

Departamento de Informática
Mestrado Integrado em Engenharia Informática

Empowering the Citizen Sensor in Smart Cities

Daniel Vieira , A73974

Joel Morais, A70841

Miguel Cunha, A78478

25 de junho de 2019

Conteúdo

1	Resumo	2
2	Introdução	2
2.1	Motivação e Objetivos	2
2.1.1	Smart City	3
2.2	Smart Citizen	4
3	Protótipo	5
3.1	Aplicação Mobile	5
3.1.1	Mockups	5
3.1.2	Front-end	6
3.1.3	Back-end	8
4	Plataforma Web	9
4.1	Front-end	9
4.2	Back-end	11
5	Conclusões e Trabalho futuro	12

1 Resumo

O projeto *Empowering the Citizen Sensor in Smart Cities*, proposto e realizado para a Unidade Curricular de *Laboratório em Engenharia Informática*, tem como objetivo o desenvolvimento de uma aplicação mobile, juntamente com uma plataforma Web e discussão de trabalho futuro. Este projeto tem como base teórica as *Smart Cities*, consideradas áreas urbanas que usam diferentes tipos de aparelhos eletrônicos para recolha de dados, de modo a gerir melhor os recursos da própria cidade. O nosso objetivo é fazer com que cada cidadão desta cidade se torne um Smart Citizen, ou seja, que através da recolha de dados destes mesmos seja possível gerir também mais eficientemente a cidade. Para isso, temos uma aplicação Mobile desenvolvida em Android Studio, que faz a recolha dos dados dos cidadãos através da API da Google Fit. Foi desenvolvida também uma plataforma Web com o mesmo objetivo, mas com a possibilidade de introduzir dados mais pessoais sobre cada cidadão. Com a análise destes dados (utilizando técnicas de Machine Learning), é possível criar Smart Citizens através de várias situações que serão abordadas.

2 Introdução

2.1 Motivação e Objetivos

Sabemos que ao longo dos anos tem havido um crescimento populacional muito acrescido. Há 200 anos havia apenas 3 cidades com mais de um milhão de habitantes, sendo estas Londres, Tóquio e Pequim. Hoje em dia já existem mais de 442 metrópoles que ultrapassam os 10 milhões de habitantes, sendo esta diferença bastante elevada. Com estes dados foi possível reconhecer que metade da população urbana vive em meios urbanos, e a ONU prevê que em 2030 essa percentagem aumente para cerca de 70%. É previsível que com este aumento cada vez maior da população mundial surjam bastantes problemas, como no trânsito, nos níveis de poluição, falta de moradias e acessos à saúde...

Com o mundo em constante evolução, todos esses problemas podem ser minimizados usando a **Tecnologia**, tecnologia essa que tem de acompanhar este mesmo mundo.

É aqui que entram as *Smart Cities*, que utilizam avanços tecnológicos de modo a tornar os meios urbanos mais eficientes e mais facilmente habitáveis, além da melhoria considerável de vida que podemos ter, usufruindo desta tecnologia inovadora.

2.1.1 Smart City

Definição

O conceito de "Cidade Inteligente" começou a tornar-se bastante popular há cerca de duas décadas, seja na comunidade científica, seja em diferentes políticas internacionais. Para melhor compreender o conceito destas cidades, é preciso primeiro reconhecer que estas mesmas são consideradas uma chave importante para um futuro tecnológico. Elas irão ter um papel importante em termos sociais, económicos e mesmo ambientais.

Na sua essência, uma cidade inteligente consiste em re-desenvolver uma área ou cidade usando diferentes tecnologias de informação e comunicação, melhorando assim a qualidade de diferentes serviços urbanos, em termos de energia, transporte, serviços públicos, etc.

Uma cidade inteligente é desenvolvida quando certo tipo de tecnologias são implantadas para mudar a natureza e a economia da infraestrutura das diferentes áreas urbanas.

As aplicações *IoT* baseadas em *Cloud* recebem e gerem dados em tempo real para ajudar certas empresas e os residentes a tomarem melhores decisões, de modo a melhorarem a sua qualidade de vida.

Este tipo de decisões tem como objetivo melhorar o congestionamento de tráfego, a interrupção de energia, a ligação à Internet, entre outros, enquanto os custos estão a ser amenizados.

Um exemplo bastante conhecido de uma aplicação tecnológica são os "veículos autónomos", que são construídos de modo a comunicarem com parquímetros e locais de carregamento EV, indicando ao condutor o ponto mais próximo disponível

Porque precisamos de Smart Cities ?

A população mundial está a crescer de uma maneira absurda e contínua, existindo possivelmente mais 2.5 biliões de pessoas em cidades urbanas nos próximos 30 anos, segundo a Gemalto.

Já existem diferentes cidades que sofrem de superlotação, piorando cada vez mais a sua situação, como Nova Iorque, Tóquio e Londres.

De acordo com o Departamento de Transporte do Reino Unido, a Grã-Bretanha é considerada um dos países mais congestionados em todo o mundo.

Por motivos como estes apresentados, as cidades são automaticamente forçadas a encontrar diferentes maneiras de remover certo tipo de pressões que os moradores e os passageiros enfrentam, através de políticas inteligentes.

As Smart Cities são seguras ?

A segurança é uma das principais preocupações em qualquer cidade, e com o fato de ser acrescentada tecnologia, esta preocupação torna-se ainda maior.

Para além que é conhecido que existem cada vez mais *cyber crimes*, juntamente com roubo de dados, ou seja, as cidades inteligentes têm de estar preparadas para enfrentar estas possíveis

ameaças.

Como foi mencionado acima, espera-se que as tecnologias ajudem os cidadãos a tomar decisões melhores. De facto, a introdução de tecnologias inteligentes tem o potencial de reduzir as fatalidades e melhorar os tempos de resposta a emergências.

Os números do relatório das cidades inteligentes do MGI descobriram que as tecnologias inteligentes poderiam reduzir as fatalidades em oito a 10 por cento e reduzir os incidentes criminais em 30 a 40 por cento.

Podemos concluir que a adoção de várias tecnologias conectadas entre si vem com certos riscos. As cidades precisam de integrar soluções que garantam um ambiente urbano seguro e protegido.

2.2 Smart Citizen

A nossa proposta será a criação de Smart Citizens, ou seja, cidadãos integrados em Smart Cities. Estes cidadãos iriam ser munidos de dispositivos conectados a uma mesma rede, com o objetivo de otimizar a eficiência de certas operações, ou seja, se eles têm os dispositivos todos conectados à mesma rede e com certo tipo de sensores, existe uma forte possibilidade de, utilizando os dados recolhidos destes sensores, ajudar as pessoas a tornarem-se mais ativas, reativas e proativas nesta comunidade inteligente.

Tal como dito, irão ser recolhidos certo tipo de dados a estes cidadãos, através do uso de sensores, como por exemplo pulseiras eletrónicas. Dados esses físicos, biológicos, sociais e biométricos.

Estes sensores irão estar a recolher os dados em tempo real, por exemplo, quando um cidadão está a fazer exercício físico ou mesmo a dormir, certos tipos de dados irão estar a ser recolhidos, como as horas de sono, a distância que o cidadão percorreu ou o número de passos que deu, entre outros. Alguns desses vão estar a ser utilizados por nós.

Visto que o cidadão não é obrigado a disponibilizar os seus dados, tem de existir uma sensibilização prévia, explicando a importância da recolha destes mesmos, ajudando no desenvolvimento da *Smart City*, através (por exemplo) da criação de perfis sobre certas zonas da cidade.

3 Protótipo

3.1 Aplicação Mobile

Como primeira fase, foi-nos proposto que desenvolvêssemos uma aplicação mobile, com o intuito de fazer uma recolha dos dados dos cidadãos. Esta recolha irá ser feita recorrendo à API da Google Fit. Teoricamente, numa cidade inteligente, os dados iriam ser provenientes de dispositivos e sensores fornecidos, neste caso, para efeitos de teste iremos recorrer à Google Fit.

Após esta recolha estar completa, irá ser desenhada a interface de modo a representar estes dados através de diferentes métodos de visualização, como gráficos de barras ou pie charts.

O utilizador (ou seja o *Smart Citizen*), após ter feito o seu login, é reencaminhado para a página principal, onde estarão representados os seus dados, sendo eles o número de passos, a distância percorrida, a média dos batimentos cardíacos, o seu peso, a quantidade de calorias gastas e as atividades realizadas. O utilizador tem também a possibilidade de selecionar uma página de definições onde tem a opção de ativar/desativar métricas que não esteja interessado em visualizar.

3.1.1 Mockups

Para uma primeira abordagem decidimos realizar 3 *mockups*, que serão explicados de seguida.

Na primeira figura, está representada a nossa página de *login*, onde o utilizador terá a opção de entrar na aplicação através do seu e-mail ou através da sua conta do facebook. Para além destas funcionalidades queremos ter um logo nesta página que será mais tarde definido. Na segunda, podemos observar a nossa página principal, onde irá estar representada toda a informação do utilizador, podemos ver todas as métricas relevantes para a nossa recolha, sendo elas a distância e o nº de passos representados por um gráfico de barras relativos a um mês. Os número de batimentos cardíacos (coração no mockup), o número de calorias (pie chart), a variação do peso (gráfico de linhas) e as atividades realizadas. Existe também a possibilidade de aceder à página das definições. A terceira figura é referente às definições , onde o utilizador pode ativar / desativar as métricas acima faladas.



Figura 1: Mockups

3.1.2 Front-end

Para o nosso front-end iremos utilizar o IDE *Android Studio*, feita para desenvolvimento Android e utilizada principalmente pela Google, sendo a linguagem utilizada o XML.

Interface

O nosso objetivo seria seguir os mockups como base para a nossa interface. Como podemos observar pela 1^a figura apresentada de seguida, o login contém os traços pretendidos, optámos por retirar o login feito por facebook, visto que não faria sentido o mesmo estar aqui representado, sendo substituído pelo login pela Google, onde poderemos ir buscar diretamente os dados à API da Google Fit. Se o login fosse feito pelo facebook não teríamos maneira de conseguir obter as métricas.

Na 2^a figura está representada a nossa página de registo. Caso o utilizador deseje fazer o seu registo através desta via, a sua conta irá ficar registada e terá mais utilidade na plataforma Web, de modo a existir uma maior interação entre a plataforma e o utilizador (que será explicado mais adiante).

Na 3^a figura podemos observar a nossa página de definições, que se encontra praticamente igual à página definida nos mockups, existindo um switch para cada métrica, havendo a possibilidade de ativar / desativar as respetivas métricas.

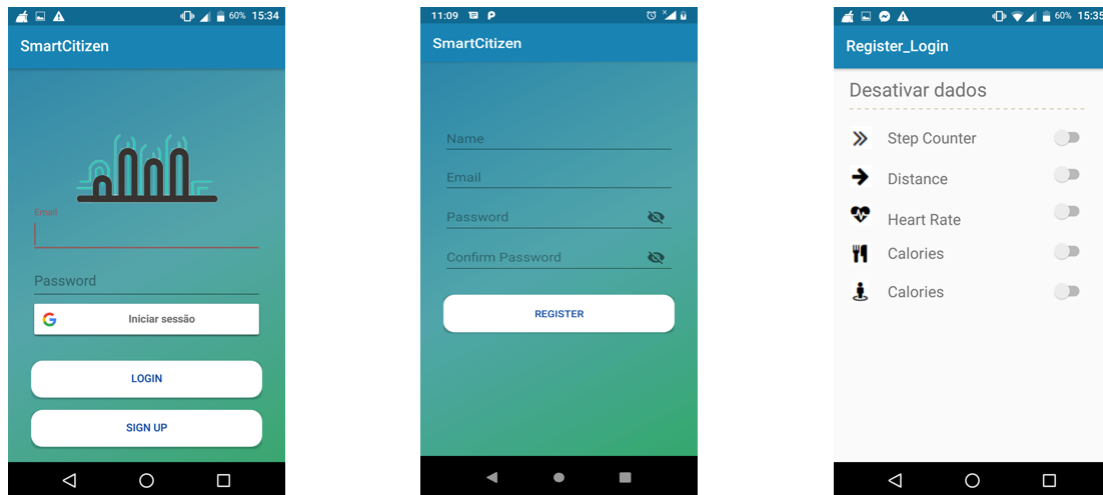


Figura 2: Mockups

De seguida podemos observar exemplos dos nossos dados, na primeira figura está representada uma variação do peso ao longo de cada semana, podendo o utilizador observar a sua variação. Na segunda estão representados o número de passos que o utilizador deu por cada dia. Na última figura é possível ver o número de calorias gastas / ingeridas e se teve um défice positivo ou negativo.

Para o utilizador mudar de dados foi optada uma solução de swipe, em vez de scroll down como demonstrado nos mockups.

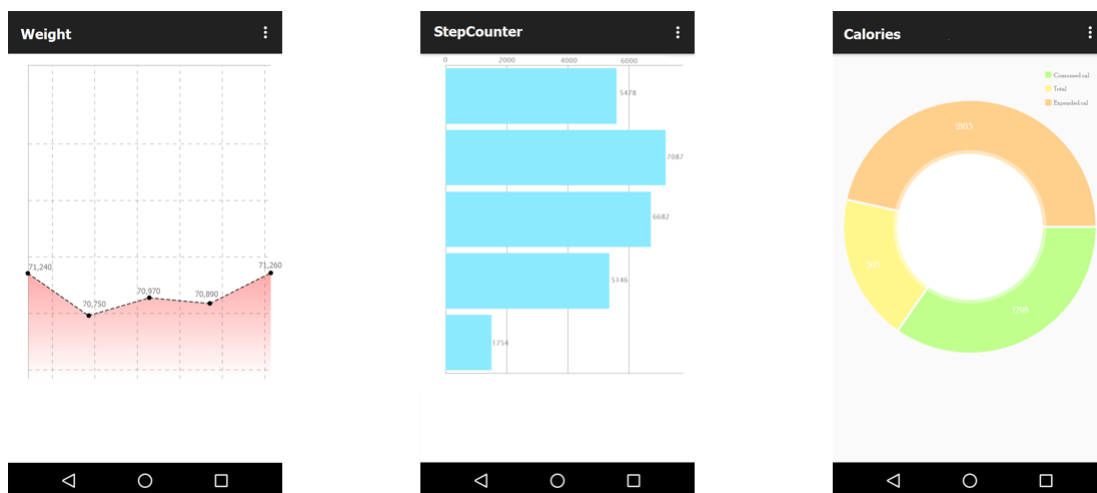


Figura 3: Dados de utilizador

3.1.3 Back-end

Para o back-end foi utilizado também o *Android Studio*. Os dados e a autenticação vão ser guardados através da plataforma *Firebase*.

Aqui o nosso objetivo será, numa primeira abordagem, recolher todos os dados do utilizador. Para isso, seguimos a documentação da Google Fit, de modo a sabermos que métricas seriam necessárias, enumeradas de seguida:

1. **TYPE_STEP_COUNT_DELTA**: Passos totais realizado pelo utilizador num espaço de 24 horas;
2. **TYPE_HEART_RATE_BPM**: Batimentos cardíacos médios do utilizador num espaço de 24 horas;
3. **TYPE_HEIGHT**: Altura do utilizador;
4. **TYPE_WEIGHT**: Peso do utilizador a cada semana;
5. **TYPE_CALORIES_EXPENDED**: Calorias gastas pelo utilizador num espaço de 24 horas;
6. **TYPE_DISTANCE_DELTA**: Distância total percorrida pelo utilizador num espaço de 24 horas;

Podemos ver na imagem seguinte um exemplo de como estão a ser recolhidos os passos totais.

```
hc.readDailyTotal(DataType.TYPE_STEP_COUNT_DELTA)
    .addOnSuccessListener(
        (OnSuccessListener) (dataSet) -> {
            total[0] =
                dataSet.isEmpty()
                    ? 0
                    : dataSet.getDataPoints().get(0).getValue(Field.FIELD_STEPS).asInt();
            System.out.println("Total steps: " + total[0]);
        })
    .addOnFailureListener(
        (e) -> {
            System.out.println("There was a problem getting the step count." + e);
        });
```

Figura 4: Dados de utilizador

Após a recolha total de todos os dados, o nosso objetivo é passar estes valores para um ficheiro .json, diretamente para o telemóvel, de modo a conseguirmos aceder aos dados que serão utilizados para criar os nossos respetivos gráficos de barras, pie charts e gráficos de linhas.

Estes gráficos foram criados recorrendo à library *MPAndroidChart*, criada especificamente para visualização gráfica.

4 Plataforma Web

4.1 Front-end

Para a parte do front-end decidimos usar HTML em conjunto com CSS.

Inicialmente, criamos uma pagina de login com o logo da nossa aplicação (logo este que estará presente em todas as paginas). Esta página tem todas as funcionalidades necessárias para o login, tais como: campos para username e password e um botão que nos permite fazer login através da google, usando uma API da google.

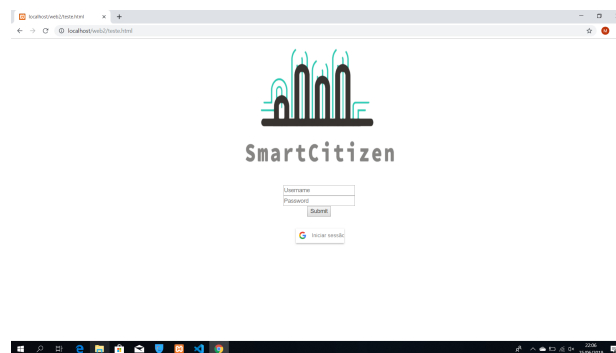


Figura 5: Pagina Inicial

Após isto, temos uma pagina inicial com vários botões que nos permitem aceder aos diversos gráficos para cada uma dos valores recolhidos ao longo dos últimos 5 dias. Temos também um botão que nos permite aceder á página das definições.

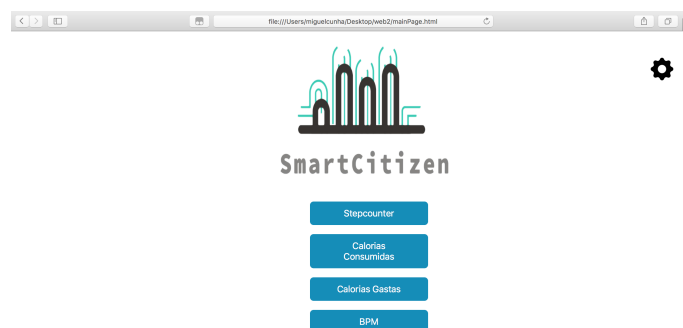


Figura 6: Pagina Inicial

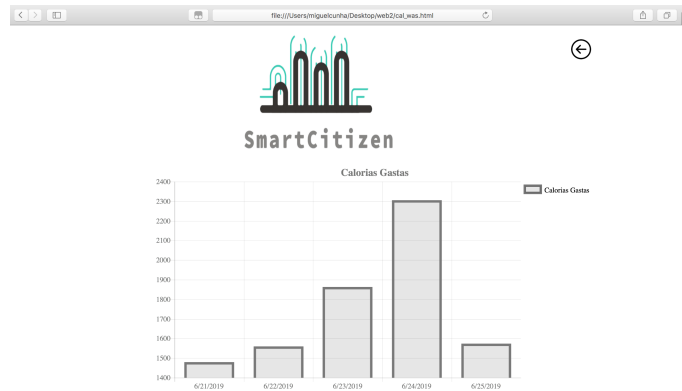


Figura 7: Grafico das calorias Gastas

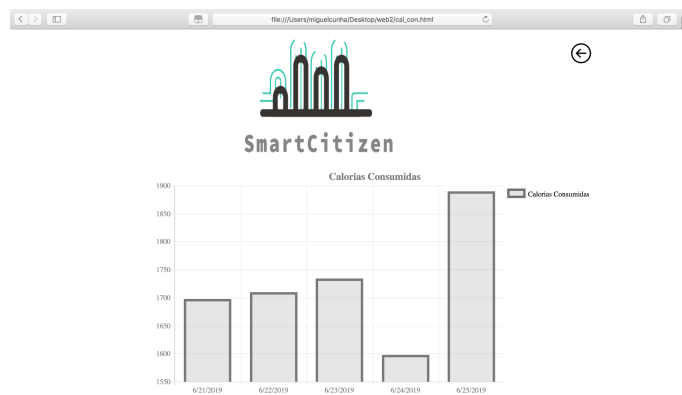


Figura 8: Grafico das calorias consumidas

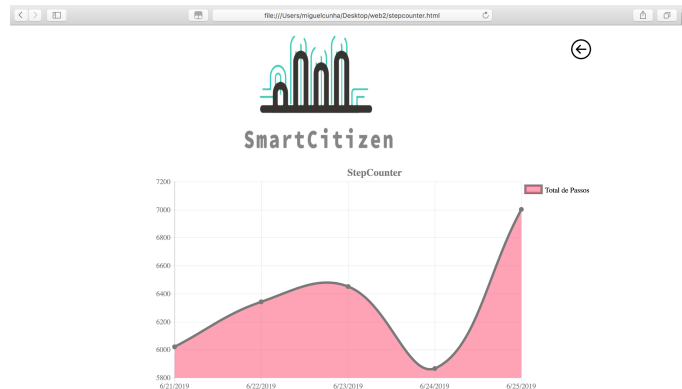


Figura 9: StepCounter

Na página das definições é possível ao utilizador seleccionar o tipo de problemas que este tem, de forma a que nos futuro possamos analisar este caso e determinar se este condutor é ou não um condutor de risco.



Figura 10: Definições

4.2 Back-end

Para a parte do back-end decidimos usar HTML em conjunto com JS.

Para que seja possível executar o login através da google, tivemos de usar a função *onSignIn* fornecida pela google. Esta função permite-nos fazer o login e obter vários dados relativamente ao utilizador que executou o login, tais como: id da conta, nome, imagem e etc...

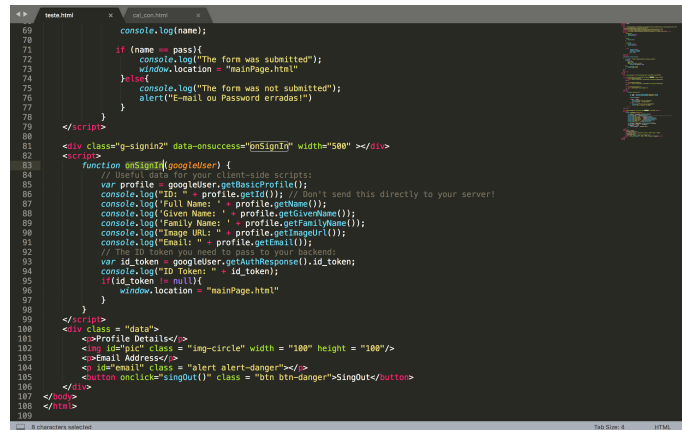


Figura 11: onSingIn

Na parte dos gráficos, usamos uma biblioteca do AJAX chamada *Chart*. Esta biblioteca permite-nos facilmente criar gráficos com os dados que bem desejamos. E tal como vimos em cima, conseguimos criar diversos tipos de gráficos.

Mas visto que não conseguimos ir buscar os dados à GOOGLE FIT, decidimos criar gráficos com dados estáticos para demonstrar que, caso conseguíssemos obter os dados que queríamos, seria bastante simples implementar os gráficos.

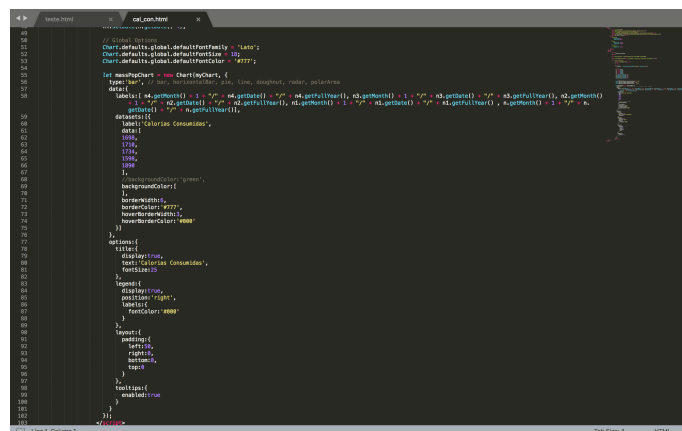


Figura 12: Exemplo de uma função de um gráfico

5 Conclusões e Trabalho futuro

O objetivo deste projeto não se baseia apenas na realização de trabalho de "Engenharia", ou seja, na realização de uma aplicação móvel e de uma página Web de modo a existir melhor

visualização de dados.

Também é para contribuir para as Smart Cities, fazendo com que os cidadãos tenham para isso um papel mais ativo.

Com a contribuição de dados extra na parte Web, os cidadãos estão a contribuir para uma recolha muito mais eficiente dos dados. Com estes dados podemos fazer certo tipo de tratamento de modo a ajudar as cidades, como dito anteriormente.

Uma das abordagens seria utilizar estes dados de modo a conseguirmos prever condutores de risco dentro deste tipo de cidades. Com certo tipo de informação, por exemplo, relativamente às diabetes de um cidadão, ou se possui problemas cardíacos, juntamente com um número elevado de informação extra. Deste modo, a nossa aplicação poderia fazer certo tipo de recomendações aos condutores, dizendo para não seguir uma certa estrada ou para não conduzir a certas horas. Tudo isto através da nossa APP, e com conexão ao Google Maps.

Existe uma infinidade de possibilidades em relação ao que se pode fazer com o tratamento devido dos dados.

Esta abordagem em relação aos condutores de risco pode também ser utilizada como um sistema classificativo para seguradoras, em que o preços relativos aos seguros seria também inflacionado de acordo com certos aspetos clínicos dos cidadãos.

Como trabalho futuro, ainda existem uma quantidade de tarefas a realizar. No que toca à aplicação mobile, alguns dados, como o batimento cardíaco (devido à falta de sensores), o peso e a altura (devido a problemas com permissões da Google) não foram recolhidos com sucesso. Este será o primeiro problema a resolver, o segundo será recolher os dados todos provenientes do ficheiro Json, apenas estamos a ir buscar os dados referentes ao dia atual, e não a dias anteriores. Com esta recolha, necessitamos de fazer o resto dos gráficos (algo que é bastante rápido, mas seria necessário ter dados anteriores, algo que não foi possível). O front-end está pronto, seja a página de login, das definições, ou a página inicial.

No que toca à plataforma Web, estamos com os mesmos problemas que na aplicação mobile. Como melhorias, seria necessário adicionar uma página personalizada para cada utilizador, tal como variadas opções para o utilizador responder, como o caso das diabetes ou dos problemas cardíacos, como falado anteriormente.

Tendo estes aspetos em consideração, podemos concluir que ainda existem diversos pontos que devem ser tratados antes de darmos o projeto por concluído. Os principais provêm da obtenção dos dados por parte da Google, seria necessário termos acesso a contas com diversos dados já existentes, visto que apenas utilizámos uma conta da Google Fit criada por nós, não tendo assim acesso a certos dados fulcrais.