

UNIVERSIDADE DE LISBOA  
FACULDADE DE CIÊNCIAS  
DEPARTAMENTO DE INFORMÁTICA



## **Camadas Acessíveis: Acessibilidade Móvel através de uma Interface Consistente**

André Ricardo Branco Santos

**Mestrado em Engenharia Informática**  
Especialização em Sistemas de Informação

Dissertação orientada por:  
Tiago João Vieira Guerreiro e André Filipe Pereira Rodrigues

2017



# Agradecimentos

O meu especial agradecimento ao meu orientador, Professor Doutor Tiago Guerreiro, cujos conhecimentos foram fundamentais no decorrer deste trabalho, assim como ao meu coorientador, André Rodrigues, por me terem guiado nesta jornada.

Ao LaSIGE pelas condições oferecidas durante a realização do meu trabalho.

À minha família, em especial aos meus pais, por todo o apoio que me deram durante os difíceis anos da faculdade, sobretudo nos últimos dois de mestrado. Sem o seu suporte e motivação durante os momentos menos bons deste percurso não teria sido possível seguir os meus sonhos.

A todas as pessoas da sala 34, que me acompanharam ao longo da concretização deste projeto. Sem eles e sem a pausa do café seria muito mais difícil.

A todos os amigos, nomeadamente à Alice e ao Gonçalo, por me convencerem a sair de casa e aproveitar a vida, impedindo que sucumbisse ao imenso trabalho para fazer e por me apoiarem sempre, mesmo não entendendo nada sobre o projeto que realizei.

À Fundação *Raquel e Martin Sain*, pela colaboração na validação do projeto, e a todos os que se disponibilizaram para realizar os testes.

Por fim gostaria de agradecer a todos os elementos da FCUL pela formação que me proporcionaram e por me ajudarem a tornar o homem que sou hoje, fazendo com que me orgulhe de mencionar o nome desta instituição.

Em suma agradeço as todos os que acreditaram em mim.



*Aos meus avós.*



# Resumo

Com a evolução tecnológica, as teclas físicas dos telemóveis foram substituídas por um ecrã maior e sensível ao tato. Esta foi uma mudança que revolucionou a maneira como interagimos com dispositivos móveis, mas que introduziu um problema de acessibilidade para quem não consegue ver o que é apresentado no ecrã. Este problema encontra-se particularmente na localização de elementos e navegação nas aplicações existentes.

Atualmente, os sistemas operativos móveis oferecem algumas soluções de acessibilidade aos seus utilizadores tais como leitores de ecrã, fontes maiores ou suporte para teclados externos. Estas soluções são um grande contributo para ajudar estes utilizadores a usar um *smartphone*, mas não resolvem totalmente o problema apresentado uma vez que o utilizador é sempre obrigado a conhecer previamente a interface ou a explorar todos os seus elementos em busca do elemento que pretende.

Neste projeto, apresentamos o *Accessible Templates*. Uma aplicação que pretende tornar mais fácil o uso de um *smartphone* através de uma interface acessível e consistente em qualquer janela do sistema. A aplicação oferece dois tipos de navegação através de botões ou através do sistema de gestos direcionais do *Talkback*. Oferece ainda um sistema de favoritos e de macros para realizar tarefas complexas automaticamente.

Foram realizados estudos com utilizadores cegos onde foi possível verificar a aceitação e facilidade de uso da aplicação desenvolvida.

**Palavras-chave:** *Smartphone*, Interface, acessibilidade, pessoas cegas, *Android*.





# Abstract

With the technological evolution, the physical keys of mobile phones have been replaced by a larger screen, sensitive to touch. This was a change that revolutionized the way we interact with mobile devices, but it has introduced an accessibility problem for those who cannot see what is displayed on the screen. This problem lies particularly in locating elements and navigating in common applications.

Currently, mobile operating systems offer some accessibility solutions to their users such as screen readers, larger fonts, or support for external keyboards. These solutions are a great help for these users when using a smartphone, but they do not fully solve the problem presented since the user is always obliged to know the interface beforehand or to explore all its elements in search of the element they want to select.

In this project, we present the Accessible Templates, an application that intends to make the use of a smartphone easier through an accessible and consistent interface in any system window. The application offers two types of navigation through buttons or through the directional Talkback gesture system. It also offers a favourites and macros system to perform complex tasks automatically.

Studies with blind users were carried out in which it was possible to verify the acceptance and ease of use of the developed application.

**Keywords:** Smartphone, Interface, accessibility, blind people, Android.



# Conteúdo

Capítulo 1	Introdução.....	1
1.1	Motivação .....	1
1.2	Objetivos.....	2
1.3	Descrição da abordagem.....	3
1.4	Contribuições.....	3
1.5	Publicações .....	4
Capítulo 2	Trabalho relacionado.....	5
2.1	Soluções que recorrem a <i>hardware</i> .....	5
2.1.1	Adereços.....	6
2.1.2	Feedback tátil .....	6
2.2	Soluções que recorrem a <i>software</i> .....	8
2.2.1	Entrada de Braille.....	8
2.2.2	Interação com o dispositivo.....	11
2.3	Sumário.....	14
Capítulo 3	Camadas Acessíveis .....	17
3.1	Requisitos .....	17
3.1.1	Funcionais .....	18
3.1.2	Não funcionais.....	18
3.2	Conceção .....	18
3.2.1	Ordenação.....	18
3.2.2	Interface.....	19
3.2.3	Navegação .....	20
3.2.4	Personalização.....	20
3.2.5	Opções da área fixa .....	21
Capítulo 4	Implementação .....	23
4.1	Serviço de acessibilidade e eventos.....	23
4.2	Componentes .....	24

4.2.1	Processador de Eventos .....	25
4.2.2	Biblioteca .....	28
4.2.3	Navegação .....	29
4.2.4	Classes auxiliares e estruturas de dados .....	32
4.2.5	View .....	33
4.2.6	Atividades.....	34
4.2.7	Serviço.....	34
4.3	Funcionalidades .....	35
Capítulo 5	Avaliação com utilizadores .....	39
5.1.1	Participantes .....	39
5.1.2	Dispositivo .....	40
5.1.3	Procedimento.....	40
5.1.4	Análise de dados.....	41
5.1.5	Resultados .....	41
5.2	Discussão .....	49
Capítulo 6	Conclusão .....	51
6.1	Benefícios .....	51
6.1.1	Adaptação.....	51
6.1.2	Interface.....	52
6.1.3	Personalização.....	52
6.1.4	Ordenação do conteúdo.....	52
6.2	Limitações .....	52
6.2.1	Explore by touch .....	53
6.2.2	Gestos avançados .....	53
6.2.3	Limitações técnicas .....	53
6.3	Trabalho futuro .....	53
6.3.1	Integração com o leitor de ecrã .....	54
6.3.2	Explorar formas novas de navegar .....	54
6.3.3	Comandos por voz.....	54

Capítulo 7 Bibliografia.....	55
Anexo.....	58
A1. Livro de códigos.....	58
A2. Guião do estudo .....	60
A3. Entrevista .....	64



# Lista de Figuras

Figura 1 Luvas Braille sem fios - [1] .....	6
Figura 2 UbiBraille - [15] .....	7
Figura 3 HoliBraille - [17] .....	7
Figura 4 Disposição dos pontos de contacto – BrailleType [21] .....	9
Figura 5 Introdução de texto – Perkinput [2] .....	10
Figura 6 Correção ao nível da célula .....	10
Figura 7 Correção ao nível da palavra .....	10
Figura 8 SlideRule [10] – Gestos de interação .....	11
Figura 9 JustSpeak – Sequência de comandos [32] .....	12
Figura 10 Ordenação original .....	19
Figura 11 Ordem alfabética .....	19
Figura 12 Esboço da interface, metade superior - Área fixa, metade inferior - Área de navegação.....	19
Figura 13 Exemplo de interação .....	24
Figura 14 Componentes principais do Accessible Templates .....	24
Figura 15 Algoritmo de navegação no conteúdo .....	26
Figura 16 Exemplo de interface da aplicação "definições" .....	28
Figura 17 Cálculo da interseção das normais de dois segmentos de reta .....	30
Figura 18 Ordenação alfabética .....	35
Figura 19 Ordenação “mais usados” .....	35
Figura 20 Ordenação “mais recentes” .....	35
Figura 21 Ordenação “interativos” .....	36
Figura 22 Ordenação “favoritos” .....	36





# Lista de Tabelas

Tabela 1 Experiência dos participantes.....	39
Tabela 2 Tarefas realizadas.....	41



# Lista de equações

Equação 1 Equação da reta AB.....	31
Equação 2 Equação da reta BC.....	31
Equação 3 Equação do declive da reta AB.....	31
Equação 4 Equação do declive da reta BC.....	31
Equação 5 Equação da reta normal da reta AB no ponto D.....	31
Equação 6 Equação da reta normal da reta BC no ponto E.....	31
Equação 7 Equação da coordenada X do ponto F.....	32



# Capítulo 1

## Introdução

Atualmente, os dispositivos móveis são uma mais valia nas nossas vidas. Estes dispositivos desempenham um grande número de tarefas para além das convencionais, efetuar chamadas ou enviar mensagens. Tarefas que antes só eram possíveis de realizar através de um computador, como aceder a redes sociais, enviar emails ou aceder a conteúdos multimédia, estão agora à nossa disposição em qualquer lugar.

### 1.1 Motivação

Hoje em dia, grande parte da população tem ao seu dispor um dispositivo móvel. Com o avançar da tecnologia, os telefones convencionais perderam as suas teclas físicas, estas foram substituídas por um ecrã maior e sensível ao tato. Uma mudança que introduziu um grande problema para pessoas com baixo ou nenhum nível de visão. Como o ecrã não possui nenhum ponto de referência por onde a pessoa se possa guiar, a localização de elementos torna-se difícil. A interação e navegação nestes dispositivos, em grande parte das aplicações existentes, foca-se fortemente na habilidade de o utilizador conseguir localizar visualmente informação apresentada no ecrã. Este problema intensifica-se ainda mais quando os utilizadores têm pouca ou nenhuma experiência em interagir com dispositivos móveis.

Para fazer face a este problema, os sistemas operativos móveis integram leitores de ecrã como o *TalkBack*<sup>1</sup> ou o *VoiceOver*<sup>2</sup>. Estas ferramentas permitem ao utilizador ter acesso a toda a informação contida no ecrã de forma auditiva, bem como interagir de forma assistida com a mesma. Estas soluções são uma grande ajuda para pessoas cegas, mas não resolvem totalmente o problema da navegação. Para conseguir interagir com os leitores de ecrã é necessário que o utilizador aprenda um conjunto de gestos necessários para navegar, o que só por si já é um desafio. Como é apresentado no estudo realizado por Rodrigues e colegas [26], estes utilizadores têm uma grande dificuldade em realizar os gestos mais simples necessários para interagir com as ferramentas de acessibilidade oferecidas. É também necessário que a interface seja conhecida previamente ou que o utilizador a explore em busca do elemento que pretende.

---

<sup>1</sup> <https://play.google.com/store/apps/details?id=com.google.android.marvin.talkback&hl>, última visita 19/9/2017

<sup>2</sup> <https://www.apple.com/accessibility/iphone/vision/>, última visita 19/9/2017

Outro problema encontrado por Rodrigues e colegas [26] na maioria das aplicações, e que se mantém com o uso dos leitores de ecrã, é a inconsistências na navegação. Nos sistemas operativos as aplicações podem manter ou perder um determinado estado conforme a memória disponível do dispositivo. Isto faz com que os passos aprendidos de uma determinada tarefa possam não ser iguais quando o utilizador voltar à aplicação. Por exemplo, estar a escrever uma mensagem, sair desse contexto, e quando volta, estar nesse ecrã em vez de estar na lista de mensagens que é o ecrã inicial da aplicação. Estas inconsistências dificultam a aprendizagem dos utilizadores em relação à posição dos elementos.

Tendo em conta as limitações existentes nestas soluções, propomos a utilização de camadas acessíveis. Esta proposta utiliza um serviço de acessibilidade para o sistema operativo *Android* e pretende que o utilizador use uma interface consistente e única para interagir com o *smartphone*. O sistema permite ao utilizador aceder a todas as funcionalidades do seu *smartphone*. Todos os elementos interativos de qualquer interface que se encontram visíveis no *smartphone* são mapeados na sua própria, que se mantém igual e consistente à semelhança dos telefones com teclados físicos. Será também possível personalizar esta interface da melhor maneira para cada utilizador, por exemplo dando sugestões de interação através de macros, que poderão ser partilhadas pelos utilizadores. Estas permitem automatizar tarefas morosas, que levam vários cliques a ser realizadas, como ativar o *WiFi* ou ver a última mensagem recebida.

## 1.2 Objetivos

O objetivo principal desta dissertação é desenvolver um sistema que facilita a adaptação e o uso de dispositivos móveis por parte de utilizadores cegos. Isto foi alcançado por:

- **Oferecer uma Interface consistente e acessível** - Oferecer uma interface consistente que facilita a adoção do dispositivo por parte do utilizador pois este apenas necessita de se familiarizar com uma interface. Esta interface permite que o utilizador usufrua de todas as funcionalidades do seu *smartphone* de uma maneira simples.
- **Permitir uma navegação personalizada** - Oferecer várias opções de ordenação do conteúdo navegado pelo utilizador para permitir diversas estratégias de navegação. Estas estratégias poderão adaptar-se a diferentes necessidades conforme a aplicação ou as necessidades do utilizador.

- **Permitir a personalização da interface pelos utilizadores** - A interface apresentada ao utilizador poderá ser personalizável tanto pelo mesmo como pela aplicação sem comprometer a sua consistência. A avaliação da interface contará com a ajuda dos utilizadores, onde estes podem escolher funções que querem ter ao seu dispor.

- **Permitir a automatização de tarefas complexas** - Utilizando um sistema de macros, a aplicação oferece ao utilizador um mecanismo fácil de realizar tarefas que de outra maneira seriam demasiado complexas. O utilizador terá a possibilidade de gravar, descarregar e partilhar estas tarefas.

### 1.3 Descrição da abordagem

A abordagem apresenta uma interface única e consistente que serve como intermediário entre o utilizador e o sistema. Esta interface permite ao utilizador tirar partido de todas as funcionalidades do seu dispositivo sem ter que decorar o posicionamento dos elementos de cada aplicação que utiliza. São oferecidos vários tipos de ordenações da lista de conteúdo para tornar o sistema mais adaptável a cada utilizador. Entre as diferentes ordenações destacam-se a que permite ordenar a lista de forma alfabética e a ordenação que permite navegar pelos elementos mais usados primeiro.

### 1.4 Contribuições

As contribuições deste projeto englobam:

1) **Nova abordagem para a acessibilidade móvel** – esta abordagem permite aos utilizadores terem de volta a consistência que tinham nas interfaces físicas sem prejudicar o acesso à variedade de aplicações e funcionalidades dos seus telemóveis;

2) **Sistema que implementa esse conceito** – a aplicação concebida durante este projeto, para o sistema operativo Android, encontra-se publicada na loja de aplicações Google Play Store <sup>3</sup>;

3) **Validação preliminar com pessoas cegas dos benefícios e desafios desta abordagem** – esta validação foi feita através de um estudo com 9 utilizadores cegos, que consistiu na realização de uma serie de tarefas seguidas de uma entrevista.

---

<sup>3</sup> <https://play.google.com/store/apps/details?id=pt.ul.fc.di.lasige.andre.tese>, última visita em 26/07/2017

## 1.5 Publicações

As contribuições apresentadas nesta dissertação resultaram num artigo científico aceite numa conferencia internacional:

- André Rodrigues, André Santos, Kyle Montague, and Tiago Guerreiro. 2017. Improving Smartphone Accessibility with Personalizable Static Overlays. In *Proceedings of the 19th International ACM SIGACCESS Conference on Computers and Accessibility - ASSETS '17*, 37–41. <https://doi.org/10.1145/3132525.3132558>



## Capítulo 2

### Trabalho relacionado

No passado, os dispositivos móveis possuíam um teclado alfanumérico e este era uma mais valia para pessoas cegas pois as suas teclas serviam de guias espaciais para o utilizador. Com a massificação dos smartphones e os seus ecrãs táteis foram introduzidos novos problemas de acessibilidade que vieram dificultar ainda mais a já difícil interação de pessoas cegas com dispositivos móveis.

O estudo [26], conduzido ao longo de oito semanas, foca nas dificuldades de adoção de smartphones por parte de pessoas cegas. Realça duas grandes preocupações destes utilizadores aquando da adoção destes dispositivos; medo do desconhecido e medo que ao utilizar um smartphone este possa ser um risco adicional à sua segurança. A maioria esperava uma adaptação mais rápida e fácil, mas tal não se verificou. Com o avançar do estudo, as dificuldades dos utilizadores evoluíram desde a dificuldade em utilizar os gestos necessários para interagir com o smartphone para a dificuldade em entender o comportamento das aplicações que podem arrancar em diferentes estados. Isto leva a que fiquem totalmente perdidos ao não encontrarem o ecrã ou elemento que estão à espera em determinado passo do caminho.

No final do estudo a maioria sentiu que usar o smartphone era mais difícil que o seu telemóvel antigo maioritariamente devido à habituação já existente na interação com o seu dispositivo. Deve-se também ao facto de não serem oferecidas ferramentas totalmente eficazes para uma adaptação mais suave e simples. Por outro lado, foram reconhecidos benefícios no uso de smartphones como poderem ser mais ativos socialmente usando aplicações sociais ou como resolver problemas do dia a dia usando aplicações utilitárias.

Tendo em conta os problemas descritos a cima, nesta secção são abordadas algumas soluções para os solucionar. Encontra-se dividida em duas partes, soluções que recorrem a *hardware* externo ao *smartphone* e soluções que recorrem somente a *software*.

#### 2.1 Soluções que recorrem a *hardware*

Neste campo existe uma grande diversidade de soluções para melhorar a interação de utilizadores cegos com dispositivos móveis seja para introdução de texto seja para leitura de ecrã. Dada uma grande parte das soluções apresentadas tirar partido de feedback tátil através de motores de vibração, esta secção divide-se em soluções que usam

*hardware* específico e soluções que modificam o dispositivo móvel como as que recorrem a feedback tátil.

### 2.1.1 Adereços

Para melhorar a experiência de utilizadores cegos com dispositivos móveis têm sido criados dispositivos específicos de raiz para combater as suas dificuldades com os dispositivos.

Estas soluções variam desde dispositivos mais comuns como telefones <sup>4</sup> que possuem teclas em braile mas têm funcionalidades reduzidas dando apenas para efetuar e receber chamadas e permitindo gravar apenas 4 números; teclados de braile [29] que permitem introduzir texto para os dispositivos; ecrãs de braile [30] que mostram em braile o que se encontra no ecrã; até dispositivos mais peculiares como luvas [1,6] para introdução de texto braile ou um dispositivo que possui diversos motores de vibração e permite ler braile em diversas partes do corpo [20] para perceção mais rápida ou pessoas que tenham pouca sensibilidade nas mãos.



Figura 1 Luvas Braille sem fios - [1]

### 2.1.2 Feedback tátil

As soluções apresentadas em [4,5] pretendem ajudar utilizadores invisuais a localizar mais facilmente objetos no ecrã de um smartphone. Consistem na criação de marcadores que delimitam áreas importantes no ecrã, por exemplo separar os números de marcação na aplicação do telefone do botão de efetuar chamada. Estes marcadores podem ser usados em conjunto com os restantes serviços de acessibilidade ou sozinhos em ambientes barulhentos onde não é possível ouvir o feedback auditivo. Os autores analisaram a interface de duas aplicações *Android*, o Telefone e o *Gmail*. Concluíram que diferentes interfaces necessitam de marcadores específicos, ou seja, é necessário que quando a aplicação arranca estes marcadores sejam ativados dinamicamente.

---

<sup>4</sup> <http://www.ownfone.com/braille>, última visita 30/09/2016

Na solução apresentada em [7] utilizam feedback tátil para complementar o uso do dispositivo. A solução pretende ajudar pessoas cegas a navegar autonomamente em museus. Esta aplicação utiliza dois motores de vibração em conjunto com marcadores visuais localizados no museu para ajudar o utilizador a navegar pelas diversas peças do museu. A aplicação utiliza ainda voz sintetizada para descrever as peças do museu e fornecer indicações de navegação. Os motores de vibração são colocados à volta de dois dedos do utilizador e vibram quando este tem que efetuar uma rotação para esquerda ou direita, é possível ainda ajustar a força da vibração.

O sistema apresentado em [15] oferece um mecanismo vibro-tátil que recria o código de seis pontos tradicionais do Braille permitindo ao utilizador ter acesso a informação de forma não auditiva. O sistema é composto por seis atuadores em que cada um representa um ponto da matriz de seis pontos. (Figura 2).



Figura 2 UbiBraille - [15]

Em [17] é apresentado um sistema para leitura e introdução de Braille. Tal como no sistema apresentado acima este faz uso de seis atuadores que representam os seis pontos da matriz usada no Braille. Podemos ver um protótipo do sistema na Figura 3.

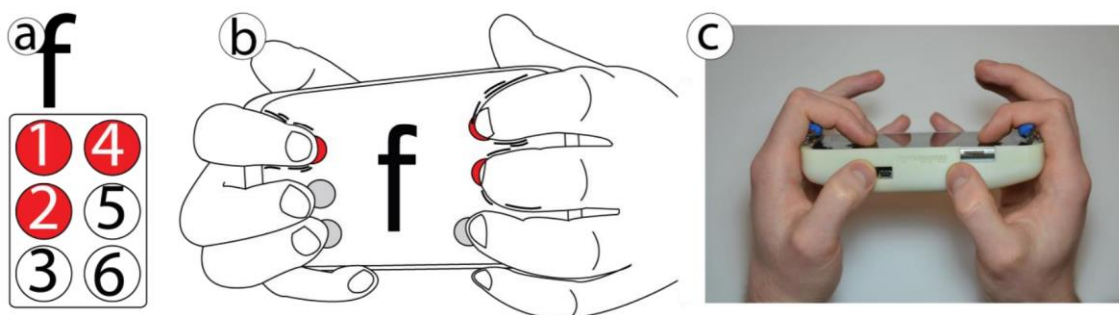


Figura 3 HoliBraille - [17]

Por fim a solução sugerida em [28] faz uso de um sistema composto por um PDA, 5 motores de vibração, um placa de circuitos e um computador em que o seu objetivo é fornecer uma navegação *eyes-free*, ou seja sem ser necessário olhar para o ecrã dos dispositivo. Os motores estão dispostos em forma de cruz, direita, esquerda, topo, base e

centro e é possível criar padrões sincronizados através dos mesmos. Estes são controlados através da placa de circuitos e do computador em que o seu propósito é fornecer informação sobre o elemento em que o utilizador está a carregar no *touchscreen*. Por exemplo num ecrã com um teclado numérico se o utilizador seleccionar o número 5 o motor do centro vibra. Esta solução não é ideal visto que necessita de demasiado hardware para além do smartphone.

Como descrito no artigo [27], o facto de ser necessário utilizar um dispositivo diferente seja em cor, tamanho ou formato pode levar a que as pessoas que necessitam deles a não se sentirem confortáveis a usá-los em público. Hoje em dia ainda existem percepções erradas sobre estes dispositivos e o seu propósito, maioritariamente por falta de informação. São sugeridas duas principais percepções, a primeira que estes dispositivos fazem com que a pessoa consiga realizar todas as suas ações com a mesma facilidade que uma pessoa normovisual faria. A segunda, que as pessoas sem os seus dispositivos não conseguem realizar qualquer tarefa.

Tendo em conta a existência destas percepções erradas podemos concluir que estes dispositivos não conseguem cumprir totalmente a sua funcionalidade e que levam as pessoas a não se sentir confortáveis em utilizá-los em público. O desenho destes dispositivos não deve contemplar só as suas funções técnicas, mas também a sua estética e aceitação social. Em conclusão, estes dispositivos só são realmente eficazes se forem utilizados eficazmente pelas pessoas no seu dia a dia.

## **2.2 Soluções que recorrem a *software***

Dado o constrangimento social e desejo demonstrado pelos utilizadores de utilizar dispositivos o mais comum [27] possível, surgiram soluções que, em vez de modificar o dispositivo em si ou utilizarem dispositivos específicos, adicionam uma camada de software extra. Esta camada pode tomar diferentes formas, ter diferentes objetivos e executar diferentes tarefas.

### **2.2.1 Entrada de Braille**

Na área de entrada de Braille, existem diversas soluções apresentadas, destacando-se o *Navtouch* [8], o *BrailleType* [21], o *Perkinput* [2] e o *B#* [16]. Onde os três primeiros são sistemas de introdução de Braille para dispositivos moveis e o quarto um sistema de sugestões para entrada de texto Braille.

Inicialmente, tentando eliminar a necessidade do uso de um teclado físico específico para Braille surgiu o *NavTap* [8], que é uma solução criada para telemóveis com teclas

físicas e que posteriormente foi adaptada para ecrãs táteis nascendo assim o *Navtouch* [8]. Esta é uma forma de entrada de texto para pessoas cegas apresentada em [8], onde os utilizadores não necessitam de decorar a posição referente a cada letra. É possível navegar através de 4 gestos direcionais (cima, baixo, esquerda, direita) pelas letras. As letras estão ordenadas alfabeticamente e organizadas em listas onde o primeiro elemento da lista é uma vogal perfazendo um total de 5 listas. Isto obriga os utilizadores a decorarem apenas o alfabeto e não a posição das letras, mesmo que não tenha memorizado a posição no alfabeto de todas as letras o utilizador pode pesquisar linearmente.

O sistema *BrailleType* [21], mapeia uma célula de braille, que pode ser vista como uma matriz de 3x2, no ecrã do dispositivo móvel. Para facilitar a procura os pontos de contacto onde o utilizador clica, estes são grandes e encontram-se em posições bem definidas e ordenados de acordo com uma célula braille (Figura 4).

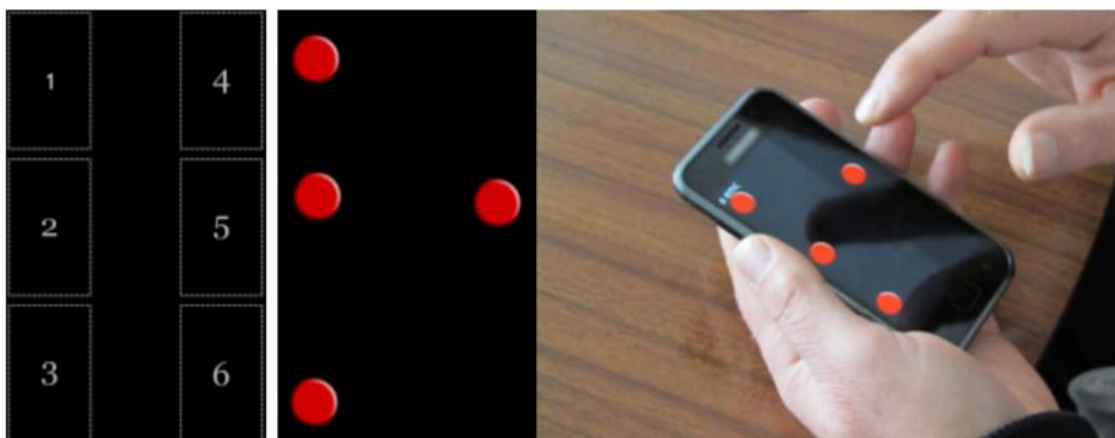


Figura 4 Disposição dos pontos de contacto – *BrailleType* [21]

Na sua utilização foi excluído o uso gestos avançados como gestos multi-toque, esta decisão facilita a aprendizagem e uso do sistema. A escrita é feita através de um único toque, para tal basta colocar o dedo no ecrã e procurar um dos pontos até ouvir uma pista auditiva. Foi implementado um temporizador configurável evitando assim seleções involuntárias.

No *Perkinput* [2], é também utilizada uma representação de uma célula Braille, mas os autores optaram por não utilizar pontos de contacto definidos. É utilizada uma técnica denominada *Input Finger Detection* em que é pedido ao utilizador para configura os pontos que deseja selecionar antes do início da utilização. A configuração permite que o utilizador escolha o número de dedos que deseja cada um representando um ponto de uma célula braille, até um total de seis. Se escolhidos menos de seis pontos o utilizador fará a introdução da célula faseadamente, por exemplo se configura três dedos de cada vez, a

introdução será feita em duas fases (Figura 5). O sistema permite ainda o uso de dois dispositivos para a introdução.



Figura 5 Introdução de texto – Perkinput [2]

Com a popularização dos sistemas de entrada de texto em braille em dispositivos com ecrã táctil surge um problema na precisão da escrita. É apresentado então o B# [16], um sistema de correção para entrada de texto em Braille oferecendo dois níveis de correção, célula a célula e palavra a palavra. Ao nível da célula o sistema tem em conta células que estão a uma distancia de pontos mais próxima (Figura 6). Ao nível da palavra o sistema tem em conta palavras que estão a uma distância de células mais próxima (Figura 7).

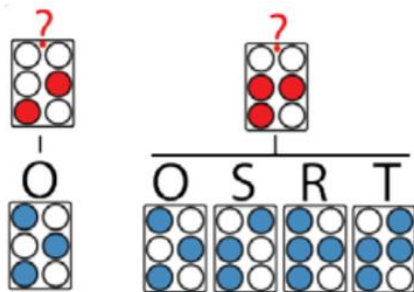


Figura 6 Correção ao nível da célula

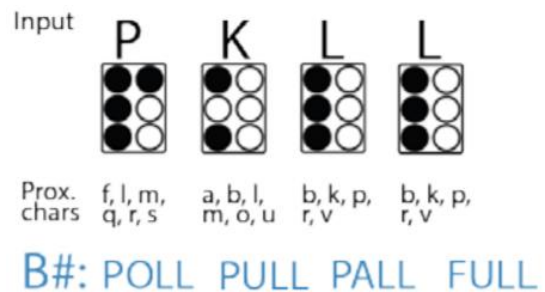


Figura 7 Correção ao nível da palavra

Oliveira et al. [22] analisaram quatro sistemas de entrada de texto, incluindo dois dos descritos, de introdução de Braille. Introdução através de um teclado virtual QWERTY com apoio do leitor de ecrã, MultiTap que tira partido de um teclado alfanumérico, o sistema *NavTouch* [8] que usa gestos direccionais para a procura de letras e o sistema *BrailleType* [21] que mapeia uma célula Braille no ecrã do dispositivo.

Foi concluído que os quatro métodos apresentam diferentes desafios, que são ultrapassadas por diferentes tipos de utilizadores, ou seja, cada utilizador tem necessidades diferentes.

## 2.2.2 Interação com o dispositivo

Outra solução anterior aos leitores de ecrã da Google e da Apple, é o *Slide Rule* apresentado em [10]. Este artigo propõe uma solução de navegação para o ecrã tátil de um smartphone. A navegação é problema para utilizadores cegos visto que a interação com estes dispositivos se basear fortemente na localização de objetos no ecrã. Este sistema oferece um conjunto de técnicas multi-toque com ajudas auditivas que facilitam o uso por parte destes utilizadores do smartphone. Oferece também uma interface totalmente não visual e usa um conjunto de 4 gestos para realizar a interação com o dispositivo, ilustrados na Figura 8; (1) interação com um dedo para percorrer listas, (2) um segundo toque para selecionar o item, (3) um *flick* multidirecional, e (4) um gesto em L para procurar itens que estão agrupados hierarquicamente.

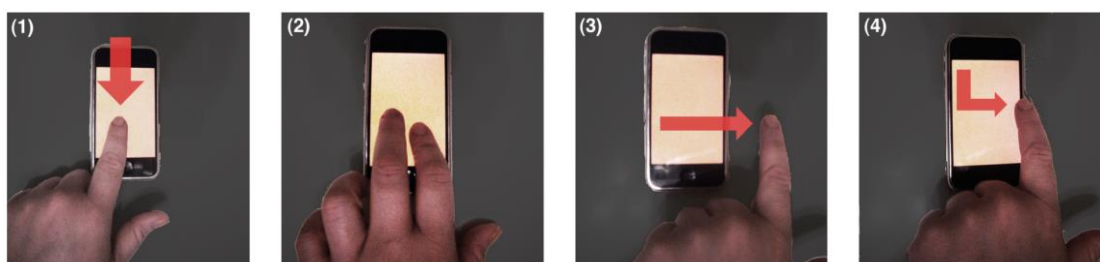


Figura 8 SlideRule [10] – Gestos de interação

Ao desenvolver esta aplicação os autores tiveram em conta 6 princípios de desenho.

- 1) Os utilizadores devem conseguir navegar pela interface sem correr o risco de efetuar alguma ação indesejada;
- 2) Como não é necessário que os itens apresentem as suas etiquetas visualmente o sistema pode usar o espaço em ecrã mais eficientemente apresentando mais itens;
- 3) Não ser necessário o utilizador selecionar o item com uma precisão extrema;
- 4) A navegação e a procura serem rápidas;
- 5) Os gestos serem intuitivos;
- 6) Permitir aos utilizadores saberem sempre onde se encontram no sistema e voltar ao ecrã inicial a qualquer momento.

Posteriormente com a generalização, propagação e evolução dos smartphones e do uso de um único ecrã tátil sem teclas físicas surgiu a necessidade de os tornar mais acessíveis. Para tal as empresas que desenvolvem os sistemas operativos criaram



softwares que leem o conteúdo que está presente no ecrã, dois casos são o *TalkBack*<sup>5</sup> e o *VoiceOver*<sup>6</sup>.

Tanto o *Talkback* como o *VoiceOver* têm como base os princípios apresentados no artigo do Slide Rule [10] e permitem navegar linearmente através de gestos pela interface caso esta seja desconhecida pelo utilizador, ou selecionar diretamente um objeto ao tocar nele. Tanto numa como noutra hipótese, seguidamente à seleção do objeto é lido o seu nome ou descrição e o tipo (botão, caixa de texto, etc...). Para efetuar a seleção, depois de escolhido o item, é necessário clicar uma segunda vez em qualquer parte do ecrã. Para utilizadores mais avançados existe um conjunto de gestos configuráveis que permitem efetuar diversas ações tais como clicar no item com dois 2 não sendo necessário selecionar previamente o item.

Surgiram ainda outras soluções para problemas mais específicos de interação com o smartphone como por exemplo as ferramentas utilitárias *eyes-free* earPod [31] e JustSpeak [32]. Apesar de terem o mesmo objetivo estas ferramentas resolvem-no de maneira diferente.

No caso do earPod [31], os autores deste sistema apresentam uma solução de menu circular *eyes-free* usando um ecrã tátil e feedback auditivo. Esta solução faz uso de um *touchpad* para a entrada dos gestos o que permite uma maior versatilidade. Os elementos do menu são mapeados em áreas físicas do *touchpad* e, quando selecionados, é reproduzida uma informação auditiva. Isto permite que os utilizadores descubram as opções ao seu próprio ritmo. A informação áudio apresentada só é reproduzida enquanto a opção do menu está selecionada. Assim, o utilizador pode ter uma navegação mais fluida e rápida na interface.



Figura 9 JustSpeak – Sequência de comandos [32]

<sup>5</sup> <https://play.google.com/store/apps/details?id=com.google.android.marvin.talkback&hl>, última visita 19/9/2017

<sup>6</sup> <https://www.apple.com/accessibility/iphone/vision/>, última visita 19/9/2017



No *JustSpeak* [32], os autores apresentam um serviço de comandos por voz para interagir com o sistema operativo *Android* sem ser necessário visualizar a interface. Este serviço permite interagir com qualquer aplicação e permite também encadear mais que um comando na mesma interação. Para efetuar o reconhecimento da voz os autores usam a API de ASR da Google que tanto pode ser usada online como offline, notando que, se o serviço estiver conectado à internet, oferece maior precisão por parte das previsões de reconhecimento de voz. Suporta dois conjuntos de comandos: 1) globais, que estão relacionados com o sistema operativo (ex: “*Open Facebook, Turn On Wifi*”); 2) locais, que estão relacionadas com a aplicação com a qual o utilizador está a interagir (ex: “*Scroll up, Click LogIn*”). O que destaca este serviço dos já existentes (Google Now, Siri, ...) é que permite a interação com qualquer aplicação existente, enquanto que as outras só permitem efetuar um conjunto predefinido de comandos, e o facto de poder encadear vários comandos numa só interação.

Tentando uma aproximação mais abrangente em termos de funções oferecidas, foram apresentados sistemas mais complexos que tentam oferecer outro tipo de interação aos utilizadores, caso do *Linear Interface* [12] e do *An Interface Agent for Non Visual* [23].

Kocielinski et al. [12] criaram uma interface linear multimodal e versátil (VMLI) que transforma a interface plana numa estrutura linear hierárquica em lista. O sistema suporta três tipos de teclados: um teclado QWERTY, um teclado virtual de Braille e um teclado físico de Braille. Esta interface consiste em apresentar todos os elementos de forma linear, mas é possível também aceder aos elementos de forma não sequencial. É possível ainda navegar pela lista através de um gesto de *gesto direcional* e ir para o início ou fim pressionando o ecrã nas suas extremidades. Para efetuar a seleção de um item é necessário clicar no centro do ecrã e seguidamente na extremidade direita para confirmação. Clicando na extremidade esquerda é possível retroceder na hierarquia. Carregar no ecrã com dois dedos ativa o modo janela que pode ser customizado pelo utilizador, esta janela só apresenta uma parte da lista.

Em [23], o autor apresenta um sistema cujo objetivo é ser um assistente autónomo que filtra informação a apresentar ao utilizador. Este assistente apresenta sugestões de conteúdo potencialmente interessante para o utilizador. Para tal, usa um modelo de previsão que faz estas sugestões e reduz a carga cognitiva do utilizador. Este pode optar por um sistema autónomo ou ter um maior controlo. É uma ferramenta que pode ser usada por qualquer browser e integrada com a maioria dos leitores de ecrã. Este assistente analisa as interações do utilizador com a interface gráfica e sugere implícita ou explicitamente diversas ações que o utilizador possa querer fazer em seguida. Estas ações

podem requerer a confirmação do utilizador para se concretizar ou ser efetuadas de maneira totalmente autónoma.

Existem ainda algumas soluções que tentam apenas mapear interfaces específicas como a de um *browser web* caso do Ember [9]. Os autores criaram uma interface para smartphones desenhada exclusivamente para pessoas cegas que tira partido do seu conhecimento em Braille. O principal objetivo desta interface é oferecer o menor número de elementos com o maior tamanho possível o que leva a que estes sejam fáceis de localizar e selecionar. Esta interface tira partido da facilidade do utilizador em localizar as extremidades do ecrã para colocar os seus botões (back, forward, home, audio). No centro do ecrã onde não existem botões é colocada uma caixa para entrada de texto. Para introduzir texto é possível utilizar um teclado de braille com feedback tátil. Para além da interação com o ecrã é possível interagir também por comandos de voz.

## 2.3 Sumário

Das soluções apresentadas anteriormente destacam-se várias características que são desejáveis encontrar atualmente num sistema de acessibilidade:

- 1) Consistência, o sistema deve apresentar interfaces consistentes para que o utilizador consiga sempre situar-se e interagir da melhor forma. Esta característica é encontrada nas soluções que recorrem a adereços ou modificação física de dispositivos - capítulo 2.1.
- 2) Acesso, os utilizadores cegos devem poder aceder a todo o conteúdo dos oferecido pelos seus dispositivos tal como as pessoas normo-visuais. Esta característica encontra-se maioritariamente nas soluções que oferecem uma camada extra de software - capítulo 2.2.
- 3) Aceitação, os utilizadores devem sentir-se confortáveis a usar os seus dispositivos - [27].
- 4) Adaptação, o sistema deve ser personalizável para se adaptar às necessidades dos utilizadores [22].
- 5) Eficiência, o sistema deve permitir uma navegação rápida e facilitar a experiência do utilizador cego.

O sistema *Accessible Templates* pretende ser uma solução que contempla estas cinco características. Oferece uma interface única, consistente e personalizável na qual é possível tirar partido de todas as funcionalidades do dispositivo (Consistência, Acesso e Adaptação). É um sistema que corre em qualquer *smartphone Android* oferecendo uma camada extra de software, não sendo necessário modificá-lo (Aceitação). Pretende ainda trabalhar em sintonia com o *Talkback* e oferecer uma maneira de os utilizadores realizarem tarefas complexas mais facilmente (Eficiência).



## Capítulo 3

### Camadas Acessíveis

Com a remoção dos teclados físicos, os telemóveis perderam também algumas características importantes para a acessibilidade. Estes teclados ofereciam uma interface consistente com vários atalhos e uma forma de navegação semelhante qualquer que fosse a aplicação. As interfaces utilizadas atualmente nos chamados *smartphones* não oferecem esta consistência, ou seja, cada aplicação usa uma interface específica. Das soluções apresentadas no capítulo 2 que procuram resolver este problema algumas apenas oferecem um conjunto de funções básicas<sup>7</sup> como efetuar chamadas, deixando de fora funções mais avançadas como instalar aplicações novas, secção 2.2. Outras modificam em demasia o dispositivo fazendo com que seja menos apelativo de usar [4,5] ou tiram partido de adereços que dificultam a sua mobilidade [1,6,20], secção 2.1.

Assim, é o nosso objetivo oferecer a consistência que encontramos em telemóveis com teclados físicos, perdida com a perda das suas teclas, permitindo ao utilizador manter controlo total sobre todo o dispositivo.

A abordagem tomada neste trabalho consiste na colocação de uma camada de interação entre o utilizador e os conteúdos interativos, permitindo abstrair e tornar consistente a interação-utilizador, independentemente da aplicação a ser utilizada. Esta abordagem inspira-se em trabalhos anteriores de adaptação de conteúdos web [3,11,13], e na experiência passada na gestão do que é apresentado pelo sistema operativo *Android* e dos eventos gerados [14,18,19,24–26]. Em particular, a adaptação permite a ordenação dos elementos da interface presente ao utilizador. Além de ordenar o conteúdo é também possível personalizar a interface do serviço/aplicação adicionando uma variedade de atalhos para aplicações, contactos e tarefas (macros). Como só é adaptada a maneira como o conteúdo é navegado é possível usá-lo em qualquer parte do sistema.

#### 3.1 Requisitos

Com base nos desafios descritos, e nas limitações e lições do trabalho relacionado foi elaborada a seguinte lista de requisitos a satisfazer pelo sistema, estes requisitos dividem-se em requisitos funcionais e não funcionais.

---

<sup>7</sup> <http://www.ownfone.com/braille>, última visita 30/09/2016

### **3.1.1 Funcionais**

- 1) Consistência, a interface apresentada ao utilizador deve ser mantida consistente independentemente da sua navegação no dispositivo.
- 2) Personalização, a interface deverá oferecer a possibilidade de ser personalizada pelo utilizador.
- 3) Controlo, o sistema apenas deve modificar a maneira como o utilizador navega pelo conteúdo da interface não restringindo as funções oferecidas pelo mesmo.
- 4) Integração, a interface deverá ser apresentada de forma a que se seja necessária uma pessoa normovisual prestar auxílio esta o consiga fazer.

### **3.1.2 Não funcionais**

- 1) Performance, melhorar a forma como o conteúdo é navegado bem como ajudar na realização de tarefas complexas.
- 2) Acessibilidade, não ser efetuadas mudanças ao hardware do dispositivo.
- 3) Abrangente, funcionar em todo o sistema independentemente das aplicações do utilizador

## **3.2 Conceção**

O sistema elaborado, tendo em conta os requisitos, apresenta uma interface personalizável e, consistente, dividida em duas áreas de maior relevo, a área fixa e a área de navegação. O sistema oferece um conjunto de ferramentas entre as quais a possibilidade de ordenar o conteúdo de várias maneiras diferentes, por exemplo por número de utilizações, atalhos personalizáveis com diferentes fins e pesquisa de elementos por voz ou texto. Não é necessário efetuar nenhuma mudança no dispositivo uma vez que funciona como uma aplicação normal. O sistema funciona em conjunto com o leitor de ecrã como uma camada adicional entre o sistema operativo e o utilizador.

### **3.2.1 Ordenação**

Com o uso desta camada é possível abstrair dos elementos que se encontram na interface original. Isto possibilita apresentar ao utilizador diferentes ordenações da mesma interface conforme o que este escolha. Estas ordenações poderão também ser aplicadas a outro tipo de listas de conteúdo além da lista da interface.

Por exemplo, podemos ver na Figura 10 como seria a navegação com a ordenação da interface original e na Figura 11 a navegação com ordenação alfabética.

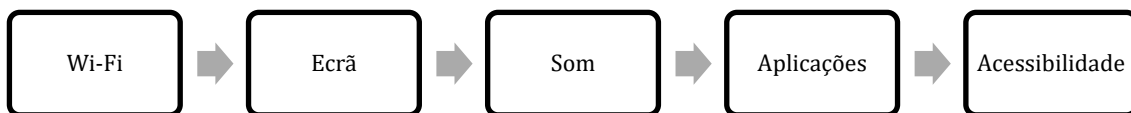


Figura 10 Ordenação original

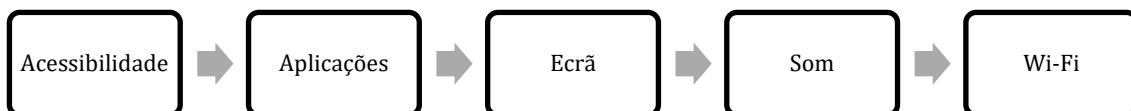


Figura 11 Ordem alfabética

### 3.2.2 Interface

A interface de utilizador da aplicação está dividida em duas partes, a área fixa e a área de navegação. Na metade superior do ecrã encontramos a área fixa onde são dispostas as opções que estão sempre acessíveis para o utilizador. Na metade inferior encontramos a área de navegação onde é possível navegar da mesma forma que no leitor de ecrã pelo conteúdo da interface da aplicação que se encontra por baixo.

#### Área Fixa

A área fixa contém as opções mais importantes para o utilizador, que este desejará ter à distância de um clique. Esta é personalizável para que possa satisfazer as necessidades de diferentes utilizadores. Para navegar nesta área o utilizador utiliza os modos oferecidos pelo leitor de ecrã. Após a seleção, que é feita através de um duplo toque, o conteúdo navegável na *área de navegação* é alterado conforme a opção selecionada e é iniciado o processo de navegação.

#### Área de Navegação

Nesta área é possível efetuar a navegação no conteúdo da mesma forma que no *Talkback*, com gestos direcionais esquerda/direita. Esta área tem como objetivo evitar erros de foco por parte do utilizador, por exemplo quando tentam fazer o gesto direcional para navegar, mas clicam por engano e acaba a ter que começar a navegação de novo.



Figura 12 Esboço da interface, metade superior - Área fixa, metade inferior - Área de navegação

Na Figura 12 podemos ver um exemplo da interface da aplicação onde na metade superior se encontra a área fixa com um conjunto de opções e na metade inferior podemos encontrar a área de navegação. Neste caso o botão de ordem permitiria a permutação das diferentes ordenações, o botão de aplicações e de contactos apresentariam uma lista do respetivo conteúdo.

### 3.2.3 Navegação

Atualmente, o leitor de ecrã *Talkback* oferece dois modos de navegação. Um modo através de gestos direcionais esquerda/direita e outro denominado *explore by touch* que ao passar o dedo lentamente pelo ecrã lê a descrição do elemento por onde o dedo passa. O modo que usa gestos direcionais permite fazer uma navegação linear da interface, enquanto que o modo *explore by touch* permite chegar ao elemento desejado imediatamente desde que o utilizador saiba a posição do mesmo. Para interfaces novas, para o utilizador este segundo modo torna-se difícil de usar, uma vez que este desconhece a posição dos elementos, ficando o utilizador limitado ao primeiro. Se o utilizador optar por usar o primeiro, ao navegar através dos gestos direcionais, existe a possibilidade de um destes gestos ser reconhecido como clique e o foco do leitor passar para o item clicado. Isto leva a que o utilizador perca a noção de onde se encontra e tenha que voltar a iniciar a navegação [26].

Na abordagem sugerimos uma navegação através de gestos direcionais esquerda e direita a partir de uma área específica. Para iniciar a navegação o utilizador terá que selecionar esta área. Isto permite evitar o problema descrito acima, uma vez que, mesmo que o utilizador clique por engano noutro botão poderá voltar ao último elemento navegado. Quando é realizado um gesto direcional, o leitor de ecrã muda o seu foco para o elemento seguinte ou anterior conforme a direção do gesto. Para permitir as diferentes ordenações oferecidas pelo sistema a navegação não poderá ser efetuada diretamente na interface da aplicação.

Com eficiência em mente, sugerimos outra forma de navegação que permite apenas com um gesto realizado de forma contínua e circular, percorrer a lista de elementos. Esta forma tira também partido do redirecionamento de foco do *Talkback*, mas não dos seus gestos de navegação. Assim o utilizador pode realizar o gesto circular em qualquer parte da área de navegação para percorrer uma lista mais rapidamente.

### 3.2.4 Personalização

Para tornar o sistema mais personalizado e funcional de acordo com as necessidades de cada utilizador são oferecidas diversas ferramentas para a sua personalização. Entre



as quais encontram-se a criação, edição e partilha de camadas, criação de atalhos para contactos, tarefas e aplicações.

### **Criação e edição de camadas**

É possível criar camadas de duas maneiras, uma maneira fácil que foca em utilizadores menos experientes, e uma maneira mais avançada, rica em opções, para utilizadores mais experientes.

A forma para utilizadores menos experientes apenas permite a escolha das opções que o utilizador pretende ver na sua camada. Estas são ordenadas de acordo com a ordem de escolha e são dispostas em linhas de 3.

Se o utilizador pretender mais controlo sobre a criação da camada poderá utilizar a forma avançada. Assim este poderá não só escolher quais as opções, mas também personalizar o número de itens por linha que pretende.

### **Partilha de camadas**

É ainda oferecida uma forma de partilha de camadas. Esta opção permite que diferentes utilizadores carreguem as camadas que criaram para uma base de dados, e que consequentemente outros os possam descarregar.

### **3.2.5 Opções da área fixa**

Estas opções têm como objetivo personalizar o conteúdo apresentado e navegado na área de navegação bem como alterar definições do sistema.

A opção *reset* de navegação permite ao utilizador retomar a navegação nos elementos da interface da aplicação que se encontra por baixo da camada. Esta opção realiza ainda um refresh na lista de elementos navegados.

Os botões de alteração de ordem permitem alterar a ordem de qualquer lista de conteúdo que esteja a ser navegada exceto listas de opções que permitam alterar o estado do sistema seja personalização da camada ou configurações. Estão disponíveis cinco ordenações diferentes, mais usados, mais recentes, ordem alfabética, favoritos e interativos.

As opções de *Aplicações* e *Contactos* alteram o conteúdo navegado na área de navegação para a lista de aplicações e com a lista de contactos respetivamente. A estas duas listas é possível aplicação as diversas ordenações oferecidas.

A opção *adicionar favoritos* permite ao utilizador guardar itens relevantes para a sua navegação, que posteriormente podem ser ordenados usando o filtro de favoritos.

As opções de pesquisa por texto e voz oferecem mais uma forma de encontrar itens permitindo ao utilizador encontrar um item específico sem este ter que navegar pela lista.

## Capítulo 4

### Implementação

Neste capítulo apresentamos o *Accessible Templates*, um sistema que permite aos utilizadores terem de volta a consistência que tinham nas interfaces físicas. O serviço de acessibilidade que está na sua base, as suas componentes e as suas funcionalidades.

#### 4.1 Serviço de acessibilidade e eventos

Os serviços de acessibilidade são uma ferramenta disponível na *framework Android* e desenhada para melhorar a interação de um utilizador com necessidades especiais com o dispositivo. Estes correm em segundo plano e recebem *callbacks* oriundas do sistema operativo quando eventos de acessibilidade (*AccessibilityEvents*) ocorrem. Estes eventos contêm informação sobre alterações na interface, se o foco do leitor de ecrã mudou, se um botão foi clicado, e ainda recolher o conteúdo que está em qualquer janela de qualquer interface do sistema. Esta recolha é restrita aos itens que se encontram visíveis no ecrã, não sendo possível aceder a todos os elementos de uma interface.

O sistema *Accessible Templates* tem como base um serviço de acessibilidade implementado no sistema operativo *Android*. O acesso privilegiado à informação gerada pelo sistema operativo, é fulcral para o funcionamento do *Accessible Templates*. De todo o tipo de eventos oriundos do sistema operativo o *Accessible Templates* utiliza os que contêm informação sobre conteúdo da janela da interface, informação relativa a interações com o dispositivo (foco, clique, gestos) e informação relativa ao *Talkback*. Especificamente esta informação oferece acesso a árvores de nós que representam a interface visível, informação sobre alterações ou mudanças de janela, comportamentos do leitor de ecrã e elementos onde o utilizador focou ou clicou.

Os eventos são recebidos pelo serviço de acessibilidade e reencaminhados para o processador de eventos através do controlador (Figura 13). Esta componente trata estes eventos e decide a melhor ação a tomar. Entre as ações possíveis encontram-se o preenchimento da lista de conteúdo da interface quando é recebido um evento de mudança de janela ou navegação nestas listas ou quando é recebido um evento de foco nos botões de navegação. Ou a atualização da descrição dos três botões de navegação quando é detetado um gesto direcional no leitor de ecrã.

A Figura 13 mostra o exemplo de interação com o dispositivo onde o utilizador está a navegar pela interface de uma aplicação. Neste caso são gerados ventos de mudança de foco do leitor de ecrã. Após a receção por parte do processador de eventos, o *Model*

atualiza devidamente o item corrente e comanda a *View* a atualizar a sua representação através do controlador.

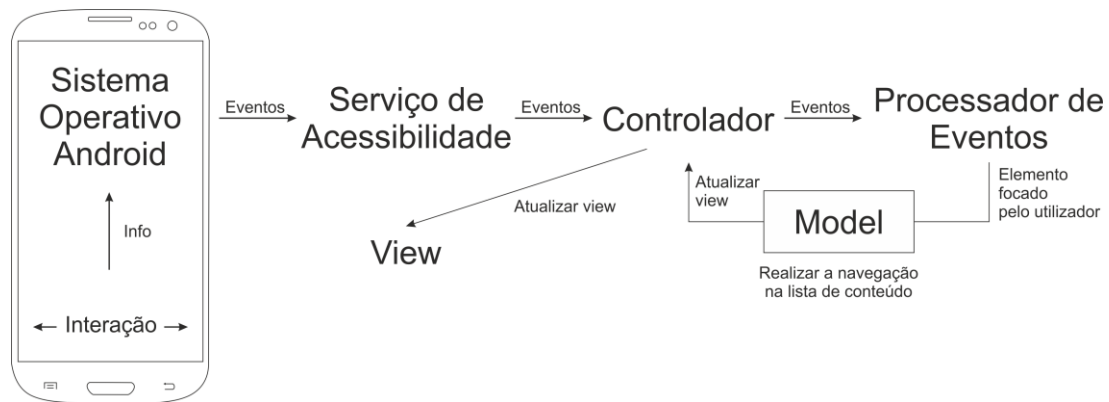


Figura 13 Exemplo de interação

## 4.2 Componentes

Nesta secção são apresentados todos os componentes que fazem parte do sistema (Figura 14). O processador de eventos é dos componentes mais importantes pois recebe os eventos provenientes do sistema operativo e decide a melhor ação a tomar. A componente responsável pela navegação é igualmente importante uma vez que não só permite a navegação pelas listas de conteúdo, como aplicar as diferentes ordenações. A biblioteca fornece um conjunto de operações que são realizadas sobre *Accessible Node Infos*. O gestor de camadas em semelhança ao das macros permite criar, editar e partilhar camadas. A componente de contactos permite gerir todas as tarefas relacionadas com contactos. A *view* é responsável pela apresentação visual e pelo tratamento da interação do utilizador com esta interface. O gestor de macros permite gravar, executar e partilhar tarefas mais complexas. O controlador é a componente que permite a comunicação entre

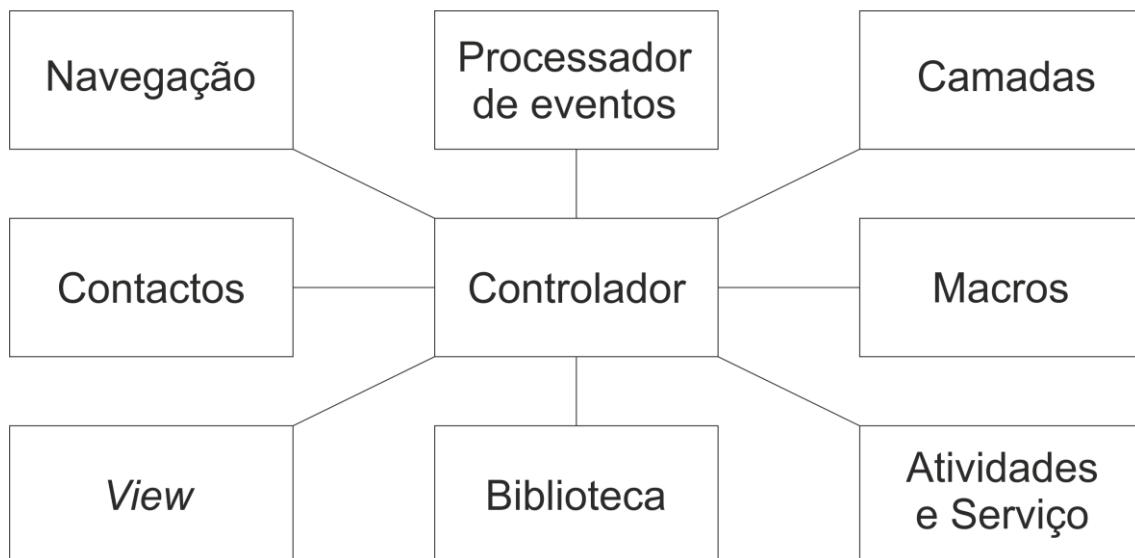


Figura 14 Componentes principais do Accessible Templates

todos os restantes componentes do sistema. As atividades implementadas servem de apoio a diversas funções das camadas do sistema. E finalmente o serviço é a classe que recebe toda a informação do sistema operativo e onde é iniciado o sistema.

### 4.2.1 Processador de Eventos

É por esta componente que passam todos os eventos de acessibilidade gerados pelo sistema operativo. Esta componente tem como principais objetivos a deteção de gestos e navegação nos botões de navegação, a deteção de mudanças de janelas e a deteção do aparecimento teclado.

#### Deteção de gestos e navegação na lista de conteúdo

Como já mencionado no capítulo anterior a área de navegação permite que o utilizador volte ao último elemento navegado a qualquer momento. Para tal é necessário ter controlo total sobre a navegação e o conteúdo navegado. Este controlo é conseguido usando uma lista, que contem os elementos da interface, criada pelo sistema e duas propriedades da API de acessibilidade do sistema operativo *Android*, *AccessibilityTraversalAfter* e *AccessibilityTraversalBefore*, que permitem redirecionar e controlar o foco do *Talkback*.

Foi ainda necessário encontrar uma forma de permitir aos utilizadores navegarem na lista de elementos e não na interface da aplicação. Para simular uma navegação circular na lista de conteúdo foram criados três botões, que se encontram escondidos do utilizador. Para que esta navegação seja possível a aplicação tira partido da navegação por gestos oferecida pelo *Talkback*. As propriedades *AccessibilityTraversalAfter* e *AccessibilityTraversalBefore*, presentes na API de acessibilidade do *Android*, são utilizadas para redirecionar o foco do *Talkback* para que este não saia destes três botões.

Quando um dos botões de navegação é focado pelo *Talkback* é dado início à execução do algoritmo para simular a navegação circular. Após o foco, existem duas possibilidades de execução. Se for o início da navegação, isto é, foi a primeira vez que os botões foram focados desde que o conteúdo da lista foi alterado. A segunda hipótese é se for o seguimento da navegação, ou seja, o foco do *Talkback* veio de um botão de navegação que foi focado imediatamente antes sem a lista de conteúdo ter sido alterada.

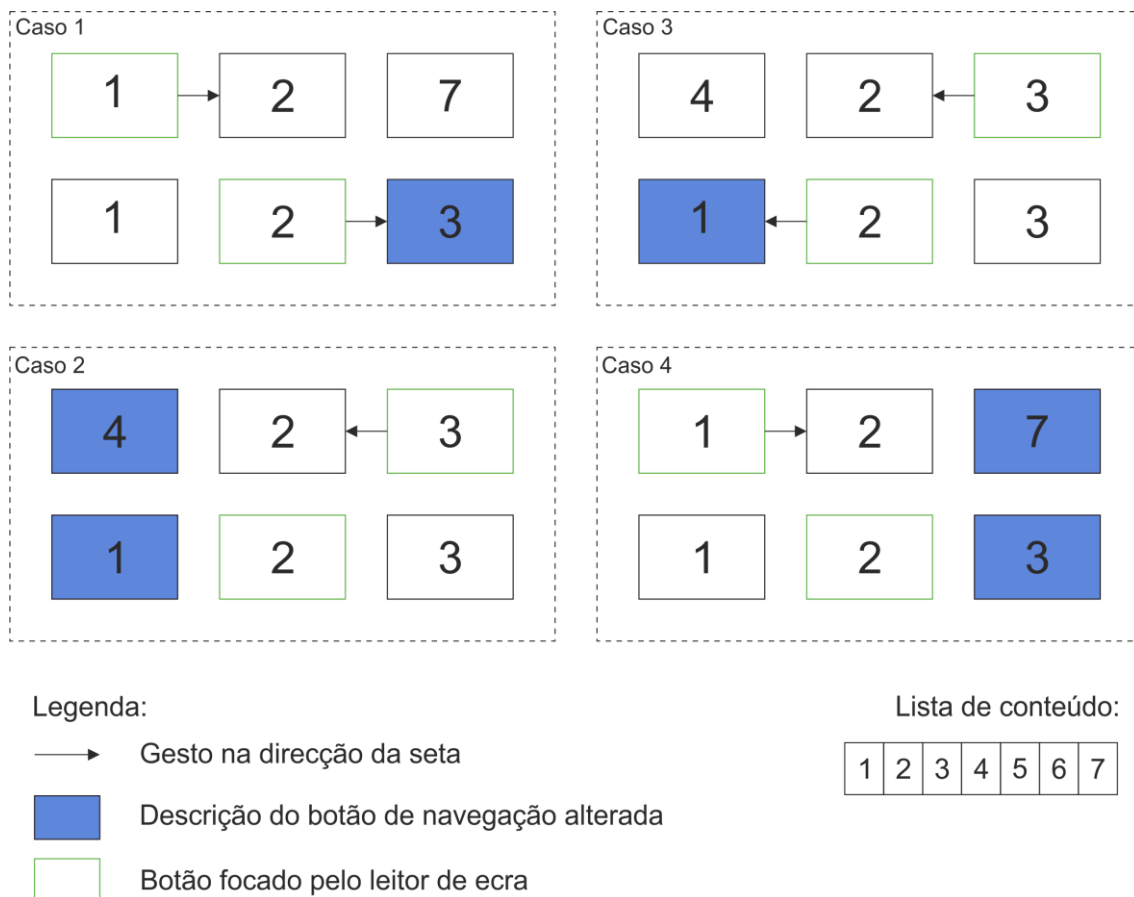


Figura 15 Algoritmo de navegação no conteúdo

Se se verificar a segunda opção o algoritmo avança para o segundo passo, se se verificar a primeira não é efetuada qualquer tarefa.

No segundo passo do algoritmo, é possível determinar o lado para o qual o utilizador efetuou o gesto. Para tal é usado o botão focado imediatamente antes e o botão focado atualmente. Por exemplo se o foco passou do botão 1 para o botão 2 andou da esquerda para a direita, se passou do botão 2 para o botão 1 significa que fez o gesto da direita para a esquerda. Após esta deteção existem três possibilidades para cada direção. É o primeiro gesto ou o gesto continua na mesma direção do anterior ou gesto foi efetuado na direção oposta do anterior. Se se verificarem a primeira ou a segunda é substituída a descrição do botão imediatamente a seguir pelo item seguinte na lista de conteúdo na direção do gesto (caso 1 e 3) da Figura 15. Se se verificar a terceira é substituído o conteúdo do botão seguinte pelo item 3 índices à frente ou atrás conforme a direção do gesto da lista de conteúdo (casos 2 e 4) da Figura 15.

### Deteção de mudança de janelas

Para saber quando é necessário atualizar a lista de conteúdo, o programa está à escuta de um evento de mudança de janela que é do tipo `TYPE_WINDOW_STATE_CHANGED`.

Sempre que este se verifica é iniciado um temporizador com 500 milissegundos que é reiniciado cada vez que se registre um novo evento. Esta abordagem permite conceder tempo ao telefone para carregar todos os elementos da interface. Quando o temporizador chegar ao fim do seu tempo é então efetuado um varrimento da interface corrente e criada uma lista de conteúdo para navegação.

No entanto este evento não cobre todos os casos de possível mudança de janela. Para os restantes casos que na sua maioria ocorrem quando a interface possui um elemento *viewpager*, são usadas duas estruturas de dados implementadas *UITREE* e *UITREEELEMENT* que representam a árvore de nós da interface. Quando é detetado um clique e existem *viewpagers* na interface da aplicação a ser navegada, os eventos usados para a deteção de mudança de interface não são lançados pelo sistema, então são usadas as estruturas de dados referidas que contêm uma cópia da árvore de elementos antes e depois do clique. Estas duas árvores são comparadas e se diferirem deteta-se que houve uma mudança e o conteúdo navegável é atualizado.

### **Deteção do aparecimento do teclado**

Outro caso especial tratado nesta componente, que obriga a uma mudança na interface do serviço ocorre quando é detetado o lançamento do teclado virtual. Esta deteção é feita através da lista de janelas oferecida pela API de acessibilidade do sistema operativo e do evento do tipo *WINDOW\_CONTENT\_CHANGE*. Sempre que este evento é detetado é percorrida a lista de janelas e verificado se a janela do teclado está presente, se sim a interface é rearranjada para que o utilizador possa escrever.

Quando o processador de eventos está em modo teclado, isto é, detetou que o teclado apareceu no ecrã, a cada toque nas teclas do teclado atualiza a lista do conteúdo de navegação. Esta função é particularmente útil quando o utilizador está a escrever em caixas de texto que oferecem sugestões de texto, por exemplo um motor de busca.

### **Outros casos tratados nesta componente**

A qualquer momento é possível minimizar a interface da aplicação e com a possibilidade é oferecida também a possibilidade de retomar o normal funcionamento do serviço. Para tal é lançada uma notificação permanente que assim que é clicada envia um evento *ANNOUNCEMENT* ao processador de eventos para relançar a interface de utilizador.

Para facilitar a perceção espacial dos utilizadores na divisão do ecrã, sempre que o foco passa de uma área para a outro o telefone vibra suavemente.

## 4.2.2 Biblioteca

É na classe Libray que são efetuadas todas as interações com os elementos de acessibilidade capturados pelo serviço. Aqui é possível obter o tipo dos eventos capturados no processador de eventos, obter descrições de nós, procurar um nó específico na interface, efetuar um clique num determinado elemento, efetuar *scroll* e obter a listas de conteúdo da interface.

### Popular a lista de conteúdo

Para popular a lista de conteúdo a árvore de elementos é percorrida recursivamente onde são guardados, por omissão, todos os nós que são visíveis para o utilizar, no momento. O método responsável por esta tarefa pode também, conforme os argumentos recebidos, guardar todos os nós da interface. Para tal é necessário efetuar *scroll* na interface à medida que são guardados os nós até chegar ao fim da interface. A estratégia escolhida de só carregar os nós visíveis não é a melhor opção para a funcionalidade de ordenação visto que esta não estará a ordenar os nós todos. Mas foi a escolhida devido a dois fatores. O primeiro é que o sistema operativo só permite o acesso a nós que estejam visíveis no ecrã e o facto de ser necessário fazer *scroll* programaticamente pode levar à existência de duplicados impossíveis de eliminar na lista de conteúdo. O segundo é que existem aplicações que baseiam o seu funcionamento numa interface que permite fazer *scroll* infinitamente, por exemplo redes sociais. Para permitir maior controlo ao utilizador sobre a forma como este aspeto é tratado no programa é possível configurar esta opção para cada aplicação individualmente.

### Descrição dos nós

Na maioria das interfaces um elemento é representado por uma hierarquia de nós. E a descrição do nó, se existir, pode estar em qualquer um destes. Esta hierarquia é constituída por um nó pai e uma lista de nós filhos. O que é feito na biblioteca é, se o nó pai não possuir descrição, percorrer a lista de filhos ordenadamente até encontrar um que possua descrição. Esta será a descrição utilizada na lista de conteúdo. Por exemplo, como podemos ver na Figura 16 o nó pai, que não possuiu descrição, é o elemento que contém os nós filhos: ícone, palavra “Aplicações” e frase “50 aplicações instaladas”. Na lista de conteúdo iria ficar a palavra “Aplicações” associada a este elemento.

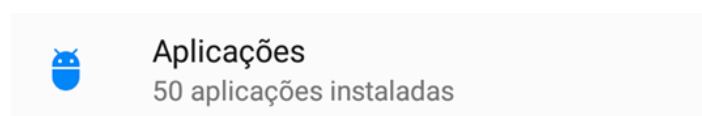


Figura 16 Exemplo de interface da aplicação "definições"



## Pesquisa do nó para o clique

Para efetuar o clique num determinado nó é usada a estrutura criada para representar um elemento da interface chamada *MyNode*. Um dos problemas encontrados nesta pesquisa foi que os nós têm, por vezes, textos e descrições iguais, o que inviabiliza a comparação usando essas características apenas. Para resolver esse problema, implementámos uma comparação que utiliza, juntamente com essas características, os nós anteriores e posteriores aos elementos a comparar. Para efetuar o clique é usada uma função da API de acessibilidade do sistema operativo que dado um determinado *AccessibilityNode* efetua o clique programaticamente.

### 4.2.3 Navegação

Esta componente é responsável pela gestão da navegação no conteúdo efetuada pela área de navegação e pela aplicação dos diversos filtros ao conteúdo.

#### Módulo de navegação

Neste módulo é mantida uma lista com as descrições a serem atribuídas dinamicamente aos botões de navegação. É possível ordenar esta lista de cinco maneiras diferentes, alfabeticamente, elementos mais usados, elementos mais recentemente usados, elementos favoritos e elementos interativos.

Para obter a lista de elementos ordenada alfabeticamente é usada uma função da classe *Collections* (Java) que ordena automaticamente a lista. Para saber quais os elementos mais usados o serviço tira partido de uma base de dados SQLite interna que é atualizada cada vez que um item é clicado. Esta base de dados mantém uma tabela por aplicação utilizando o seu *package* como nome. Em cada entrada é usado a classe, texto e descrição do elemento a guardar como chave e o numero de interações com este determinado elemento como valor. Para o filtro de mais recentes é usada a mesma base de dados com uma entrada onde é possível saber quando foi a última vez que o elemento foi utilizado. Os elementos favoritos são igualmente guardados numa base de dados, mas estes são escolhidos explicitamente pelo utilizador. Finalmente, para a escolha dos elementos interativos é usada uma propriedade dos elementos chamada *clickable*, só os elementos em que se verifique esta propriedade são apresentados.

#### Detetor de cliques

Neste módulo são tratados todos os cliques efetuados nos botões de navegação, é possível tratar cliques simples bem como cliques longos. Este funciona em paralelo com o módulo de navegação e tem como base do seu funcionamento uma propriedade,

*CurrentNavigable*, que é atribuída sempre que o conteúdo é alterado ou é necessário entrar num menu com funções diferentes. Esta propriedade indica que tipo de conteúdo está a ser navegado e o modo como este deve ser tratado após um clique. Por exemplo, se a propriedade for do tipo "*INTERFACE*", quando houver um clique este componente sabe que tem que reencaminhar este clique para o elemento da interface com a mesma descrição do botão de navegação que foi selecionado. Outro exemplo, se a propriedade for do tipo "*TEMPLATE*" esta componente sabe que o utilizador está a navegar pelo menu de camadas que em seguida, conforme o item selecionado, terá que atualizar a lista de conteúdo com o menu seguinte.

### Módulo de deteção gestual

Este módulo oferece uma navegação diferente e independente da utilizada pelo leitor de ecrã *Talkback*. Em vez de o utilizador recorrer a gestos direcionais usa gestos circulares para navegar pelo conteúdo. Com o algoritmo implementado é possível fazer o gesto em qualquer parte da área da navegação e com qualquer raio desde que este seja circular, isto facilita bastante especialmente para pessoas com pouca perceção espacial pelo facto de o utilizador não necessitar de efetuar o movimento em relação a um ponto central.

Para a realização do algoritmo é implementada a interface *OnGenericMotion Listener*, oferecida pelo sistema *Android* que se encontra à escuta de gesto feitos sobre a área de navegação. Este componente possui dois métodos principais um que capta os dados e outro que calcula o movimento a efetuar. O método que capta os dados recebe um evento de movimento do qual são usados dois componentes, a sua ação e as suas coordenadas. As ações podem ser de *ENTER*, significando o início do gesto, *MOVE*, significando a continuação do gesto e de *EXIT*, significando o fim do gesto. Quando é recebido um *ENTER* são inicializadas as variáveis a utilizar nos cálculos. Quando é recebido um *EXIT* estas são limpas.

Quando é recebido um *MOVE* as coordenadas XY do movimento são guardadas numa estrutura de dados até que se atinja um número de pontos correspondente à precisão desejada, nesta implementação serão três pontos. Quando são captados três pontos é efetuada a média das suas coordenadas, que são registadas. Este processo é repetido o número de vezes da precisão, ou seja, três. No final foram processados nove pontos três por cada ponto, passando ao próximo passo.

Antes de iniciar os cálculos do passo é verificado se algum par de pontos é colinear, se sim este conjunto

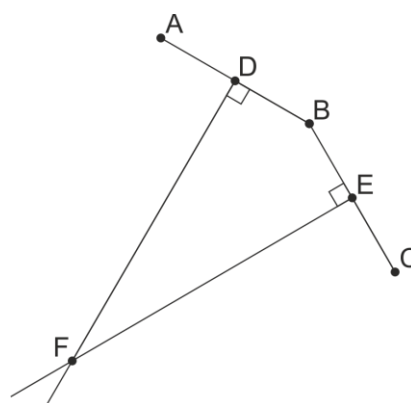


Figura 17 Cálculo da interseção das normais de dois segmentos de reta

é descartado. Para encontrar o ponto F a partir dos três pontos A, B e C é necessário calcular a interseção das normais que passam pelos pontos centrais dos segmentos de reta formados por AB e BC.

Duas retas podem ser definidas através de 3 pontos, AB e BC, a primeira passa por A e B, a segunda por B e C. As equações das retas AB e BC são:

$$y_B = m_{AB}(x_B - x_A) + y_A$$

*Equação 1 Equação da reta AB*

$$y_C = m_{BC}(x_C - x_B) + y_B$$

*Equação 2 Equação da reta BC*

De onde podemos deduzir o declive de ambas as retas representado por  $m_{AB}$  e  $m_{BC}$ .

$$m_{AB} = \frac{y_B - y_A}{x_B - x_A}$$

*Equação 3 Equação do declive da reta AB*

$$m_{BC} = \frac{y_C - y_B}{x_C - x_B}$$

*Equação 4 Equação do declive da reta BC*

O ponto F que simboliza a interseção das duas normais que passam nos pontos centrais D e E. A perpendicular de uma reta com declive  $m$  tem declive  $-1/m$ , logo temos que as equações das retas perpendiculares a AB e BC que passam pelos pontos D e E são:

$$y = -\frac{1}{m_{AB}} \left( x - \frac{x_A + x_B}{2} \right) + \frac{y_A + y_B}{2}$$

*Equação 5 Equação da reta normal da reta AB no ponto D*

$$y = -\frac{1}{m_{BC}} \left( x - \frac{x_B + x_C}{2} \right) + \frac{y_B + y_C}{2}$$

*Equação 6 Equação da reta normal da reta BC no ponto E*

Resolvendo a equação seguinte encontramos a coordenada X do ponto F. Finalmente para encontrar a coordenada Y basta substituir o valor de X e resolver uma das equações acima.

$$x = \frac{m_{AB}m_{BC}(y_A - y_C) + m_{BC}(x_A + x_B) - m_{AB}(x_B + x_C)}{2(m_{BC} - m_{AB})}$$

*Equação 7 Equação da coordenada X do ponto F*

Após o cálculo deste ponto F é comparado a coordenada Y dos pontos A e C para saber se o gesto que originou este conjunto de pontos foi feito numa trajetória ascendente ou descendente, e comparadas as coordenadas X para saber para que lado foi o gesto. Finalmente são comparadas as coordenadas do ponto encontrado F com as coordenadas do primeiro ponto capturado para chegar à primeira estimativa da direção real do gesto.

Como os pontos registados pelo sistema estão demasiados juntos existem demasiados erros no cálculo descrito a cima. Para resolver o problema foi implementado um sistema de votos que tem em consideração diferentes resultados e atribui-lhes diferentes pesos até chegar ao resultado final. Estes diferentes resultados são um histórico dos cálculos, os resultados do cálculo descrito a cima e um histórico da decisão tomada no final. Ao resultado do cálculo é atribuído o maior peso, o peso atribuído a ambos os históricos é dependente do seu tamanho.

#### **4.2.4 Classes auxiliares e estruturas de dados**

##### **Template**

Este componente é responsável pela criação e edição de camadas bem como partilhar e guardar na memória do dispositivo. As camadas encontram-se codificadas em JSON.

##### **Contactos**

Esta componente oferece um conjunto de operações sobre a lista de contactos. Permite ter acesso à lista de contactos que possuem um número de telefone gravados no dispositivo, bem como efetuar chamadas e enviar mensagens de texto para os mesmos. É também possível através de um numero de telefone encontrar o nome do contacto e vice-versa. Por fim este componente é também responsável por manter uma base de dados dos contatos mais utilizados.

## Macros

Com esta componente é possível gravar, executar e partilhar tarefas complexas. As macros são gravadas em formato JSON facilitando a sua partilha.

Para efetuar a gravação é necessário entrar em modo de gravação que minimiza a interface da aplicação bem como todo o processamento de eventos. Após o início são gravados todos os elementos que sejam clicados entre o início e o fim da tarefa.

## JSONUtils

Esta componente oferece um conjunto de funções para transformar diferentes estruturas de dados em JSON preparado para partilhar através da firebase.

## MyNode

Esta estrutura de dados foi criada para representar um elemento da árvore da interface. A estrutura contém o texto e descrição do elemento bem como a sua classe e pacote. Para comparação entre MyNodes a estrutura possui também nó seguinte e o nó anterior na árvore.

A classe oferece métodos de conversão de *AccessibilityNodeInfo* (estrutura usada pelo sistema) e de JSON para MyNode. É possível ainda exportar o conteúdo da estrutura para JSON.

## UITree e UITreeElement

Esta estrutura tem como base um *HashMap* e tem como objetivo guardar uma árvore de elementos de interface. Para a chave é usada a descrição do elemento e valor o *UITreeElement*. Um *UITreeElement* possui uma descrição, um pai e uma lista de filhos do seu tipo. Estes elementos servem de base para o método de comparação. Neste método são corridos todos os nós da raiz até as folhas, se for encontrado algum nó diferente significa que a árvore de elementos da interface mudou.

## 4.2.5 View

### Métodos de desenho da UI

Este componente oferece quatro métodos para quatro desenhos diferentes de interface, interface normal, interface com teclado, interface para introdução de texto e interface para gravação de macros. Para este desenho é necessário um objeto JSON que contém a informação para o desenho dos botões da área fixa, posição, tamanho e função.

## Parâmetros do Windows manager

Como se trata de uma interface que necessita de estar sempre visível, é necessário usar um *WindowManager* oferecido pelo sistema operativo *Android* configurado com uma flag *TYPE\_ACCESSIBILITY\_OVERLAY*. Esta flag permite que a interface do sistema esteja sempre por cima de todas as janelas do dispositivo. O gestor recebe um layout desenhado em XML que contem apenas os três botões de navegação. O uso do ficheiro é necessário porque o uso do *AccessibilityTraversalAfter* e *AccessibilityTraversalBefore* necessita que os identificadores dos botões sejam conhecidos em tempo de compilação. O *Talkback* deixa de funcionar quando estes ids são atribuídos em tempo de execução.

A classe implementa também um *listener* de cliques que recebe todos os cliques efetuados nos botões da área fixa. É possível alterar o comportamento desta função quando o programa está em modo de edição de camada. Neste caso quando o utilizador clica em algum botão em vez da função normal depara-se com um menu de possíveis alteração para o botão.

Finalmente esta componente possui também um gestor que trata da função de minimizar a interface através do lançamento de uma notificação permanente que permite quando é clicada arrancar novamente a interface.

### 4.2.6 Atividades

Existem cinco atividades implementadas no projeto *Laucher, Tutorial, Ativar o Serviço, Voice Search e BackListener*. A atividade que é lançada quando a aplicação arranca é o *Launcher*, o seu objetivo é verificar se o tutorial já foi efetuado ou não. Em caso afirmativo, arranca o ativar serviço em caso negativo arranca o tutorial. Como o nome indica, a atividade *Tutorial* mostra um tutorial das principais funcionalidades e de como usar a aplicação. Enquanto que a atividade *Ativar o Serviço* verifica se as permissões necessárias foram concedidas, se isto não se verificar pede ao utilizador para as conceder. Já com a aplicação a correr o *Voice Search* oferece uma fachada para aceder pesquisa de voz da aplicação. Por fim o *BackListener* está à escuta dos toques no botão de voltar atrás para efetuar esta operação na camada e não no sistema.

### 4.2.7 Serviço

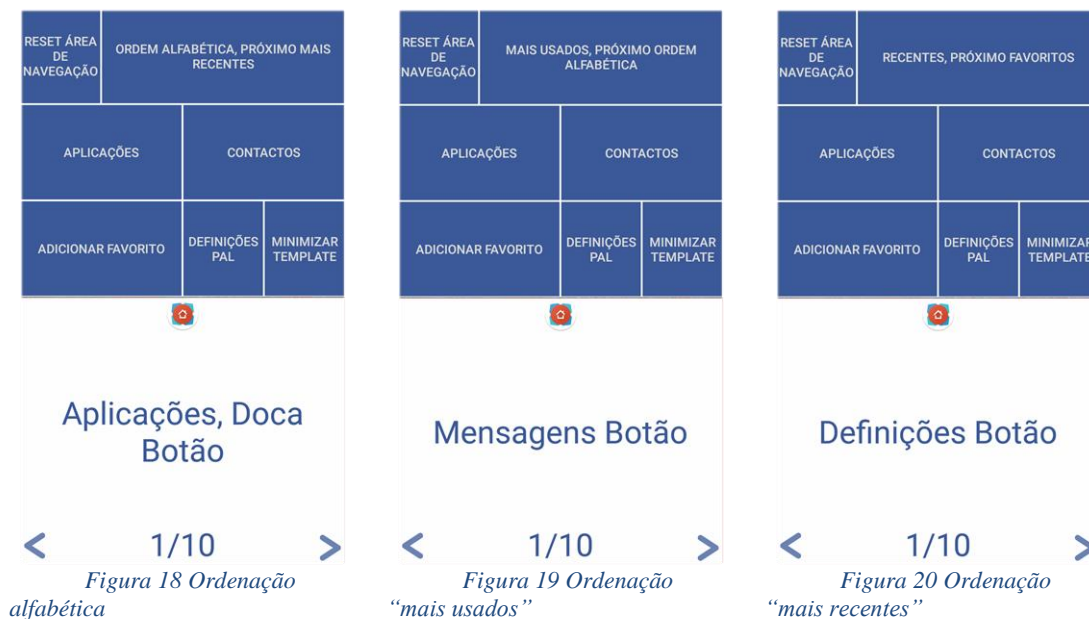
Ao arrancar, esta componente verifica novamente se existem todas as permissões necessárias e em caso positivo mostra a interface de utilizador, em caso negativo fornece ao utilizador uma mensagem explicando que é necessário reiniciar e conceder as permissões.

## 4.3 Funcionalidades

### Ordenação das listas

Esta funcionalidade permite ordenar qualquer interface ou lista de contatos e aplicações conforme a opção escolhida. Para a ativar basta seleccionar no botão que se encontra na interface para o efeito. É possível ter dois tipos de botões para esta funcionalidade, o primeiro é o que podemos ver nas capturas de ecrã e permite ter acesso a todas as ordenações num só botão. Para alternar entre ordenações basta clicar no mesmo. O segundo tipo é um botão individual para cada tipo de ordenação que permite ativar e desativar a ordenação.

Nas capturas de ecrã apresentadas a baixo podemos ver a mesma lista de conteúdo ordenada de três maneiras diferentes, neste caso o utilizador encontra-se na página inicial do dispositivo. Na Figura 18 podemos ver que a lista se encontra por ordem alfabética, na Figura 19 a lista encontra-se com o elemento mais usado primeiro e finalmente na Figura 20 encontra-se com o ultimo elemento a ser usado.



Quando ativado o filtro de itens interativos só são apresentados os itens com os quais o utilizador pode interagir, por exemplo botões ou caixas de texto (Figura 21). A única opção de ordenação que pode ser afetada direta e ativamente pelo o utilizador é a ordenação por favoritos que permite ao utilizador adicionar os elementos que quiser à lista de favoritos (Figura 22).

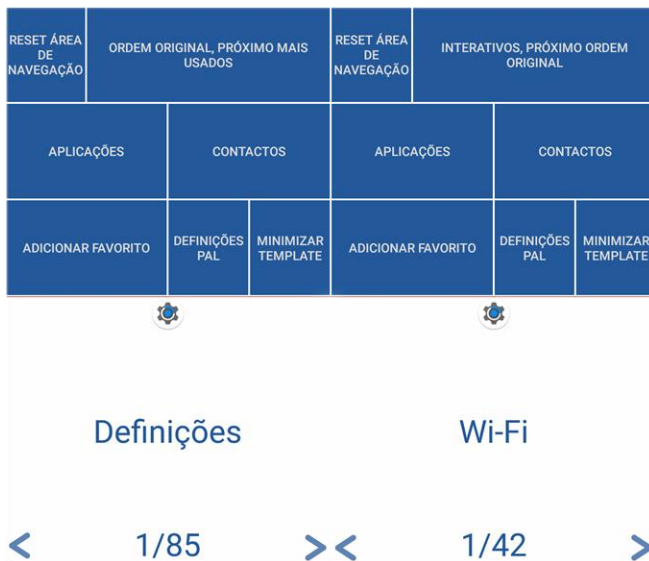


Figura 21 Ordenação “interativos”



Figura 22 Ordenação “favoritos”

### Criação de atalhos

Esta funcionalidade permite adicionar diversos tipos de atalhos à área fixa. Estes atalhos podem ter como fim abrir aplicações, enviar mensagens de texto ou telefonar a contactos ou executar tarefas. Para utilizar esta funcionalidade o utilizador necessita de aceder às definições da aplicação e seleccionar a opção adicionar atalho. Seguidamente basta seleccionar o local na área fixa onde pretende colocar o botão que acionará o atalho.

### Edição de camadas

Esta funcionalidade oferece um maior grau de personalização ao utilizador, uma vez que permite editar a posição e a função dos botões bem como elimina-los. Para iniciar a edição basta seleccionar a opção de editar no menu das definições. Após o início, para editar um botão basta seleccioná-lo e em seguida será apresentada uma lista de opções de edição. Nesta lista podemos encontrar as opções substituir, adicionar um botão a seguir ou antes e eliminar. A opção substituir permite como o nome indica substituir a função do botão por outra à escolha do utilizador. A opção de adicionar permite adicionar um botão com uma opção nova no lugar imediatamente antes ou imediatamente depois do botão seleccionado. Por fim opção de eliminar permite eliminar o botão da camada. Para terminar a edição basta clicar no botão de definições. Após terminar é possível ativar, guardar ou descartar a camada editada.



## **Criação de camadas**

No modo fácil da criação o utilizador apenas pode escolher o número de opções e a sua ordem. Para iniciar a criação neste modo é necessário ir ao menu de definições da aplicação e dentro deste menu selecionar a opção “Criar camada”, em seguida selecionar “modo fácil”. Após a seleção é apresentada a lista de opções que poderão ser colocadas na área fixa que o utilizador poderá selecionar pela ordem que desejar. Para terminar a criação basta selecionar a opção “terminar” existente na lista. No final é possível descartar, guardar, partilhar e ativar a camada acabada de criar. As camadas criadas neste modo possuem sempre três opções por linha.

O modo avançado oferece ao utilizador mais controlo sobre a criação da camada. Neste modo em adição à escolha da ordem e do número de funções é possível escolher o número de botões por linha. Para iniciar a criação neste modo, à semelhança do modo fácil, é necessário ir ao menu de definições da aplicação e dentro deste menu selecionar a opção “Criar camada”, em seguida selecionar “modo avançado”. Após a seleção é pedido ao utilizador que selecione o número de linhas que pretende que pode ser entre um e cinco. Em seguida o utilizador escolhe o número de botões que pretende por linha, este número também pode variar entre um e cinco. No terceiro passo são escolhidas as funcionalidades pretendidas para cada posição. A criação termina quando todos os lugares estão preenchidos. No final, como acontece com o modo fácil, é possível descartar, guardar, partilhar e ativar a camada acabada de criar.

## **Lista de aplicações e lista de contactos**

Ao utilizar estas funcionalidades o utilizador tem acesso à lista de aplicações instaladas no sistema e à lista dos seus contactos. Em ambas as listas é possível aplicar as ordenações faladas a cima. Através da lista de aplicações é possível iniciar qualquer aplicação onde quer que o utilizador se encontre no dispositivo sem necessitar de voltar à página inicial. Através da lista de contactos, à semelhança da lista de contactos acessível em qualquer parte do dispositivo, é possível efetuar chamadas de voz e enviar mensagens de texto.

## **Pesquisa por texto e voz**

Estas funcionalidades permitem ao utilizador procurar um item específico na lista de conteúdo, seja esta formada por elementos da interface da aplicação navegada ou aplicações ou contactos. Desta forma não necessitam de navegar por toda a lista.



## Capítulo 5

### Avaliação com utilizadores

Para avaliar a aplicação, foi realizado um estudo de laboratório onde nove pessoas utilizaram o *Accessible Templates*. Os dados recolhidos através de uma entrevista realizada após o teste foram analisados qualitativamente para analisar os benefícios e falhas da aplicação.

#### 5.1.1 Participantes

Para a realização do estudo foram recrutadas nove pessoas que possuíam experiência no uso de um smartphone com recurso a um leitor de ecrã. Estes utilizadores são capazes de realizar tarefas básicas num smartphone, como fazer ou receber chamadas e enviar ou receber mensagens.

Os participantes tinham idades compreendidas entre 27 e 58 anos ( $M=43.3$ ,  $SD=10$ ). A sua experiência no uso e realização de tarefas com smartphone era variada, como pode ser observado na Tabela 1. Os dois utilizadores mais experientes e que são capazes de realizar todas as tarefas da Tabela 1 são instrutores de acessibilidade que dão aulas a outras pessoas cegas ou com algum tipo de deficiência visual sobre como usar leitores de ecrã. Os participantes fazem parte de uma instituição social local, todos os participantes são legalmente cegos.

ID	Experiencia (meses)	Tarefas						
		Ouvir musica	Receber emails	Enviar emails	Navegar na internet	Usar aplicações sociais	Instalar aplicações	Uso de aplicações de acessibilidade
P1	30							
P2	72							
P3	48							
P4	0,5							
P5	72							
P6	36							
P7	48							
P8	12							
P9	6							

Tabela 1 Experiência dos participantes

### 5.1.2 Dispositivo

Para efetuar os testes foi usado um smartphone Samsung Galaxy S3 com *Android* Nougat versão 7.1.1, com o serviço a correr em primeiro plano. O serviço serve de proxy entre o conteúdo do ecrã e o leitor de ecrã. Como leitor de ecrã foi usado o *Talkback*.

### 5.1.3 Procedimento

Cada participante recebeu uma breve introdução sobre o objetivo do estudo, explorar a aplicação *Accessible Templates* para melhorar o suporte de acessibilidade em smartphones, e sobre as funcionalidades mais importantes do sistema. Seguidamente foi realizado um breve questionário para avaliar a sua experiência no uso de smartphones. Neste questionário figuravam perguntas sobre a frequência do uso do smartphone e do uso de internet, bem como a habilidade de realização das tarefas apresentadas na Tabela 1.

Após o questionário, foi feita uma breve descrição do sistema e das suas funcionalidades. Foi explicada a existência das duas áreas e o propósito de cada uma. No final da explicação os utilizadores puderam explorar as opções da área física para se ambientarem à interface.

Posteriormente, os participantes foram guiados na realização de um conjunto de oito tarefas, apresentadas na

Tabela 2. Antes de realizarem cada tarefa receberam uma breve introdução sobre a tarefa e sobre o ecrã onde se encontravam. Foi permitido aos participantes fazer perguntas livremente e receber ajuda sempre que necessário. Cada participante realizou as tarefas descritas na

Tabela 2 pela mesma ordem. A realização de todas as tarefas demorou cerca de 30 minutos.

Finalmente, após a realização das tarefas, foi conduzida uma entrevista semiestruturada com duração entre 10 e 15 minutos. Os participantes foram inquiridos sobre a sua opinião em relação à interface consistente do *Accessible Templates* bem como as suas opções de personalização. Fizeram ainda parte da entrevista questões sobre os desafios que os utilizadores enfrentam hoje em dia, bem como desafios que enfrentaram quando começaram a utilizar dispositivos móveis.

<b>Característica</b>	<b>Tarefa</b>
Interface Consistente	Explorar todas as opções disponíveis na interface fixa na parte superior do ecrã.

Área de Navegação	Focar a área de navegação e navegar pelo conteúdo para encontrar a aplicação <i>Definições</i> .
Ordem Alfabética	Usar o filtro de ordem Alfabética para encontrar a opção <i>Acessibilidade</i> .
Aplicações e Ordem de frequentes	Usar o atalho para a lista de aplicações e ordenar por frequência para entrar na aplicação mais usada.
Contatos	Usar o atalho para a lista de contatos e efetuar uma chamada para o contato mais usado
Adicionar um item aos favoritos	Marcar a opção <i>Som</i> nas definições como favorito. E verificar com o filtro de favoritos que a opção se encontra em primeiro lugar na lista de conteúdo.
Atalho Novo	Usar as definições do serviço para adicionar um atalho para um contacto. E em seguida efetuar uma chamada para o mesmo usando o atalho.
Atalho para uma tarefa	Usar as definições do serviço para adicionar um atalho para uma tarefa. E em seguida efetuar a tarefa usando o atalho.

*Tabela 2 Tarefas realizadas*

#### **5.1.4 Análise de dados**

As entrevistas foram gravadas em formato áudio e posteriormente transcritas para a análise. Inicialmente, duas entrevistas foram codificadas por dois investigadores para gerar indutivamente o livro de códigos, incluído em anexo. Os investigadores usaram posteriormente o mesmo livro para codificar as restantes entrevistas independentemente e chegaram a um nível de concordância de *Cohen* de  $k=0.71$ .

#### **5.1.5 Resultados**

Depois de analisados os dados emergiram cinco temas distintos pelos quais esta secção se encontra dividida. Em “Desafios no uso de um smartphone” são discutidas as visões/opiniões dos participantes nos desafios com que se depararam quando começaram a utilizar um *smartphone*. “Personalizar, organizar e simplificar” apresenta uma discussão

sobre a interface do *Accessible Templates*. As principais vantagens apresentadas pelo sistema são apresentadas em “Aprendizagem e Autonomia” os últimos dois temas são “Preocupações e limitações” e “Melhorar o *Accessible Templates*”, onde os participantes expressaram as suas preocupações com o *Accessible Templates* e como o *Accessible Templates* podia ser melhorado. Nesta secção serão referenciados como “peritos” os utilizadores que conseguem realizar seis ou sete das tarefas da Tabela 1.

### **Desafios no uso de um smartphone**

Como podemos ver em trabalhos anteriores, desenvolvidos por Rodrigues e colegas [26], e como foi referido pela maioria dos participantes existem diversas dificuldades na adaptação ao uso de um smartphone.

Quando inquiridos sobre as suas experiências, alguns não conseguiram lembrar-se especificamente a origem dos problemas (aconteceu há mais de três anos), contudo, conseguiram transmitir a grande frustração que sentiram no início do uso do smartphone.

A acessibilidade nos dispositivos móveis melhorou muito nos últimos anos, mas ainda não chegamos ao ponto onde começar a usar e adaptar-se a um dispositivo móvel seja considerado uma tarefa fácil. Um dos participantes menos experientes explicou as suas dificuldades iniciais e como necessitou da ajuda de outra pessoa cada vez que se deparava com algo de novo.

*“O primeiro desafio era não o (dispositivo) atirar contra a parede. (...), pois eu não sabia nada como estou a dizer. Depois tentei procurar sozinha, mas depois a minha filha lá se chateava comigo que eu alterei uma coisa que não sabia que tinha alterado. E, portanto, eu acho que esse método dos botões levava menos raspanetes de certeza.” P4*

Apesar de querer explorar as funcionalidades do dispositivo independentemente, o utilizador P4 sentiu que era perigoso fazê-lo porque podia editar alguma definição ou configuração inadvertidamente.

Do ponto de vista dos utilizadores peritos a maior parte dos problemas que incomodam utilizadores inexperientes vem do facto de estes terem pouca orientação espacial. Com o corrente modo funcionamento dos sistemas operativos móveis, quando abertas as aplicações podem apresentar diferentes estados o que em conjunto com interfaces dinâmicas e complexas leva os utilizadores a sentirem-se muitas vezes perdidos e frustrados.

*“Agora se nós não temos a noção do espaço, a gente anda para ali esquerda, direita, esquerda, direita, esquerda, esquerda, direita. (...) se não tivermos esta orientação, se não conseguirmos visualizar na nossa cabeça. É como o caminho daqui para o metro, se não tiver o caminho na cabeça, ando para aí a apalpar e a andar à volta de postes.” P5*

Aprender a utilizar um leitor de ecrã num smartphone pode ser bastante difícil. Os utilizadores menos experientes tendem a ter muitas dificuldades em perceber o que está a acontecer, visto que a interface parece alterar-se completamente sempre que uma opção é seleccionada. Isto leva a que a navegação se torne confusa. Outro fator que torna a navegação complexa é a existência de uma grande variedade de gestos diferentes.

*“O Facebook, o email, essas chatices de deslizar os dedos com dois, com um, para cima e para baixo, (...) e irrita-me. Portanto acho que os vossos botões são uma mais valia.” P4*

Em suma para utilizadores inexperientes todos estes desafios levam a uma falta de autonomia no uso do dispositivo. Mesmo utilizadores experientes não são necessariamente peritos. Em alguns casos mesmo depois de anos de uso, estes utilizadores ainda sentem dificuldades com a navegação até ao ponto onde a descoberta se torna um problema. No caso do utilizador P1, ele ainda só consegue realizar tarefas básicas e tem dificuldades em encontrar as opções que procura.

*“(..), eu às vezes recebo as mensagens com as faturas e queria encaminhar para a minha esposa para ela as pagar. E eu não consigo fazer isso, e eu tenho que estar digamos a decorar os números das referências e depois escrever uma mensagem de novo para a minha esposa.” P1*

### **Personalizar, organizar e simplificar**

As áreas dedicadas com opções fixas e navegação oferecem a estrutura necessária à forma como conteúdo pode ser acedido. Estas opções oferecem uma sensação de segurança pelo facto de estarem sempre presentes no mesmo local. Como explica o utilizador P8 pode ainda ser uma mais valia na rapidez com que os utilizadores utilizam o dispositivo.

*“Sim, facilita. Sabendo que elas (opções fixas) estão cá no topo, não é preciso andar a procura na interface onde tudo está espalhado. Ali vai mais rápido, tem mais acesso rápido.” P8*

Com a inclusão de uma área do ecrã dedicada exclusivamente à navegação e estabelecendo uma regra clara de como o conteúdo é estruturado, é mais fácil para os utilizadores perceberem a interface.

*“Por exemplo nessa área de navegação quando tivesse na aplicação das mensagens, provavelmente apareceria lá, chegaria ao botão de reencaminhar.” P1*

A possibilidade de ordenar o conteúdo sempre que o utilizador desejar foi vista como uma maneira de facilitar a interação e promover a descoberta das interfaces. Para alguns utilizadores isto significa simplesmente que teriam que perder menos tempo a aprender os diferentes gestos de navegação para opções avançadas. Enquanto que para outros esta abordagem permitiria que conseguissem ter diferentes estratégias de navegação dependendo da aplicação.

*“Hoje pode ser por ordem alfabética, mas depois posso mudar os mais usados, dependendo da aplicação do que eu queira fazer.” P3*

Os participantes admitiram que diferentes utilizadores têm diferentes necessidades de interação. Além disso que as necessidades podem facilmente mudar. Para alguns utilizadores a possibilidade de personalizar a área fixa e poder adicionar ou remover diferentes tipos de atalhos foi uma característica importante.

*“Eu penso que, realmente poder adicionar quer os contactos, quer as tarefas, acho que isso facilita, porque, tenho logo ali uma porta aberta para aquilo que pretendemos. (...). Por exemplo eu agora poderia ter com certa frequência contactos com o Tiago, mas depois, por algum motivo não seria tão pertinente. E em vez de ter ali muitos botões fixos eu poderia remover alguns que já não fossem úteis.” P1*



## Aprendizagem e Autonomia

O *Accessible Templates* foi visto como um sistema com uma interface mais simples para interagir com smartphones, e também mais fácil de aprender. Os participantes acreditam que teriam passado por menos problemas de aprendizagem se tivessem tido acesso ao *Accessible Templates* durante a seu período de adaptação.

*“Facilitava muito. É o que eu estava a ver. Eu tinha o (telemóvel) de teclas normal, de um momento para o outro avariou, fiquei... Nem sabia atender o telefone [com o smartphone novo]. (...). Mas de inicio, isto [o sistema Accessible Templates] seria uma grande ajuda.” P8*

Os participantes, nomeadamente o utilizador P3, reportaram que, para pessoas com dificuldades adicionais seja por serem menos peritas em tecnologia, mais idosas ou terem pouca orientação espacial iria ser benéfico o uso desta interface em contraste com o uso de apenas o leitor de ecrã.

*“Iria ajudar imenso. Mais propriamente até para as pessoas mais idosas, digamos assim. Têm muitas dificuldades com o tal smarphone. Porque? Por causa disso mesmo né, as coisas nunca estão no mesmo sitio, o botão varia de um lado para o outro. E aqui eu acredito que irá ajudar imenso.” P3*

Dois participantes peritos mencionaram que o *Accessible Templates* deveria ser incorporado como um modo fácil no leitor de ecrã padrão do sistema operativo *Android*. Isto poderia ser uma ajuda para utilizadores inexperientes ou com mais dificuldades na ambientação à navegação com o leitor de ecrã. O utilizador P5 realçou que poderia não ser uma solução definitiva, mas sim uma solução para facilitar a adaptação ao funcionamento do dispositivo.

*“Vocês, venderem esta app ao Talkback era uma mais valia. Ou seja, o utilizador começaria por aqui. A partir do momento que acha que já está familiarizado com o sistema, pronto vou passar ao original.” P5*

Contudo, o mesmo utilizador também acredita que pessoas com dificuldades adicionais não se vão adaptar completamente à navegação padrão do leitor de ecrã e que

o *Accessible Templates* acabaria por ser usado como solução definitiva e não como plataforma de aprendizagem e habituação ao dispositivo.

*“Acho que esta aplicação não só no início, mas para sempre. Há pessoas que vai ser sempre. Pessoas que chegaram muito tarde, que não aprenderam o braille, que informática não é fácil. Ou seja, que se vêm obrigadas a utilizar um ecrã. Um telefone sem teclas, acho que não no início, mas para sempre será uma mais valia. Ou seja, usarem sempre este.” P5*

Como efeito secundário, a consistência do *Accessible Templates* conferida pela interface acessível sempre presente, oferece aos utilizadores uma sensação de segurança e de independência quando exploraram o sistema. Estes sabem que se cometerem algum erro durante a navegação podem sempre voltar à área de navegação.

*“ (área fixa) Sim pelo menos é uma coisa que, dá-nos um pouco mais de segurança para podermos seleccionar outras coisas.” P2*

Para alguns participantes as opções de ordenação tornam o sistema em mais que uma simples plataforma de aprendizagem. Estes acreditam que a possibilidade de ordenar o conteúdo de diferentes formas e adicionar atalhos seria uma melhoria na performance de utilização do dispositivo.

*“Uma pessoa estando habituado a esse menu faz as coisas muito mais rápidas e com os atalhos que tem facilita imenso.” P3*

### **Preocupações e Limitações**

O sistema *Accessible Templates* impossibilita o uso de diversas ferramentas oferecidas pelo leitor de ecrã nativo do sistema. Ferramentas tais como o *touch to explore* e os gestos avançados são impossíveis de usar uma vez que a interface do *Accessible Templates* se encontra em primeiro plano.

Para um dos instrutores este facto é visto como uma limitação, pois vai contra as normas estabelecidas pelos leitores de ecrã dos smartphones. Como tal ele acredita que para utilizadores como boa orientação espacial e que tenham conhecimento da interface

(peritos) não existe grande benefício em relação ao leitor padrão. Mas não é surpresa nenhuma que todos os participantes gostariam de utilizar o *Accessible Templates* no seu dispositivo menos os instrutores.

*“Para alguém que já sabe usar o touch to explore ele vai usar touch to explore. (...). É que ali há outro problema, é que vocês quando estão a dividir o ecrã, estão a bloquear os gestos, standard de utilização do Android, (...).” P7*

Para utilizadores menos experientes é bastante normal recorrer a ajuda de pessoas normovisuais para realizar tarefas mais difíceis no seu dispositivo. Os participantes descreveram vários cenários onde recorrem a essa ajuda para ultrapassar um problema. Um participante revelou preocupação sobre a possibilidade de utilizadores normovisuais não serem capazes de oferecer esta ajuda se o *Accessible Templates* estivesse a correr no dispositivo. Na altura do estudo o sistema ainda não oferecia feedback visual para pessoas normovisuais, neste momento é mostrado o nome do item focado bem como o ícone da aplicação onde se encontra e ainda o número de itens da lista e o número do item corrente.

*“É assim, dado que estas aplicações que nos facilitam a vida, complicam um bocado a vida às pessoas que veem quando depois vão interagir com o equipamento. As pessoas acham muito estranho por exemplo o Talkback, (...). Mas quando o sistema está ativo, se eu passar o telemóvel para as mãos de uma pessoa normovisual o que é que ela observa.” P1*

### **Melhorar o Accessible Templates**

Quando inquiridos sobre como melhorar o *Accessible Templates* todos os utilizadores sugeriram mais opções de customização. Alguns sugeriram a possibilidade de mudar o tamanho ou posição das áreas dedicadas, outros um botão específico para cada ordem em vez de um botão que circula por todas. Ainda foi mencionado o desejo de poder criar os seus próprios atalhos/tarefas.

*“Adicionar mais tarefas. Que tipo de tarefas? Isso depois depende do que as pessoas mais usam no telefone. Ou senão dar a hipótese da própria pessoa poder criar aquilo que mais lhe dá jeito.” P6*

Muitos participantes mencionaram que o Accessible Templates deveria ser incorporado no leitor de ecrã. Para um utilizador perito isto significava que o Accessible Templates não só suportaria os gestos standard, mas também permitiria a criação de atalhos usando gestos.

*“Mas por exemplo quero abrir o whatsapp, faço o W e ele abre” P7*

Os dois instrutores expressaram como o uso de guias ou interfaces físicas pode ser uma grande ajuda na adaptação à navegação num dispositivo móvel. Um pensa que seria bom ter de volta as interfaces físicas antigas e sugere uma interface física com um teclado com setas direcionais permitindo aos utilizadores navegar com o uso das mesmas. O outro instrutor sugeriu o uso de uma linha física em conjunto com *Accessible Templates* para reforçar e facilitar a distinção entre a área física e a área de navegação.

*“Este ecrã adicionalmente ainda se poderia por tipo uma linha física para dividir. (...). Uma linha quase impercetível. Quase só percetível ao tato. Para ter noção, aqui estou na área de navegação, aqui estou na área de cima.” P5*

## 5.2 Discussão

O sistema recebeu uma aceitação positiva dos utilizadores menos experientes e que têm mais dificuldade em navegar nos seus dispositivos.

O facto de o sistema mudar a forma como os utilizadores interagem com o dispositivo é visto como uma vantagem pela maioria dos utilizadores, mas não pelos utilizadores mais experientes. Os utilizadores menos experientes reconhecem os benefícios e a ajuda da utilização deste sistema no seu dia-a-dia. Os utilizadores mais experientes, que já estão habituados a utilizar os gestos avançados do leitor de ecrã, não sentiram grande vantagem em mudar a forma de interagir com os seus dispositivos. Um deles acha mesmo que a maneira como os utilizadores menos experientes interagem com o dispositivo não deveria ser alterada, pois se estes não estão confortáveis a utilizar o leitor de ecrã atual também não iriam estar usando o sistema.

Os utilizadores menos experientes concordaram que o sistema seria uma grande ajuda aquando da sua adaptação aos dispositivos móveis. Outro dos utilizadores experientes achou ainda que para além de ser uma ajuda na aprendizagem e adaptação, poderia ser uma maneira definitiva para interagir com o dispositivo para pessoas com pouca noção espacial ou pessoas mais idosas.

Todos concordaram que a possibilidade de ordenar a interface conforme a necessidade de cada utilizador melhoraria a navegação em relação à forma existente.



## Capítulo 6

### Conclusão

Os dispositivos móveis estão totalmente enraizados nas nossas vidas, como tal são desenvolvidos a pensar numa grande variedade de tipos de utilizadores. O desenvolvimento já tem em conta as diferentes necessidades de acessibilidade, mas ainda não corresponde totalmente à necessidade dos utilizadores. Por isso é necessário criar condições para que todos os utilizadores possam usufruir totalmente, da melhor maneira e da forma mais autónoma possível.

Neste projeto apresentamos um sistema acessível que tem como objetivo facilitar o uso de smartphones por parte utilizadores cegos. Para tal é oferecida uma camada intermédia entre o sistema operativo e o utilizador que adapta a forma como o utilizador interage com conteúdo do dispositivo móvel.

A camada apresenta duas áreas distintas que dividem o ecrã do dispositivo em duas partes iguais. A área fixa contém as opções sempre acessíveis ao utilizador e a área de navegação permite ao utilizador navegar pelo conteúdo da aplicação que se encontra por baixo. Durante a avaliação do sistema foi efetuado um estudo com nove pessoas para validar os seus benefícios e encontrar limitações.

#### 6.1 Benefícios

Ao efetuar o estudo, os utilizadores menos experientes reconheceram que utilizar o sistema seria uma mais valia para a sua interação diária com dispositivos móveis. Usando este sistema em conjunto com o leitor de ecrã existem diversos benefícios para o utilizador, tais como o facto de ter um conjunto de opções sempre à distância de um clique ou poder navegar no conteúdo da interface de uma aplicação de diversas formas. É possível ainda personalizar as opções da área fixa.

##### 6.1.1 Adaptação

Diversos utilizadores referiram durante o estudo que a utilização do *Accessible Templates* teria facilitado a sua adaptação ao funcionamento do dispositivo móvel. Um dos instrutores reconheceu ainda que o sistema seria bastante útil para utilizadores inexperientes que estivessem a começar e que seria uma mais valia na sua adaptação. Mas que alguns poderiam não conseguir evoluir até ao uso exclusivo do leitor de ecrã.

### **6.1.2 Interface**

As opções da área fixa, que podem variar desde listas de aplicações e contactos acessíveis em qualquer lugar até atalhos para tarefas complexas, oferecem uma maior segurança ao utilizador que está a utilizar o dispositivo visto estarem presentes na área fixa.

A área de navegação oferece uma segurança adicional, uma vez que mantém o item corrente caso o utilizador clique por engano noutra local do ecrã. A partir desta área é possível navegar de duas maneiras distintas. A maneira habitual do *Talkback*, utilizando gestos direcionais, que tira partido da habituação que os utilizadores já possuem com este método. Enquanto que a maneira com gestos circulares requer mais prática, mas permite uma navegação mais rápida pelo conteúdo.

### **6.1.3 Personalização**

A área fixa é personalizável para poder satisfazer as necessidades de diversos utilizadores. Outra opção de personalização oferecida pelo sistema é a opção de ordenar a lista de conteúdo navegável de diferentes formas. Esta opção permite ao utilizador uma maior liberdade na navegação nas interfaces, uma vez que, à distancia de um clique pode aceder a vários tipos de ordenações incluindo itens mais usados ou mais recentes.

### **6.1.4 Ordenação do conteúdo**

O sistema permite que o conteúdo seja ordenado de diferentes formas, possibilitando a adaptação às necessidades dos diferentes utilizadores. Para alguns utilizadores, o facto de poder ordenar as listas através de um clique num botão significa que perderiam menos tempo a aprender os diferentes gestos para opções avançadas enquanto que para outros esta abordagem permitiria que conseguissem ter diferentes estratégias de navegação dependendo da aplicação.

## **6.2 Limitações**

Durante o estudo foram também encontradas algumas limitações apontadas pelos utilizadores mais experientes. Como o sistema foi desenvolvido para utilizadores inexperientes e com dificuldades em usar o leitor padrão, este não é o mais adequado para utilizadores peritos. Estes utilizadores sentem que perdem alguma autonomia oferecida pelo *explore by touch* e pelos gestos avançados do leitor de ecrã. Uma vez que, ao usar uma interface que tapa por completo a interface da aplicação de baixo os utilizadores não conseguem usar estas ferramentas.



### **6.2.1 Explore by touch**

Esta ferramenta permite ao utilizador explorar a interface que se encontra em primeiro plano no dispositivo. Uma vez que o Accessible Templates está em primeiro plano quando é executado, ao utilizar esta ferramenta o utilizador está efetivamente a explorar a interface do sistema e não a da aplicação onde se encontra.

### **6.2.2 Gestos avançados**

Estes gestos em grande parte permitem agilizar a navegação na interface da aplicação, por exemplo ir para o primeiro ou último item de uma lista. Mais uma vez como o Accessible Templates se encontra em primeiro plano quando é executado, ao utilizar esta ferramenta o utilizador está efetivamente a realizar as ações na interface do sistema e não na aplicação onde se encontra.

### **6.2.3 Limitações técnicas**

Existem dois tipos de interfaces de aplicações que tornam a recolha do conteúdo problemática. Estes tipos são interfaces com listas e interfaces com separadores.

Quando as aplicações possuem interfaces que não se encontram totalmente no ecrã, ou seja, possuem listas ou separadores o sistema não tem acesso a toda a informação relativa à interface. O sistema possui duas estratégias para ultrapassar este caso, percorrer a interface toda quando o utilizador entra na interface ou ir percorrendo à medida que o utilizador navega. Ambas as estratégias possuem a mesma limitação, a possibilidade de apresentar itens repetidos. O sistema possui um mecanismo para limitar esta ocorrência, mas com interfaces dinâmicas a sua eficácia não é total. Isto ocorre com maior frequência em listas.

## **6.3 Trabalho futuro**

Tendo em conta as limitações apresentadas na secção anterior são sugeridas algumas propostas de trabalho futuro como forma de aperfeiçoar o sistema desenvolvido. Um aspeto importante para utilizadores peritos, que foi referido durante o estudo, é melhorar a integração com o leitor de ecrã. Outros aspetos que complementaríamos o sistema seriam a inclusão de novas formas de efetuar a navegação pelo conteúdo e a integração de comandos de voz.

### **6.3.1 Integração com o leitor de ecrã**

Uma maior integração com o leitor de ecrã significaria que os utilizadores poderiam usufruir dos gestos avançados que já conhecem. Uma vez que estes afetariam a lista de conteúdo da aplicação em vez de afetarem a interface do sistema. Desta maneira os utilizadores podiam tirar total partido de todos as funcionalidades dos dois sistemas.

### **6.3.2 Explorar formas novas de navegar**

Atualmente o sistema oferece apenas dois modos de efetuar a navegação pelo conteúdo. Efetuando gestos direcionais ou efetuando um gesto circular na área de navegação. Poderiam ser exploradas outras formas de navegação como por exemplo em vez de uma navegação linear navegar uma navegação baseada no conteúdo. Por exemplo navegar de paragrafo em paragrafo em numa lista ordenada alfabeticamente navegar pelas primeiras começadas por cada letra do alfabeto.

### **6.3.3 Comandos por voz**

Outra possibilidade que está indiretamente ligada com a navegação seria a integração de comandos por voz com a pesquisa por voz oferecida pelo sistema, à semelhança do que é feito pelo *Just Speak* [32]. Estes comandos permitiriam a realização de diferentes tarefas como por exemplo efetuar chamadas, abrir aplicações, etc. A inclusão desta funcionalidade permitiria o uso de um grande numero de funcionalidade do dispositivo única e exclusivamente através da voz.

## Capítulo 7

### Bibliografía

1. Sang Sup An, Jae Wook Jeon, Seongil Lee, Hyuckyeol Choi, and Hoo-Gon Choi. 2004. A pair of wireless braille-based chording gloves. *Computers Helping People with Special Needs*: 490–497.
2. Shiri Azenkot, Jacob O. Wobbrock, Sanjana Prasain, and Richard E. Ladner. 2012. *Input finger detection for nonvisual touch screen text entry in Perkinput*. Canadian Information Processing Society. Retrieved September 19, 2017 from <http://dl.acm.org/citation.cfm?id=2305297>
3. Yevgen Borodin, Jalal Mahmud, I. V. Ramakrishnan, and Amanda Stent. 2007. The HearSay non-visual web browser. In *Proceedings of the 2007 international cross-disciplinary conference on Web accessibility (W4A) - W4A '07*, 128–129. <https://doi.org/10.1145/1243441.1243444>
4. Maria Claudia Buzzi, Marina Buzzi, Barbara Leporini, and Maria Teresa Paratore. 2013. Vibro-tactile enrichment improves blind user interaction with mobile touchscreens. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 8117 LNCS, PART 1: 641–648. [https://doi.org/10.1007/978-3-642-40483-2\\_45](https://doi.org/10.1007/978-3-642-40483-2_45)
5. Maria Claudia Marina Buzzi, Francesco Donini Buzzi, Barbara Leporini, and Maria Teresa Paratore. 2013. Haptic Reference Cues to Support the Exploration of Touchscreen Mobile Devices by Blind Users. *Proceedings of the Biannual Conference of the Italian Chapter of SIGCHI*: 28:1–28:8. <https://doi.org/10.1145/2499149.2499156>
6. Myung Chul Cho, Kwang Hyun Park, Soon Hyuk Hong, Jae Wook Jeon, Sung Il Lee, Hyuckyeol Choi, and Hoo Gon Choi. 2002. A pair of Braille-based chord gloves. *Proceedings - International Symposium on Wearable Computers, ISWC 2002-Janua*: 154–155. <https://doi.org/10.1109/ISWC.2002.1167238>
7. Giuseppe Ghiani, Barbara Leporini, and Fabio Paternò. 2008. Vibrotactile feedback as an orientation aid for blind users of mobile guides. *MobileHCI '08: Proceedings of the 10th international conference on Human computer interaction with mobile devices and services*: 431–434. <https://doi.org/10.1145/1409240.1409306>
8. Tiago Guerreiro, Paulo Lagoá, Hugo Nicolau, Pedro Santana, and Joaquim Jorge. 2008. Mobile text-entry models for people with disabilities. *Proceedings of the 15th European conference on Cognitive ergonomics the ergonomics of cool interaction - ECCE '08*: 1. <https://doi.org/10.1145/1473018.1473067>
9. Isha Singh Jassi, S. Ruchika, Susmitha Pulakhandam, Subhayan Mukherjee, T. S. Ashwin, and G. Ram Mohan Reddy. 2014. Ember : A Smartphone Web Browser Interface for the Blind. *Proceedings of the 7th International Symposium on Visual Information Communication and Interaction - VINCI '14*: 106–112. <https://doi.org/10.1145/2636240.2636847>
10. Shaun K. Kane, Jeffrey P. Bigham, and Jacob O. Wobbrock. 2008. Slide rule: making mobile touch screens accessible to blind people using multi-touch interaction techniques. *Proceedings of the 10th international ACM SIGACCESS conference on Computers and accessibility - Assets '08*: 73–80. <https://doi.org/10.1145/1414471.1414487>
11. Andrea Kennel, Louis Perrochon, and Alireza Darvishi. 1996. WAB: World Wide

- Web Access for Blind And Visually Impaired Computer Users. *ACM SIGCAPH Computers and the Physically Handicapped*, 55: 10–15. <https://doi.org/10.1145/231674.231675>
12. Daniel Kocielinski and Jolanta Brzostek-Pawłowska. 2013. Linear Interface for Graphical Interface of Touch-Screen: a Pilot Study on Improving Accessibility of the Android- Based Mobile Devices. *Proceedings of the 15th international conference on Human-computer interaction with mobile devices and services - MobileHCI '13*: 546. <https://doi.org/10.1145/2493190.2494439>
  13. Jalal U. Mahmud, Yevgen Borodin, and I. V. Ramakrishnan. 2007. CSurf: A Context-Driven Non-Visual Web-Browser. In *Proceedings of the 16th international conference on World Wide Web - WWW '07*, 31. <https://doi.org/10.1145/1242572.1242578>
  14. Kyle Montague, André Rodrigues, Hugo Nicolau, and Tiago Guerreiro. 2015. TinyBlackBox: Supporting Mobile In-The-Wild Studies. In *Proceedings of the 17th International ACM SIGACCESS Conference on Computers & Accessibility - ASSETS '15*, 379–380. <https://doi.org/10.1145/2700648.2811379>
  15. Hugo Nicolau, João Guerreiro, Tiago Guerreiro, and Luís Carriço. 2013. UbiBraille: Designing and Evaluating a Vibrotactile BrailleReading Device. In *Proceedings of the 15th International ACM SIGACCESS Conference on Computers and Accessibility - ASSETS '13*, 1–8. <https://doi.org/10.1145/2513383.2513437>
  16. Hugo Nicolau, Kyle Montague, Tiago Guerreiro, João Guerreiro, and Vicki L. Hanson. 2014. B#: Chord-based Correction for Multitouch Braille Input. In *Proceedings of the 32nd annual ACM conference on Human factors in computing systems - CHI '14*, 1705–1708. <https://doi.org/10.1145/2556288.2557269>
  17. Hugo Nicolau, Kyle Montague, Tiago Guerreiro, André Rodrigues, and Vicki L. Hanson. 2015. HoliBraille: Multipoint Vibrotactile Feedback on Mobile Devices. In *Proceedings of the 12th Web for All Conference on - W4A '15*, 1–4. <https://doi.org/10.1145/2745555.2746643>
  18. Hugo Nicolau, Kyle Montague, Tiago Guerreiro, André Rodrigues, and Vicki L. Hanson. 2015. Typing Performance of Blind Users. In *Proceedings of the 17th International ACM SIGACCESS Conference on Computers & Accessibility - ASSETS '15*, 273–280. <https://doi.org/10.1145/2700648.2809861>
  19. Hugo Nicolau, Kyle Montague, Tiago Guerreiro, André Rodrigues, and Vicki L. Hanson. 2017. Investigating Laboratory and Everyday Typing Performance of Blind Users. *ACM Transactions on Accessible Computing* 10, 1: 1–26. <https://doi.org/10.1145/3046785>
  20. Satoshi Ohtsuka, Nobuyuki Sasaki, Sadao Hasegawa, and Tetsumi Harakawa. 2008. Body-braille system for disabled people. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 5105 LNCS: 682–685. [https://doi.org/10.1007/978-3-540-70540-6\\_98](https://doi.org/10.1007/978-3-540-70540-6_98)
  21. João Oliveira, Tiago Guerreiro, Hugo Nicolau, Joaquim Jorge, and Daniel Gonçalves. 2011. BrailleType: Unleashing Braille over Touch Screen Mobile Phones. . Springer, Berlin, Heidelberg, 100–107. [https://doi.org/10.1007/978-3-642-23774-4\\_10](https://doi.org/10.1007/978-3-642-23774-4_10)
  22. João Oliveira, Tiago Guerreiro, Hugo Nicolau, Joaquim Jorge, and Daniel Gonçalves. 2011. Blind People and Mobile Touch-based Text-Entry: Acknowledging the Need for Different Flavors. In *The proceedings of the 13th international ACM SIGACCESS conference on Computers and accessibility - ASSETS '11*, 179. <https://doi.org/10.1145/2049536.2049569>
  23. Yury Puzis. 2012. An Interface Agent for Non-visual, Accessible Web

- Automation. *Adjunct Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology*: 55–58. <https://doi.org/10.1145/2380296.2380319>
24. André Rodrigues and Tiago Guerreiro. 2014. SWAT: Mobile System-Wide Assistive Technologies. *Proceedings of the 28th International BCS Human Computer Interaction Conference on HCI 2014 - Sand, Sea and Sky - Holiday HCI*, 341–346. Retrieved September 19, 2017 from <http://dl.acm.org/citation.cfm?id=2742991>
  25. André Rodrigues, Kyle Montague, Hugo Nicolau, João Guerreiro, and Tiago Guerreiro. In-context Q&A to Support Blind People Using Smartphones. Retrieved September 15, 2017 from [http://web.ist.utl.pt/joao.p.guerreiro/publications/context-q-support\\_crc.pdf](http://web.ist.utl.pt/joao.p.guerreiro/publications/context-q-support_crc.pdf)
  26. André Rodrigues, Kyle Montague, Hugo Nicolau, and Tiago Guerreiro. 2015. Getting Smartphones to Talkback: Understanding the Smartphone Adoption Process of Blind Users. *ASSETS 2015: The 17th International ACM SIGACCESS Conference of Computers and Accessibility*: 23–32. <https://doi.org/10.1145/2700648.2809842>
  27. Kristen Shinohara and Jacob O. Wobbrock. 2011. In the Shadow of Misperception: Assistive Technology Use and Social Interactions. *Proceedings of the 2011 annual conference on Human factors in computing systems - CHI '11*: 705. <https://doi.org/10.1145/1978942.1979044>
  28. Koji Yatani and Khai N Truong. 2009. SemFeel: A User Interface with Semantic Tactile Feedback for Mobile Touch-screen Devices. *System*: 111–120. <https://doi.org/10.1145/1622176.1622198>
  29. Xuan Zhang, Cesar Ortega-sanchez, and Iain Murray. 2007. A hardware based braille note taker. In *Proceedings - 2007 3rd Southern Conference on Programmable Logic, SPL'07*, 131–135. <https://doi.org/10.1109/SPL.2007.371736>
  30. Feng Zhao, Keishi Fukuyama, and Hideyuki Sawada. 2009. Compact Braille display using SMA wire array. *Proceedings - IEEE International Workshop on Robot and Human Interactive Communication*, c: 28–33. <https://doi.org/10.1109/ROMAN.2009.5326301>
  31. Shengdong Zhao, Pierre Dragicevic, Mark Chignell, Ravin Balakrishnan, and Patrick Baudisch. 2007. earPod: Eyes-free Menu Selection using Touch Input and Reactive Audio Feedback. *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '07*: 1395. <https://doi.org/10.1145/1240624.1240836>
  32. Yu Zhong, T V Raman, Casey Burkhardt, Fadi Biadsy, and Jeffrey P Bigham. 2014. JustSpeak: Enabling Universal Voice Control on Android. *Proceedings of the 11th Web for All Conference on - W4A '14*: 1–4. <https://doi.org/10.1145/2596695.2596720>

# Anexo

## A1. Livro de códigos

A seguinte tabela representa o livro de códigos utilizado para codificar as entrevistas realizadas durante o estudo de avaliação.

<b>1. Past Challenges</b>	1.1 Apps	1.1.1 Messaging
		1.1.2 Location
	1.2 Navigation	
	1.3 Spatial abilities	
	1.4 Interface	
	1.5 Autonomy	
	1.6 Labeling	
	1.7 Updates	
1.8 Internet		
<b>2. Features</b>	2.1 Consistency	2.1.1 Dedicated areas
		2.1.2 Fixed Options
	2.2 Personalization	
	2.3 Shortcuts	2.3.1 Task
		2.3.2 App
		2.3.3 Contact
	2.4 Order and Filtering	2.4.1 Order
2.4.2 Favorites		
2.5 Dismissable		
<b>3. Benefits &amp; Expectations</b>	3.1 Learnability	
	3.2 Usefulness	
	3.3 Independence	
	3.4 Confidence boost	
	3.5 Performance	
	3.6 Overcoming Challenges	
<b>4. Concerns and Challenges</b>	4.1 Dedicated areas	
	4.2 Learnability	
	4.3 Sighted Assistance	
	4.4 Crowded interface	
	4.5 Compatibility	
	4.6 Usefulness	
<b>5. Usage and Desires</b>	5.1 Desire	5.1.1 Fixed order
		5.1.2 Customization Dedicated Areas

		5.1.3 New features
		5.1.4 Physical interface
	5.2 Usage	5.2.1 Intent to use
		5.2.2 Intent to advise usage
		5.2.3 No intent to use
<b>6. Expertise</b>	6.1 Novice	
	6.2 Expert	

## **A2. Guião do estudo**

### **INTRODUÇÃO**

Antes de mais obrigado por estar aqui hoje e aceitar participar neste trabalho. O meu nome é

[nome] e sou [profissão] no/a [universidade]. Nós já colaboramos com a Fundação há alguns anos e temos como principal objetivo tornar os telemóveis mais fáceis de usar, principalmente os dispositivos mais recentes, os smartphones.

Nos telemóveis antigos a navegação era feita sempre da mesma maneira utilizando os mesmos controlos sempre disponíveis em todas as aplicações. O mesmo não sucede nos atuais smartphones onde cada aplicação pode ter uma estrutura única e consequente uma navegação diferente.

Para facilitar a navegação em smartphones desenvolvemos um sistema que adapta todos os ecrãs de todas as aplicações e os permite navegar todos da mesma forma.

### **ESTUDO**

Temos como objetivo para este estudo perceber as vantagens e desvantagens desta abordagem quando comparado com o método atual de navegação. Para tal queria pedir a sua colaboração para testar o sistema. Iremos lhe mostrar todas as funcionalidades do sistema e concluir a sessão com um questionário sobre a sua experiência.

Até agora, alguma questão?

### **PRÉ-QUESTIONÁRIO**

Antes de começarmos, vou-lhe fazer algumas perguntas [preencher em <https://docs.google.com/forms/d/e/1FAIpQLSceNfA7DdFhIjh-GX7BWHou1ynSPx1A1AGiSpRK2AgUZa150g/viewform> ].

### **SESSÃO**

O nosso sistema divide o ecrã do smartphone em duas metades. Na metade superior do ecrã estão as diversas opções que definimos para estarem sempre disponíveis, semelhante a quando tínhamos telemóvel com teclas físicas onde uma tecla abria sempre os contactos ou as mensagens. Na metade inferior do ecrã encontra-se a área dedicada à navegação onde apenas gestos são reconhecidos (e.g. swipes esquerda/direita, duplo toques para selecionar).

Para experimentarmos, nós definimos as opções que estão sempre disponíveis. Primeiro vou-lhe pedir que explore as opções fixas disponíveis no topo do ecrã. Para as



navegar pode seleccionar o topo do ecrã e navegar com swipes ou pode explorar arrastando o dedo no topo do ecrã.

Nós iremos explicar e experimentar cada um em detalhe.

Para navegar o conteúdo do ecrã precisa de fazer um toque na área inferior do ecrã e navegar usando swipes. É possível também seleccionar a primeira opção fixa Área de Navegação que foca o primeiro elemento do conteúdo e navegar. Para já vou-lhe pedir que encontre a opção: Área de Navegação, que é a primeira opção na metade superior do ecrã. Neste momento está no homescreen vamos experimentar e encontrar a app “Definições” e seleccionar. Verificar que as primeiras opções das definições quando explora são Definições e Definições de Pesquisa.

Agora vou-lhe pedir que encontre a segunda opção fixa, para voltar a navegar nas opções fixas basta fazer um toque na parte superior do ecrã:

A segunda opção disponível permite seleccionar como o conteúdo é ordenado. Neste momento possui seis opções diferentes. Quando a opção é focado lê qual a ordenação actual e de seguida se seleccionada qual a próxima. Cada vez que clica o conteúdo do ecrã é reordenado. As seis ordenações possíveis são: a Original, Mais Usadas, Alfabética, Recentes, Favoritos e Interactivos (apenas lista botões) . Vamos primeiro ordenar Alfabeticamente. Lembra-se que as primeiras opções das Definições eram Definições e Definições de Pesquisa, agora como estão ordenadas alfabeticamente a primeira será Acerca e a segunda Acessibilidade vou-lhe pedir que volte então à Área de Navegação e encontre Acessibilidade.

A próxima opção fixa é a lista de aplicações, “Aplicações”:

Esta opção fixa é um atalho para a lista de aplicações, agora vou-lhe pedir que selecione. Se reparar a lista de aplicações está organizada de forma Alfabética, o que provavelmente não é o mais interessante. Vou-lhe pedir que mude o filtro de ordenação para as aplicações mais Mais Usadas e descubra a app mais usada deste dispositivo.

Agora na lista aparece em primeiro lugar a app das Mensagens que é a app que mais usamos neste telemóvel. Vou-lhe pedir que entre na app.

A próxima opção fixa é a lista de contactos, “Contactos”:

Esta opção permite rapidamente de qualquer ecrã aceder para ligar a um contacto. Neste momento estamos na app das mensagens, mas vou-lhe pedir usando a opção fixa para encontrar e ligar ao contacto Tiago. É de notar que o primeiro contacto da lista é o Tiago que é o contacto que mais frequentemente foi ligado no telemóvel que é o filtro que temos atualmente.

A próxima opção fixa é adicionar aos favoritos, para explicar vou pedir que entre na aplicação Definições:

Está agora ordenada por ordem de frequência. Mas imaginemos que apesar de não usar muitas vezes eu queria que a primeira opção fosse a opção Sons.

Para isso primeiro precisava de navegar o conteúdo até encontrar a opção de Som.

Agora que encontrou precisa de ir às opções fixas e encontrar a opção Adicionar Favorito. Selecione.

Agora selecione o filtro Favoritos. Apartir deste momento sempre que entrar na app de Definições com o filtro Favoritos a primeira opção seria Sons

As últimas duas opções fixas disponíveis são Definições PAL e Minimizar. Como mencionamos no início, estas opções são só algumas das que podemos adicionar. Vamos agora adicionar mais algumas opções.

Primeiro vamos seleccionar o Definições Pal:

A primeira opção permite-nos adicionar uma nova opção fixa, um atalho

Ao seleccionar de seguida tem de escolher em que posição quer o novo atalho. Vamos escolher antes da opção de Contactos.

Neste momento estão disponíveis três tipos de atalhos, para: Contatos, Aplicações ou Tarefas.

Vamos primeiro seleccionar criar um atalho para o contacto Tiago.

Agora nas opções fixas podemos seleccionar o botão Tiago para fazer uma chamada a qualquer momento.

Podemos criar um atalho semelhante para qualquer aplicação como Alarme, Whatsapp ou Facebook. Mas mais interessante podemos criar atalhos para tarefas. Vamos voltar às Definições Pal e adicionar um novo atalho para uma tarefa.

Vamos adiciona-lo como o primeiro botão antes de Área de Navegação .

De seguida vamos seleccionar uma tarefa. Neste momento temos duas disponíveis mas será possível para qualquer utilizador criar e partilhar novas tarefas. As tarefas disponíveis são Abrir Ultima Mensagem e Abrir Ultimo Email. Vamos seleccionar Abrir Última Mensagem.

Agora quando for área superior o seleccionar a primeira opção de abrir mensagem irá automaticamente abrir a última mensagem recebida. Vamos experimentar.

Terminamos a apresentação de todas as funcionalidades disponíveis de momento. Tem alguma dúvida ou algo que queira experimentar?

#### DEBRIEFING

Para terminar queria apenas fazer umas perguntas. Vou-lhe pedir autorização para gravar a entrevista.

[https://docs.google.com/document/d/16GJEY1XRivN4m3077w3wBIEzUN2CDA5\\_HpouIUzFxI/edit](https://docs.google.com/document/d/16GJEY1XRivN4m3077w3wBIEzUN2CDA5_HpouIUzFxI/edit)

Obrigado pela sua participação!

### A3. Entrevista

1. Que considerações tem do PAL?
  - a. O que achou da <funcionalidade>? O que gostou e o que não gostou sobre a <funcionalidade>? Achou útil, quando? Ou onde?
    - i. Opções fixas
    - ii. Área de navegação dedicada
    - iii. Ordenação
    - iv. Atalhos
    - v. Possibilidade de personalizar a área fixa
2. Que desafios enfrenta hoje em dia quando usa o seu smartphone?
  - a. O PAL teria uma influência positiva ou negativa?
3. Quais foram os principais desafios que enfrentou ao começar a usar um smartphone?
  - a. O PAL teria uma influência positiva ou negativa?
4. Que problemas encontrou a usar o PAL?
5. O PAL muda a maneira como interage com o seu smartphone. Prevê algum problema ou benefício em relação a essa mudança?
6. O que melhoraria no PAL?
  - a. Tentou várias opções de ordenação. Gostaria de ter mais algum tipo de ordenação disponível?
  - b. Que outras tarefas gostaria de ter ao seu dispor nos atalhos?
  - c. A metade superior do ecrã tem sempre disponível um conjunto fixo de opções, além das que já estão disponíveis, quais gostaria de ver?
7. Gostaria de ter o PAL no seu dispositivo? Aconselharia o seu uso a alguém?
8. Outros? Comentários?