Relatório técnico de desenvolvimento – App Vitoria

Jorge Sá Silva, Marcelo Fernandes, João Sobral, Afonso Nunes, André Rodrigues, Fernando Boavida.

Universidade de Coimbra, Faculdade de Ciências e Tecnologia.

1. Introdução

A chegada da Internet das Coisas (IoT) está a tornar-se uma realidade: uma rede global de sensores, telemóveis, computadores e outros dispositivos que podem analisar o mundo com exatidão e comunicar entre si em tempo real. De acordo com recentes estudos, espera-se que o mercado de comunicação Machine-to-Machine (M2M) sem fios represente quase 196 mil milhões de dólares de lucro em 2025, seguindo uma taxa de crescimento anual composta de 21% durante os próximos anos. Da utilização destes diversos elementos computacionais surge o conceito de sistema cíber-físico (CPS), que consiste na percepção e controlo de fenómenos físicos através destas redes de dispositivos interligados, que trabalham em conjunto para atingir objetivos comuns. Estes sistemas cíber-fisicos representam uma confluência das áreas de robótica, das redes de sensores sem fios, da computação móvel e da IoT para conseguir ambientes altamente monitorizados, e facilmente controláveis e adaptáveis.

Apesar destas ferramentas interligadas e inteligentes comunicarem umas com as outras sem intervenção humana, a tecnologia é sempre feita por humanos e para humanos. Desta forma, para promover a criação de sistemas que sejam úteis à pessoa comum, não basta simplesmente considerar a heterogeneidade e a integração de ferramentas inteligentes. No entanto, fora da área de "e-health", cujo objetivo principal é o de monitorizar pacientes, existe ainda pouco trabalho científico que se foque nos efeitos do contexto humano na malha de controlo dos CPSs. De facto, um importante elemento frequentemente não considerado nos atuais sistemas cíber-físicos é o utilizador humano.

Por outro lado, sistemas que considerem o contexto humano irão tornar-se progressivamente mais importantes, e a maioria das futuras tecnologias irão convergir para uma consciência deste contexto humano. Quando todos estes sensores e dispositivos móveis começarem a ser usados para detetar e compreender a natureza humana, os humanos tornar-se-ão parte integral da IoT e dos CPSs convergindo-se, então, para os "Human-in-the-loop Cyber-Physical Systems" (HiTLCPS). A presença e comportamento humanos já não serão vistos como fatores externos e desconhecidos, mas tornar-se-ão numa parte fulcral do sistema. Concretamente, os crescentes avanços na área da Inteligência Artificial (AI) e nas áreas afins, como a Aprendizagem Máquina (ML) e a Ciência de Dados, juntamente com a rápida penetração de dispositivos móveis na vida de cada um de nós, impulsionou o desenvolvimento de sistemas denominados de Assistentes Pessoais Inteligentes (IPAA), em que as técnicas de AI são utilizadas para criar sistemas inteligentes e virtuais capazes de fornecer às pessoas assistência

personalizada e adaptativa (por exemplo: Apple Siri, Amazon Alexa). Os IPAAs podem monitorar as acções do indivíduo e produzir modelos comportamentais sobre a forma como o indivíduo se sente, dos seus fatores motivacionais, das suas intenções e desejos, e até da sua personalidade e da forma como este irá reagir em situações futuras. Esses modelos são essenciais para que um IPAA seja, de facto, considerado útil e forneça o "efeito de adaptação" desejado a cada indivíduo. Capacitados por esses modelos personalizados de utilizador, os IPAAs podem prever contextos específicos e estados psicológicos de cada ser humano, e agir de forma adequada e seletiva para cada indivíduo.

Por outro lado, hoje em dia, a Internet das Coisas (IoT) oferece oportunidades sem precedentes para rastrear a movimentação do ser humano, o seu ambiente envolvente, os seu bio-sinais e muito mais, tudo em tempo real. Além disso, juntamente com os avanços em AI / ML, os wearables prometem atingir um novo nível de conexão e de integração do próprio indivíduo na Internet.

O projeto Resilience4COVID propôs-se desenvolver um sistema de monitorização do comportamento e do estado emocional dos cidadãos nacionais no âmbito da pandemia COVID-19. A aplicação Vitória insere-se, assim, no contexto do projeto Resillience4 COVID, e teve como objetivo encontrar métricas e técnicas preditivas dos padrões de comportamento das pessoas durante e após uma pandemia. A aplicação recolhe, de forma anonimizada, informação através dos sensores do telemóvel, e, utilizando técnicas de aprendizagem automática, pretende compreender o impacto da pandemia no dia-a-dia do cidadão, nomeadamente no seu estado emocional, no stress, nos seus padrões de sono, na actividade física, na socialização e bem-estar. Para além disso, foi implementado um IPAA de forma a alertar o indivíduo para comportamentos de risco que ele teve e a aconselhar padrões comportamentais seguros.

2. Arquitectura

A arquitetura do sistema VITORIA é baseada numa aplicação para *smartwatch* e *smartphone*. A arquitetura completa da aplicação pode ser visualizada na Figura 1. O sistema é constituído por um servidor de *backend* baseado na arquitetura FIWARE [1].

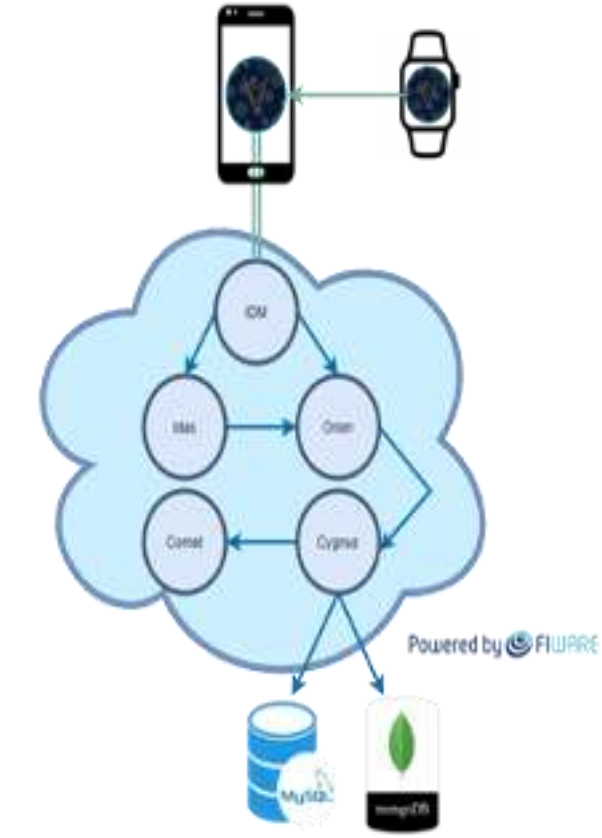

Figura 1- Arquitetura do Vitoria.

Os componentes do FIWARE utilizados no sistema foram o IDM, o IDAS, o ORION, o Cygnus e o Comet. As funcionalidades especificas de cada modulo são as seguintes:

- IDM (Identity Manager): este modulo é responsável pela gestão da segurança e da privacidade do sistema. Todas as comunicações feitas com os outros módulos são geridas por este modulo. Este implementa um sistema de autenticação oauth2.0 [2] e permite estabelecer políticas de acesso, que podem restringir a comunicação com os módulos internos a grupos específicos de utilizadores ou *endpoints* específicos.
- IDAS: este modulo é responsável pelas comunicações com dispositivos de recursos mais limitados (IoT), uma vez que estes na maioria das vezes suportam protocolos de comunicação diferentes (p.e. UL2.0, LwM2M, etc.). Este módulo é ainda responsável por "traduzir" estas comunicações para o protocolo de comunicação NGSI10 (o protocolo suportado pelo ORION) [3].
- ORION: este módulo gere o contexto, na criação e gestão das entidades representativas de entidades físicas, assim como cria subscrições para as mesmas de forma a capturar mudanças nessas entidades. Um exemplo real ilustrativo desta situação é na criação de uma subscrição para uma base de dados de alarmes que apenas guarda valores acima de um determinado "*threshold*".
- Cygnus: este módulo tem a seu cargo a criação de conexões entre o ORION e outros módulos, como por exemplo a base de dados e o COMET.
- COMET: este módulo faz a agregação histórica da informação guardada na base de dados. Permite, por exemplo, combinar a informação por dia ou por ocorrência, permitindo transformar *queries* complexos e demorados em simples pedidos a um *webservice*.

Mais informação especifica sobre cada um dos módulos pode ser encontrada no *site* da Fundação FIWARE [1].

Adicionalmente o sistema é também constituído por duas instâncias de uma base de dados: uma instância da base de dados MySQL server[4] e uma instância da base de dados MongoDB[5]. A duplicação da base de dados permite- obter uma replica da informação, em tempo real, de forma a não perder nenhum dado caso ocorra uma eventualidade inesperada.
No caso das aplicações para telemóveis e para *smartwatch*, ambas são implementadas para o sistema operativo Android [6] (no caso do *smartwatch* para Android WearOs [7]).
A aplicação de *smartwatch* comunica com o *smartphone* diretamente fazendo uso das capacidades de Bluetooth de ambos. Esta comunicação é feita usando a API (Interface de programação da aplicação) fornecida pelo sistema operativo para esta finalidade.
Ambas as aplicações possuem uma base de dados local que permite armazenar a informação recolhida, quando a comunicação dos dispositivos com a Internet não é possível.

3. Armazenamento

O armazenamento da aplicação VITORIA é feito em duas instâncias: pelo armazenamento de longa duração e pelo armazenamento temporário. O armazenamento de longa duração é suportado no servidor central da aplicação, onde os dados são armazenados durante o decorrer do estudo. O armazenamento temporário é o armazenamento realizado nos dispositivos móveis (*smartphone* e *smartwatch*) até estes dispositivos móveis se sincronizarem com o servidor; nesse momento os dados são apagados dos dispositivos móveis.
O armazenamento de longa duração é, conforme referido anteriormente, suportado em duas instâncias de base de dados: uma instância de MySQL Server[4] e uma instância de MongoDB[5]. A duplicação da base de dados permite ter uma replica da informação, em tempo real, de forma a não se perder nenhum dado caso ocorra algum problema. No entanto, por questões de segurança, apenas a base de dados MySQL é utilizada para suportar o processamento dos dados. Em cenários de deteção de alguma irregularidade procede-se ao acesso da base de dados MongoDB para se poder realizar as devidas verificações e reposições.
Ambas as bases de dados possuem também restrições de acesso. Ou seja, apenas um elemento da equipa de desenvolvimento teve acesso às mesmas durante o decorrer do projeto. Adicionalmente, ambas as bases de dados estão limitadas a uma rede interna, isto é, apenas podem ser acedidas de outras máquinas que se encontrem na mesma rede local (p.e. os outros módulos do sistema) para se aumentar, assim, o nível de segurança e proteção dos dados.
Durante este estudo foram também utilizadas duas bases de dados temporárias, que apenas existem até a informação poder ser transmitida para a sua localização final. A primeira base de dados temporária é uma base de dados que armazena os dados capturados nos *smartwatches*. Apesar do *smartwatch* ter capacidades de comunicação com a Internet para realizar pequenas tarefas, a Google recomenda que se use uma arquitetura distribuída, para tarefas mais complexas em que o *smartwatch* comunica com o *smartphone* e este realiza as comunicações com os recursos da Internet. Assim

sendo, a base de dados do *smartwatch* apenas existe até se estabelecer uma comunicação estável entre o *smartwatch* e o smartphone. Após isto, os dados são transmitidos para a base de dados temporária do *smartphone* e apagados da base de dados do *smartwatch*. A segunda base de dados temporária é a base de dados do *smartphone* que armazena todos os dados transmitidos do *smartwatch* bem como os dados capturados no *smartphone*. Esta base de dados é eliminada após os dados serem transmitidos para o servidor principal.
Adicionalmente, foram também tomadas outras medidas de segurança no caso das bases de dados temporárias: todos os dados são anonimizados antes de serem armazenados nestas base de dados; as bases de dados são encriptadas; e estas bases de dados apenas existem no contexto da aplicação. Isto é, estas bases de dados apenas podem ser acedidas a partir do código incorporado na aplicação VITORIA, não podendo ser acedidas por outras aplicações.

4. Aquisição de dados

A aquisição dos dados na aplicação VITORIA é feita através dos 2 tipos de dispositivos móveis: os *smartphones* e os *smartwatches*.
Os dados recolhidos através do smartwatch são os seguintes:

- Atividade: é utilizada a Google Activity Recognition API (API de reconhecimento de atividade) [8]. Esta API utiliza os sensores dos dispositivos móveis para fornecer uma classificação da atividade física. A lista completa das classificações fornecidas pode ser consultada na documentação respetiva [9]. A taxa de aquisição deste tipo de dados é controlada pela própria API (i.e é apenas obtido um novo valor quando é detetada uma atividade diferente da última atividade detetada).
- Número de Passos: é utilizado o sensor incorporado no *smartwatch* para obter o número de passos dados pelo utilizador. A taxa de aquisição destes dados é de 30 segundos.
- Ritmo Cardíaco: recorreu-se ao sensor incorporado no *smartwatch* para obter o ritmo cardíaco. A taxa de aquisição destes dados é de 30 segundos, sendo que, a cada 30 segundos, uma amostra de 10 segundos de duração é capturada.
- Acelerómetro: é utilizado o sensor incorporado para detetar os valores do acelerómetro do *smartwatch*. A taxa de aquisição destes dados é de 30 segundos, isto é, a cada 30 segundos é armazenada uma amostra dos últimos 100 valores do acelerómetro.
- Giroscópio: através do sensor incorporado conseguiu-se detetar os valores do giroscópio do *smartwatch*. A taxa de aquisição destes dados é de 30 segundos, isto é, a cada 30 segundos é armazenada uma amostra dos últimos 100 valores do giroscópio.

Apesar da utilização de um *smartwatch* enriquecer o funcionamento da aplicação VITORIA, a aplicação do *smartphone* foi construída de maneira a conseguir operar sozinha. Desta forma muitos dos dados do *smartwatch* são também adquiridos no *smartphone*, nomeadamente: Atividade, Acelerómetro e Giroscópio. No entanto também são adquiridos dados específicos do smartphone:

- Luminosidade: é utilizado o sensor de luminosidade para capturar a luz ambiente. A taxa de aquisição deste sensor é de 30 segundos.

- Amplitude de Som: é utilizado o microfone do *smartphone* para capturar o nível de som ambiente. A taxa de aquisição deste sensor é de 30 segundos.
- WIFI: estes dados correspondem aos resultados de *scans* WIFI do smartphone. Através destes facilmente se consegue perceber se o indivíduo se encontra em espaços interiores ou exteriores. Dos resultados fornecidos pelo *scan* do WIFI apenas são recolhidos a força do sinal e o endereço MAC dos mesmos. A taxa de aquisição destes valores é de 1 vez a cada 5 minutos.
- Bluetooth: estes dados correspondem aos resultados dos scans Bluetooth realizados pelo *smartphone*. Através destes scans o sistema consegue aferir o número de pessoas nas proximidades dos utilizadores. Dos resultados fornecidos apenas são recolhidos a força do sinal, o endereço MAC dos mesmos e o tipo de dispositivo[10]. A taxa de aquisição destes valores é de 1 vez a cada 5 minutos.
- Localização discreta: Aquando da configuração da aplicação é pedido ao utilizador que escolha o nome da sua rede WIFI de casa. Esta informação não é guardada no servidor, e é apenas utilizada para se obter a localização discreta do utilizador. Isto é, quando é feito um *scan* as redes WIFI disponíveis, avalia-se a existência do nome da rede de casa do utilizador nos resultados do scan. Desta forma é possível distinguir entre duas localizações discretas: "Home" e "Other". A taxa de aquisição deste dado é a mesma do scan WIFI.
- Localização Precisa: recorrendo às capacidades de GPS do *smartphone*, e utilizando o serviço Nominatim API (OpenStreetsMaps)[11], foi possível detetar a cidade, concelho e freguesia dos utilizadores. Para suportar a privacidade dos participantes, estes valores são posteriormente trocados por uma *hash* SHA256[12], conforme será explicado mais à frente. A taxa de aquisição deste valor é em função da receção de um novo valor de GPS que pode ocorrer numa de três situações: após mais de 5 minutos desde a última atualização; o utilizador movimentou-se durante mais de 10m; ou quando qualquer outra aplicação faça um pedido e receba uma atualização do valor do GPS. É importante realçar que nenhum valor de GPS é guardado em nenhuma base de dados, quer no *smartphone* ou no servidor central. Estes valores são apenas utilizados para se obter uma localização discreta por freguesia.
- Aplicação em primeiro plano: de forma a obter um histórico de utilização das aplicações do *smartphone* por parte do utilizador, é também capturado e guardado o nome da aplicação que se encontra em primeiro plano. A taxa de aquisição deste valor é de 30 segundos.
- Sensor de proximidade: este valor está muitas vezes associado ao sensor de luminosidade presente nos telemóveis. É normalmente utilizado para detetar quando o telemóvel se encontra próximo da cara do utilizador. Na aplicação VITORIA o valor deste sensor é recolhido com o intuito de detetar quando o telemóvel se encontra dentro do bolso/carteira dos utilizadores. A taxa de aquisição deste sensor é de 30 segundos.
- Ecrã bloqueado: este valor é recolhido com o objetivo de determinar se o utilizador se encontra a utilizar ativamente o *smartphone*. É apenas detetado um de dois valores ("bloqueado" ou "não bloqueado"), e a taxa de aquisição é de 30 segundos.
- Último Alarme: recorre-se ao relógio do *smartphone* para detetar a data e hora do próximo alarme definido pelo utilizador. Este valor é utilizado para auxiliar o sistema de deteção de sono.
- Utilização de Aplicações: são também armazenadas as métricas de utilização das aplicações por parte do utilizador. Estas correspondem a uma lista das aplicações

utilizadas ao longo do dia e a duração da sua utilização. A taxa de aquisição destes valores é uma vez a cada hora.

5. Privacidade

Anonimização:

Toda a informação recolhida pelo sistema é guardada no *smartphone* ou no *smartwatch* do participante, anonimizada e enviada através de ligação encriptada ponto-a-ponto para o servidor central onde é agrupada com informação de outros participantes, reforçando, assim, a anonimização.
O primeiro passo para se proceder à anonimização dos dados, resulta do facto de que nenhum dado pessoal é pedido ao utilizador no momento do login. O utilizador realiza o login através da sua conta do Facebook e a implementação deste mecanismo foi feita recorrendo a um SDK (Software Development Kit) do próprio Facebook. Desta forma a aplicação VITORIA não tem qualquer acesso à informação do utilizador.
Após o login a aplicação apenas realiza a gestão da autenticação através do mesmo SDK, recorrendo à gestão do *"lifecycle"* de *tokens* de autenticação. Desta forma o utilizador não precisa de disponibilizar nenhum *username* ou email.
Aquando do login é gerado um código aleatório e único através de um mecanismo de *hash*, utilizando SHA256 [12]. Este código serve como identificador das instâncias do utilizador nas bases de dados. Este procedimento permite que a informação fique agrupada por utilizador, mas ao mesmo tempo permite anonimizar os dados uma vez que o id gerado não é reversível, e não é utilizada nenhuma informação pessoal para o gerar.
Relativamente aos dados adquiridos, são igualmente tomadas medidas de segurança adicionais de forma a anonimizar os dados, como já foi explicado anteriormente.

Localização:

Os valores recebidos de localização precisa da API Nominatim [11] são trocados por *hashes SHA256* [12]. Esta mecanismo de hash por si só já é irreversível; porém procede-se ainda à adição de um mecanismo de segurança complementar que consiste em criar uma *hash* de 64 caracteres e truncar essa *hash* aos primeiros 32 caracteres. Isto torna completamente impossível reverter a *hash*, enquanto se mantém as propriedades desejadas de cada *hash* ser única.
Os valores de *hash* são criados a partir da conjunção de um valor que é específico ao utilizador (mas que não o identifica) e do nome das cidades, concelhos e freguesias, respetivamente. Isto permite que os valores sejam repetíveis, mas únicos por utilizador. Consegue-se assim aferir que um utilizador passou um determinado tempo numa localização, mas nunca é possível afirmar que dois utilizadores estiveram na mesma localização. Adicionalmente, como os valores são por utilizador, permanecem constantes para cada localização ao longo do tempo. Assim, é possível aferir as movimentações dos utilizadores entre freguesias ou concelhos, mas não é possível aferir quais são esses concelhos.
Adicionalmente nenhum dado do GPS é armazenado, quer nos dispositivos móveis (*smartphone*/*smartwatch*) quer no servidor central.

Microfone:

A aplicação VITORIA utiliza o microfone apenas para capturar o ruido ambiente. Apesar da permissão do microfone o possibilitar, nenhum áudio é capturado ou guardado pela aplicação. A aplicação recorre à função de captura da amplitude de som, disponibilizada pela API *MediaRecorder* do Android [13]. Esta API devolve um valor entre 0 e 32768, que depois é convertido para dBs através da seguinte expressão matemática:

$$Amplitude_{DBs} = 20 \times \log_{10}(32768) + 20 \times \log_{10}\left(\frac{value}{32768}\right)$$

Onde "*value"* corresponde ao valor recebido pela API do Android. Estes valores são posteriormente armazenados na base de dados local até serem enviados para o servidor central. Quando a aplicação VITORIA não é capaz de aceder ao microfone, ou este está a ser usado por outra aplicação, é armazenado o valor *-10*.
Adicionalmente os dados recebidos do microfone do *smartphone* não são armazenados, sendo apenas processados em tempo real e apenas a média do ruido ambiente é armazenada.

6. Testes

Os testes foram realizados por um conjunto de voluntários em momentos pré-determinados da pandemia. Pretendeu-se avaliar o nível de risco de contágio para cada participante em alturas de confinamento e de não-confinamento. Em cada uma destas alturas, o estudo temporal foi divido em períodos sem alertas e com alertas personalizados. Para este último caso a IPAA alertava sempre que uma situação de risco ocorria para aquele indivíduo. Assim, para estes casos, além de compreender as motivações, contexto e estado emocional de cada participante, o sistema suportou também ações pró-ativas.

Referências

[1] F. Foundation, "Fiware Catalogue," 2020. https://www.fiware.org/developers/catalogue/ (accessed Nov. 12, 2020).
[2] IETF OAuth Working Group., "OAuth 2.0," 2020. https://oauth.net/2/ (accessed Nov. 12, 2020).
[3] F. Foundation, "NGSIV2 IMPLEMENTATION NOTES," 2020. https://fiware-orion.readthedocs.io/en/master/user/ngsiv2_implementation_notes/index.html (accessed Nov. 12, 2020).
[4] Oracle Corporation and/or its affiliates, "MySQL," 2020. https://www.mysql.com/ (accessed Nov. 12, 2020).
[5] I. MongoDB, "Mongo DB," 2020. https://www.mongodb.com/ (accessed Nov. 12, 2020).
[6] Google, "Crie o que quiser no Android | developers Android," 2020. https://developer.android.com/ (accessed Nov. 12, 2020).
[7] Google, "Wear OS by Google | Android Developers," 2020. https://developer.android.com/wear (accessed Nov. 12, 2020).

[8] Google, “Activity Recognition API,” 2020. https://developers.google.com/location-context/activity-recognition (accessed Nov. 12, 2020).
[9] Google, “DetectedActivity | Google APIs for Android,” 2020. https://developers.google.com/android/reference/com/google/android/gms/location/DetectedActivity (accessed Nov. 12, 2020).
[10] Google, “Bluetooth Major Class | Android Developers,” 2020. https://developer.android.com/reference/android/bluetooth/BluetoothClass.Device.Major (accessed Nov. 13, 2020).
[11] Nominatim, “Nominatim API,” 2020. https://nominatim.org/release-docs/develop/api/Overview/ (accessed Nov. 13, 2020).
[12] C. Paar and J. Pelzl, *Understanding cryptography: a textbook for students and practitioners*. Springer Science & Business Media, 2009.
[13] Google, “MediaRecorder | Android Developers,” 2020. https://developer.android.com/reference/android/media/MediaRecorder (accessed Nov. 18, 2020).