quatro diferentes, todos eles cruciais para suportar os *module definitions* abordados na secção 6.3. Estes podem ser divididos em dois grupos devido aos seus diferentes comportamentos. Enquanto dois deles são usados para calcular a distância entre uma entidade e uma passadeira, os outros dois usam estes para fazer o cálculo das distâncias para todas as entidades registadas na aplicação.

Começando pelo primeiro grupo

- **closerCar** - Responsável por calcular a distância entre um carro e uma passadeira:

  ```
  CREATE DEFINER=‘root‘@‘localhost‘ PROCEDURE ‘closerCar‘(in cross_id int ,in car_id int)
  ```

  Além disso esta é responsável por adicionar um carro ao histórico da passadeira num determinado dia.

- **closerPedestrian** - Responsável por calcular a distância entre um peão e uma passadeira:

  ```
  CREATE DEFINER=‘root‘@‘localhost‘ PROCEDURE ‘closerPedestrian‘(in cross_id int ,
  in ped_id int)
  ```

  Ta como a anterior, esta também vai inserir o peão ao histórico da passadeira.

Quanto ao segundo grupo temos:

- **closerCars** - Este procedure é responsável por calcular a distância de todos os carros a uma determinada passadeira. Além deste cálculo este procedure também altera o estado da passadeira.

```
CREATE DEFINER=‘root‘@‘localhost‘ PROCEDURE ‘closerCars‘(in cross_id int)
```

- **closerPedestrians** - Este aplica o mesmo processo que o anterior mas dirige-se aos peões:

```
CREATE DEFINER=‘root‘@‘localhost‘ PROCEDURE ‘closerPedestrians‘(in cross_id int)
```

7 Conclusão e Análise Crítica

Concluida a primeira fase deste projeto, podemos afirmar que o mesmo se encontra num excelente caminho para se tornar bastante próspero. Esta afirmação é fundamentada pela avaliação feita pelo grupo que depois de todo o trabalho e esforço estamos motivados e seguros que podemos prosseguir com o desenvolvimento do projeto.

Esperamos agora um desenvolvimento desafiante e bastante edificante em todos os aspetos.