



**NOVA**  
NOVA SCHOOL OF  
SCIENCE & TECHNOLOGY

DEPARTAMENTO DE  
ENGENHARIA ELETROTÉCNICA E DE  
COMPUTADORES

DIOGO FILIPE CURTO MARQUES DA SILVA

# UMA ARQUITETURA PARA PROMOVER A SEGURANÇA DE PACIENTES COM DEMÊNCIA UTILIZANDO MÉTODOS DE INTELIGÊNCIA ARTIFICIAL PARA OTIMIZAR BATERIA EM DISPOSITIVOS MÓVEIS

MESTRADO INTEGRADO EM ENGENHARIA ELETROTÉCNICA E  
COMPUTADORES

Universidade NOVA de Lisboa  
novembro, 2021





# UMA ARQUITETURA PARA PROMOVER A SEGURANÇA DE PACIENTES COM DEMÊNCIA UTILIZANDO MÉTODOS DE INTELIGÊNCIA ARTIFICIAL PARA OTIMIZAR BATERIA EM DISPOSITIVOS MÓVEIS

**DIOGO FILIPE CURTO MARQUES DA SILVA**

**Orientador:** Ricardo Luís Rosa Jardim Gonçalves,  
Professor Catedrático, Universidade NOVA de Lisboa

**Coorientador:** Fernando Luís Lourenço Ferreira,  
Professor Auxiliar Convocado, Universidade NOVA de Lisboa

## **Júri:**

**Presidente:** Nome do(a) presidente,  
Professor(a) Catedrático(a), FCT-NOVA

**Arguentes:** Nome de um dos arguentes,  
Professor Associado, Outra Universidade  
Nome do outro arguente,  
Professor Auxiliar, Ainda Outra Universidade

**Orientador:** Nome do orientador presente nas provas,  
Professor Associado, FCT-NOVA

**Membros:** Mais um membro do júri,  
Professora Catedrática, Outra Universidade  
E ainda mais um membro do júri,  
Professor Auxiliar, Ainda Outra Universidade



**Uma arquitetura para promover a segurança de pacientes com demência utilizando métodos de inteligência artificial para otimizar bateria em dispositivos móveis**

Copyright © Diogo Filipe Curto Marques da Silva, Faculdade de Ciências e Tecnologia, Universidade NOVA de Lisboa.

A Faculdade de Ciências e Tecnologia e a Universidade NOVA de Lisboa têm o direito, perpétuo e sem limites geográficos, de arquivar e publicar esta dissertação através de exemplares impressos reproduzidos em papel ou de forma digital, ou por qualquer outro meio conhecido ou que venha a ser inventado, e de a divulgar através de repositórios científicos e de admitir a sua cópia e distribuição com objetivos educacionais ou de investigação, não comerciais, desde que seja dado crédito ao autor e editor.



Dedicado aos meus pais.



## AGRADECIMENTOS

Começo por agradecer ao meu orientador Ricardo Gonçalves, e em especial ao meu co-orientador Fernando Ferreira por todo o apoio que me deu nesta caminhada. Quero agradecer também à Faculdade pela formação que me proporcionou e por todas as vivências que me deu ao longo destes anos.

O maior agradecimento é sem dúvida para os meus pais por todo o investimento que fizeram em mim, por estarem sempre ao meu lado e me apoiarem em tudo. Sem eles nada disto seria possível e foi graças a eles que me tornei na pessoa que sou.

Por fim, quero agradecer a todos os meus familiares, amigos que fui fazendo ao longo desta caminhada que é a vida e também à minha namorada, pois todos eles foram importantes.



“If there is effort, there is always accomplishment.”  
(Jigoro Kano).



## RESUMO

O aumento em proporção da população envelhecida é uma realidade em culturas desenvolvidas, o que coloca novos desafios aos indivíduos e à sociedade. Um dos mais preocupantes tem que ver com a tendência de crescimento dos casos de demência em todo o mundo, já que a sua incidência aumenta com o envelhecimento.

Uma pessoa com demência enfrenta diversos desafios que podem limitar a sua capacidade de ter uma vida normal e causar preocupações à família e aos cuidadores. A arquitetura proposta visa aumentar ao máximo a duração da bateria do dispositivo de contato do indivíduo com a demência, aumentando a possibilidade de este ser encontrado e resgatado em caso de emergência. Desta forma pretende-se melhorar a qualidade de vida do paciente e da comunidade, através da detecção de padrões de atividade e de uma gestão energética ativa dos recursos de um *smartphone*. Nesse sentido, faz-se recolha e análise de dados com o objetivo de otimizar os serviços disponíveis, reagindo de maneira adequada, de forma a otimizar o consumo de energia, aumentando assim a janela de tempo disponível para a comunicação entre paciente e cuidadores.

A estratégia proposta tem um impacto potencial na autonomia dos dispositivos, evitando situações perigosas e regulando a comunicação entre a pessoa e os cuidadores, especialmente se a pessoa inicia atividades imprevisíveis, como por exemplo, um comportamento errante que é característico deste tipo de patologias. Assim, ao aumentar a possibilidade de a pessoa ser contactada e localizada poderá ser decisivo para esta ser resgatada e a sua vida salva.

Os resultados obtidos foram positivos, uma vez que a arquitetura utilizada se revelou eficiente no objetivo proposto de maximizar a duração da bateria num smartphone sem comprometer a localização do utilizador.

**Palavras chave:** Demência, Doença, Cuidador, Dispositivos Móveis e Consumo Energético.

## ABSTRACT

The emergence of a relatively large proportion of aging population is a reality in developed cultures, that poses challenges to individuals and to the society. Therefore, it is noticed a tendency of growth in dementia cases worldwide as its incidence increases with aging.

A person with dementia faces diverse challenges that may limit their ability to perform a normal life and cause stress to family and caregivers. The proposed architecture aims to maximize the device's battery life of the individual with dementia, increasing the possibility of being found and rescued in an emergency. This way it's intended improve the quality of life for the patient and the community through the detection of activity patterns and an active energy management of the smartphone resources. In this sense, data is collected and analysed with the objective of optimizing the services available, as well as the current circumstances of that individual, reacting appropriately to optimize the energy consumption, thus increasing the window of time available for communication between patient and caregivers.

The proposed strategy has a potential impact in autonomy of the devices, preventing hazardous situations and regulating the communication with the person and caregivers, especially if the person starts unpredictable activities as in the case of wandering behaviour that is characteristic of this type of pathologies. So, by increasing the possibility of the person being contacted and located, it could be decisive for them to be rescued and their life saved.

The results were positive since the architecture used, proved to be efficient in the proposed objective of maximising battery life in a smartphone without compromising the user's location.

**Keywords:** Dementia, Disease, Caregiver, Mobile Devices and Energy Consumption.

# ÍNDICE

<b>1</b>	<b>INTRODUÇÃO</b> .....	<b>1</b>
1.1	Motivação .....	1
1.2	A epidemiologia da demência .....	2
1.2.1	Doença de Alzheimer .....	2
1.2.2	Demência vascular .....	3
1.2.3	Demência frontotemporal .....	3
1.2.4	Demência de corpos de Lewy .....	3
1.2.5	Doença de Parkinson.....	4
1.3	Objetivo do presente estudo.....	4
1.3.1	Perguntas de investigação.....	5
1.3.2	Hipótese da dissertação .....	5
1.4	Organização do documento .....	5
<b>2</b>	<b>ESTADO DA ARTE</b> .....	<b>7</b>
2.1	Dispositivos e serviços existentes .....	7
2.1.1	<i>Smartphones</i> .....	7
2.1.2	<i>Wearables</i> .....	9
2.1.3	Serviços e produtos disponíveis.....	11
2.2	Tecnologias existentes .....	14
2.2.1	Sensores .....	14
2.2.2	Gestão energética.....	18

2.2.3	Algoritmos de aprendizagem ( <i>Machine Learning</i> ).....	21
<b>3</b>	<b>METODOLOGIA E DESENVOLVIMENTO.....</b>	<b>27</b>
3.1	Dispositivos e ferramentas utilizadas .....	27
3.1.1	<i>Smartphone</i> utilizado .....	27
3.1.2	<i>Software</i> utilizado .....	28
3.2	Avaliação dos consumos energéticos .....	31
3.3	Perfis e dados para análise .....	33
3.3.1	Perfis .....	33
3.3.2	Dados para CART .....	34
3.4	Software CART.....	36
3.4.1	Separador <i>Model</i> .....	36
3.4.2	Separador <i>Categorical</i> .....	37
3.4.3	Matriz de custos .....	38
3.4.4	Outras definições da modelação .....	38
3.5	Android Studio.....	39
3.5.1	Interface do usuário (IU).....	39
3.5.2	Serviços <i>wi-fi</i> , <i>bluetooth</i> , dados móveis e GPS.....	41
3.5.3	Acelerómetro como contador de passos.....	43
3.5.4	Temporizador .....	44
3.5.5	Serviço da bateria .....	45
3.5.6	Envio de SMS.....	45
3.5.7	Gestor de perfis .....	46
3.5.8	Árvore de decisão .....	47
<b>4</b>	<b>RESULTADOS E VALIDAÇÃO.....</b>	<b>49</b>
4.1	Análise da implementação no CART.....	49

4.1.1	Matriz de Custos.....	49
4.1.2	Resultados dos conjuntos de treino e de teste .....	50
4.1.3	Árvore de decisão .....	52
4.2	Aplicação Android.....	55
5	<b>CONCLUSÕES E TRABALHO FUTURO .....</b>	<b>59</b>



## ÍNDICE DE FIGURAS

Figura 1-1 — Evolução prevista para os casos de demência até 2050.....	1
Figura 2-1 — Martin Cooper, inventor do telemóvel.....	8
Figura 2-2 — Mercado dos <i>smartphones</i> .....	9
Figura 2-3 — <i>Smartwatch</i> .....	10
Figura 2-4 — <i>Smartbands</i> .....	11
Figura 2-5 — Pulseira myMDlife.....	12
Figura 2-6 — Palmilha GPS SmartSole.....	12
Figura 2-7 — <i>Wearables</i> da AngelSense.....	13
Figura 2-8 — Pulseira do programa Estou Aqui da PSP.....	13
Figura 2-9 — Exemplo de uma rede <i>wi-fi</i> .....	15
Figura 2-10 — Exemplo de uma arquitetura <i>master-slave</i> .....	16
Figura 2-11 — Esquema de 3 eixos de um acelerómetro.....	17
Figura 2-12 — Esquema de triangulação GPS.....	17
Figura 2-13 — Gráfico representativo do consumo energético de diferentes sensores.....	20
Figura 2-14 — Consumo energético sem transmissão de dados.....	20
Figura 2-15 — Consumo energético com consumo de dados.....	21
Figura 2-16 — Gráfico resultante da lógica Fuzzy.....	22
Figura 2-17 — Modelo de um neurónio numa RN.....	23
Figura 2-18 — Exemplo de uma árvore de decisão.....	23
Figura 3-1 — BQ Aquaris M5: o dispositivo utilizado.....	28
Figura 3-2 — Janela principal do Android Studio.....	30
Figura 3-3 — Ambiente de trabalho do CART.....	31
Figura 3-4 — Modelação das variáveis de entrada e da variável <i>target</i> .....	36
Figura 3-5 — Separador <i>Categorical</i> .....	37

Figura 3-6 — Matriz de custos do problema.....	38
Figura 3-7 — Número de casos necessários para construção da árvore.....	39
Figura 3-8 — Interface do usuário.....	40
Figura 3-9 — Classe <i>BaseController.java</i> : classe abstrata para os serviços.....	41
Figura 3-10 — Serviço <i>wi-fi</i> (classe <i>WifiController.java</i> ).....	42
Figura 3-11 — Serviço do acelerómetro (classe <i>AcelerometerController.java</i> ).....	43
Figura 3-12 — Serviço do temporizador (classe <i>TimerController.java</i> ).....	44
Figura 3-13 — Função para obter o nível da bateria do <i>smartphone</i> . ....	45
Figura 3-14 — Função para enviar uma mensagem de texto.....	46
Figura 3-15 — Configuração dos perfis.....	46
Figura 3-16 — Gestor de perfis ( <i>ProfileController.java</i> ).....	47
Figura 3-17 — Interface para as condições de um nó. ....	47
Figura 3-18 — Definição da raiz e de um nó da árvore.....	48
Figura 3-19 — Ramificação de cada nó.....	48
Figura 4-1 — Tabela de previsão de sucesso do conjunto de treino.....	51
Figura 4-2 — Tabela de previsão de sucesso do conjunto de teste.....	51
Figura 4-3 — Importância das variáveis na árvore de decisão. ....	52
Figura 4-4 — Escolha da árvore de decisão.....	53
Figura 4-5 — Árvore de decisão.....	54
Figura 4-6 — Perfis na IU. ....	55

## ÍNDICE DE TABELAS

Tabela I — Consumo energético de sensores em percentagem. ....	19
Tabela II — Média de percentagem de bateria restante após 1h de teste.....	32
Tabela III — Definição e características dos perfis.....	33
Tabela IV — 10 dos 300 casos para serem analisados pelo CART. ....	35
Tabela V — Matriz de custos utilizada no CART. ....	50
Tabela VI — Testes finais de 1h. ....	56
Tabela VII — Testes finais de 6h. ....	56



## ÍNDICE DE EQUAÇÕES

Equação i — Energia consumida no momento. ....	19
Equação ii — Energia consumida num período de tempo.....	19
Equação iii — Entropia num nó. ....	24
Equação iv — Ganho de informação. ....	24
Equação v — Critério de Gini.....	25
Equação vi — Equação para calcular magnitude. ....	43
Equação vii — Cálculo de $\Delta$ Magnitude.....	44



## SIGLAS

<b>A-GPS</b>	<i>Assisted Global Positioning System.</i>
<b>AP</b>	<i>Access Point.</i>
<b>API</b>	<i>Application Programming Interface.</i>
<b>BSS</b>	<i>Basic Service Set.</i>
<b>IDE</b>	<i>Integrated Development Environment.</i>
<b>GPS</b>	<i>Global Positioning System.</i>
<b>GSM</b>	<i>Global System for Mobile Communications.</i>
<b>JMV</b>	<i>Java Virtual Machine.</i>
<b>ML</b>	<i>Machine Learning.</i>
<b>NFC</b>	<i>Near Field Communication.</i>
<b>OMS</b>	Organização Mundial da Saúde.
<b>PSP</b>	Polícia de Segurança Pública.
<b>RN</b>	Redes Neurais.
<b>SDK</b>	<i>Software Development Kit.</i>
<b>SMS</b>	<i>Short Message Service.</i>
<b>SO</b>	Sistema Operativo.
<b>SSID</b>	<i>Service Set Identifier.</i>



# INTRODUÇÃO

## 1.1 Motivação

Atualmente, em todo o mundo, estima-se que existam cerca de 50 milhões de pessoas com demência. Este número segundo a Organização Mundial da Saúde (OMS), pode ser superior a 80 milhões em 2030 e superior a 150 milhões em 2050, o que representa um triplicar de casos nos próximos 30 anos. Esta evolução pode ser observada no gráfico Figura 1-1.

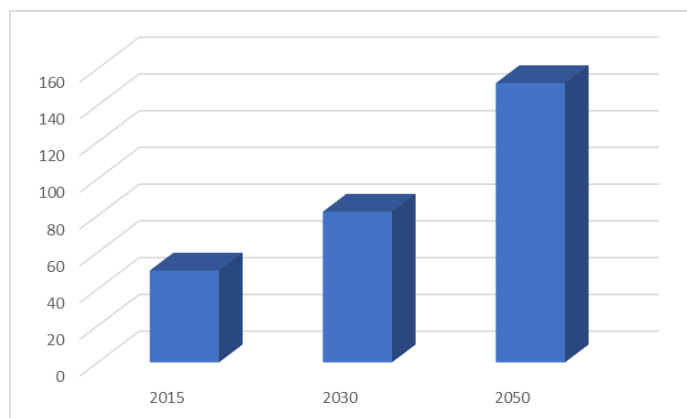


Figura 1-1 — Evolução prevista para os casos de demência até 2050.

A demência é a sétima causa de morte mais frequente em todo o mundo. É uma doença que afeta o cérebro e o seu normal funcionamento. Tem implicações graves para o doente, mas também causa perturbações aos familiares e amigos, que na maioria dos casos são os cuidadores informais.

Os cuidadores passam frequentemente por situações de *stress* físico, emocional e financeiro. É uma doença que acarreta com muitos custos, que são suportados maioritariamente pelos familiares e amigos. O custo aferido no ano de 2015 superou os 800 mil milhões de dólares em todo o mundo.

Apesar de estarem a aumentar mundialmente os casos de demência, prevê-se que a maioria das ocorrências venham a incidir com maior frequência em países subdesenvolvidos. Estes factos, levaram a OMS a declarar a demência como uma prioridade de saúde pública, no seu plano de ação global para o período de 2017 a 2025, e podem ser encontrados no seu documento oficial [1].

Com este trabalho pretende-se contribuir para o bem-estar dos doentes e cuidadores, possibilitando-lhes uma vida com menos preocupações e maior segurança.

## 1.2 A epidemiologia da demência

A demência é uma síndrome, frequentemente de natureza crónica ou progressiva, em que existe uma deterioração de funções cognitivas, que vai além do envelhecimento natural. Pode afetar funções cognitivas como a memória, a compreensão, a capacidade de aprendizagem, a fala e a orientação.

Apesar de uma maioria significativa dos casos de demência se manifestar em pessoas com uma idade superior a 65 anos, estes podem ocorrer em pessoas mais jovens.

Alguns dos principais fatores de risco são a inatividade física, o consumo de tabaco, a obesidade, a diabetes e a hipertensão. Estes podem provocar os primeiros sintomas de demência, que apesar de subtis, são muito comuns:

- Perda de memória;
- Confusão (perder-se em locais familiares, também conhecido por *wandering*);
- Alterações de personalidade (alguma apatia).

Existem, contudo, vários tipos de demência como será possível observar já de seguida[2].

### 1.2.1 Doença de Alzheimer

A doença de Alzheimer é o tipo de demência mais comum, representando entre 60% e 70% dos casos. É uma doença de progressão lenta, em que as células cerebrais vão regredindo em tamanho e número, dificultando gradualmente a comunicação entre células. Esta progressiva morte celular, expressa-se na crescente incapacidade em recordar informações ou memorizar as atuais. À medida que o cérebro fica mais danificado podem-se perder mais capacidades cognitivas [3].

A doença pode incidir em qualquer faixa etária, apesar de ser mais comum a partir dos 65 anos. Os principais sintomas são o esquecimento de pessoas ou locais conhecidos, a dificuldade na fala, problemas de memória, alterações emocionais e ainda dificuldade e demora em realizar atividades de rotina [4].

### **1.2.2 Demência vascular**

A demência vascular é o segundo tipo de demência mais comum. Divide-se em vários subtipos, todos eles associados a problemas de circulação sanguínea no cérebro. Os dois subtipos mais comuns são a demência por multienfartes e a doença de Binswanger.

A demência por multienfartes deve-se a múltiplos enfartes cerebrais, que são também conhecidos por acidentes isquêmicos transitórios (AIT). Estes diversos enfartes, quando ocorrem no córtex cerebral provocam danos na capacidade de aprendizagem, na memória e na fala.

Já a doença de Binswanger, também conhecida como doença vascular subcortical, apresenta sintomas muito similares à doença de Alzheimer, mas é provocado por danos cerebrais na substância branca do cérebro [5].

### **1.2.3 Demência frontotemporal**

É atribuído o nome demência frontotemporal às demências em que os lobos frontais ou temporais do cérebro se degeneram. Consoante os lobos afetados, os sintomas podem variar.

Enquanto nos lobos frontais as capacidades de controlo de emoções e o comportamento podem sofrer alterações, nos lobos temporais são as capacidades como a fala que ficam afetadas, podendo o doente ter dificuldades em falar ou compreender palavras [6].

### **1.2.4 Demência de corpos de Lewy**

Ao se depositarem nas células nervosas, os corpos de Lewy (uma proteína) destroem-nas. Estas mortes celulares provocam sintomas muito parecidos com os da doença de Alzheimer. É muito difícil distinguir as duas doenças, no entanto, de uma forma geral as alucinações tendem a ocorrer mais cedo na demência por corpos de Lewy, a capacidade para desenhar fica

mais prejudicada e as funções mentais também ficam afetadas denotando grandes oscilações num curto período de tempo [7].

### 1.2.5 Doença de Parkinson

Tal como na doença de Alzheimer, os sintomas manifestam-se de uma forma lenta e gradual na doença de Parkinson. Esta patologia resulta de uma redução dos níveis de dopamina, que é provocada pela morte de mais de 70% das células cerebrais que a produzem. É desconhecida a razão para a morte destas células. Os primeiros sintomas podem ser tremores, problemas na fala e rigidez nas articulações, mas em casos mais avançados é comum desenvolver demência [8].

## 1.3 Objetivo do presente estudo

Como foi possível verificar anteriormente, a demência, apesar de incidir com maior frequência num determinado tipo de população, pode aparecer em qualquer região do globo e em qualquer família. Por isso seria importante encontrar uma solução prática, funcional e de uso imediato.

Sendo a perda de memória e o *wandering* dois dos principais sintomas comuns aos vários tipos de demência, é de extrema importância que os doentes sejam acompanhados diariamente. Este acompanhamento fica muitas vezes ao cuidado de familiares ou amigos que não têm a formação nem a disponibilidade para cuidar destes pacientes.

O *wandering* afeta uma grande percentagem de pessoas com Alzheimer. Quando as pessoas estão nesta situação perdem a noção de onde estão, para onde vão ou em alguns casos até de quem são. Isto representa um perigo enorme e pode mesmo colocá-las em risco de vida, já que podem não ter noção dos perigos ou podem não conseguir regressar a casa ou à respetiva instituição. Por essa razão as primeiras horas são críticas e as pessoas devem ser encontradas nas primeiras 72 horas, no máximo.

Nesse sentido o *smartphone*, quer pelos seus diversos sensores e antenas quer pelas suas aplicações é normalmente uma das melhores chances de localizar uma pessoa. Contudo a duração da bateria é muitas vezes um problema, uma vez que o *smartphone* só consegue ser localizado se estiver ligado.

No âmbito desta dissertação foi também realizado o *short paper* para a 11<sup>a</sup> conferência internacional de *e-Health MCCSIS (Multi Conference on Computer Science and Information Systems)* presente no apêndice A.1.

### 1.3.1 Perguntas de investigação

- Como se pode maximizar a duração de bateria de um *smartphone* mantendo alguma informação sobre a localização de doentes com Alzheimer?
- Podem as técnicas de *machine learning* representar uma melhoria na gestão energética do *smartphone*?

### 1.3.2 Hipótese da dissertação

Esta dissertação pretende encontrar uma estratégia para que o *smartphone* consiga preservar a sua bateria ao máximo, mas mantendo sempre algum tipo de localização ativo, aumentando assim as hipóteses de uma pessoa ser localizada por um maior período de tempo e assim ser resgatada com vida, e em suma salvar-se uma vida.

Para isso foi criada uma aplicação para controlar os vários sensores do *smartphone*. Após analisar alguns consumos energéticos, foram utilizados conceitos de *machine learning*, nomeadamente árvores de decisão, para tentar encontrar uma forma eficiente de realizar a gestão energética do dispositivo. Depois procedeu-se à implementação da árvore de decisão na aplicação e tiraram-se as devidas conclusões.

Isto permitirá uma gestão um pouco mais eficiente da bateria e facilitará a localização dos pacientes por um período de tempo mais prolongado.

## 1.4 Organização do documento

Esta dissertação, é composta por mais quatro capítulos, dos quais se segue uma breve descrição:

- Capítulo 2: São apresentados alguns dispositivos e tecnologias existentes na área desta dissertação e ainda são apresentados alguns conceitos fundamentais para um entendimento do tema abordado;
- Capítulo 3: Mostra-se todo o trabalho que foi desenvolvido, todos os passos dados na construção do modelo de tratamento de dados e construção da aplicação;

- Capítulo 4: São apresentados e analisados os resultados com as devidas justificações;
- Capítulo 5: É o capítulo onde são tiradas as conclusões finais e os caminhos que podem ser seguidos na sequência deste estudo.

## ESTADO DA ARTE

### 2.1 Dispositivos e serviços existentes

Quando se pensa em dispositivos que acompanhem as a população no dia-a-dia, é impossível não pensar nos *smartphones* e também em alguns *wearables*, como *smartwatches*. Este tipo de dispositivos acompanham uma grande parte da população diariamente, e têm muitos sensores e funcionalidades que permitem ao utilizador ser facilmente localizável e contactável.

Estas funcionalidades são essenciais para o objetivo pretendido, uma vez que, um doente com demência tende a abandonar o local onde está e a perder-se com enorme facilidade.

#### 2.1.1 *Smartphones*

Como a palavra inglesa indica, um *smartphone* é um telefone inteligente. Os telemóveis existem desde os anos 70 do século XX, começaram por ser pouco portáteis tal como se observa na Figura 2-1, pesando o primeiro dispositivo quase 1 quilograma. Houve, desde então, uma grande evolução a nível tecnológico que permitiu o desenvolvimento de dispositivos com mais funcionalidades e mais leves [9].

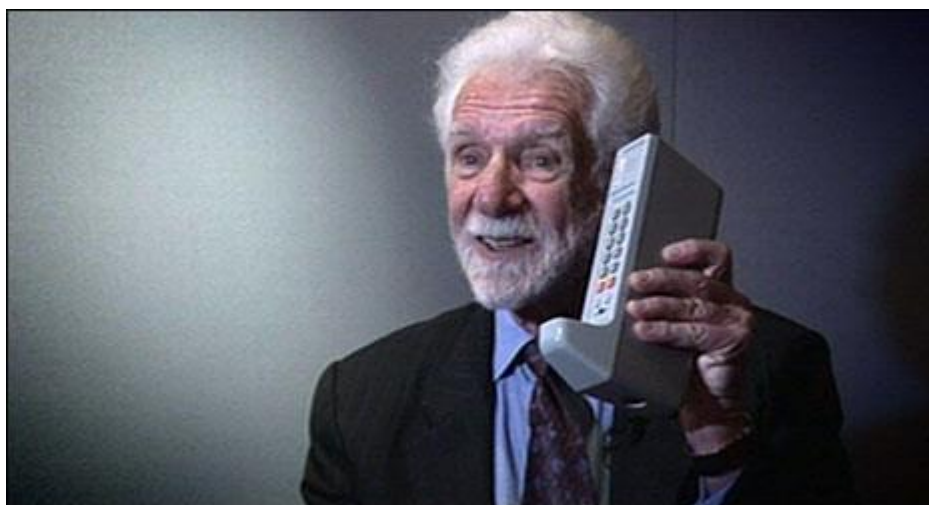


Figura 2-1 — Martin Cooper, inventor do telemóvel [9].

Inicialmente, os telemóveis apenas realizavam chamadas de voz, no entanto, foram ganhando novas capacidades como enviar mensagens escritas. Com o aparecimento da rede de 3ª geração, os telemóveis tornaram possível realizar videochamadas em qualquer lugar, uma vez que, as transmissões de dados entre dispositivos ficaram muito mais rápidas.

Com os telemóveis a ganhar cada vez mais funcionalidades, como ir a páginas *web*, enviar *e-mails* e agendar tarefas, o termo telemóvel começou a cair em desuso, tornando-se o termo *smartphone* mais adequado, tendo em conta as capacidades destes dispositivos.

Atualmente, os *smartphones* conseguem realizar grande parte das tarefas que se realizam em computadores. Através de sensores como os acelerómetros e antenas GPS (*Global Positioning System*) os *smartphones* conseguem apresentar uma localização bastante exata do utilizador. É também possível obter a localização de um utilizador através das redes sem fios (*wi-fi*, *Wireless Fidelity*), mesmo sem estar ligado a estas [10].

Existem presentemente dois grandes concorrentes no mercado dos *smartphones*, a Google LLC e a Apple Inc. (Figura 2-2), que dominam o mercado uma vez que produzem os principais sistemas operativos (SO) para este tipo de dispositivo [11].

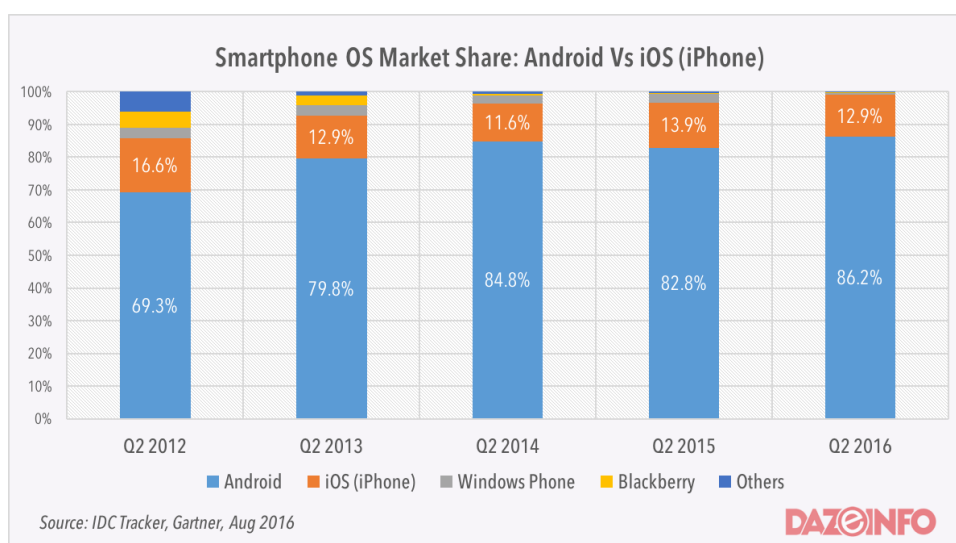


Figura 2-2 — Mercado dos *smartphones*.

A Google LLC desenvolve o SO Android, baseado em Linux que é o mais utilizado em todo o mundo, já que quase todas as empresas que desenvolvem hardware optam por utilizar o SO Android por ser *open source*. Com esta estratégia *open source*, a Google LLC domina o mercado com mais de 80% dos *smartphones* em todo o mundo a utilizarem Android [12].

Já a Apple Inc., desenvolve o iOS que apenas é utilizado nos dispositivos da sua marca. Apesar deste facto, a Apple Inc. detém uma fatia importante do mercado e é a empresa mais valiosa em todo o mundo, avaliada em mais de 900 mil milhões de dólares [13].

## 2.1.2 *Wearables*

Com a evolução tecnológica, tem sido possível criar dispositivos cada vez mais pequenos. Esta portabilidade cada vez maior, permite às empresas desenvolvedoras, tornar produtos de *hardware* em peças ou acessórios de vestuário.

Os *smartwatches* e as *smartbands* são dos dispositivos tecnológicos mais utilizados na categoria dos *wearables*.

### 2.1.2.1 *Smartwatches*

Os *smartwatches* são aparelhos computacionais criados para serem utilizados no pulso, com o objetivo de substituir os relógios tradicionais, como o presente na Figura 2-3. Baseados frequentemente em SO móveis como o Android, muitos destes relógios inteligentes permitem

o emparelhamento com um *smartphone*, e através dessa ligação receber no pulso todo o tipo de notificações.



Figura 2-3 — *Smartwatch* [14].

Contudo, a maioria dos *smartwatches* permite o funcionamento independente. Através dos mais variados sensores é possível ter acesso a quase todas as funcionalidades. Com as antenas GPS (*Global Positioning System*), o relógio consegue saber a sua localização e com isso fazer o *tracking* constante do utilizador. Com os acelerómetros, também é possível contar os passos dados e detetar outros movimentos (muitos contam braçadas dadas ao nadar) [15].

### 2.1.2.2 *Smartbands*

Vistas muitas vezes como *fitness trackers*, as *smartbands*, presentes na Figura 2-4, são também aparelhos computacionais que podem ser usados no pulso. A maioria das empresas opta por criar estes *fitness trackers* em formato de pulseira.



Figura 2-4 — *Smartbands* [16].

Estes dispositivos partilham, por norma, muitos dos sensores dos *smartwatches*, mas são raros os que possuem antenas de GPS e *Wi-fi*. São usados os acelerómetros para contar passos e detetar hábitos de sono, e ainda sensores biométricos para medir a pulsação [15].

### 2.1.3 Serviços e produtos disponíveis

Há alguns serviços ou produtos disponíveis, principalmente nos países desenvolvidos, que deixem os cuidadores menos preocupados para o caso do doente se perder. Como foi visto anteriormente, este é um dos maiores problemas de quem sofre de demência, nomeadamente de Alzheimer.

Na Europa e Estados Unidos da América existem alguns produtos com serviços e subscrições associadas. Um desses programas é o myMDlife, que tem por base uma bracelete (presente na Figura 2-5) com um número de série e um pin, e possibilita ao doente ou a quem encontre o doente, aceder a uma plataforma online proprietária da empresa onde é possível ver várias informações e avisar os cuidadores ou médicos responsáveis [17]. O produto está disponível por cerca de 70 dólares, mas sendo necessários encargos extraordinários caso se pretenda ter acesso à plataforma [18].



Figura 2-5 — Pulseira myMDlife.

Existem também alguns dispositivos mais inovadores como o GPS SmartSole. Este equipamento é uma palmilha para usar nos sapatos, que possui GPS. Através da aplicação os cuidadores podem ter acesso à localização do doente. A bateria pode ser carregada por indução e tem uma duração até 48h. O produto presente na Figura 2-6 tem um valor de venda ao público de 299 dólares e uma subscrição associada de quase 25 dólares, que permite o acesso à aplicação da empresa [19].



Figura 2-6 — Palmilha GPS SmartSole.

Também nos Estados Unidos da América, existe uma empresa, a AngelSense, que comercializa diversos *wearables*, como cintos e camisolas (Figura 2-7) que possuem GPS e enviam a informação para uma aplicação da empresa, na qual os cuidadores podem ter acesso à

localização do dispositivo caso paguem uma mensalidade. Estes *wearables* têm também características interessantes como possuir microfones e colunas para realizar uma comunicação entre o cuidador e o doente [20].



Figura 2-7 — *Wearables* da AngelSense.

Em Portugal o panorama é muito menos tecnológico. O programa Estou Aqui, da Polícia de Segurança Pública (PSP), tem como missão garantir a segurança de adultos, que por várias razões, sofram de desorientação na via pública. Este programa é gratuito, devendo os doentes ou os cuidadores fazer o pedido de uma pulseira. Tal como no programa da myMDlife, a pulseira presente na Figura 2-8 possui um número identificador, mas com a indicação para ligar para o número de emergência 112. A pulseira tem uma validade de 2 anos devendo ser renovada ao chegar ao fim do tempo útil do programa. Esta pulseira não conta com nenhum tipo de localizador [21]. Também da parte das Associações que lidam com pacientes com Alzheimer, não existe indicação de nenhum dispositivo tecnológico a ser utilizado por estes pacientes, sendo as soluções conhecidas baseadas em pulseiras com um número de telefone.



Figura 2-8 — Pulseira do programa Estou Aqui da PSP.

## 2.2 Tecnologias existentes

### 2.2.1 Sensores

Como já foi referido, os *smartphones* e outros *wearables* têm no seu interior vários sensores que permitem ajudar na identificação da localização do dispositivo, e consequentemente da pessoa que o tem na sua posse.

Alguns permitem ter acesso a essa informação de uma forma direta, como por exemplo o GPS, enquanto outros o permitem indiretamente ou não de uma forma tão precisa, como são exemplo os sensores *wi-fi*, *bluetooth* ou o acelerómetro.

#### 2.2.1.1 *Wi-fi (Wireless Fidelity)*

O *wi-fi* é uma tecnologia que permite que vários dispositivos tecnológicos, como computadores ou *smartphones*, estabeleçam uma rede sem fios entre si e se conectem à *internet*, normalmente através de um *Access Point* (AP).

Quando os dispositivos se conectam a um AP formam uma rede, normalmente denominada de *Basic Service Set* (BSS), como a rede da Figura 2-9. Como num local pode haver mais do que uma BSS, estas devem estar identificadas. As redes são identificadas por um conjunto de caracteres denominado *Service Set Identifier* (SSID)[22].

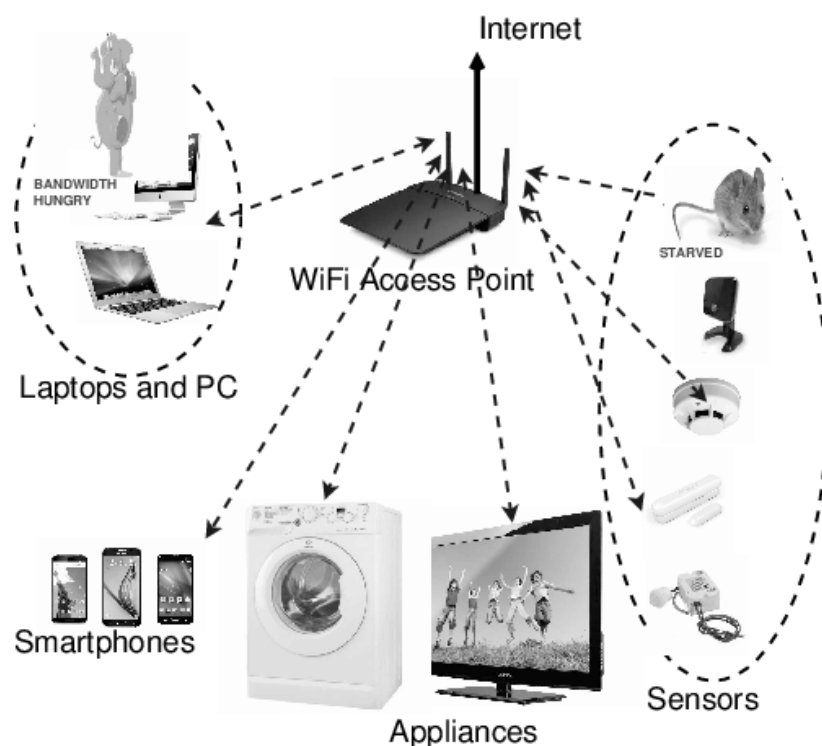


Figura 2-9 — Exemplo de uma rede *wi-fi* [23].

### 2.2.1.2 Bluetooth

O *bluetooth* é uma tecnologia de comunicação sem fios com baixo consumo energético que permite a troca de dados entre dispositivos próximos. Quando dois ou mais dispositivos comunicam entre si formam uma rede *piconet*. Esta rede é caracterizada por uma comunicação do tipo *master-slave*, representada na Figura 2-10. Numa *piconet* apenas um dispositivo pode assumir o papel de *master*, no entanto podem existir até 7 *slaves*.

A tecnologia tem estado em constante evolução e várias versões têm surgido ao longo dos últimos anos. Com as diferentes versões foram feitas diversas melhorias, ao nível da velocidade (na versão 1 a velocidade era até 721Kb/s e na versão 5 pode chegar aos 50Mb/s) e também em relação ao consumo energético [24].

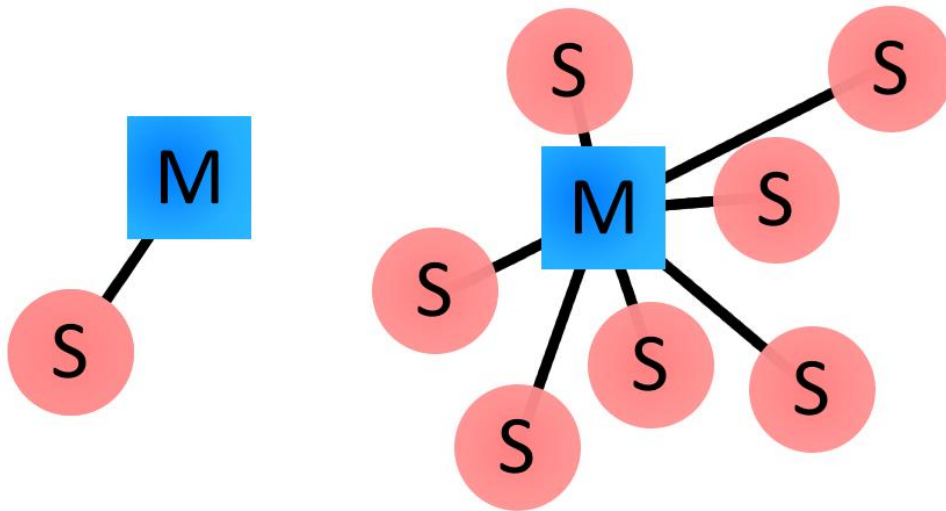


Figura 2-10 — Exemplo de uma arquitetura *master-slave* [25].

### 2.2.1.3 Acelerómetro

O acelerómetro é um sensor que, tal como o nome indica, mede a aceleração de um dispositivo. Este é um sensor de consumo energético muito reduzido. Pode ser utilizado para diversas funções num *smartphone*, desde saber a orientação do ecrã, até algo mais complexo como contar passos.

Um acelerómetro mede a força de aceleração na unidade G. Os acelerómetros presentes em *smartphones* são compostos por 3 eixos (Figura 2-11) através das quais são feitas as medições e obtidos os valores correspondentes. Com estes valores é possível depois efetuar as operações consoante as necessidades [26].

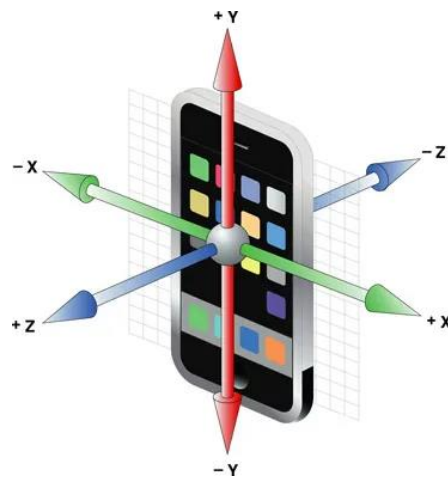


Figura 2-11 — Esquema de 3 eixos de um acelerômetro [27].

#### 2.2.1.4 GPS (*Global Positioning System*)

O GPS tem como objetivo determinar a localização geográfica de uma forma precisa. Para isso são utilizados satélites na órbita do planeta que transmitem informação e permitem medir a distância entre os satélites e o utilizador [28].

Para realizar este cálculo e conseguir uma medição precisa, são necessários pelo menos três satélites. Ao saber a distância aproximada a cada um dos satélites, é possível fazer uma triangulação e com isso obter uma localização bastante precisa. Esta triangulação pode ser observada na Figura 2-11.

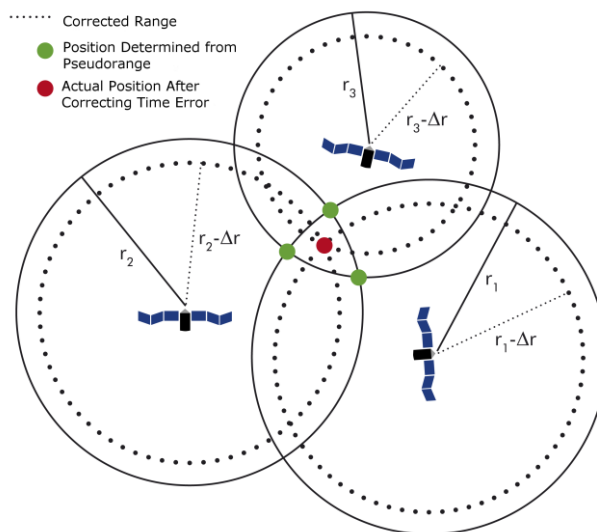


Figura 2-12 — Esquema de triangulação GPS [29].

Atualmente, quase todos os *smartphones* têm antenas GPS, sejam antenas mais tradicionais, ou antenas que necessitam do auxílio de dados móveis ou *wi-fi*, a que normalmente se dá o nome de A-GPS (*Assisted GPS*).

#### 2.2.1.5 Outros sensores

Num *smartphone* existem outros sensores como o de orientação, o de proximidade ou ainda o de NFC (*Near Field Communication*). Atualmente estes sensores estão presentes em quase todos os *smartphones* e são sensores com diversas funcionalidades muito úteis no dia-a-dia de um utilizador.

O sensor de orientação ou rotação é útil ao acelerómetro por exemplo, já que permite saber para onde o utilizador está virado, funcionando como uma bússola [30].

O sensor de proximidade permite saber, entre outras coisas, se um *smartphone* está dentro de algum bolso ou se está virado para baixo numa mesa. Ou seja, consegue medir a proximidade a um objeto [31].

Já O NFC permite a comunicação entre dispositivos muito próximos (por norma até 10 cm). Uma das mais importantes funcionalidades deste sensor é permitir o pagamento em terminais multibanco [32].

### 2.2.2 Gestão energética

Existe literatura sobre vários estudos sobre gestão energética em *smartphones*. Para o objetivo da investigação desenvolvida é importante ter uma ideia do consumo energético dos sensores, para no futuro gerir da melhor forma a sua utilização, maximizando o tempo de vida das baterias.

Para calcular o consumo de energia dos vários sensores podem ser utilizadas as seguintes fórmulas matemáticas:

Equação i — Energia consumida no momento.

$$P_{sensor} = P_{energia} \times T_{tempo}$$

Equação ii — Energia consumida num período de tempo.

$$P_{sensor}(j) = \sum_{t=1}^n P_{sensor}(t)$$

A Equação i representa a energia consumida por um sensor num determinado momento. Já a Equação ii indica o valor de energia consumida por um sensor num determinado período de tempo.

Diferentes sensores apresentam diferentes consumos, como seria de esperar. Na Tabela I é possível verificar como os diferentes sensores têm variações no consumo conforme a situação em que se encontram.

Tabela I — Consumo energético de sensores em percentagem.

Sensor	Movimento Interior	Parado Interior	Movimento Exterior	Parado Exterior
Acelerómetro	6,35%	5,95%	8%	6,6%
Proximidade	6,7%	5,9%	5,9%	6,9%
Orientação	13,23%	9,7%	5,9%	6,9%
GPS	46%	45,22%	53%	51%

Ao analisar a tabela acima, observa-se que o GPS consome bastante energia quando comparado com os outros sensores. No entanto este é o que fornece informações mais exatas da localização (exterior) do dispositivo.

O acelerómetro é muito útil para perceber se o equipamento se encontra em movimento, e consome substancialmente menos que o GPS. Já o sensor de proximidade deteta a presença de objetos ou equipamentos perto do dispositivo através do campo eletromagnético sem ser necessário contacto físico. Enquanto o de orientação permite saber para onde o equipamento está virado, como uma bússola.

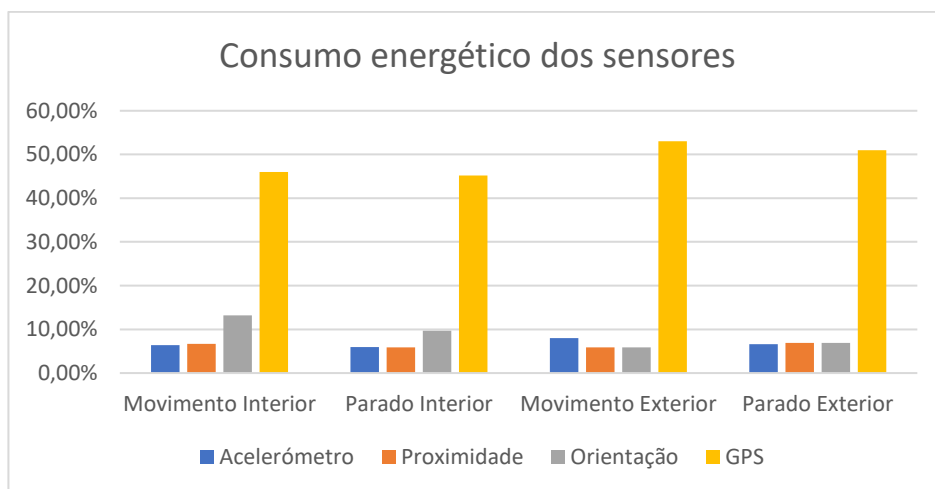


Figura 2-13 — Gráfico representativo do consumo energético de diferentes sensores.

Na Figura 2-13 é mais fácil comparar visualmente o consumo dos diferentes sensores e observar as diferentes oscilações dos seus gastos energéticos.

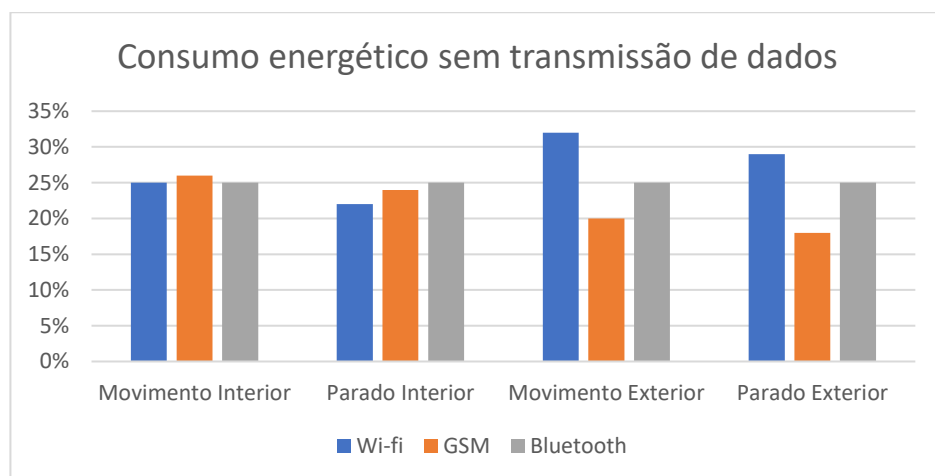


Figura 2-14 — Consumo energético sem transmissão de dados.

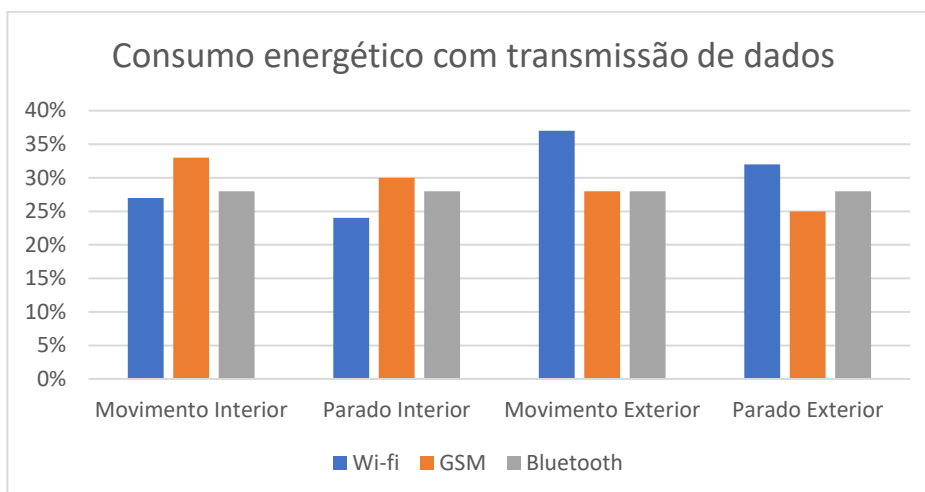


Figura 2-15 — Consumo energético com consumo de dados.

Na Figura 2-14 e na Figura 2-15 podemos ver o consumo de antenas como a *wi-fi*, GSM (*Global System for Mobile Communications*) e *bluetooth*. Verifica-se que no caso do *bluetooth* os consumos são muito similares, não existindo oscilações consoante as situações, apenas importando o facto de transmitir informação ou não. Já com o *wi-fi* e o GSM os resultados obtidos fazem sentido, uma vez que o sinal *wi-fi* tem por norma mais interferências no exterior, originando assim um maior consumo. Enquanto o GSM tem mais dificuldade em espaços interiores com paredes, daí apresentar consumos mais elevados nessas situações.

Estes dados podem ser encontrados no documento “*Sensors are Power Hungry*” [33].

### 2.2.3 Algoritmos de aprendizagem (*Machine Learning*)

*Machine Learning* (ML) é um termo usado na computação, que usualmente se associa à aprendizagem computacional ou ao reconhecimento de padrões. Existem dois tipos de técnicas de classificação, as supervisionadas e as não supervisionadas.

A supervisionada utiliza os dados já conhecidos, dos quais se conhece o resultado, para treinar um modelo que possa gerar previsões para novos dados. É um tipo de aprendizagem muito útil quando se sabe o que esperar nos resultados.

A técnica de classificação não supervisionada tenta encontrar padrões em dados de entrada, no entanto, sem saber previamente os resultados esperados. Serve, por exemplo, para agrupar informação [34].

### 2.2.3.1 Classificadores *Fuzzy*

A classificação *fuzzy* consiste num modelo de regras em que as variáveis de entrada estão representadas num intervalo entre 0 e 1, como está representado na Figura 2-16. As regras são descritas por operadores lógicos e criam relações entre conjuntos difusos. Estas regras são muitas vezes atribuídas por quem está a tratar da informação. São estas regras que determinam, numa grande maioria dos casos, a eficiência deste modelo de classificação [35].

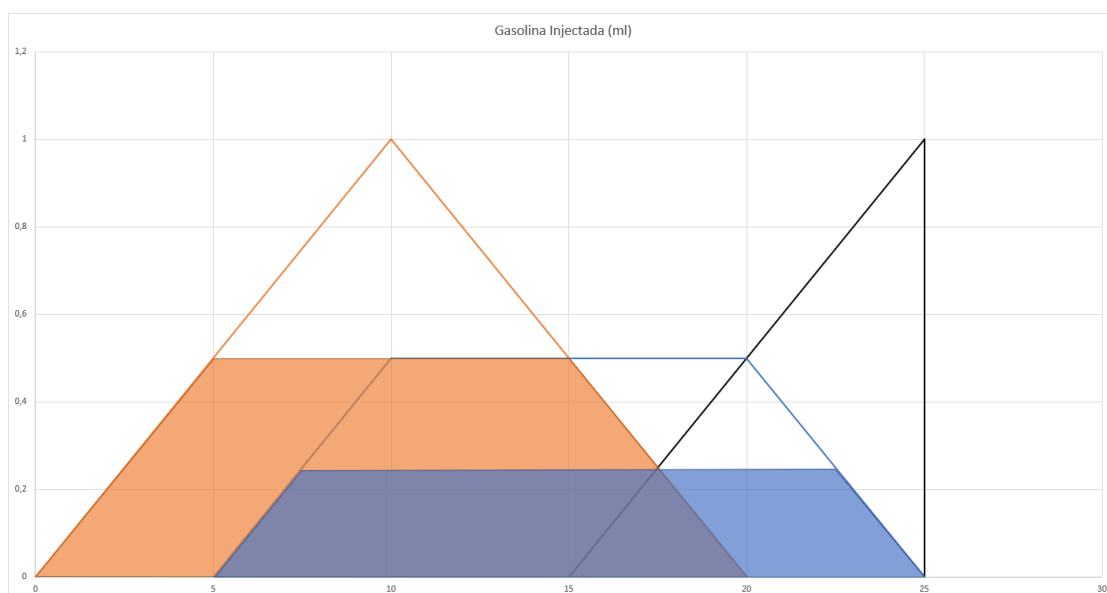


Figura 2-16 — Gráfico resultante da lógica Fuzzy.

### 2.2.3.2 Redes Neurais (RN)

As redes neuronais tentam replicar o cérebro humano nas suas capacidades em classificar informação ou identificar padrões. Tal como no cérebro, uma RN possui neurónios capazes de processar informações. Numa RN os neurónios são compostos por um conjunto de sinapses (ligações caracterizadas com pesos), um somador que adiciona os sinais de entrada aos pesos e por uma função de ativação que tem o objetivo de limitar a amplitude de saída do neurónio [36]. Este modelo pode ser observado na Figura 2-17.

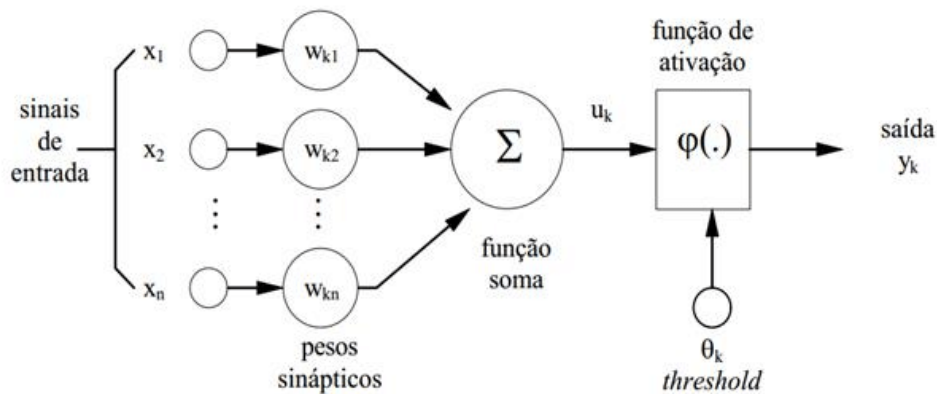


Figura 2-17 — Modelo de um neurónio numa RN [37].

### 2.2.3.3 Árvores de decisão

As árvores de decisão são um algoritmo para tomar decisões ou classificar algo. O nome advém da sua morfologia, já que faz lembrar uma árvore com ramos e folhas no final como pode ser observado na Figura 2-18.

Uma árvore de decisão é composta por nós e por folhas em que os nós podem dar origem a outros nós ou a folhas. Já as folhas são a terminação do ramo da "árvore" e o local onde é feita a classificação.

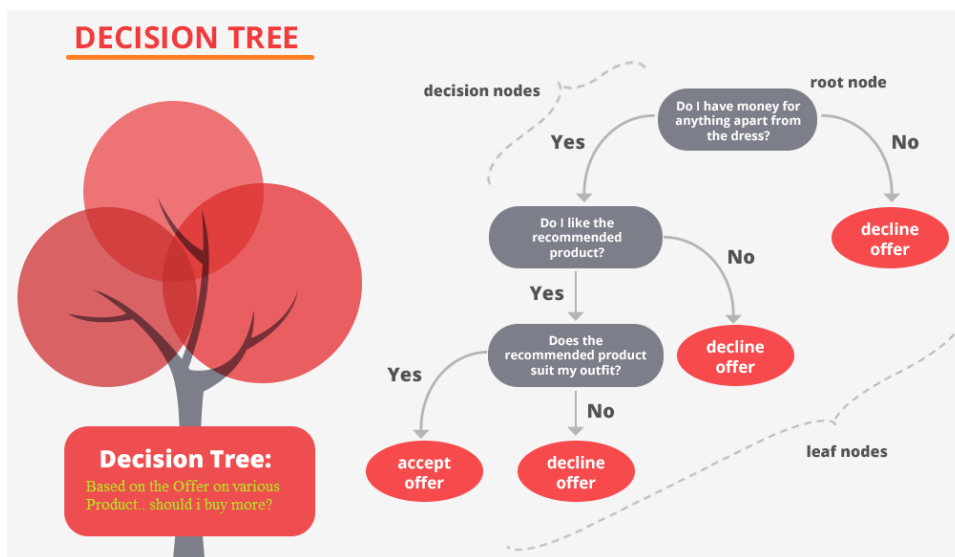


Figura 2-18 — Exemplo de uma árvore de decisão [38].

A árvore é construída nó a nó em que cada um representa uma decisão que é tomada consoante os dados do conjunto. Este processo de construção da árvore dá-se o nome de indução e muitas vezes exige um enorme poder computacional. Esta divisão do conjunto inicial

em subconjuntos vai aumentando a cada nó e estes vão ficando cada vez mais puros. Considera-se que um subconjunto é mais puro à medida que contém menos classes da variável *target*. Para realizar estes cálculos e fazer a tomada de decisões é utilizado o critério da impureza, que tem como objetivo minimizar a impureza em cada nó. Para isso são muito importantes dois conceitos, a entropia e o ganho de informação [39].

### 2.2.3.3.1 Entropia e ganho de informação

A entropia é uma forma de calcular a pureza de cada um dos subconjuntos de uma árvore de decisão. Por outras palavras é uma forma de calcular a probabilidade de um elemento escolhido aleatoriamente do conjunto, ser avaliado incorretamente caso fosse escolhido aleatoriamente ao ser feita a divisão do subconjunto.

A entropia de um nó é dada por:

Equação iii — Entropia num nó.

$$i(t) = - \sum_{i=1}^N p_i \cdot \log_2 p_i$$

Na Equação iii está presente a forma de calcular a entropia de um nó.  $N$  é o número de classes e  $p_i$  é a probabilidade da classe  $i$  [40].

Já na Equação iv resulta o ganho de informação.

Equação iv — Ganho de informação.

$$I(C; A) = H(C) - \sum_i p(t_i^A) H(C|t_i^A)$$

O ganho de informação baseia-se na redução da entropia após uma divisão do conjunto, consoante a aplicação de uma determinada regra. Essa redução da entropia torna o conjunto mais homogéneo [40].  $H(C)$  representa a entropia original do nó,  $p(t_i^A)$  a probabilidade da partição  $i$  e  $H(C|t_i^A)$  a entropia dessa mesma partição [40].

De uma forma geral pode-se dizer que uma árvore de decisão se baseia em encontrar regras através das variáveis do modelo que retornam o maior ganho de informação com o mínimo de entropia.

### 2.2.3.3.2 Validação cruzada e *overfitting*

Existem várias técnicas para avaliar a precisão de um modelo, e uma das mais utilizadas é a técnica de validação cruzada (*V-Fold Cross Validation*). A técnica consiste em dividir o conjunto de dados em dois subconjuntos, um de treino e outro de teste.

Por norma são utilizados 90% dos casos para testar o modelo e 10% para realizar os testes. Um modelo que acerta demasiado nas decisões que deve tomar nem sempre deve ser considerado bom, já que se pode estar perante um problema de *overfitting*.

O *overfitting* acontece quando o modelo está demasiado adaptado ao conjunto de treino, mas falha em generalizar quando é testado em dados desconhecidos. Existem algumas técnicas para evitar esta situação e uma das mais utilizadas é a poda. A poda, tal como o nome indica, consiste em “podar” ramos da árvore de decisão de forma a não a deixar crescer até ao fim, ou seja, a possibilitar uma generalização do problema evitando assim o *overfitting* [41].

### 2.2.3.3.3 Critério de Gini

O critério de Gini ou a impureza de Gini, implementado no *software* CART, é um critério de medição de quantas vezes um elemento de um conjunto que é escolhido aleatoriamente, será classificado incorretamente caso seja classificado aleatoriamente de acordo com a classificação do subconjunto [42].

Equação v — Critério de Gini.

$$i(t) = 1 - \sum_{Classes} (p(Classe|t))^2$$

A impureza de Gini pode ser calculada através da Equação v [40]. Varia entre 0 e 1 já que é uma probabilidade e quanto mais perto de 1 é o valor, mais puro é o nó.



## METODOLOGIA E DESENVOLVIMENTO

Neste capítulo será apresentado todo o trabalho desenvolvido. Começa por ser feita uma referência aos materiais e ferramentas utilizadas, de seguida são feitos alguns testes para testar o consumo energético com a finalidade de criar uma tabela de dados para ser analisada posteriormente pelo CART. Por fim, é explicado como foi criada a aplicação pretendida de acordo com os resultados obtidos no CART.

### 3.1 Dispositivos e ferramentas utilizadas

Para realizar o estudo e conseguir obter os resultados desejados, foi necessário um equipamento para conseguir analisar consumos energéticos, e foi também preciso encontrar *software* indicado para uma correta análise e tratamento dos dados, e por fim para o desenvolvimento posterior de uma aplicação que consiga colocar em prática o sistema criado após esse tratamento da informação.

#### 3.1.1 *Smartphone* utilizado

O *smartphone* utilizado para realizar os vários testes foi o dispositivo da marca BQ, modelo Aquaris M5, na Figura 3-1. Este *smartphone* tem a versão 7.1.2 do SO Android, e está equipado com os vários sensores necessários para a realização deste estudo.



Figura 3-1 — BQ Aquaris M5: o dispositivo utilizado.

Está equipado com *wi-fi*, *bluetooth*, GPS e também antenas 4G. Possui ainda giroscópio, bússola, sensor de proximidade e acelerómetro. Atualmente há equipamentos que já possuem contador de passos, no entanto uma grande parte ainda não está equipada com esse tipo de sensores, muito menos os *smartphones* da população com idade mais avançada, que muitas vezes tem equipamentos mais antigos. Por essa razão será utilizado o acelerómetro para a função de contar passos.

O *smartphone* tem uma bateria de 3120mAh (miliampere por hora, medida convencional para medir a capacidade das baterias dos *smartphones*), e tem capacidade para suportar várias horas com uma utilização muito intensiva (em que seja necessária uma localização constante) ou capacidade para um a dois dias de duração num uso mais comum [43].

### 3.1.2 *Software* utilizado

Como foi possível verificar na Figura 2-2 o Android é o SO mais utilizado em todo o mundo e existem diversas ferramentas para a criação de aplicações. Já as ferramentas para criar algoritmos de aprendizagem não são tão populares, mas são cada vez mais e mais utilizadas.

O SO Android é baseado num *kernel/Linux open source* em que o principal desenvolvedor é a Google. O Android foi criado tendo em vista a utilização em dispositivos móveis com ecrã tátil.

Como o seu código é aberto aos desenvolvedores, qualquer pessoa com uma ferramenta de desenvolvimento pode criar a sua aplicação e utilizá-la no seu dispositivo [44].

### 3.1.2.1 Android Studio e Java

A ferramenta escolhida para programar a aplicação para o *smartphone* Android foi o Android Studio.

O Android Studio é o *software* oficial da Google para desenvolver aplicações para o seu sistema operativo Android, baseado no IntelliJ IDEA, um Ambiente de Desenvolvimento Integrado (*Integrated Development Environment - IDE*) para máquinas virtuais de Java (*Java Virtual Machines - JMV*) [45]. A primeira versão oficial (versão 1.0) foi lançada em dezembro de 2014. Está disponível para os sistemas operativos Windows, macOS e Linux e a sua interface pode ser observada na Figura 3-2.

As linguagens de programação suportadas pelo software para o desenvolvimento de aplicações são o Java, o Kotlin e o C++. O *Kit* de Desenvolvimento de *Software*, conhecido por SDK (*Software Development Kit*), é necessário para compilar a aplicação e este foi recebendo várias atualizações ao longo dos anos e também novas funcionalidades. Estas funcionalidades chegam nos pacotes de imagem do sistema, não mais que *Application Programming Interfaces* (API's) [46].

Para desenvolver a aplicação optou-se por utilizar a linguagem Java. É uma das linguagens mais utilizadas em todo o mundo, tem muita informação disponível e muitas ferramentas já desenvolvidas que são úteis para o problema. O Java é uma linguagem de programação orientada a objetos, que possui facilidade na criação de programas multitarefa e também na organização do código [47].

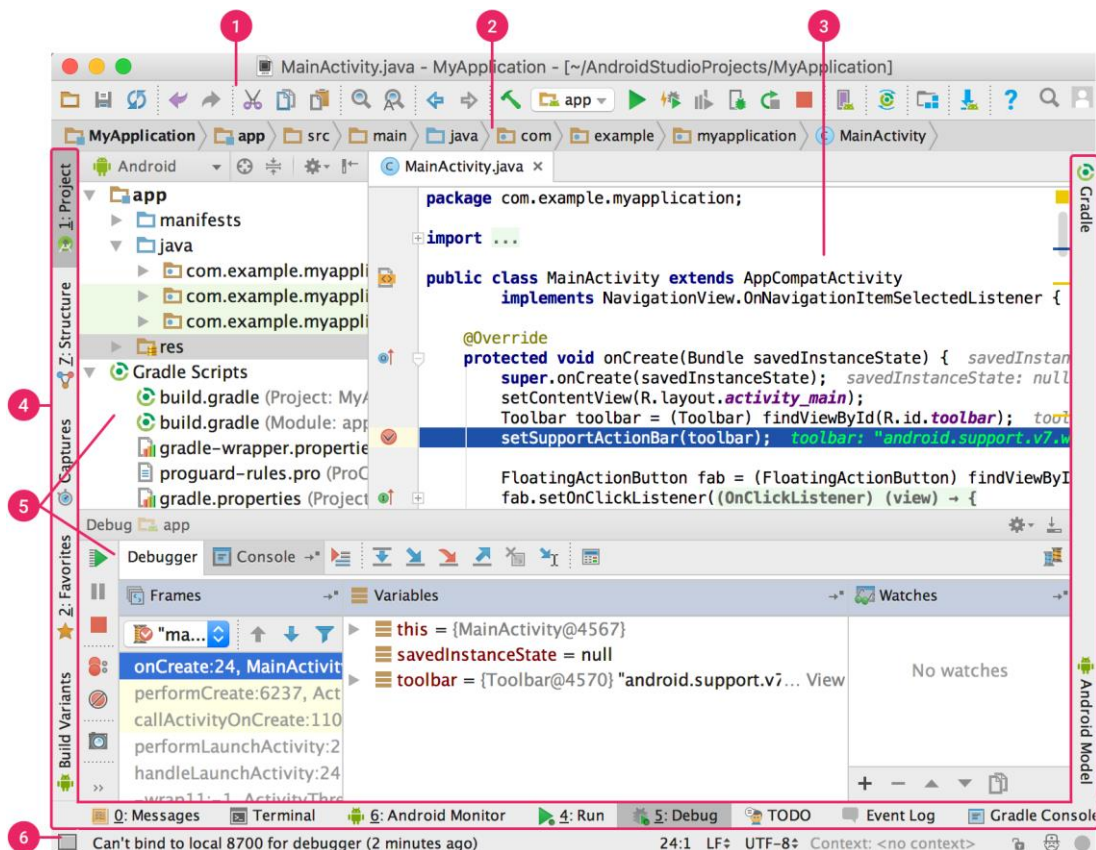


Figura 3-2 — Janela principal do Android Studio.

A cada API mais recente vão existindo novas funcionalidades, ou as que já existem vão sendo aplicadas de novas formas. Isto faz com que nem todos os smartphones ou outros dispositivos Android suportem todas as API's. Isto pode ter um grande impacto na versão que deve ser escolhida quando se começa a desenvolver uma aplicação em Android [48].

A API escolhida foi a 21 já que engloba quase a totalidade dos dispositivos Android utilizados atualmente, mesmo alguns modelos mais antigos, que até poderão ser os mais utilizados pela população alvo desta dissertação [49].

Por ser o software oficial para o desenvolvimento das aplicações em Android e por todas as atualizações e funcionalidades em constante evolução, o Android Studio revela-se a escolha mais acertada para a criação de aplicações para *smartphones* com o SO Android.

### 3.1.2.2 CART

Já para a construção de uma árvore de decisão que originasse um bom modelo de gestão energética, a escolha recaiu para o CART.

O CART é uma ferramenta de árvores de decisão da Salford Systems, que procura automaticamente padrões e relações importantes num conjunto de dados. As árvores geradas podem ser utilizadas para gerar modelos preditivos bastante precisos e confiáveis.

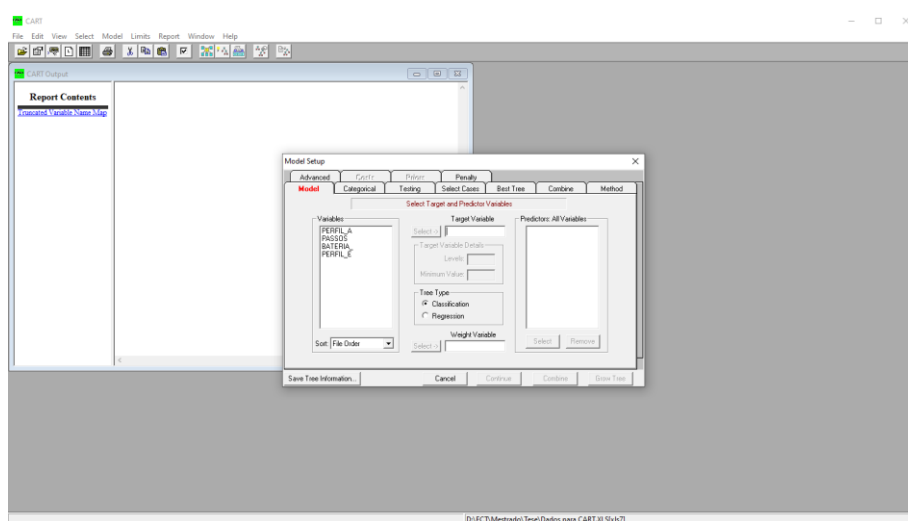


Figura 3-3 — Ambiente de trabalho do CART.

Na Figura 3-3 pode-se observar o ambiente de trabalho do CART. O *software* tem todas as ferramentas associadas às árvores de decisão, desde a validação cruzada às técnicas de poda para evitar o *overfitting* [50].

## 3.2 Avaliação dos consumos energéticos

Para confirmar algumas das informações relativas ao consumo energético fizeram-se testes para os vários sensores do *smartphone*. Para isso foram repetidos 3 vezes para cada antena pequenos testes de *stress* da bateria em condições semelhantes com o objetivo de aferir o impacto que estas antenas têm no consumo energético.

Estes testes foram todos realizados nas mesmas condições:

- Ecrã sempre ligado;
- Luminosidade do ecrã no máximo;
- Todas as aplicações do *smartphone* fechadas;
- Nível inicial da bateria com carga no máximo (100%).

Cada um dos testes teve a duração de uma hora, com o objetivo de verificar o possível impacto energético de ter o *wi-fi*, o *bluetooth* e o GPS ativos. Para isso foram criadas aplicações que ativassem através de botões as antenas pretendidas para o teste. Foram realizados testes com todas desligadas, com cada uma delas ligadas e com todas ligadas.

Inicialmente foi testada uma aplicação em que nada era ativado. Continha apenas um "hello world" e uma instrução para o ecrã ficar sempre ligado. Depois foi testada uma aplicação que ligava apenas o *wi-fi*, seguida de outra que ligava apenas o *bluetooth* e por fim uma que ativava os serviços de localização e mantinha o GPS ligado. É possível ver capturas de ecrã destas aplicações criadas para o efeito de teste no apêndice A.3.

Tabela II — Média de percentagem de bateria restante após 1h de teste.

<i>Wi-fi</i>	<i>Bluetooth</i>	GPS	Média de % bateria após 1h
D	D	D	93
L	D	D	89
D	L	D	90
D	D	L	88
L	L	L	86

Na Tabela II as antenas ligadas estão representadas pela cor verde e as desligadas pela cor vermelha. O resultado dos testes comprova numa grande parte o que já estava documentado no subcapítulo 2.2.2, já que se observa que há um impacto direto no consumo energético por parte destes componentes do *smartphone*. Nota para a diferença do GPS para os restantes não se ter revelado tão substancial como o esperado, talvez por o tempo de atualização da localização escolhido não ter sido o mais exigente para a bateria (o tempo escolhido foi de 30 segundos).

Estes testes permitiram assim confirmar, que um sistema que consiga tornar mais eficiente o consumo energético e que mantenha de alguma forma um acompanhamento da localização do *smartphone* pode ser bastante útil para os doentes com Alzheimer e outras demências.

### 3.3 Perfis e dados para análise

Com a ajuda dos resultados da Tabela II o passo seguinte foi arranjar uma forma de simplificar o sistema para ser implementado no CART e por conseguinte tirar melhores e mais eficientes conclusões. Assim, foram criados perfis que representam os diferentes estados das antenas.

Para a criação dos perfis foi também considerada a antena GSM dos dados móveis já que esta também ajuda na localização do *smartphone*. De notar o facto de esta não ter sido considerada para os testes anteriores porque se pressupõe que o utilizador do *smartphone* poderá querer ter uma grande parte do tempo os dados móveis ligados. Com o aumentar dos limites dos dados móveis, cada vez menos se justifica desligar os mesmos exceto em casos de otimização de consumo energético, que como se verá já de seguida nos perfis será o caso.

Para além desta situação o Android não permite a uma aplicação ligar ou desligar os dados móveis. Apenas permite recorrer ou não aos dados móveis para obter informação. Esta opção foi removida na versão 5.0 do Android (API 21) [51].

#### 3.3.1 Perfis

Com base nos dados recolhidos em 3.2 foram criados cinco perfis para fazer a gestão energética do *smartphone*. Cada um destes perfis tem as antenas ligadas associadas e também uma duração (em minutos).

Tabela III — Definição e características dos perfis.

Perfil	Wi-fi	Bluetooth	Dados móveis	GPS	Duração (m)
0	D	D	D	D	30
1	L	D	D	D	15
2	D	L	L	D	5
3	L	D	L	D	5
4	L	D	D	L	2

Como é possível observar na Tabela III existem cinco perfis numerados entre 0 e 4. A cor vermelha representa mais uma vez que a antena está desligada e a cor verde representa que a antena está ativa. A duração representa o tempo que o perfil está ativo até ser feita uma nova escolha pelo sistema.

No perfil 0 está tudo desligado e a duração do perfil é de 30 minutos. Este perfil foi concebido para situações em que o paciente pode estar num local seguro como a sua casa e

não se desloca durante muito tempo, como por exemplo o período da noite. Daí a duração deste perfil ser a mais longa.

Já no perfil 1 apenas o *wi-fi* está ligado e a duração é um pouco mais curta, ou seja, as atualizações já se tornam mais frequentes. Este perfil foi criado para situações em que o paciente possa estar em casa ou noutra local com acesso *wi-fi* que seja conhecido.

O perfil 2 é um pouco diferente dos restantes, uma vez que é o único que tem o *bluetooth* ativo. Para além do *bluetooth* tem também os dados móveis para conseguir obter algum tipo de localização, mas sem ter de recorrer às antenas GPS. O perfil 2 foi pensado para situações em que o doente se possa deslocar num carro equipado com *bluetooth* em que se desloca por vezes distâncias grandes, mas sem dar muitos passos. Daí a sua duração ser de 5 minutos.

O perfil 3 é uma versão mais precisa do perfil 1 em termos de localização da pessoa, uma vez que para além da rede *wi-fi* recorre também aos dados móveis para obter a localização. Por ser um perfil que necessariamente também vai ser mais prejudicial para o nível da bateria do *smartphone*, a sua duração é também de 5 minutos e por isso mais reduzida que o perfil 1.

Por fim, o perfil 4 é o que terá um custo energético maior, daí a sua duração ser a mais curta de todos os perfis. Neste perfil está ativo o GPS que é a forma de obtenção de localização mais precisa e é ainda auxiliado pelo *wi-fi*.

### 3.3.2 Dados para CART

Depois da definição dos perfis que serão a variável *target* da árvore de decisão, foi necessário definir outras variáveis para as tomadas de decisão. Estas variáveis devem ser importantes para o problema e foi por essa razão que as variáveis definidas para a análise do problema foram as seguintes:

- Perfil anterior;
- Passos;
- Nível da bateria.

O perfil anterior deve ser considerado importante já que se pode ter por base um critério de continuidade, ou seja, se o utente está em casa a dormir isso deve ser tido em consideração e deve existir uma probabilidade de continuar no mesmo estado. Tal como o caso em que uma

pessoa está na rua e o smartphone está a recorrer ao GPS. São duas situações que exemplificam este critério.

Os passos são extremamente importantes, uma vez que são um bom input do nível de deslocamento da pessoa. Existem alguns estudos sobre os passos médios que idosos dão num determinado período de tempo, e a sua cadência por minuto. Num estudo da Universidade de Massachusetts Amherst [52], concluiu-se que (dependendo sempre da sua condição física claro) os idosos podem chegar a cadências de mais de 100 passos por minuto. Como este estudo pretende englobar uma grande variedade de casos este fator foi tido em grande consideração. No ficheiro Excel os passos vão até 250, já que a partir desse valor o que a aplicação deverá fazer é sempre o mesmo.

A última variável que também tem uma grande importância é a do nível da bateria do *smartphone*. Há situações em que o sistema sabe que existem níveis de bateria mais "confortáveis" e pode tirar partido disso para escolher um perfil que apesar de ser pior a nível energético pode devolver uma localização mais precisa, ou então a situação inversa. Foi tida em consideração uma situação específica para classificação inicial dos dados. Quando a bateria é inferior a 10% o sistema opta por ativar o perfil 4 para ter uma localização precisa e enviar uma SMS (*Short Message Service*), uma mensagem de texto) com a localização do doente.

Para os dados a fornecer ao *software* CART foi criada uma tabela com dados aleatórios, em que apenas a variável *target* foi escolhida tendo em conta o que se pretendia da aplicação. Portanto, apenas as variáveis perfil anterior, passos e bateria foram geradas aleatoriamente no Microsoft Excel através da função: =RAND().

Tabela IV — 10 dos 300 casos para serem analisados pelo CART.

Perfil atual	Passos	Bateria atual	Perfil escolhido
4	98	63	4
0	140	27	1
0	226	84	4
3	217	99	4
0	133	34	3
2	30	5	4
2	140	25	4
2	180	84	4
1	208	86	4
3	250	67	4
4	159	84	3
...	...	...	...

Foram criados 300 casos para análise e na Tabela IV pode ser visto um pequeno excerto da mesma. Os dados foram gerados aleatoriamente para tentar englobar o máximo de situações possíveis. Na variável *target* tentaram-se criar poucos erros para ver se a árvore conseguia aprender ao analisar os casos e para evitar um problema de *overfitting*.

## 3.4 Software CART

Ao colocar os dados no CART foi necessário modelar corretamente a informação para a criação da árvore.

### 3.4.1 Separador *Model*

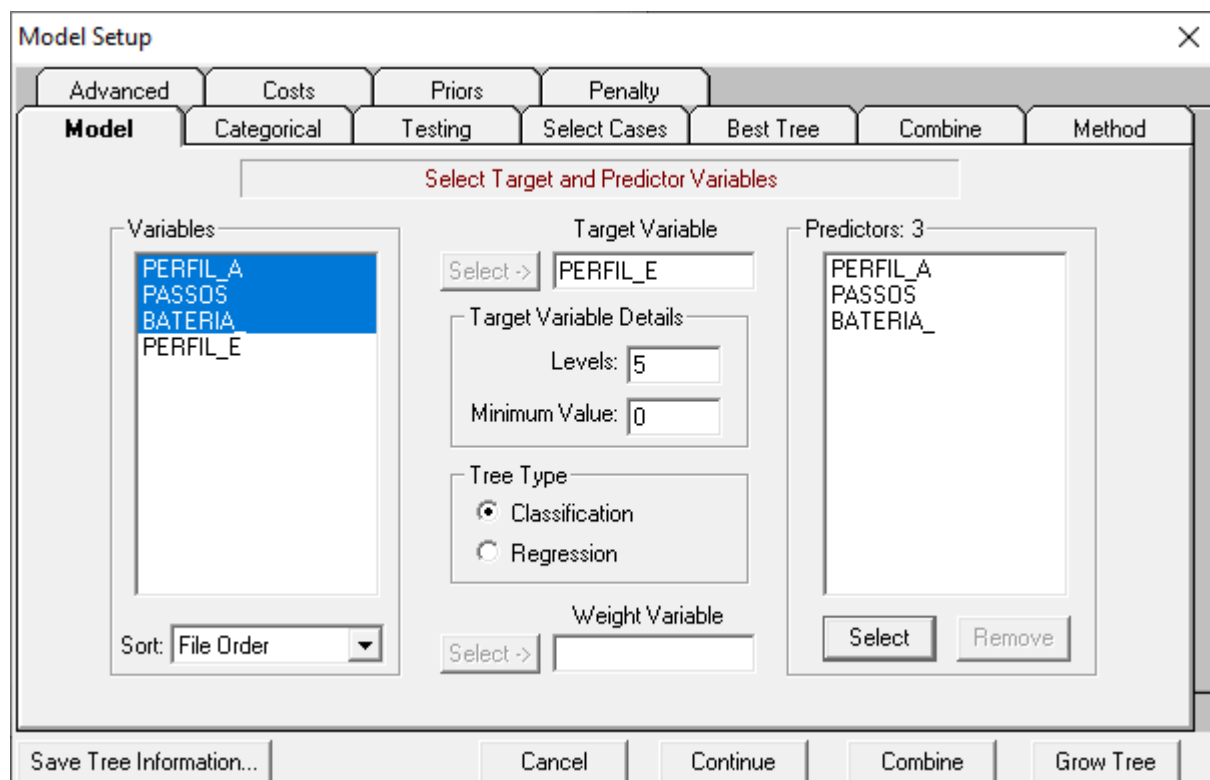


Figura 3-4 — Modelação das variáveis de entrada e da variável *target*.

O CART identifica automaticamente através da primeira linha das colunas as variáveis presentes no ficheiro, que se encontram no setor *Variables* da Figura 3-4. Depois foi necessário definir quais são as de entrada e qual delas é a *target*. Neste caso foram definidas nos setores

*Predictors* e no setor *Target Variable* respetivamente. As variáveis PERFIL\_A, PASSOS e BATERIA\_ foram definidas como *Predictors* e a variável PERFIL\_E definida como *target*. Foi ainda preciso configurar o número de perfis que existem e em que número os perfis começam, daí o setor *Target Variables Details* na Figura 3-4 estar definido com 5 *Levels* e 0 no *Minimum Value*.

### 3.4.2 Separador *Categorical*

No separador *Categorical* são definidas quais as variáveis de entrada que são categóricas. Neste caso foi apenas definida a variável PERFIL\_A que corresponde ao perfil anterior.

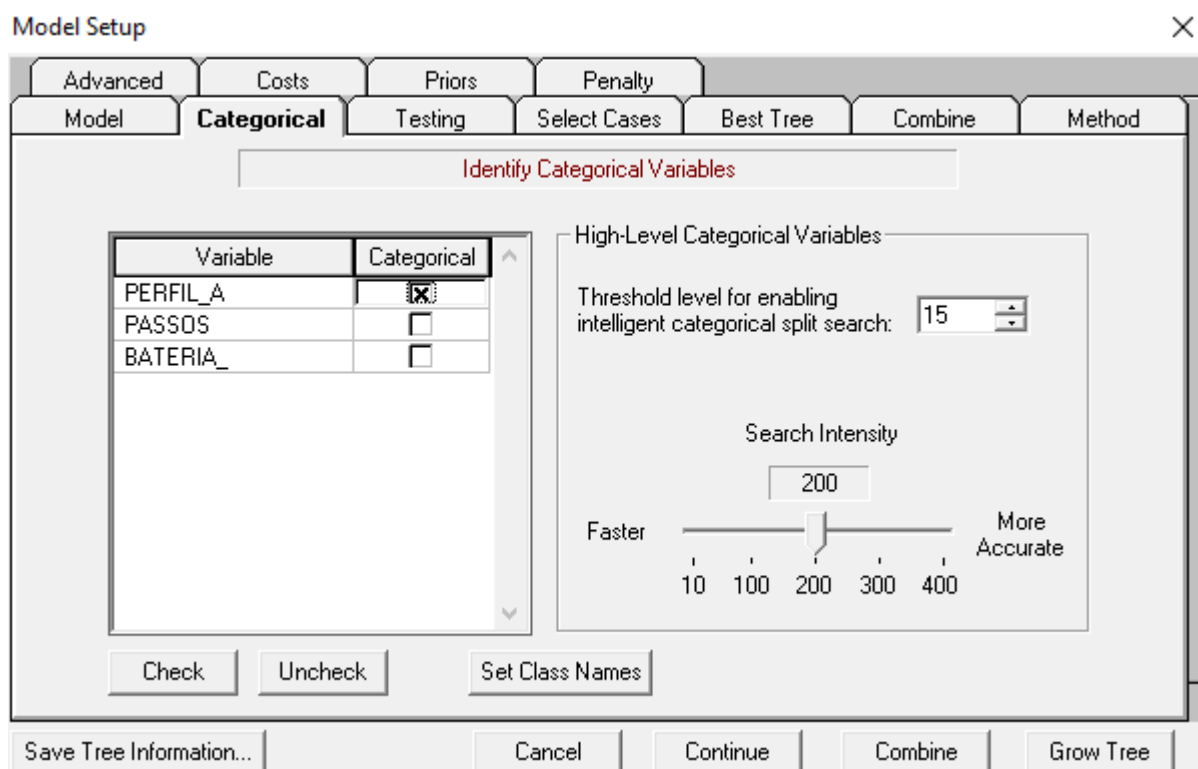


Figura 3-5 — Separador *Categorical*.

A variável categórica é uma variável discreta. As variáveis PASSOS e BATERIA\_ podem ter o valor de qualquer número natural e não se pretende que sejam categorizadas, daí não terem sido escolhidas como categóricas.

### 3.4.3 Matriz de custos

Como há erros nas decisões que a árvore toma, que têm maior impacto que outras, é muito importante decidir quais os custos que esses erros têm. É precisamente isso que se define numa matriz de custos.

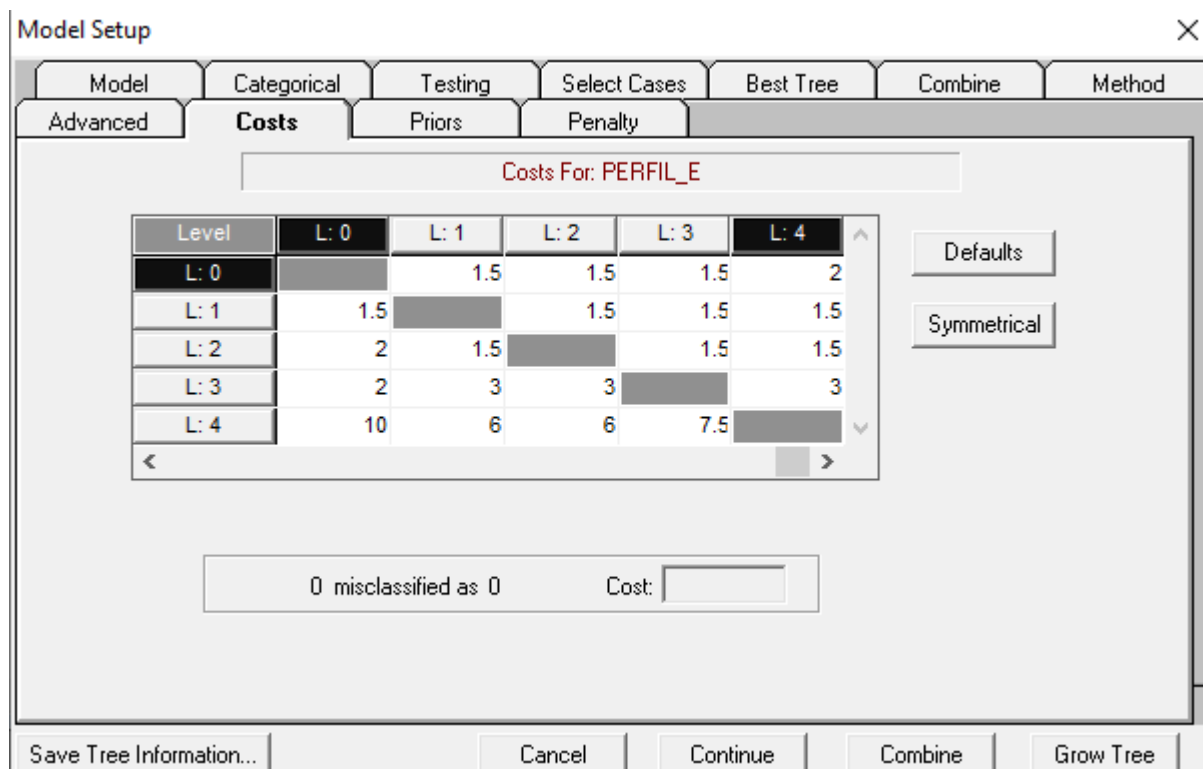


Figura 3-6 — Matriz de custos do problema.

Para definir a matriz de custos da Figura 3-6 foram feitos vários ajustes a fim de obter os resultados observados no capítulo 4.1.1, ou seja, fizeram-se vários testes com diferentes custos para ajustar a árvore ao problema e com isso conseguir melhores resultados.

### 3.4.4 Outras definições da modelação

Para a criação da árvore de decisão foram ainda definidos o critério de Gini para a escolha dos nós e o método de validação cruzada para testar a árvore. Estes métodos foram abordados nos capítulos 2.2.3.3.3 e 2.2.3.3.2 respetivamente.

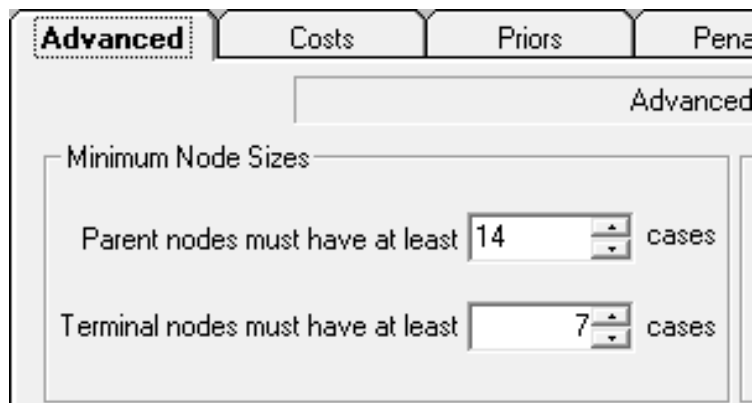


Figura 3-7 — Número de casos necessários para construção da árvore.

Na Figura 3-7 é possível ver ainda outro parâmetro importante que foi definido. O número de casos necessários para ser criada uma folha e um nó. É usual o número de casos comuns necessários para ser criado um nó ser o dobro dos casos para ser criada uma folha, daí a escolha de 14 casos para a criação de um nó e 7 casos para a criação de uma folha.

## 3.5 Android Studio

Após ter uma árvore já definida começou a ser desenvolvida a aplicação final para a resolução do problema proposto nesta dissertação.

### 3.5.1 Interface do usuário (IU)

A aplicação tem como conceito fundamental ser autónoma e não necessitar da ajuda do utilizador para ativar antenas, no entanto é uma mais-valia e algo que pode ser útil para um doente que esteja perdido, ter botões de fácil interação para ligar e desligar o *wi-fi*, o *bluetooth* ou o GPS. Ao ter esta funcionalidade o utilizador pode aumentar ou diminuir a qualidade do acompanhamento feito pela aplicação em alturas de maior ou menor necessidade.

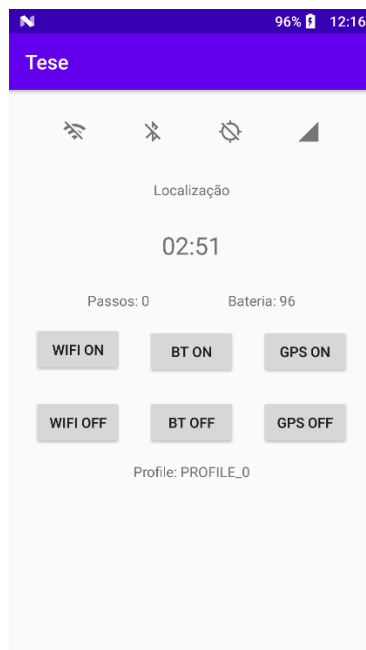


Figura 3-8 — Interface do usuário.

Numa IU pretende-se que seja de fácil perceção e interação caso exista a necessidade. Foi isso que se tentou criar nesta aplicação. Na Figura 3-8 está uma captura de ecrã da aplicação em funcionamento.

Na parte superior estão pequenos ícones com imagens a representar o *wi-fi*, o *bluetooth*, o GPS e os dados móveis (da esquerda para a direita respetivamente). Estas imagens mudam consoante o estado das antenas; se estão ativas ficam coloridas, como se pode observar na Figura 4-6.

Por baixo dos ícones encontra-se o local onde é mostrada a localização atual, seguido do temporizador, que faz uma contagem decrescente com o tempo pré-determinado pelo perfil ativo, e de algumas das informações necessárias para a árvore de decisão, nomeadamente os passos dados e o nível da bateria.

Após estas informações importantes estão os botões para o utilizador poder interagir com as diferentes antenas, e por fim está a informação do perfil atual.

Esta interface foi criada através de um ficheiro .xml, que pode ser consultado no apêndice A.4.

### 3.5.2 Serviços *wi-fi*, *bluetooth*, dados móveis e GPS

Para os serviços, que são as classes Java que têm o objetivo de controlar o estado das antenas e tudo o que se relaciona com elas, criou-se uma classe abstrata. Esta é uma classe genérica para todos os outros serviços, já que todos eles partilham as mesmas funções.

```
public abstract void turnON();

public abstract void turnOFF();

public abstract boolean isEnabled();

public abstract void setImage(boolean isEnabled);

public void setImage() { setImage(isEnabled()); }

public void changeState(){
    if (!isEnabled()) {
        turnON();
    } else {
        turnOFF();
    }
}
```

Figura 3-9 — Classe *BaseController.java*: classe abstrata para os serviços.

Na Figura 3-9 é possível ver as funções comuns aos vários serviços:

- *TurnON*- função para ligar a antena;
- *TurnOFF*- função para desligar a antena;
- *IsEnabled*- Função para verificar o estado da antena;
- *SetImage*- Função para colocar a imagem certa na IU;
- *ChangeState*- Função para alterar o estado da antena (ligar ou desligar).

```

@Override
public void turnON(){
    showToast( msg: "Wi-Fi On");
    wifiManager.setWifiEnabled(true);
    setImage(true);
}

@Override
public void turnOFF(){
    showToast( msg: "Wi-Fi Off");
    wifiManager.setWifiEnabled(false);
    setImage(false);
}

@Override
public boolean isEnabled() { return wifiManager.isWifiEnabled(); }

@Override
public void setImage(boolean isEnabled) {
    if (isEnabled)
        image.setImageResource(R.drawable.ic_wifi_on);
    else
        image.setImageResource(R.drawable.ic_wifi_off);
}

```

Figura 3-10 — Serviço *wi-fi* (classe *WifiController.java*).

Na Figura 3-10 está o serviço *wi-fi* onde é possível ver as funções presentes na classe *BaseController.java*. Para o serviço *wi-fi* é essencial chamar uma classe do Android com o nome de *Wifi Manager*, que é responsável pelas funções relativas ao *wi-fi*. Como se pode ver na Figura 3-10 esta classe é chamada, por exemplo, na função *turnON* para ativar a antena na quarta linha.

O serviço *bluetooth* é semelhante, mas a classe que é chamada para controlar a antena tem o nome de *BluetoothAdapter*.

O serviço GPS é um pouco diferente, já que como foi referido no capítulo 3.3 os dados móveis não podem ser ativados sem ser pelas próprias definições do sistema. No entanto tornou-se esta situação através dos modos de funcionamento dos serviços de localização.

Os serviços de localização têm diversos modos que podem ser escolhidos. Para este problema foram utilizados dois modos:

- `PRIORITY_HIGH_ACCURACY`- que é o modo de alta precisão em que é utilizada a antena GPS. Este modo tem uma precisão de cerca de 10 metros;
- `PRIORITY_BALANCED_POWER_ACCURACY`- modo em que são utilizados os dados móveis e o *wi-fi* se disponível para tentar obter uma localização aproximada. Este modo tem uma precisão de cerca de 100 metros[53].

Este modo é um parâmetro de entrada do serviço GPS e é recebido consoante o perfil que foi selecionado como é possível observar no capítulo 3.5.7.

### 3.5.3 Acelerómetro como contador de passos

O serviço do acelerómetro é responsável pela contagem de passos do utilizador. Para recorrer ao acelerómetro foi necessário utilizar a classe Java, *Sensor Manager*, que tem acesso aos sensores do acelerómetro.

```
@Override
public void onSensorChanged(SensorEvent sensorEvent) {
    if(sensorEvent!=null){
        float x_acceleration = sensorEvent.values[0];
        float y_acceleration = sensorEvent.values[1];
        float z_acceleration = sensorEvent.values[2];

        double Magnitude = Math.sqrt(x_acceleration*x_acceleration + y_acceleration*y_acceleration + z_acceleration*z_acceleration);
        double MagnitudeDelta = Magnitude - MagnitudePrevious;
        MagnitudePrevious = Magnitude;

        if (MagnitudeDelta > 5){
            stepCount++;
        }
        textSteps.setText("Passos: " + stepCount.toString());
    }
}
```

Figura 3-11 — Serviço do acelerómetro (classe *AcelerometerController.java*).

Como foi observado no capítulo 2.2.1.3 o acelerómetro tem um esquema de 3 eixos em que cada eixo devolve um valor de aceleração na direção correspondente. Para ter acesso a esses valores de aceleração, estes são armazenados nas variáveis do tipo *float* (variáveis que permitem números decimais e também valores negativos) *x\_acceleration*, *y\_acceleration* e *z\_acceleration*, que podem ser encontradas na Figura 3-11.

Equação vi — Equação para calcular magnitude.

$$Magnitude = \sqrt{x\_acceleration^2 + y\_acceleration^2 + z\_acceleration^2}$$

Para calcular o valor da magnitude do movimento utilizou-se a fórmula da Equação vi. Esta fórmula permite depois calcular a variação da aceleração através subtração da aceleração atual com a aceleração anterior.

Equação vii — Cálculo de  $\Delta$ Magnitude.

$$\Delta\text{Magnitude} = \text{Magitude} - \text{MagitudePrevious}$$

Depois de obter o valor  $\Delta$ Magnitude da Equação vii pode ser considerado que foi dado um passo caso esse valor seja superior a 5.

Importante referir que cada vez que o perfil muda (mesmo que seja para manter o perfil) o contador de passos reinicia, já que a arquitetura está feita com esse intuito.

### 3.5.4 Temporizador

O temporizador também tem uma classe própria, a classe *TimerController.java*. Nesta classe foi contruído um temporizador simples com as funções de *start*, *pause* e *reset*.

```
public void startTimer(){
    mEndTime = System.currentTimeMillis() + mTimeLeftInMillis;

    mcountDownTimer = new CountDownTimer(mTimeLeftInMillis, countDownInterval: 1000) {
        @Override
        public void onTick(long millisUntilFinished) {
            mTimeLeftInMillis = millisUntilFinished;
            updateCountdownText();
        }

        @Override
        public void onFinish() {
            mTimerRunning = false;
            mainActivity.getNewProfile();
        }
    }.start();

    mTimerRunning = true;
}

public void pauseTimer(){
    if (mcountDownTimer != null) {
        mcountDownTimer.cancel();
        mTimerRunning = false;
    }
}

public void resetTimer(int timeMinutes){
    startTimeInMillis = timeMinutes * 60 * 1000;
    mTimeLeftInMillis = startTimeInMillis;
    updateCountdownText();
}
```

Figura 3-12 — Serviço do temporizador (classe *TimerController.java*).

As funções para começar, pausar e reiniciar o temporizador estão representadas na Figura 3-12. Na função *startTimer* o temporizador é iniciado com intervalos de 1 segundo, e continua a atualizar o tempo enquanto o temporizador não pausa ou chega ao final. A função *pauseTimer* é onde é possível pausar o temporizador. De notar que é boa prática, para evitar problemas no correto funcionamento da aplicação, pausar o temporizador antes de o reiniciar. Por fim, a função *resetTimer* recebe o tempo em minutos do gestor de perfis (no capítulo 3.5.7) e converte-o para milissegundos. Existe ainda uma função com o nome de *updateCountdownText*, que é responsável por atualizar o tempo na IU com o formato de *minutos:segundos*.

### 3.5.5 Serviço da bateria

O serviço da bateria, atua como monitor do nível de bateria do *smartphone*, já que cada vez que há uma alteração no valor, recebe essa informação e atualiza-a na IU.

```
public int getBatteryLevel(){  
  
    BatteryManager bm = (BatteryManager)context.getSystemService(BATTERY_SERVICE);  
  
    int percentage = bm.getIntProperty(BatteryManager.BATTERY_PROPERTY_CAPACITY);  
  
    return percentage;  
  
}
```

Figura 3-13 — Função para obter o nível da bateria do *smartphone*.

Na Figura 3-13 está a função para obter o valor percentual da bateria. Para isso recorreu-se à classe de Android *Battery Manager*, que é responsável por todas as informações relacionadas com a bateria do dispositivo. A função retorna o valor da bateria.

### 3.5.6 Envio de SMS

Para ter uma forma de conseguir enviar a informação da localização para um contato de um cuidador ou familiar, criou-se uma função na classe principal (*MainActivity.java*) para enviar um SMS ao contato desejado. Essa função pode ser encontrada na Figura 3-14 e recorre à classe do Android *SMS Manager*.

```

public void sendSMS (){
    try {
        SimpleDateFormat formatter = new SimpleDateFormat( pattern: "dd/MM/yyyy HH:mm:ss");
        String gpsInfo = gpsLocal.getText().toString();
        String message = String.format("Perfil: %s\nData: %s\n%s;", currentProfile.toString(), formatter.format(new Date()), gpsInfo);
        Log.d( tag: "testSms", message); //TODO: Para remover no futuro
        SmsManager mySmsManager = SmsManager.getDefault();
        mySmsManager.sendTextMessage( destinationAddress: "+351[REDACTED]", scAddress: null, message, sentIntent: null, deliveryIntent: null);
        showToast( msg: "Mensagem enviada");
    } catch (Exception e) {
        showToast( msg: "Erro ao enviar mensagem");
    }
}
}

```

Figura 3-14 — Função para enviar uma mensagem de texto.

Na mensagem é enviado o perfil ativo, a hora do envio e a localização atual do doente. A mensagem é enviada no serviço da bateria quando há uma alteração do estado, está pré-definido para enviar a partir dos 10%, no entanto esse valor pode ser alterado.

### 3.5.7 Gestor de perfis

Os perfis foram criados no Android Studio de acordo com os dados da Tabela III. A forma como foram implementados pode-se observar na Figura 3-15

```

public static ProfileController getProfile(ProfileEnum profileEnum){
    switch (profileEnum){
        case PROFILE_0:
            return new ProfileController( _hasWifi: false, _hasBluetooth: false, _hasMobileData: false, _hasGps: false, _timerMinutes: 30);
        case PROFILE_1:
            return new ProfileController( _hasWifi: true, _hasBluetooth: false, _hasMobileData: false, _hasGps: false, _timerMinutes: 15);
        case PROFILE_2:
            return new ProfileController( _hasWifi: false, _hasBluetooth: true, _hasMobileData: true, _hasGps: false, _timerMinutes: 5);
        case PROFILE_3:
            return new ProfileController( _hasWifi: true, _hasBluetooth: false, _hasMobileData: true, _hasGps: false, _timerMinutes: 5);
        case PROFILE_4:
            return new ProfileController( _hasWifi: true, _hasBluetooth: false, _hasMobileData: false, _hasGps: true, _timerMinutes: 2);
        default:
            return null;
    }
}
}

```

Figura 3-15 — Configuração dos perfis.

Ao serem definidos os perfis, são configurados também os estados das antenas, quais estão ativas ou desativadas. Para além dessa informação é também logo definido o tempo em minutos associado a cada perfil. Este tempo é recebido pelo serviço do temporizador como foi visto no capítulo 3.5.4.

```

public void setSettings(WifiController wifiController, BluetoothController bluetoothController, GpsController gpsController,
    GpsController gpsMDCController, TimerController timerController, AccelerometerController accelerometerController){
    if (hasWifi != wifiController.isEnabled())
        wifiController.changeState();

    if (hasBluetooth != bluetoothController.isEnabled())
        bluetoothController.changeState();

    gpsMDCController.turnOFF();
    gpsController.turnOFF();

    if (hasGps) {
        gpsController.turnON();
    } else if (hasMobileData) {
        gpsMDCController.turnON();
    }

    timerController.pauseTimer();
    timerController.resetTimer(timerMinutes);
    timerController.startTimer();
    accelerometerController.resetStepsCount();
}

```

Figura 3-16 — Gestor de perfis (*ProfileController.java*).

A acrescentar à classe *Profile.java* da Figura 3-15 existe ainda outra classe importante para os perfis, *ProfileController.java*, que ativa todas as antenas e definições consoante o perfil escolhido pela árvore.

A função que coloca as definições corretas do perfil escolhido é a função *setSettings* da Figura 3-16. Muda se necessário os estados do *wi-fi*, do *bluetooth*, dos dados móveis e do GPS. Reinicia o temporizador com o tempo associado ao perfil, e ainda reinicia o contador de passos.

### 3.5.8 Árvore de decisão

A árvore de decisão no Android Studio foi construída com base na árvore gerada pelo CART. O Android Studio tem classes e funções para uma implementação correta das árvores de decisão.

```

public interface INodeCondition {

    public boolean get(int battery, int steps, ProfileEnum profileEnum);

}

```

Figura 3-17 — Interface para as condições de um nó.

Para criar a árvore de decisão, começou-se por criar uma interface com as informações que podem ser necessárias para um nó ser criado. Essa interface está na Figura 3-17. Posteriormente, na classe *DecisionTree.java* foi construída a árvore de decisão.

```

root = new Node(new INodeCondition() {
    @Override
    public boolean get(int battery, int steps, ProfileEnum profileEnum) {
        return steps <= 156;
    }
});

Node node_2 = new Node(new INodeCondition() {
    @Override
    public boolean get(int battery, int steps, ProfileEnum profileEnum) {
        return battery <= 10;
    }
});

```

Figura 3-18 — Definição da raiz e de um nó da árvore.

Na Figura 3-18 está presente uma parte da função que gera a árvore de decisão. Consegue-se observar a criação da raiz (o primeiro nó da árvore) e ainda um outro nó. Cada função associada ao nó retorna a condição do mesmo, que como se pode observar no exemplo da raiz a condição que retorna é os passos serem menores ou iguais a 156.

```

node_8.addLeftNode(leaf_profile_1)
      .addRightNode(node_9);

```

Figura 3-19 — Ramificação de cada nó.

Por fim, na Figura 3-19 está representada a ramificação de um nó. O nó ramifica-se em dois nós, que podem ser outros nós ou folhas (nós terminais), como o nó esquerdo da figura.

Tanto os nós como as suas ramificações, foram criadas caso a caso com as condições que resultaram do CART.

Os códigos da atividade principal da aplicação e o seu manifesto (ficheiro em que são dadas as autorizações da aplicação) podem ser consultados no apêndice A.4.

## RESULTADOS E VALIDAÇÃO

Neste capítulo serão apresentados e analisados os resultados do trabalho desenvolvido. Em primeiro lugar será apresentado o estudo desenvolvido no CART e os resultados obtidos, bem como a árvore de decisão que foi gerada. Depois será analisado o funcionamento da aplicação e por fim serão apreciados os resultados dos testes da implementação proposta.

### 4.1 Análise da implementação no CART

Como foi possível verificar nos capítulos 3.3 e 3.4 a implementação na aplicação CART foi realizada com os dados da Tabela IV. Foram feitos vários testes com os dados e com diferentes matrizes de custo para tentar chegar a percentagens de acerto de cerca de 70%. Tentaram-se evitar percentagens muito elevadas para prevenir o *overfitting*, que ao acontecer pode causar problemas quando a árvore é testada em novos casos[54].

#### 4.1.1 Matriz de Custos

A matriz de custos que melhor endereça o caso de estudo e por isso a que foi escolhida para a resolução do problema, encontra-se na Tabela V.

Tabela V — Matriz de custos utilizada no CART.

Perfil	0	1	2	3	4
0		1,5	1,5	1,5	2
1	1,5		1,5	1,5	1,5
2	2	1,5		1,5	1,5
3	2	3	3		3
4	10	6	6	7,5	

Na Matriz de custos a coluna a laranja representa o perfil que deve ser selecionado pela árvore, enquanto a linha a cinzento tem os perfis que podem ser selecionados. Os valores do resto da tabela representam o custo que está associado a uma tomada de decisão errada, os quais têm o nome de pesos e são valores relativos. A diagonal da tabela a cinzento são os casos de acerto que obviamente não têm um custo associado.

Para este problema, os maiores custos estão associados aos casos em que deveria haver um melhor acompanhamento do utilizador do *smartphone* e em que tal não acontece. É exemplo disso a linha do perfil 4 (o que tem melhor acompanhamento, mas também maior consumo energético), em que o custo de falhar o perfil é elevado, já que é importante acompanhar a localização do paciente, mesmo que se consuma mais bateria.

No entanto, no sentido inverso o mesmo acontece, ou seja, como se pretende com esta dissertação definir os melhores métodos para poupar a bateria, o custo de ligar o GPS no perfil 4 quando não seria necessário também é maior.

Existem ainda uns pesos intermédios para diferenciar os casos em que os dados móveis são utilizados para a localização (perfis 2 e 3) ou não (perfis 0 e 1).

#### 4.1.2 Resultados dos conjuntos de treino e de teste

Como foi referido no capítulo 3.4.4 foi utilizada a técnica de validação cruzada para criar a árvore de decisão.

Na validação cruzada existem dois conjuntos, o de teste e o de treino, e o CART permite observar os resultados para estes dois conjuntos. Na Figura 4-1 estão os resultados do conjunto de treino representados numa tabela de previsão de sucesso.

Learning Sample Prediction Success Table

Actual Class	Total Cases	Percent Correct	Predicted Class				
			Class 0 N=15	Class 1 N=47	Class 2 N=53	Class 3 N=53	Class 4 N=132
0	7	100,000	7	0	0	0	0
1	50	70,000	8	35	2	5	0
2	43	79,070	0	4	34	5	0
3	60	56,667	0	8	10	34	8
4	140	88,571	0	0	7	9	124

Figura 4-1 — Tabela de previsão de sucesso do conjunto de treino.

Como se pode ver pelos dados da tabela os pesos atribuídos pela matriz de custo fizeram com que a percentagem de acerto fosse elevada nos casos mais críticos como pretendido. Ainda de notar que mesmo com pesos muito pequenos o perfil 0 é sempre escolhido corretamente.

O perfil 1 tem alguns casos em que não acerta no perfil que deveria ser escolhido, mas é uma situação normal e não muito problemática. É normal já que os pesos atribuídos não foram grandes e as escolhas que foram tomadas pela árvore, ou promovem um melhor acompanhamento do paciente ou então quando opta pelo perfil 0 provavelmente será em casos em que a necessidade de acompanhamento não se revela muito importante.

Test Sample Prediction Success Table

Actual Class	Total Cases	Percent Correct	Predicted Class				
			Class 0 N=14	Class 1 N=33	Class 2 N=47	Class 3 N=57	Class 4 N=149
0	7	100,000	7	0	0	0	0
1	50	50,000	7	25	3	10	5
2	43	62,791	0	4	27	10	2
3	60	45,000	0	4	13	27	16
4	140	90,000	0	0	4	10	126

Figura 4-2 — Tabela de previsão de sucesso do conjunto de teste.

Na Figura 4-2 estão os resultados do conjunto de teste. Verifica-se uma percentagem de acerto inferior, o que é normal acontecer, já que a árvore tem mais facilidade em acertar com o conjunto que está a treinar.

Contudo, volta a acertar em todos os casos do perfil 0 e mantém uma boa percentagem de acerto no perfil 4. Há mais erros em especial nos perfis 1 e 3, mas mais uma vez não se revela muito problemático, uma vez que na maioria dos casos é mantido ou até melhorado o acompanhamento da localização, mesmo que se possa comprometer um pouco a bateria.

A percentagem de acerto da árvore de decisão no conjunto de treino é de 78%, o que é um valor muito bom uma vez que se fosse um pouco superior já poderia representar um problema de *overfitting*. Já a percentagem de acerto do conjunto de teste é de 70,67%. Como era de esperar é um valor inferior ao conjunto de treino, ainda assim superior aos 70% propostos.

### 4.1.3 Árvore de decisão

Antes de se observar a árvore de decisão, na figura abaixo pode-se observar a importância de cada uma das variáveis na construção da árvore resultante de todo o processo de estudo e desenvolvimento.

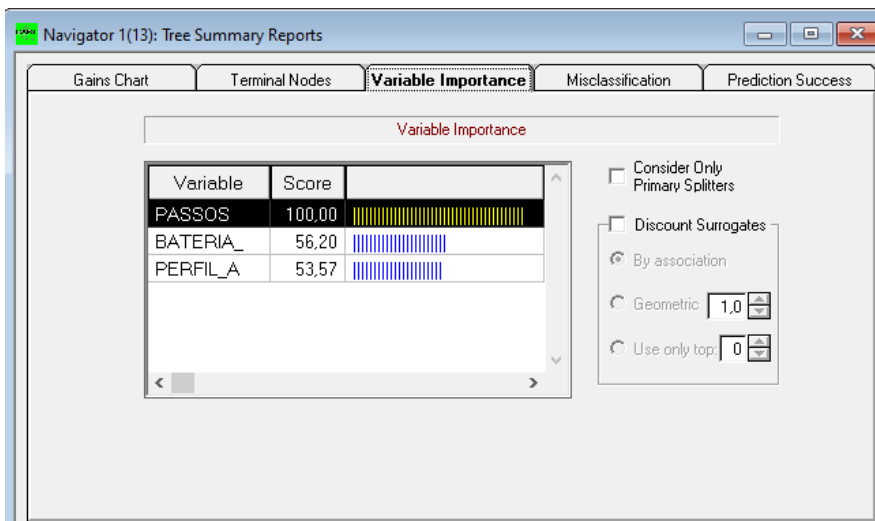


Figura 4-3 — Importância das variáveis na árvore de decisão.

Como se verifica é atribuído um *score* às variáveis que determina a sua importância. A variável PASSOS é a mais importante (com um score de 100) e como será possível observar já de seguida na árvore, é, por esse motivo, o critério inicial no nó raiz.

As variáveis BATERIA\_ e PERFIL\_A partilham praticamente a mesma preponderância com um score muito próximo de 56,2 e de 53,57 respetivamente.

Alguns dados mais pormenorizados da árvore e dos erros de classificação podem ser encontrados no apêndice desta dissertação.

Na abaixo Figura 4-4 pode-se observar a árvore de decisão escolhida pelo CART para solucionar o problema.

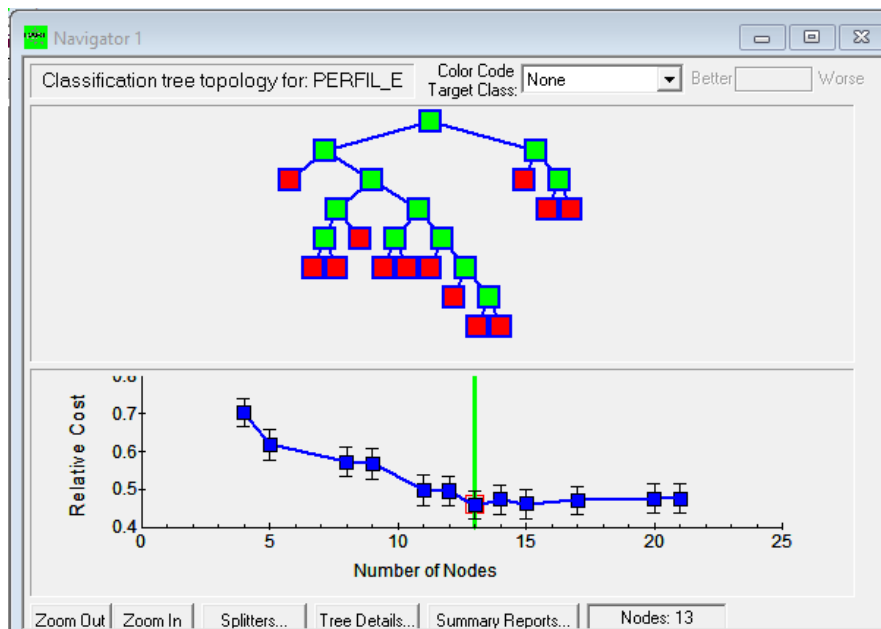


Figura 4-4 — Escolha da árvore de decisão.

Como se verifica no gráfico inferior da imagem que relaciona o número de nós ao custo relativo, a árvore escolhida tem 13 nós terminais e é a que tem um menor custo relativo.

Observa-se que por exemplo a árvore com 15 folhas aproxima-se no acerto, mas mesmo assim é um valor de acerto inferior. O gráfico comprova ainda que as árvores com menos nós, muitas vezes são pouco eficazes, mas que a partir de um certo número essa eficácia de acerto não aumenta, e pelo contrário até diminui.

Na Figura 4-5 é possível ver com maior pormenor toda a árvore de decisão, com todos os seus nós e folhas.

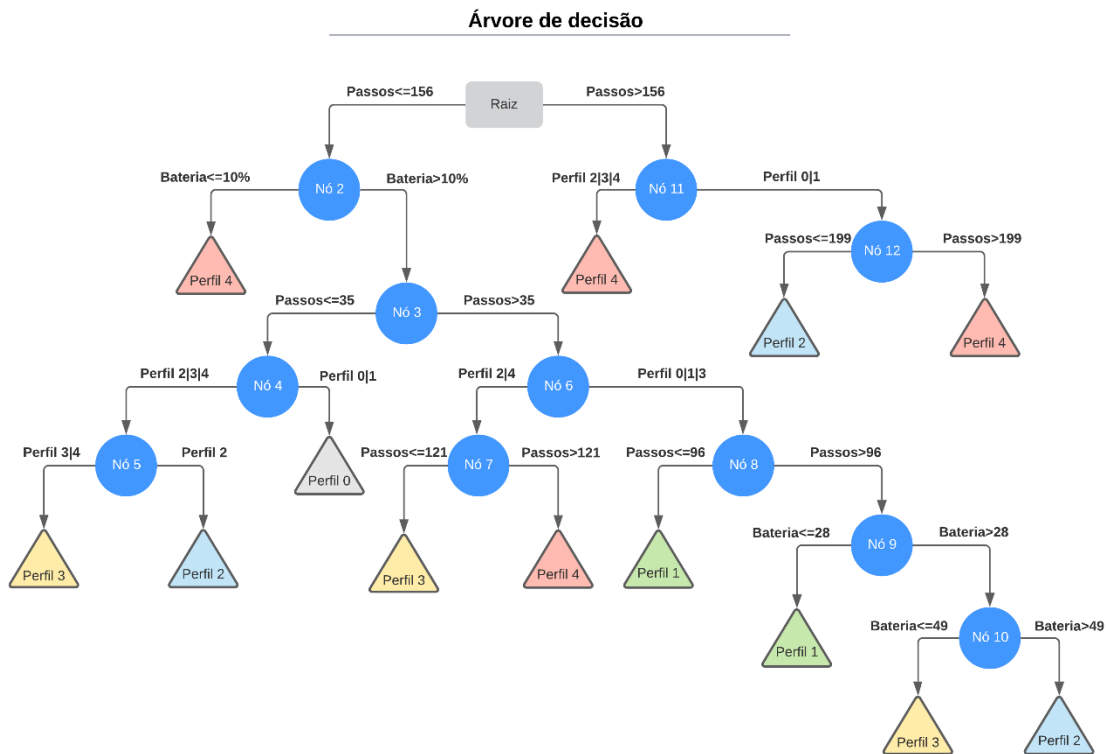


Figura 4-5 — Árvore de decisão.

Na árvore de decisão acima os vários nós estão representados com os círculos azuis, e as folhas representadas com os triângulos coloridos. As folhas retratam o perfil selecionado pela árvore.

Como se verifica há perfis que são escolhidos em mais situações, caso do perfil 4 que é escolhido em quatro situações distintas, normalmente por serem dados muitos passos, o que significa que a pessoa se deslocou bastante no tempo em que o perfil anterior esteve ativo, ou por a bateria estar a chegar ao fim e aí a aplicação tentará manter uma localização mais fidedigna e enviar essa informação para o cuidador ou contato de emergência.

O perfil 0 apenas é selecionado numa situação, quando os passos são inferiores a 35, a bateria superior a 10% e o perfil anterior era 0 ou 1. Portanto, o perfil 0 está mais associado a situações de repouso por exemplo.

Os perfis 1,2 e 3 são ativados em situações mais variadas, o que faz sentido que aconteça já que também partilham algumas das antenas utilizadas e têm consumos energéticos mais parecidos.

## 4.2 Aplicação Android

No capítulo 3.5 abordou-se a implementação realizada no Android Studio para desenvolver a aplicação pretendida para o controlo energético do *smartphone*. Neste capítulo ir-se-ão mostrar os resultados práticos dessa implementação.

Na figura acima observam-se os diferentes perfis ativos na aplicação em funcionamento.

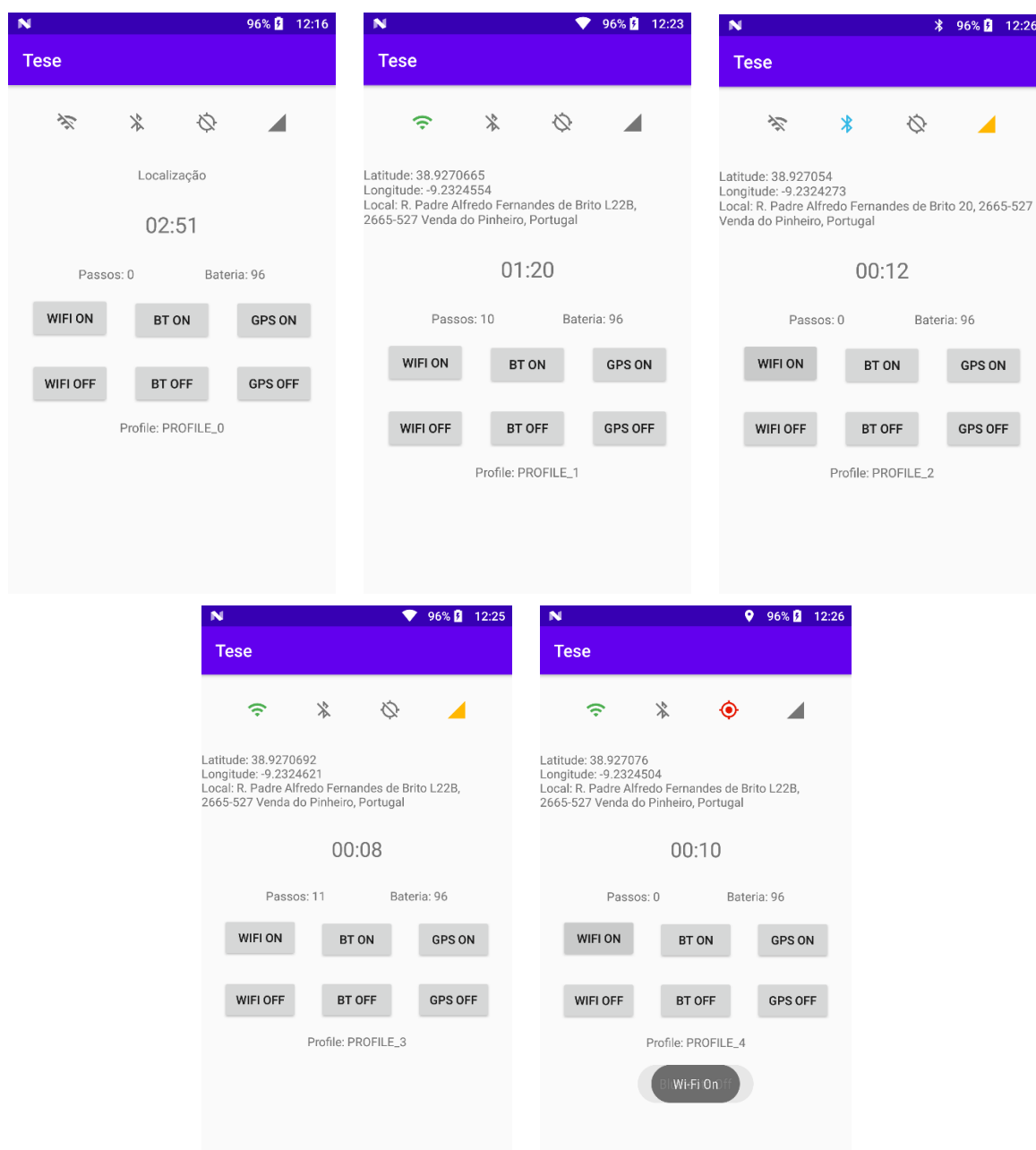


Figura 4-6 — Perfis na IU.

Como era previsto a aplicação mostra as antenas que estão a ser utilizadas em cada perfil através do símbolo colorido e o perfil atual é mostrado na parte inferior da aplicação após os botões.

Para testar a eficácia da aplicação realizaram-se mais dois conjuntos de testes. Tal como os testes realizados no capítulo 3.2 estes também foram repetidos diversas vezes e os valores apresentados são uma média dos valores obtidos.

Os testes foram todos realizados nas mesmas condições dos anteriores:

- Ecrã sempre ligado;
- Luminosidade do ecrã no máximo;
- Todas as aplicações do *smartphone* fechadas;
- Nível inicial da bateria com carga no máximo (100%).

Realizaram-se dois conjuntos de testes, uma com a duração de 1 hora e outra com a duração de 6h para ser mais fácil distinguir os resultados. Cada bateria de testes possuiu três testes distintos, um com as antenas todas desligadas, outro com as antenas todas ligadas e ainda um terceiro teste com a aplicação em funcionamento onde se tentou variar os perfis ao longo do teste para simular uma utilização normal.

Tabela VI — Testes finais de 1h.

<b>Teste</b>	<b>Média de % de bateria após 1h</b>
Tudo desligado	93
Tudo ligado	86
Aplicação	90

Tabela VII — Testes finais de 6h.

<b>Teste</b>	<b>Média de % de bateria após 6h</b>
Tudo desligado	41
Tudo ligado	30
Aplicação	34

Nas tabelas acima verifica-se que a aplicação faz exatamente o pretendido, confirmando assim a hipótese desta dissertação. Tanto nos testes de 1 hora como nos testes de 6 horas o consumo médio estabeleceu-se num valor intermédio entre os outros dois testes, que representam as situações mais comuns, que se traduzem em, ou não haver qualquer tipo de serviço

de localização ou então ter as antenas todas ativas para conseguir obter uma localização do utilizador do *smartphone*.

Verificaram-se poupanças médias de 4% tanto no teste de 1 hora como no teste de 6 horas. Os valores percentuais da bateria consumida revelam algum nível de poupança, que não sendo muito grande, pode ser importante em algumas situações limite.

É natural que consoante os perfis que acabam por ser escolhidos pela árvore de decisão o consumo possa variar mais do que as outras situações testadas, contudo não deverá ultrapassar os consumos de ter as antenas todas ligadas, sendo assim um resultado bastante positivo.

Concluindo, os resultados obtidos são bastante positivos, uma vez que é notável uma melhoria na gestão energética do *smartphone* com a implementação do sistema de decisão aqui apresentado, permitindo assim confirmar a hipótese proposta por esta dissertação.

Os resultados discriminados dos testes podem ser consultados nas tabelas do apêndice A.5.



## CONCLUSÕES E TRABALHO FUTURO

Nesta dissertação procurou-se responder a um problema que afeta milhões de pessoas com Alzheimer e outras demências em todo o mundo, o *wandering* e os riscos associados, que incluem esquecimento sobre a localização ou o destino, desorientação e possível risco de vida. Para tentar ajudar na resolução do problema procurou-se perceber se o *smartphone* poderia ser uma boa solução, já que apesar de ser um dos dispositivos eletrônicos com capacidades de localização mais utilizados em todo o mundo, são conhecidas as suas limitações em termos de autonomia da bateria.

Com o objetivo de contornar este problema, a hipótese proposta passou por estudar o impacto energético de cada um dos sensores e componentes relevantes para a comunicação e localização, e de que forma podem as técnicas de *Machine Learning* promover a melhor gestão energética do *smartphone* com vista à maximização do tempo em que o dispositivo está ativo, e que por esse motivo a pessoa se encontra contactável e localizável. Para tal, foram utilizadas árvores de decisão que se revelaram muito úteis para este tipo de análise de dados, uma vez que com poucos parâmetros de entrada e criteriosamente selecionados, os resultados foram bastante positivos.

Os testes realizados permitiram verificar a mais-valia que é implementar um sistema de inteligência artificial na gestão do consumo energético de um dispositivo como o *smartphone*, adaptável a outros dispositivos móveis, como por exemplo os *smartwatches*. Tendo consciência do quão importante é garantir a pessoa localizável mais um minuto que seja, verificaram-se melhorias ao nível do consumo energético do *smartphone* pela técnica de comutação de perfis de acordo com as circunstâncias. A utilização racional dos perfis revelou-se um fator de regulação dos consumos, que no cômputo geral, se tornaram mais próximos de valores em

que apenas uma antena está ativa, ao contrário da resolução menos criteriosa de manter todas as antenas ativas.

Estes resultados confirmam assim a hipótese proposta para a dissertação como resposta às perguntas de investigação, e lançam assim os alicerces de possíveis investigações futuras.

No futuro, este sistema implementado poderá ser melhorado quer ao nível do tratamento dos dados quer no desenvolvimento da aplicação. Em relação ao tratamento dos dados uma solução poderá passar por serem acrescentadas mais condições de entrada na árvore de decisão na tentativa de obter melhores resultados, como por exemplo condições ambientais ou comportamentos de risco. Na aplicação, também poderão ser realizadas algumas melhorias, quer ao nível da interface de utilizador, se bem que as funções fundamentais podem funcionar de forma invisível em background. Esse é também um dos motivos pelo qual a interação com o utilizador não foi um ponto chave desta dissertação. De qualquer forma, novas funcionalidades que possam ser implementadas podem vir a tornar a aplicação ainda mais eficiente.

## BIBLIOGRAFIA

- [1] W. H. Organization, "A public health priority," 2015.
- [2] "dementia @ www.who.int." [Online]. Available: <https://www.who.int/en/news-room/fact-sheets/detail/dementia>.
- [3] T. Epidemiology and A. Disease, "Epidemiologia da Demência e da Doença de Alzheimer em Portugal : Estimativas da Prevalência e dos Encargos Financeiros com a Medicação," pp. 182–189, 2015.
- [4] "alzheimers-disease @ www.dementia.org.au." [Online]. Available: <https://www.dementia.org.au/about-dementia/types-of-dementia/alzheimers-disease>.
- [5] "vascular-dementia @ www.dementia.org.au." [Online]. Available: <https://www.dementia.org.au/about-dementia/types-of-dementia/vascular-dementia>.
- [6] "frontotemporal-dementia @ www.dementia.org.au." [Online]. Available: <https://www.dementia.org.au/information/about-dementia/types-of-dementia/frontotemporal-dementia>.
- [7] "lewy-body-disease @ www.dementia.org.au." [Online]. Available: <https://www.dementia.org.au/about-dementia/types-of-dementia/lewy-body-disease>.
- [8] "parkinson @ www.saudecuf.pt." [Online]. Available: <https://www.saudecuf.pt/mais-saude/doencas-a-z/parkinson>.
- [9] "Meet Marty Cooper - the inventor of the mobile phone @ news.bbc.co.uk." [Online]. Available: [http://news.bbc.co.uk/2/hi/programmes/click\\_online/8639590.stm](http://news.bbc.co.uk/2/hi/programmes/click_online/8639590.stm).
- [10] "iOS Security Contents," no. November, pp. 1–95, 2018.
- [11] A. N. Mohmedhussen, "Comparison of Android and iPhone Operating System," vol. 167, no. 2, pp. 6–11, 2017.
- [12] "Worldwide Market Share in OS @ www.moontechnolabs.com." [Online]. Available: <https://www.moontechnolabs.com/apple-vs-android-comparative-study-2017/>.
- [13] "Top companies in the world by market value 2018 @ www.statista.com," 2018. [Online]. Available: <https://www.statista.com/statistics/263264/top-companies-in-the-world-by-market-value/>.

- [14] "Amazfit @ en.amazfit.com." [Online]. Available: <http://en.amazfit.com/>.
- [15] M. Mardonova and Y. Choi, "Review of Wearable Device Technology" 2018.
- [16] "Xiaomi @ www.mi.com." [Online]. Available: <https://www.mi.com/global/>.
- [17] "myMDlife @ mymdlifewixsite.com." [Online]. Available: <http://mymdlifewixsite.com/mymdlifeflive>.
- [18] "Emergency Medical Alert Bracelet @ www.alzstore.com." [Online]. Available: <https://www.alzstore.com/alzheimers-alert-id-bracelets-p/0910.htm>.
- [19] "GPS Smart Sole @ gpssmartsole.com." [Online]. Available: <http://gpssmartsole.com/gpssmartsole/>.
- [20] "AngelSense @ www.angelsense.com." [Online]. Available: <https://www.angelsense.com/>.
- [21] "Estou aqui adultos PSP @ estouaquiadultos.mai.gov.pt." [Online]. Available: <https://estouaquiadultos.mai.gov.pt/Pages/Home.htm>.
- [22] "wifi @ www.infowester.com." .
- [23] "An-example-of-IoT-scenario-in-home-WiFi-network-consisting-of-sensors-and-devices @ www.researchgate.net." .
- [24] "bluetooth @ www.infowester.com." .
- [25] "how-bluetooth-works @ learn.sparkfun.com." .
- [26] "Acelerómetro @ www.tme.eu." .
- [27] "10-Best-Accelerometer-Apps @ turbofuture.com." .
- [28] "navgps-1 @ www.eso.org." .
- [29] "how-does-gps-work @ insights.globalspec.com." .
- [30] "Orientation Sensor @ www.w3.org." .
- [31] "what-a-proximity-sensor @ www.fierceelectronics.com." .
- [32] "nfc @ www.infowester.com." .
- [33] I. Khan, S. Khusro, S. Ali, and J. Ahmad, "Sensors are Power Hungry : An Investigation of Smartphone Sensors Impact on Battery Power from Lifelogging Perspective," *Bahria Univ. J. ICT*, vol. 9, no. 2, pp. 8–19, 2016.
- [34] "machine-learning @ www.mathworks.com." [Online]. Available: <https://www.mathworks.com/discovery/machine-learning.html>.
- [35] J. M. Fonseca, "Genetic and Fuzzy." .
- [36] "Modelo-nao-linear-de-um-neuronio @ www.researchgate.net." [Online]. Available: <https://www.researchgate.net/figure/Figura-1-Modelo-nao-linear-de-um-neuronio->

Haykin-2001\_fig1\_282647489.

- [37] J. M. Fonseca, "Redes Neurais." pp. 1–27.
- [38] "understanding-decision-trees-with-python @ datascience.foundation."
- [39] "O que é árvore de decisão (decision tree)? @ cienciaenegocios.com."
- [40] J. M. Fonseca, "PS - Modulo 4 - Arvores de decisão.pdf."
- [41] "3-techniques-to-avoid-overfitting-of-decision-trees @ towardsdatascience.com."
- [42] M. S. Lauretto, "Árvores de Decisão 1 Definições," no. 1, pp. 1–27, 2010.
- [43] "bq\_aquaris\_m5@ www.gsmarena.com."
- [44] "Android is for everyone @ www.android.com."
- [45] "discover-intellij-idea @ www.jetbrains.com."
- [46] Developers, "Intro @ Developer.Android.Com." 2020.
- [47] "java @ www.lenovo.com."
- [48] "platforms @ developer.android.com."
- [49] "index @ apilevels.com."
- [50] I. Cart, "Introducing CART ®."
- [51] "ServiceLogin @ issuetracker.google.com."
- [52] E. J. Aguiar *et al.*, "Normative peak 30-min cadence (Steps per minute) values for older adults: Nhanes 2005–2006," *J. Aging Phys. Act.*, vol. 27, no. 5, pp. 625–632, 2019.
- [53] "LocationRequest @ developers.google.com."
- [54] "Overfitting – what and why? @ blog.knoldus.com."
- [55] W. H. O. Alzheimer's Disease International, *Dementia: A Public Health Priority*. World Health Organization, 2012.
- [56] J. T. O'Brien and A. Thomas, "Vascular dementia," *Lancet*, vol. 386, no. 10004, pp. 1698–1706, Oct. 2015.
- [57] D. Weisman and I. McKeith, "Dementia with Lewy Bodies," *Semin. Neurol.*, vol. 27, no. 1, pp. 042–047, Feb. 2007.
- [58] F. Luis-Ferreira, D. Rodrigues, D. Pereira, J. Sarraipa, and R. Jardim-Gonçalves, "Assessment of Risk for Alzheimer Patients by Monitoring Heartbeat and Identifying Falls," in *Proceedings of 10th International Conference on e-Health 2018*, 2018.
- [59] L. Corral, A. B. Georgiev, A. Sillitti, and G. Succi, "A method for characterizing energy consumption in Android smartphones," in *2013 2nd International Workshop on Green and Sustainable Software (GREENS)*, 2013, pp. 38–45.
- [60] G. P. Perrucci, F. H. P. Fitzek, and J. Widmer, "Survey on Energy Consumption Entities on the Smartphone Platform," in *2011 IEEE 73rd Vehicular Technology Conference (VTC Spring)*, 2011, pp. 1–6.

- [61] F. Luis-Ferreira, J. Calado, A. Artifice, J. Sarraipa, and R. Jardim-Goncalves, "Personalised Technological Architectures to Assist Dementia Patients based on Energy Efficiency," in *ICIST 2019 - 9th International Conference on Information Society and Technology*, 2019.

## APÊNDICE

Este capítulo de apêndices pretende apresentar algumas informações ou resultados mais extensos sobre o trabalho que foi desenvolvido, que também são relevantes e que podem ajudar em futuras investigações.

### A.1 Short paper

*Short paper* submetido para a conferência eHealth 2019.

## **ENERGY MANAGEMENT FOR DEVICES SUPPORTING DEMENTIA PATIENTS**

Diogo Silva

CTS, UNINOVA, Dep.º de Eng.ª Electrotécnica Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa  
2825-529 Caparica, Portugal

Fernando Luís-Ferreira

CTS, UNINOVA, Dep.º de Eng.ª Electrotécnica Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa  
2825-529 Caparica, Portugal

João Sarraipa

CTS, UNINOVA, Dep.º de Eng.ª Electrotécnica Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa  
2825-529 Caparica, Portugal

Ricardo Goncalves

CTS, UNINOVA, Dep.º de Eng.ª Electrotécnica Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa  
2825-529 Caparica, Portugal

## **ABSTRACT**

The advances in medicine and the improvements in healthcare have provided treatment and cure for many pathologies and awarded a larger expectancy of life. It represents a considerable improvement in ensuring treatment for diseases otherwise mortal and thus becoming, in some cases, chronic disorders that a person can live with for many years. This evolution has a tremendous impact in the overall healthcare environment and thus promoting longer life expectancy especially in developed countries. It is necessary, however, to consider that by increasing the width of the top level of the aging pyramid, it is therefore expected an increased incidence of the ageing related disorders. In that matter, the raising prevalence of Dementia cases is noticeable and is among the most prominent worries for the health community, as expressed in the World Health Organization reports. It is therefore of major relevance to provide assistance to citizens with dementia, especially in what concerns with losing cognitive functions such as the notion of self, awareness of current location and identifying the need to return home or to known places. The present research work is based on the usage of smartphones to track people and has a specific target, in this matter, in extending battery life and for that matter the time a person can be located and assisted. The strategy, applicable to smartphones or other devices, relies on ensuring that energy is wisely managed so that batteries have an extended duration, increasing the chances that devices are still connected and accessible when most needed, such as while a person is experiencing a wandering episode. The aim of the present work is to focus on energy consumption to increase the chances to locate a person that is lost, with limited cognitive functions and most likely at risk for health or when even survival could be endangered.

## **KEYWORDS**

Dementia, Wandering, Smartphone App, Energy optimization

## **1. INTRODUCTION**

Today the expectancy of life has reached levels that were unthinkable centuries ago. The major part of the population in developed countries have access to healthcare that allows prevention and treatment for most of the pathologies once considered deadly, sooner or later. It is thus expectable that with the extended lifetime the prevalence of aging associated pathologies becomes more problematic for people and the society. Dementia, in its various forms is a major problem recognized by the World Health Organization as a public health priority. Dementia is becoming a problem for healthcare systems that will become worse with time as people's lives will become longer. The benefit of better healthcare is putting a burden on the quality of life for those living longer and a challenge to support healthy longer life. Dementia is a syndrome that affects memory, thinking, behavior and ability to perform everyday activities. The number of people living with dementia worldwide is currently estimated at 35.6 million. This number will double by 2030 and more than triple by 2050 [55]. There are several types of dementia, depending on the causes and symptoms but the consequences are rather similar and the problems they rise are almost the same. Alzheimer is the most prevalent with about 60 to 70% of the cases. Then it comes Vascular Dementia as the second most frequent cause for dementia, about 15% of the cases but, unlike Alzheimer, without any licensed treatment [56], The other types are front temporal dementia, Dementia of Lewy corps which is a neurodegenerative disorder resulting in slowly progressive and unrelenting dementia until death

[57] and the well-known Parkinson's disease. Those types of dementia share common characteristics that emerge from cognitive impairments. It is likely that patients in different stages present different symptoms and that, as time passes, the patterns and activity level of the disease are changing, in most of the cases getting worse over time. The common behaviors that worry carers and families are the tendency to forget basic information, such as name and address and, for that reason, the tendency to wander without known destination as the person forgets what was doing or where is home. Those are risky situations that can be addressed by devices that a person can carry and help in getting their location.

## **2. ANALYSIS OF ENERGY CONSUMPTION**

Currently, energy consumption is a hotly contested topic all over the world. The objective is to find effective methods to extend the life of the devices in each load. Patients with dementia are no exception and, due to the problems in remembering paths and places, it is important that they can be located. For that it is necessary that devices have tracking information and, more important than that, devices must be working. For these patients that need daily follow-up due to lack of orientation it is important that unless they are accompanied, it is possible to reach them by any available device. In order to keep their independence and continue to become integrated in society, it is better to use electronic devices that most people use in daily routine and, at the same time, keep them trackable and connected to the caregivers. The easiest devices to find by the end consumer that meet these requirements are smartphones. For this reason, it's decisive to optimize resources for a better battery management. That is the aim of the current work.

In previous works [58] research has been focused in the devices and applications that can provide location and help the patients with the adequate features to detect risks and provide timely communication of detected risks and location, to carers and family. The problems then identified were related with the extension of the absence of notice and the time a person can be lost by wandering in unknown places. That significantly arises the question that, when the person most need it is possible that the devices are shut down by lack of energy.

A reasonable approach to address this problem is to manage, optimized through machine learning, the usage of sensors and antennas in smartphones. In our research study, as observed in the following graphs, energy consumption of the sensors and antennas may vary in different situations.

The consumption patterns will take in account not only the devices but also the applications being used and the result, in terms of consumption of both software applications and hardware devices being used for the same objective. It was noticed a dramatic increase in energy consumption once the smartphone actually starts transmitting data over the Wi-Fi antenna. The value climbs almost three times if we run an application such as Google Maps, which uses GPS and Wi-Fi simultaneously, while rendering images dynamically [59]. In Figure 1 it is possible to observe our early basis of work by observing the different consumption patterns for devices in a smartphone (Nokia N95) that gives clues about where energy is spent and what benefits can be obtained with a proper management of those devices [60].

Technology		Action	Power [mW]	Energy [J]
Wireless data	Bluetooth	BT off	12	
		BT on	15	
		BT connected and idle	67	
		BT discovery	223	
		BT receiving	425	
		BT sending	432	
	WiFi IEEE802.11 (infrastructure mode)	In connection	868	8.2
		In disconnection	135	0.4
		Idle	58	
		Idle in power save mode	26	
		Downloading@4.5Mbps	1450	
	WiFi IEEE802.11 (ad hoc)	Sending @ 700 kB/s	1629	
		Receiving	1375	
		Idle	979	
	2G	Downloading@44Kbps	500	
Handover 2G->3G		1389	2.4	
3G	Downloading@1Mbps	1400		
	Handover 3G->2G	591	2.5	

Figure 1 - Energy Consumption in smartphone communication devices

It is observed that the use of GPS causes major losses of battery when compared with other sensors like accelerometers that are known to provide guidance for short periods or time or small to medium displacements by evaluating relative displacements. So, it is deductible that its usage should be kept to strictly necessary. It is still interesting to note that GPS, although more precise on the outside, consumes more energy in this situation. There are still other antennas that help locating the device such as the Wi-Fi antenna. Figure 2 presents the different energy consumption patterns for a user moving and standing either inside or outside a building.

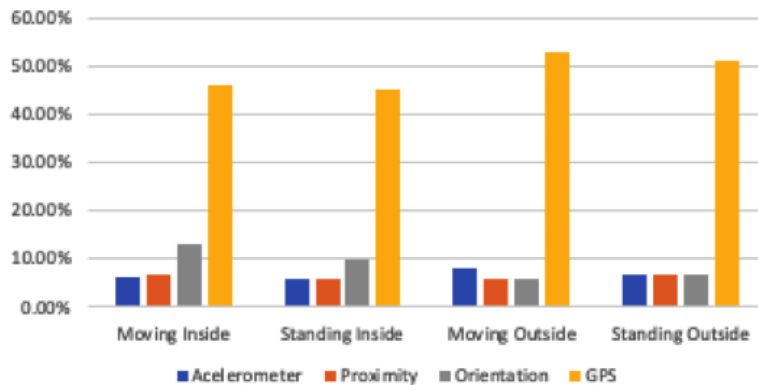


Figure 2 - Energy Consumption with data transmission

In the previous chart, it becomes evident that the Wi-Fi, GSM and Bluetooth antennas consume less energy than GPS. For this reason, they can be used, when near to other recognizable devices, to locate the patient's device it can thus be applied to smartwatches that can use the smartphone location devices. Another important strategy to be applied in these cases is the intermittent activation of the GPS that can minimize the impact of that device in energy consumption patterns and still ensure that location is stored and provided from time to time. Table 1 presents the usage of different devices according to the present situation, making possible to extend the battery life by switching on and off the sensors that are not needed in the present moment. The different devices in a smartphone have different consumption levels meaning different weight in the overall consumption of a device's battery. The consumption patterns also vary according to the activity a person is executing while indoors or outside and such results can be observed next in Table 1 [33].

Table 1 - Sensors relative Energy consumption

Sensor	Interior Movement	Standing Inside	External Movement	Standing Outside
Accelerometer	6,35%	5,95%	8%	6,6%
Proximity	6,7%	5,9%	5,9%	6,9%
Orientation	13,23%	9,7%	5,9%	6,9%
GPS	46%	45,22%	53%	51%

Beyond comparison, there are different profiles of consumption according to behavioural status. This is the case, for the same person, that moves, goes outside or returns inside buildings. Those patterns as observed next in Figure 3.

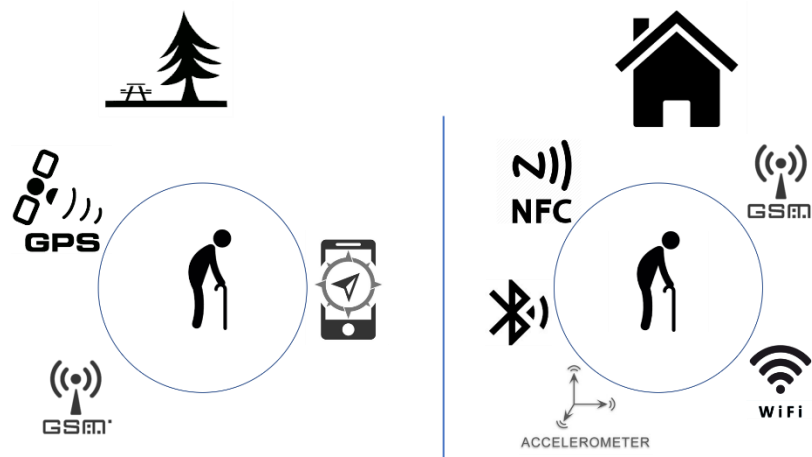


Figure 4 - Devices selected according to location

The energy management information becomes of major importance by allowing an optimization of resources in smartphones for people who are in situations that need constant monitoring. The present work comes as a continuation of the early work on the development of personalized technological architectures to assist dementia patients based on energy consumption profiles which become essential for the successful implementation of the current work where those profiles are in fact brought to implementation [61].

### 3. CONCLUSION

The presented research is being carried in the scope of an ongoing MSc in Electronics and Computer Science and is applied in the research work for the CARELINK AAL project. Work carried so far passed early stages as it was verified the high potential of energy management strategies to extend the autonomy of mobile devices. This smart management of energy is of most importance for patients with Dementia, as sometimes, cognitive problems appear without notice in episodes where the person manifests lack of orientation, lost sense of time and place and in general inability to return home or to a known place. In these cases, people also tend to distance themselves from a non-controllable situation and refuse to admit their situation and thus are not likely to ask for help even while carrying a communications device. In those cases, it becomes important to have an active device to make possible location and contact, to make possible reaching this people while in a situation that can be of risk or even life threatening. It is known that the first 72 hours that a person is lost are critical, with time passing, without any option to locate them time runs against the possibilities of a good outcome. The most common and used solutions for communication and location services are in fact the smartphones but other customized dispositive can also be used. The study hereby presented addresses the need to extend, as much as possible, the autonomy of the devices which, when dry out of energy, become completely useless for tracking and rescue. From the current tests it Is

possible to envisage an extension of battery above 20% making a contributing factor to promote the possibility that lost people can be found and become safe before it is too late.

## ACKNOWLEDGEMENT

The authors acknowledge the European Commission for its support and partial funding and the partners of the research project CARELINK, AAL-CALL-2016- 049 funded by AAL JP, and co-funded by the European Commission and National Funding Authorities of Ireland, Belgium, Portugal and Switzerland.

## REFERENCES

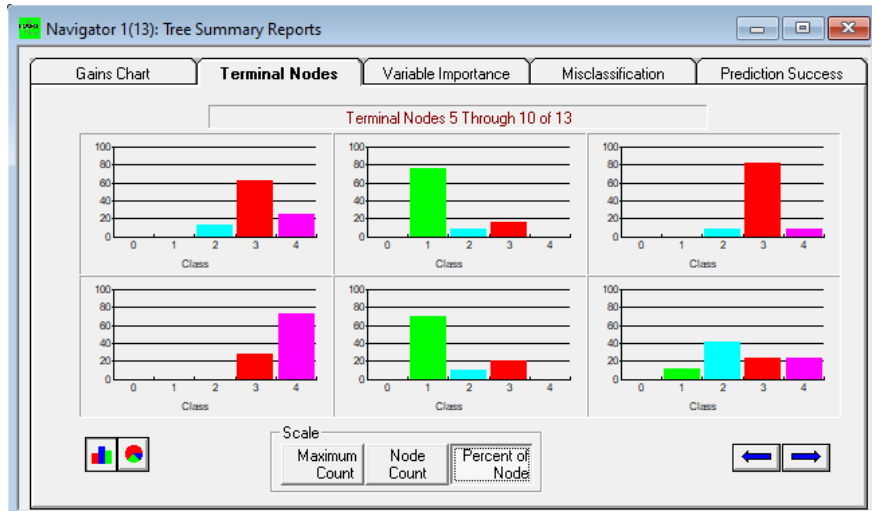
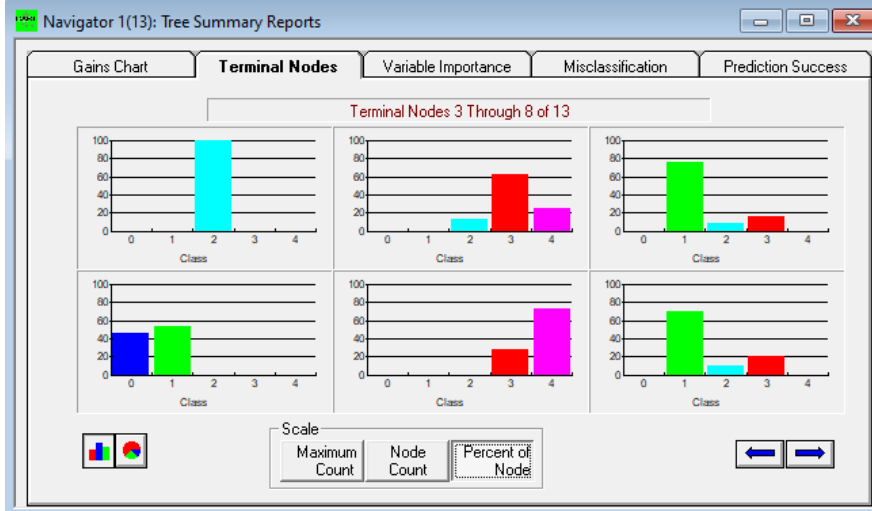
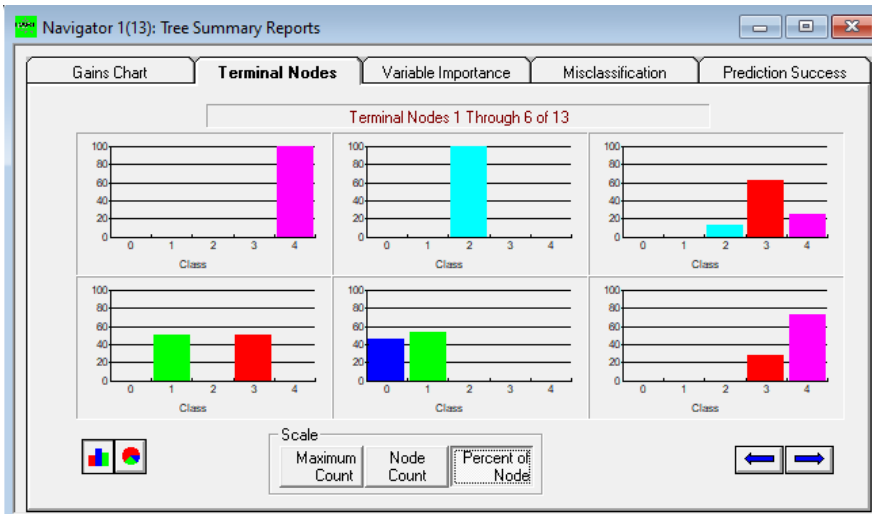
- [1] W. H. Organization, "A public health priority," 2015.
- [2] "dementia @ www.who.int." [Online]. Available: <https://www.who.int/en/news-room/fact-sheets/detail/dementia>.
- [3] T. Epidemiology and A. Disease, "Epidemiologia da Demência e da Doença de Alzheimer em Portugal : Estimativas da Prevalência e dos Encargos Financeiros com a Medicação," pp. 182–189, 2015.
- [4] "alzheimers-disease @ www.dementia.org.au." [Online]. Available: <https://www.dementia.org.au/about-dementia/types-of-dementia/alzheimers-disease>.
- [5] "vascular-dementia @ www.dementia.org.au." [Online]. Available: <https://www.dementia.org.au/about-dementia/types-of-dementia/vascular-dementia>.
- [6] "frontotemporal-dementia @ www.dementia.org.au." [Online]. Available: <https://www.dementia.org.au/information/about-dementia/types-of-dementia/frontotemporal-dementia>.
- [7] "lewy-body-disease @ www.dementia.org.au." [Online]. Available: <https://www.dementia.org.au/about-dementia/types-of-dementia/lewy-body-disease>.
- [8] "parkinson @ www.saudecuf.pt." [Online]. Available: <https://www.saudecuf.pt/mais-saude/doencas-a-z/parkinson>.
- [9] "Meet Marty Cooper - the inventor of the mobile phone @ news.bbc.co.uk." [Online]. Available: [http://news.bbc.co.uk/2/hi/programmes/click\\_online/8639590.stm](http://news.bbc.co.uk/2/hi/programmes/click_online/8639590.stm).
- [10] "iOS Security Contents," no. November, pp. 1–95, 2018.
- [11] A. N. Mohmedhussen, "Comparison of Android and iPhone Operating System," vol. 167, no. 2, pp. 6–11, 2017.
- [12] "Worldwide Market Share in OS @ www.moontechnolabs.com." [Online]. Available: <https://www.moontechnolabs.com/apple-vs-android-comparative-study-2017/>.
- [13] "Top companies in the world by market value 2018 @ www.statista.com," 2018. [Online]. Available: <https://www.statista.com/statistics/263264/top-companies-in-the-world-by-market-value/>.
- [14] "Amazfit @ en.amazfit.com." [Online]. Available: <http://en.amazfit.com/>.
- [15] M. Mardonova and Y. Choi, "Review of Wearable Device Technology and Its," 2018.
- [16] "Xiaomi @ www.mi.com." [Online]. Available: <https://www.mi.com/global/>.
- [17] "myMDlife @ mymdlifewixsite.com." [Online]. Available: <http://mymdlifewixsite.com/mymdlifeflive>.
- [18] "Emergency Medical Alert Bracelet @ www.alzstore.com." [Online]. Available: <https://www.alzstore.com/alzheimers-alert-id-bracelets-p/0910.htm>.
- [19] "GPS Smart Sole @ gpssmartsole.com." [Online]. Available:

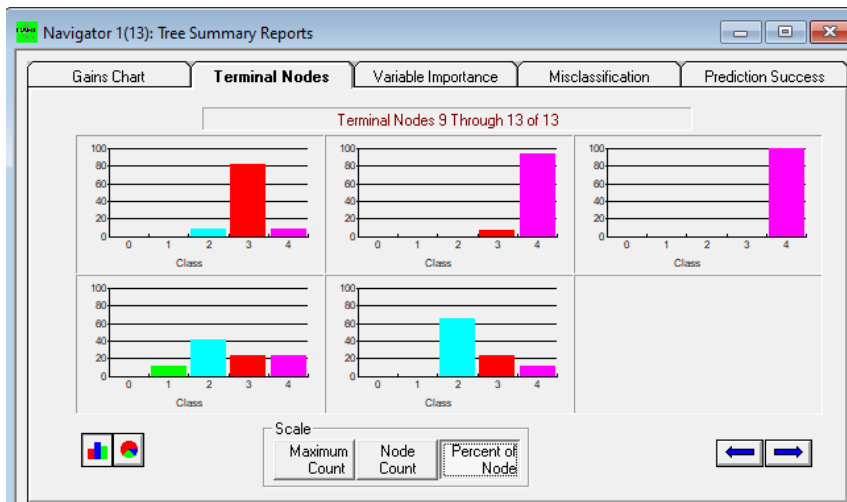
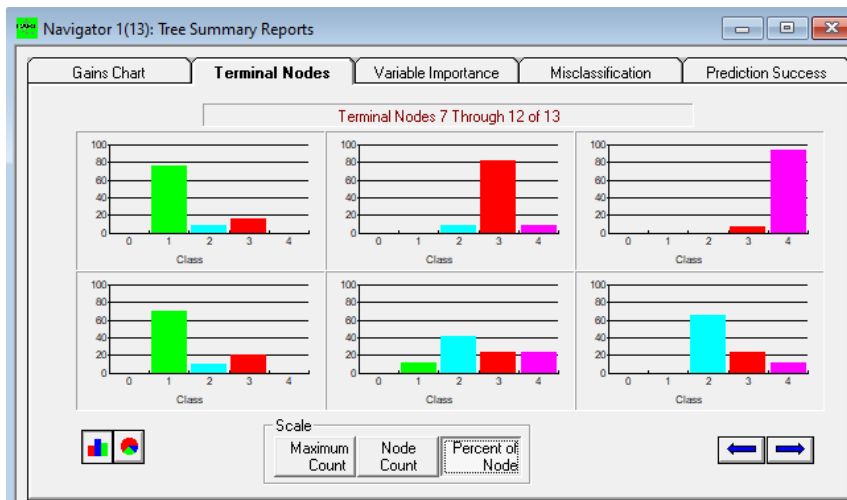
- <http://gpssmartsole.com/gpssmartsole/>.
- [20] "AngelSense @ [www.angelsense.com](http://www.angelsense.com)." [Online]. Available: <https://www.angelsense.com/>.
- [21] "Estou aqui adultos PSP @ [estouaquiadultos.mai.gov.pt](https://estouaquiadultos.mai.gov.pt)." [Online]. Available: <https://estouaquiadultos.mai.gov.pt/Pages/Home.htm>.
- [22] "wifi @ [www.infowester.com](http://www.infowester.com)." .
- [23] "An-example-of-IoT-scenario-in-home-WiFi-network-consisting-of-sensors-and-devices @ [www.researchgate.net](http://www.researchgate.net)." .
- [24] "bluetooth @ [www.infowester.com](http://www.infowester.com)." .
- [25] "how-bluetooth-works @ [learn.sparkfun.com](http://learn.sparkfun.com)." .
- [26] "Acelerómetro @ [www.tme.eu](http://www.tme.eu)." .
- [27] "10-Best-Accelerometer-Apps @ [turbofuture.com](http://turbofuture.com)." .
- [28] "navgps-1 @ [www.eso.org](http://www.eso.org)." .
- [29] "how-does-gps-work @ [insights.globalspec.com](http://insights.globalspec.com)." .
- [30] "Orientation Sensor @ [www.w3.org](http://www.w3.org)." .
- [31] "what-a-proximity-sensor @ [www.fierceelectronics.com](http://www.fierceelectronics.com)." .
- [32] "nfc @ [www.infowester.com](http://www.infowester.com)." .
- [33] I. Khan, S. Khusro, S. Ali, and J. Ahmad, "Sensors are Power Hungry : An Investigation of Smartphone Sensors Impact on Battery Power from Lifelogging Perspective," *Bahria Univ. J. ICT*, vol. 9, no. 2, pp. 8–19, 2016.
- [34] "machine-learning @ [www.mathworks.com](http://www.mathworks.com)." [Online]. Available: <https://www.mathworks.com/discovery/machine-learning.html>.
- [35] J. M. Fonseca, "Genetic and Fuzzy." .
- [36] "Modelo-nao-linear-de-um-neuronio @ [www.researchgate.net](http://www.researchgate.net)." [Online]. Available: [https://www.researchgate.net/figure/Figura-1-Modelo-nao-linear-de-um-neuronio-Haykin-2001\\_fig1\\_282647489](https://www.researchgate.net/figure/Figura-1-Modelo-nao-linear-de-um-neuronio-Haykin-2001_fig1_282647489).
- [37] J. M. Fonseca, "Redes Neurais." pp. 1–27.
- [38] "understanding-decision-trees-with-python @ [datascience.foundation](http://datascience.foundation)." .
- [39] "O que é árvore de decisão (decision tree)? @ [cienciaenegocios.com](http://cienciaenegocios.com)." .
- [40] J. M. Fonseca, "PS - Modulo 4 - Arvores de decisão.pdf." .
- [41] "3-techniques-to-avoid-overfitting-of-decision-trees @ [towardsdatascience.com](http://towardsdatascience.com)." .
- [42] M. S. Lauretto, "Árvores de Decisão 1 Definições," no. 1, pp. 1–27, 2010.
- [43] "bq\_aquaris\_m5@ [www.gsmarena.com](http://www.gsmarena.com)." .
- [44] "Android is for everyone @ [www.android.com](http://www.android.com)." .
- [45] "discover-intellij-idea @ [www.jetbrains.com](http://www.jetbrains.com)." .
- [46] Developers, "Intro @ [Developer.Android.Com](http://Developer.Android.Com)." 2020.
- [47] "java @ [www.lenovo.com](http://www.lenovo.com)." .
- [48] "platforms @ [developer.android.com](http://developer.android.com)." .
- [49] "index @ [apilevels.com](http://apilevels.com)." .
- [50] I. Cart, "Introducing CART ®." .
- [51] "ServiceLogin @ [issuetracker.google.com](http://issuetracker.google.com)." .
- [52] E. J. Aguiar *et al.*, "Normative peak 30-min cadence (Steps per minute) values for older

- adults: Nhanes 2005-2006," *J. Aging Phys. Act.*, vol. 27, no. 5, pp. 625–632, 2019.
- [53] "LocationRequest @ developers.google.com." .
- [54] "Overfitting – what and why? @ blog.knoldus.com." .
- [55] W. H. O. Alzheimer's Disease International, *Dementia: A Public Health Priority*. World Health Organization, 2012.
- [56] J. T. O'Brien and A. Thomas, "Vascular dementia," *Lancet*, vol. 386, no. 10004, pp. 1698–1706, Oct. 2015.
- [57] D. Weisman and I. McKeith, "Dementia with Lewy Bodies," *Semin. Neurol.*, vol. 27, no. 1, pp. 042–047, Feb. 2007.
- [58] F. Luis-Ferreira, D. Rodrigues, D. Pereira, J. Sarraipa, and R. Jardim-Gonçalves, "Assessment of Risk for Alzheimer Patients by Monitoring Heartbeat and Identifying Falls," in *Proceedings of 10th International Conference on e-Health 2018*, 2018.
- [59] L. Corral, A. B. Georgiev, A. Sillitti, and G. Succi, "A method for characterizing energy consumption in Android smartphones," in *2013 2nd International Workshop on Green and Sustainable Software (GREENS)*, 2013, pp. 38–45.
- [60] G. P. Perrucci, F. H. P. Fitzek, and J. Widmer, "Survey on Energy Consumption Entities on the Smartphone Platform," in *2011 IEEE 73rd Vehicular Technology Conference (VTC Spring)*, 2011, pp. 1–6.
- [61] F. Luis-Ferreira, J. Calado, A. Artifice, J. Sarraipa, and R. Jardim-Goncalves, "Personalised Technological Architectures to Assist Dementia Patients based on Energy Efficiency," in *ICIST 2019 - 9th International Conference on Information Society and Technology*, 2019.

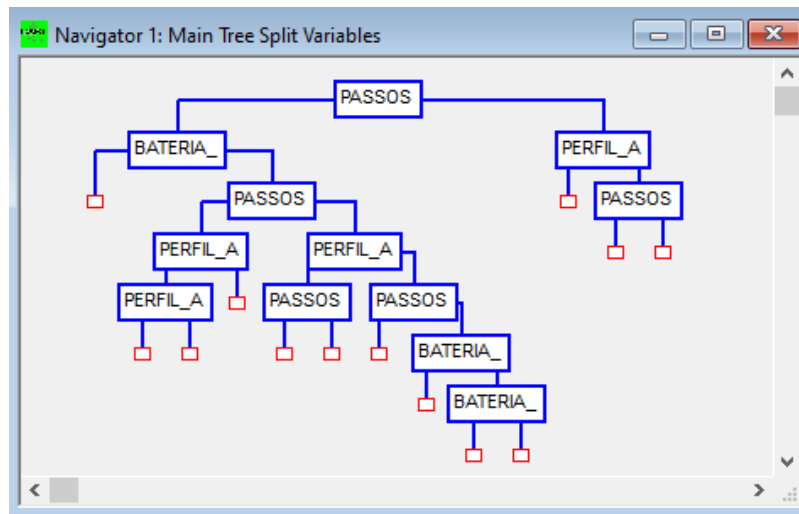
## A.2 Resultados CART

Porcentagem de cada classe nas folhas:





Variável de entrada escolhida em cada nó da árvore:



### A.3 Aplicações de teste para Android Studio

Capturas de ecrã das várias aplicações criadas para os testes das antenas e sensores utilizados na aplicação final:

- Bateria;



```
Health: 2
Level: 96%
Plugged: 2
Status: 2
Present: true
Technology:
Temperature: 28°C
Voltage: 4317mV
```

- Temporizador;



09:35

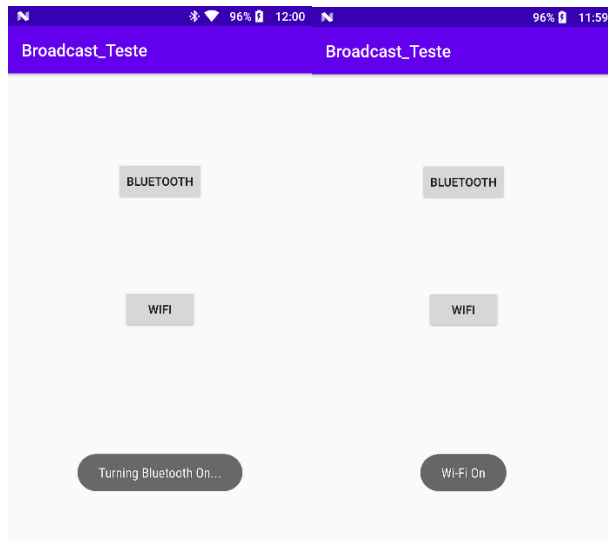
09:53



- GPS;



- *Wi-fi e bluetooth;*



## A.4 Código relevante não mencionado

Código do ficheiro .xml, usado para a criação da IU da aplicação.

```
?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:an-
droid="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/textEstado"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="29dp"
        android:text="Estado"
        android:textSize="24sp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/gpsLocal" />

    <ImageView
        android:id="@+id/imageWifi"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="29dp"
        app:layout_constraintEnd_toStartOf="@+id/imageBluetooth"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:srcCompat="@drawable/ic_wifi_on" />

    <ImageView
        android:id="@+id/imageBluetooth"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="29dp"
        app:layout_constraintEnd_toStartOf="@+id/imageGPS"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintStart_toEndOf="@+id/imageWifi"
        app:layout_constraintTop_toTopOf="parent"
        app:srcCompat="@drawable/ic_bluetooth_on" />

    <ImageView
        android:id="@+id/imageGPS"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
```

```

        android:layout_marginTop="29dp"
        app:layout_constraintEnd_toStartOf="@+id/imageMobileData"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintStart_toEndOf="@+id/imageBluetooth"
        app:layout_constraintTop_toTopOf="parent"
        app:srcCompat="@drawable/ic_gps_on" />

<TextView
    android:id="@+id/textPassos"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginBottom="15dp"
    android:text="TextView"
    app:layout_constraintBottom_toTopOf="@+id/wifiONButton"
    app:layout_constraintEnd_toStartOf="@+id/textBateria"
    app:layout_constraintHorizontal_bias="0.5"
    app:layout_constraintStart_toStartOf="parent" />

<TextView
    android:id="@+id/textBateria"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginBottom="17dp"
    android:text="TextView"
    app:layout_constraintBottom_toTopOf="@+id/gpsONButton"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.5"
    app:layout_constraintStart_toEndOf="@+id/textPassos"
    tools:ignore="NotSibling" />

<Button
    android:id="@+id/wifiONButton"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginBottom="24dp"
    android:text="wifi on"
    app:layout_constraintBottom_toTopOf="@+id/wifiOFFButton"
    app:layout_constraintEnd_toStartOf="@+id/btONButton"
    app:layout_constraintHorizontal_bias="0.5"
    app:layout_constraintStart_toStartOf="parent" />

<Button
    android:id="@+id/wifiOFFButton"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="wifi off"
    app:layout_constraintBaseline_toBaselineOf="@+id/btOFFButton"
    app:layout_constraintEnd_toStartOf="@+id/btOFFButton"
    app:layout_constraintHorizontal_bias="0.5"
    app:layout_constraintStart_toStartOf="parent" />

<Button
    android:id="@+id/btONButton"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="17dp"
    android:text="bt on"
    app:layout_constraintEnd_toStartOf="@+id/gpsONButton"
    app:layout_constraintHorizontal_bias="0.5"

```

```

        app:layout_constraintStart_toEndOf="@+id/wifiONButton"
        app:layout_constraintTop_toBottomOf="@+id/textBateria" />

<Button
    android:id="@+id/btOFFButton"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="22dp"
    android:text="bt off"
    app:layout_constraintEnd_toStartOf="@+id/gpsOFFButton"
    app:layout_constraintHorizontal_bias="0.5"
    app:layout_constraintStart_toEndOf="@+id/wifiOFFButton"
    app:layout_constraintTop_toBottomOf="@+id/btONButton" />

<Button
    android:id="@+id/gpsONButton"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="65dp"
    android:text="gps on"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.5"
    app:layout_constraintStart_toEndOf="@+id/btONButton"
    app:layout_constraintTop_toBottomOf="@+id/textEstado" />

<Button
    android:id="@+id/gpsOFFButton"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="22dp"
    android:text="gps off"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.5"
    app:layout_constraintStart_toEndOf="@+id/btOFFButton"
    app:layout_constraintTop_toBottomOf="@+id/gpsONButton" />

<TextView
    android:id="@+id/gpsLocal"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="36dp"
    android:text="Localização"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.5"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/imageBluetooth" />

<TextView
    android:id="@+id/textProfile"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="15dp"
    android:text="TextView"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.5"

```

```

        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/btOFFButton" />

<ImageView
    android:id="@+id/imageMobileData"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="29dp"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.5"
    app:layout_constraintStart_toEndOf="@+id/imageGPS"
    app:layout_constraintTop_toTopOf="parent"
    app:srcCompat="@drawable/ic_mobiledata_on" />

</androidx.constraintlayout.widget.ConstraintLayout>

```

Código da atividade principal do programa (Main Activity).

```

package com.example.tese;

import android.Manifest;
import android.content.Intent;
import android.content.IntentFilter;
import android.content.SharedPreferences;
import android.content.pm.ActivityInfo;
import android.content.pm.PackageManager;
import android.os.Bundle;
import android.telephony.SmsManager;
import android.util.Log;
import android.view.View;
import android.view.WindowManager;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;
import androidx.core.content.ContextCompat;

import com.google.android.gms.location.LocationRequest;

import java.text.SimpleDateFormat;
import java.util.Date;

public class MainActivity extends AppCompatActivity {

    private static final String PREFERENCES_NAME = "prefs";
    private TextView textEstado, textPassos, textBateria, gpsLocal,
textProfile;
    private ImageView imageWifi, imageBluetooth, imageGPS, imageMobileData;
    private Button wifiONButton, wifiOFFButton, btONButton, btOFFButton,
gpsONButton, gpsOFFButton;

    // Controllers
    private WifiController wifiController;

```

```

private BluetoothController bluetoothController;
private GpsController gpsController, gpsMDCController;
private BatteryController batteryController;
private AcelerometerController acelerometerController;
private TimerController timerController;
private ProfileController profileController = null;

private ProfileEnum currentProfile = ProfileEnum.PROFILE_0;

private DecisionTree decisionTree;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    // Screen orientation
    setRequestedOrientation(ActivityInfo.SCREEN_ORIENTATION_PORTRAIT);
    // Screen always on
    getWindow().addFlags(WindowManager.LayoutParams.
Params.FLAG_KEEP_SCREEN_ON);

    textEstado = findViewById(R.id.textEstado);
    textBateria = findViewById(R.id.textBateria);
    textPassos = findViewById(R.id.textPassos);
    textProfile = findViewById(R.id.textProfile);
    gpsLocal = findViewById(R.id.gpsLocal);
    wifiONButton = findViewById(R.id.wifiONButton);
    wifiOFFButton = findViewById(R.id.wifiOFFButton);
    btONButton = findViewById(R.id.btONButton);
    btOFFButton = findViewById(R.id.btOFFButton);
    gpsONButton = findViewById(R.id.gpsONButton);
    gpsOFFButton = findViewById(R.id.gpsOFFButton);

    //Imagens
    imageWifi = findViewById(R.id.imageWifi);
    imageBluetooth = findViewById(R.id.imageBluetooth);
    imageGPS = findViewById(R.id.imageGPS);
    imageMobileData = findViewById(R.id.imageMobileData);

    //Controllers
    wifiController = new WifiController(this, imageWifi);
    bluetoothController = new BluetoothController(this, imageBlue-
tooth);
    gpsController = new GpsController(this, MainActivity.this, imageGPS,
gpsLocal, LocationRequest.PRIORITY_HIGH_ACCURACY);
    gpsMDCController = new GpsController(this, MainActivity.this, image-
MobileData, gpsLocal, LocationRequest.PRIORITY_BALANCED_POWER_ACCURACY);
    batteryController = new BatteryController(this, MainActivity.this,
textBateria);
    acelerometerController = new AcelerometerController(this,
textPassos);
    timerController = new TimerController(this, textEstado, MainActiv-
ity.this);
    //mobileDataController = new MobileDataController(this,

```

```

imageMobileData);

    //Decision Tree
    decisionTree = new DecisionTree();
    decisionTree.generateDecisionTree();

    wifiController.setImage();
    bluetoothController.setImage();
    gpsController.setImage();
    gpsMDCController.setImage();

    acelerometerController.resetStepsCount();
    profileController = Profile.getProfile(currentProfile);
    profileController.setSettings(wifiController, bluetoothController,
gpsController, gpsMDCController, timerController, acelerometerController);
    textProfile.setText("Profile: " + currentProfile);

    acelerometerController.registerStepsListener();

    ActivityCompat.requestPermissions(MainActivity.this, new
String[]{Manifest.permission.SEND_SMS, Manifest.permission.READ_SMS}, Pack-
ageManager.PERMISSION_GRANTED);
}

@Override
protected void onStart() {
    super.onStart();

    //TIMER
    timerController.onStartTimer();

    registerReceiver(batteryController.getBatteryLevelReceiver(), new
IntentFilter(Intent.ACTION_BATTERY_CHANGED));

    //permissão GPS
    if (ContextCompat.checkSelfPermission(this, Manifest.permission.AC-
CESS_FINE_LOCATION) != PackageManager.PERMISSION_GRANTED) {
        gpsController.pedirAcessoLocalizacao();
    }

    //BOTÕES
    wifiONButton.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            wifiController.turnON();
        }
    });

    wifiOFFButton.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            wifiController.turnOFF();
        }
    });

    btONButton.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            bluetoothController.turnON();

```

```

    }
    });

    btOFFButton.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            bluetoothController.turnOFF();
        }
    });

    gpsONButton.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            gpsController.turnON();
        }
    });

    gpsOFFButton.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            gpsController.turnOFF();
        }
    });
}

public void getNewProfile() {
    currentProfile = decisionTree.getProfile(batteryController.getBatteryLevel(),
    acelerometerController.stepCount, currentProfile);
    profileController = Profile.getProfile(currentProfile);
    profileController.setSettings(wifiController, bluetoothController,
    gpsController, gpsMDCController, timerController, acelerometerController);
    textProfile.setText("Profile: " + currentProfile);
}

public void sendSMS () {
    try {
        SimpleDateFormat formatter = new SimpleDateFormat("dd/MM/yyyy
HH:mm:ss");
        String gpsInfo = gpsLocal.getText().toString();
        String message = String.format("Perfil: %s\nData: %s\n%s;",
currentProfile.toString(), formatter.format(new Date()), gpsInfo);
        Log.d("testSms", message); //TODO: Para remover no futuro
        SmsManager mySmsManager = SmsManager.getDefault();
        mySmsManager.sendTextMessage("+351968721402", null, message,
null, null);
        showToast("Mensagem enviada");
    } catch (Exception e) {
        showToast("Erro ao enviar mensagem");
    }
}

@Override
public void onRequestPermissionsResult(int requestCode, @NonNull
String[] permissions, @NonNull int[] grantResults) {
    if(requestCode == gpsController.LOCATION_REQUEST_CODE){

```

```

        if (grantResults.length > 0 && grantResults[0] == Pack-
ageManager.PERMISSION_GRANTED) {
            //Permissão concedida
            gpsController.turnON();
        } else {
            //Permissão não concedida
            gpsController.pedirAcessoLocalizacao();
        }
    }
}

@Override
protected void onStop() {
    super.onStop();

    unregisterReceiver(batteryController.getBatteryLevelReceiver());

    //ACELEROMETRO
    SharedPreferences prefs = getSharedPreferences(PREFS_NAME,
MODE_PRIVATE);
    SharedPreferences.Editor editor = prefs.edit();
    editor.clear();
    editor.putInt("stepCount", acelerometerController.stepCount);
    editor.apply();

    // TIMER
    timerController.onStopTimer();
}

@Override
protected void onResume() {
    super.onResume();

    SharedPreferences sharedPreferences = getPreferences(MODE_PRIVATE);
    acelerometerController.stepCount = sharedPreferences.getInt("step-
Count", 0);
}

@Override
protected void onPause() {
    super.onPause();

    SharedPreferences sharedPreferences = getPreferences(MODE_PRIVATE);
    SharedPreferences.Editor editor = sharedPreferences.edit();
    editor.clear();
    editor.putInt("stepCount", acelerometerController.stepCount);
    editor.apply();
}

private void showToast(String msg) { Toast.makeText(this, msg,
Toast.LENGTH_SHORT).show(); }
}

```

### Código do manifesto da aplicação:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.tese">

    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"
/>
    <uses-permission android:name="android.permission.BLUETOOTH" />
    <uses-permission android:name="android.permission.BLUETOOTH_ADMIN" />
    <uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
    <uses-permission android:name="android.permission.CHANGE_WIFI_STATE" />
    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"
/>
    <uses-permission android:name="android.permission.READ_PHONE_STATE" />
    <uses-permission android:name="android.permission.SEND_SMS" />
    <uses-permission android:name="android.permission.RECEIVE_SMS" />
    <uses-permission android:name="android.permission.READ_SMS" />

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">

        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER"
/>
            </intent-filter>
        </activity>
    </application>

</manifest>
```

## A.5 Tabelas de resultados dos testes de bateria

Testes de uma hora:

Teste Tudo Desligado	Bateria após 1h
1	92
2	93
3	93

<b>Teste Tudo Ligado</b>	<b>Bateria após 1h</b>
1	86
2	87
3	85

<b>Teste Aplicação</b>	<b>Bateria após 1h</b>
1	92
2	90
3	89

Teste de seis horas:

<b>Teste Tudo Desligado</b>	<b>Bateria após 6h</b>
1	40
2	39
3	43

<b>Teste Tudo Ligado</b>	<b>Bateria após 6h</b>
1	30
2	29
3	30

<b>Teste Aplicação</b>	<b>Bateria após 6h</b>
1	31
2	39
3	32



<a>  
2021  
</a>

DIOGO SILVA

UMA ARQUITETURA PARA PROMOVER A SEGURANÇA DE PACIENTES COM  
DEMÊNCIA UTILIZANDO MÉTODOS DE INTELIGÊNCIA ARTIFICIAL PARA  
OTIMIZAR BATERIA EM DISPOSITIVOS MÓVEIS