

---

# Guia de Usabilidade para Desenvolvimento da Aplicações Móveis

---



# Ficha Técnica

## Guia de Usabilidade Para Desenvolvimento de Aplicações Móveis

---

### Produzido por:

AMA - Agência para a Modernização Administrativa, IP.

TicAPP - Centro de Competências Digitais da Administração Pública,

Palácio Foz, Praça dos Restauradores, 2ºPiso

1250-187 Lisboa

<https://usabilidade.gov.pt/guia-apps>



Este Trabalho tem uma Licença Creative Commons  
Atribuição • Não Comercial • Compartilha Igual 4.0 Internacional.

**Versão 1.0 (Agosto 2019)**

# Índice



<b>1. Introdução</b>	<b>4</b>
<b>2. Boas Práticas Gerais</b>	<b>6</b>
2.1. Tipos de aplicações .....	6
2.2. Tipo de Aplicação Recomendada .....	9
2.3. Acessibilidade .....	11
2.4. Partilha de dados e API .....	12
2.5. Considere tecnologias emergentes para casos complexos .....	12
<b>3. Componentes da Interface</b>	<b>13</b>
3.1. Apresentação de Conteúdo .....	13
3.2. Tipografia .....	14
3.3. Navegação .....	17
3.4. Tabelas .....	22
3.5. Formulários .....	25
3.6. Notificações .....	30
3.7. Toque .....	31
3.8. Permissões .....	39
3.9. Cache ou conteúdos offline .....	40
3.10. Pagamentos .....	41
3.11. Pesquisa .....	41
3.12. Botões primários de Call-to-Action .....	45
<b>4. Referências</b>	<b>47</b>

# 1. Introdução



A utilização de dispositivos móveis tem vindo a crescer e em 2018, segundo a [Statista](#), 48,2% do tráfego de Internet foi realizado em dispositivos móveis. Com este número significativo de utilização de dispositivos móveis, quando um serviço da Administração Pública Portuguesa cria uma aplicação móvel para disponibilizar os seus serviços, deve ter em conta a usabilidade dessa aplicação por forma a garantir o acesso fácil e universal à aplicação e aos serviços públicos nela disponibilizados, permitindo dessa forma a sua adoção pelos cidadãos e empresas.

Na nossa análise, identificámos que grande parte dos guias de usabilidade existentes estão focados na navegação em computadores desktop e não abordam as especificidades presentes em dispositivos móveis. Por esta razão, considerámos ser de enorme relevância criar o presente guia no qual apresentamos o que consideramos as boas práticas de usabilidade para aplicações móveis, como complemento às já disponibilizadas em [usabilidade.gov.pt](https://usabilidade.gov.pt).

Este documento apresenta um conjunto de boas práticas gerais e um conjunto de boas práticas ao nível dos componentes da interface das aplicações.

Para além dos exemplos e boas práticas apresentados neste guia, é recomendado que sejam seguidas as recomendações do fabricante do sistema operativo para dispositivos móveis. Pode consultar estas recomendações online, nos sites dos principais fornecedores de dispositivos móveis:

- Apple iOS – [Human Interface Guidelines](#)
- Google Android – [Material Design](#)

Como pressuposto, e de acordo com o DL n.º 83/2018, **uma Aplicação Móvel** é o “software de aplicação concebido e desenvolvido para utilização pelo público em geral, em dispositivos móveis, nomeadamente telemóveis inteligentes (smartphones)

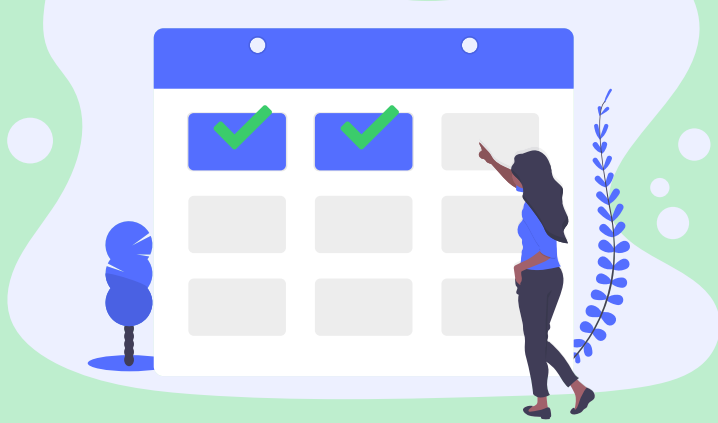
## 1. Introdução

e tablets, excluindo o software que controla esses dispositivos (sistemas operativos móveis) e o hardware;” .

Como nota adicional a esta introdução, gostaríamos de salientar que tal como para outras soluções tecnológicas, no caso da criação de uma aplicação móvel no âmbito de serviços da Administração Pública Portuguesa, deverão ser seguidas as especificações estipuladas no [Regulamento Nacional de Interoperabilidade Digital \(RNID\)](#).

## 2.

# Boas Práticas Gerais



Os cidadãos, individualmente ou no seio das empresas esperam uma boa experiência na utilização de aplicações móveis dos serviços da Administração Pública Portuguesa. A concretização deste objetivo passa por um bom funcionamento das aplicações e dos serviços nelas disponibilizados, independentemente do dispositivo ou navegador pelo qual acedem. As aplicações móveis apresentam-se como forma de simplificar o acesso aos serviços públicos porque têm, em regra, um foco concreto num determinado serviço ou tarefa relacionado com um evento de vida ou necessidade do cidadão ou empresa.

Como boa prática devemos considerar os diversos tipos de aplicações móveis que podem ser criadas, avaliando as suas características e quais as vantagens e as desvantagens para a aplicação concreta que pretendemos desenvolver.

De seguida iremos detalhar quais os tipos de aplicações móveis que devem ser analisados à luz dos requisitos de cada aplicação e serviços a disponibilizar ao cidadão ou empresa.

---

## 2.1. TIPOS DE APLICAÇÕES

---

### 2.1.1. Aplicação Web

---

As aplicações web (Web Apps) são na realidade web sites otimizados para dispositivos móveis e que apresentam uma experiência semelhante às aplicações nativas, mas através do navegador de Internet (Chrome, Safari, Edge, etc.).

Considerámos duas técnicas principais para o desenvolvimento deste tipo de aplicação:

1. *Aplicação Web Responsiva*: um site que responde às dimensões do navegador em qualquer ponto. Esta resposta às dimensões do navegador é feita de forma fluída e flexível.
2. *Aplicação Web Adaptativa*: um site que se adapta à largura do navegador em pontos específicos. Este tipo de site só foca em pontos específicos de largura e apenas se adapta nesses pontos à mudança de largura.<sup>1</sup>

Tendo em conta estas duas técnicas, a nossa recomendação é a cd que a aplicação web seja desenvolvida como uma aplicação web responsiva. Este desenvolvimento deverá seguir ainda a estratégia de [Progressive enhancement](#).

### **Progressive enhancement**

O *progressive enhancement* é uma estratégia para o design web que dá ênfase ao conteúdo principal da página. Esta estratégia passa por adicionar progressivamente mais nuances, camadas de apresentação e recursos sobre o conteúdo principal, de acordo com a velocidade e condições de acesso de dados disponível em cada momento. O benefício desta estratégia é permitir que todos os utilizadores possam ter acesso ao conteúdo e à funcionalidade base de uma página da web, usando qualquer navegador ou ligação à Internet, além de fornecer uma versão mais completa da página para aqueles utilizadores que possuam navegadores de Internet mais avançados ou que tenham disponível uma maior largura de banda.

#### **Vantagens:**

- O utilizador não precisa de instalar a aplicação no dispositivo móvel;
- Os custos de desenvolvimento são mais baixos do que no desenvolvimento de uma aplicação nativa, porque são utilizadas linguagens de programação comuns de desenvolvimento de aplicações web;
- É mais fácil disponibilizar as atualizações porque estas poderão ser automaticamente disponibilizadas e não estão dependentes da loja de aplicações.

#### **Desvantagens:**

- Não é possível utilizar todas as funcionalidades avançadas e nativas do dispositivo móvel.

---

<sup>1</sup>. Ver exemplo em <https://css-tricks.com/wp-content/uploads/2015/11/rwd-vs-adapt-example.gif>

## 2.1.2. Aplicação Nativa

---

Uma aplicação nativa (App Nativa) é definida como uma aplicação cujo código de programação é específico para o dispositivo móvel. São disso exemplo o uso de Objective C para sistema operativo iOS ou de Java para o sistema operativo Android.

### Vantagens:

- Desempenho rápido e um nível alto de fiabilidade;
- Disponibiliza uma ligação fácil a muitas das funcionalidades nativas do dispositivo, como a câmara, contactos, calendário e outras características como o giroscópio, gps, touch Id, etc.
- Algumas aplicações podem ser utilizadas sem ligação à Internet.

### Desvantagens:

- O desenvolvimento destas aplicações apresenta custos de desenvolvimento mais elevados, bem como será necessário mais tempo para o seu desenvolvimento, por causa da sua dependência da linguagem específica do sistema operativo do dispositivo.
- O desenvolvimento destas aplicações obriga a duplicar a aplicação noutras linguagens de programação para conseguir apresentar noutros dispositivos baseados em sistemas operativos distintos.

## 2.1.3. Aplicação Híbrida

---

Uma aplicação híbrida (App Híbrida) é muito semelhante a uma App Nativa, no sentido em que pode ser encontrada e instalada através da loja de aplicações (*App store*) do fabricante do sistema operativo do dispositivo (por exemplo: *PlayStore* da Google). A diferença entre as aplicações nativas e as aplicações híbridas encontra-se no processo de desenvolvimento.

### Vantagens:

- As aplicações Híbridas são desenvolvidas utilizando linguagens de programação comuns ao desenvolvimento de aplicações web como HTML 5, CSS e Javascript.
- São utilizáveis em todos os tipos de dispositivo e o código precisa de ser escrito apenas uma vez, em que apenas o contexto de execução é nativo ao sistema operativo do dispositivo.
- Os custos e o tempo necessários ao desenvolvimento deste tipo de aplicações



são menores do que no caso das aplicações nativas.

### **Desvantagens:**

- Apesar do desempenho deste tipo de aplicações ser semelhante ao das aplicações nativas, no caso das aplicações com necessidades de gráficos de alta qualidade, ocorre uma degradação do desempenho da aplicação.
- Precisam de um plugin nativo (o contexto de execução) para conseguir aceder às funcionalidades nativas do dispositivo como por exemplo o *touch ID* e a câmara. Se não existir um plugin nativo, será necessário desenvolvê-lo, o que poderá ser uma tarefa mais complexa.
- Todas as aplicações híbridas estão dependentes de *frameworks* e bibliotecas *open source*, como o *Ionic* ou o *Cordova*.

### **2.1.4. Progressive Web Applications**

---

Algumas tecnologias emergentes permitem evoluir as aplicações web e torná-las em aplicações web progressivas (*progressive web applications* - PWA). Estas tecnologias disponibilizam funcionalidades que até agora estavam disponíveis apenas para aplicações nativas ou híbridas.

Exemplos de funcionalidades que poderemos tirar partido com o uso de *pwa*:

- adicionar ícones ao ecrã principal do telemóvel
- enviar notificações *push* ao utilizador
- disponibilizar um serviço quando o utilizador está sem ligação à Internet
- integrar um serviço com outras partes do telemóvel, como por exemplo a câmara fotográfica

A utilização de *PWAs* permite que os utilizadores deixem de estar dependentes da instalação de novas versões das aplicações para terem acesso a novas funcionalidades e correção de bugs.

---

## **2.2. TIPO DE APLICAÇÃO RECOMENDADA**

---

Na grande maioria das situações e contextos de disponibilização de aplicações móveis pelos serviços públicos, recomendamos a utilização de uma aplicação web progressiva, que para o utilizador, é essencialmente um web site acessado através do navegador do

dispositivo.

Como referido anteriormente o processo de desenvolvimento é menos dispendioso e é mais fácil melhorar e manter uma aplicação web, comparativamente com aplicações móveis nativas ou híbridas. Por outro lado, também se consegue mais facilmente oferecer ao utilizador uma experiência mais consistente, sendo indiferente a forma como acedem ao serviço.

No entanto podem existir alguns casos em que uma aplicação web progressiva não é a melhor solução para as necessidades dos serviços a disponibilizar na aplicação. Nos pontos seguintes iremos detalhar estas situações.

### 2.2.1. Quando optar por aplicações nativas ou híbridas

---

Há situações em que uma aplicação nativa ou híbrida é a única solução como por exemplo:

- O utilizador necessita de recolher e guardar dados, mas não lhe é possível ter uma ligação de Internet que seja estável, sendo necessário o funcionamento em modo offline, e uma PWA não é opção.
- O serviço só funciona se tiver uma iteração constante no dispositivo do utilizador, por exemplo, aplicações de *fitness* e saúde.

Outra situação, em que poderá fazer sentido utilizar uma aplicação nativa, será quando a mesma só necessita de funcionar num dispositivo. Exemplos práticos:

- Todos os utilizadores utilizam o mesmo dispositivo padrão departamental.
- Usar uma capacidade específica do dispositivo.

Salvo exceções do tipo das acima referidas, é bastante difícil justificar a opção de aplicações híbridas/nativas no desenvolvimento de uma aplicação, para disponibilização de um serviço, para o cidadão comum. Ainda assim, será necessário seguir boas práticas gerais e boas práticas de serviço.

#### Se está limitado por funcionalidades do dispositivo

Se a sua aplicação nativa requer uma funcionalidade que se encontra disponível num pequeno grupo de dispositivos, então terá que apresentar a aplicação como *progressive enhancement*.

Nesta situação, apresentamos a aplicação aos utilizadores que possuem esses

dispositivos, uma vez que é a forma mais fácil de completarem a sua tarefa. Mesmo assim, será necessário, apresentar aos restantes utilizadores que não possuem estes dispositivos, com uma forma simples de conseguirem completar as suas tarefas.

### 2.2.2. Quando evitar aplicações nativas ou híbridas

---

Como referido anteriormente, as aplicações nativas ou híbridas não são a melhor forma de disponibilizar serviços. É mais simples e menos dispendioso disponibilizar um website responsivo. Um site responsivo, desenvolvido utilizando boas práticas web, funciona em todos os navegadores e dispositivos móveis.

Ao construir uma aplicação nativa ou híbrida, será necessário suportar todos os sistemas operativos para os dispositivos móveis mais utilizados nas localizações alvo da aplicação. Isto pode tornar-se muito dispendioso quer na construção quer na manutenção das aplicações.

Tipicamente, existem outras abordagens possíveis, para ultrapassar algumas das limitações das aplicações web, do que desenvolver uma aplicação nativa ou híbrida, como as já referidas aplicações web progressivas.

Explore sempre as alternativas possíveis antes de decidir desenvolver aplicações nativas ou híbridas – e só o faça se não encontrar outra forma de ir ao encontro das necessidades do utilizador.

---

## 2.3. ACESSIBILIDADE

---

De acordo com o DL n° 83/2018, deve considerar-se acessibilidade como: “« os princípios e técnicas a observar na conceção, construção, manutenção e atualização de sítios web e aplicações móveis de forma a tornar os seus conteúdos mais acessíveis aos utilizadores, em especial a pessoas com deficiência;”

Neste sentido as aplicações móveis a disponibilizar no âmbito dos serviços da Administração Pública Portuguesa deverão cumprir as normas enunciadas no decreto lei acima referido e deverão ainda usar todas as ferramentas e guias de orientação disponibilizados no site [www.acessibilidade.gov.pt](http://www.acessibilidade.gov.pt).

## 2.4. PARTILHA DE DADOS E API

---

Uma das boas práticas que os serviços públicos devem seguir, é disponibilizar o acesso aos dados e às *Application Programming Interfaces (API)* dos seus serviços a outras entidades para que estas possam utilizar os dados e os serviços para construir serviços complementares ou adicionais aos serviços da entidade.

Esta abordagem também pode funcionar se identificar uma necessidade que, por constrangimentos diversos, não possa ser respondida pela sua entidade. Ao disponibilizar o acesso aos dados, pode estar a estimular o mercado a criar um serviço que possa responder a essa necessidade.

Um exemplo desta prática é o *Transport of London* que abriu o acesso das suas APIs, de forma a que algumas empresas privadas tivessem a possibilidade de desenvolver aplicações relacionadas com a rede de transportes públicos em Londres, como é o caso da aplicação *CityMapper*.

---

## 2.5. CONSIDERE TECNOLOGIAS EMERGENTES PARA CASOS COMPLEXOS

---

Um ponto relevante em termos de boas práticas é manter-se a par dos novos desenvolvimentos tecnológicos. Uma das formas de o conseguir é seguindo blogs e outras informações online, como por exemplo:

- [Informação para developers da plataforma Google](#)
- [Informação para developers da plataforma Apple](#)
- [Novidades em experiência de utilização](#)
- [Tailor your PWA for Windows](#)
- [Site de PWA da Google](#)
- [Progressive Web Apps: Escaping tabs without losing our soul \(blog\)](#)
- [Getting started with Progressive Web Apps \(blog\)](#)

# 3.

## Componentes da Interface



---

### 3.1. APRESENTAÇÃO DE CONTEÚDO

---

Um dos casos mais comuns de consumo de conteúdos em dispositivos móveis é a utilização dos mesmos como forma de entretenimento ou para ocupar um tempo de espera (ex. enquanto espera uma consulta ou por um meio de transporte). Nestes cenários os utilizadores valorizam bastante a rapidez do acesso à informação, ficando impacientes com aplicações que desperdiçam o seu tempo.

Desta forma, no que diz respeito à apresentação de conteúdo é recomendado manter o conteúdo para dispositivos móveis curto e sucinto, atendendo ao seu contexto de utilização.

Em termos de boas práticas para evitar desconforto nas situações acima descritas, devemos considerar as seguintes:

- Reduzir o conteúdo, para apresentar apenas o que é estritamente necessário. Muitas vezes os utilizadores preferem um resumo do conteúdo que aborda apenas os temas principais, especialmente no caso de notícias.

---

Fontes e Recursos Adicionais:

<https://www.nngroup.com/articles/mobile-content/>

<https://www.nngroup.com/articles/condense-mobile-content/>

- Caso exista informação secundária que possa ser pertinente, deverá ser colocada num segundo ecrã mais fácil de consultar; mas apenas por quem estiver mesmo interessado.

## 3.2. TIPOGRAFIA

A distância a que o utilizador segura o dispositivo móvel, enquanto o utiliza, é variável e depende do contexto de utilização e do tamanho do dispositivo.

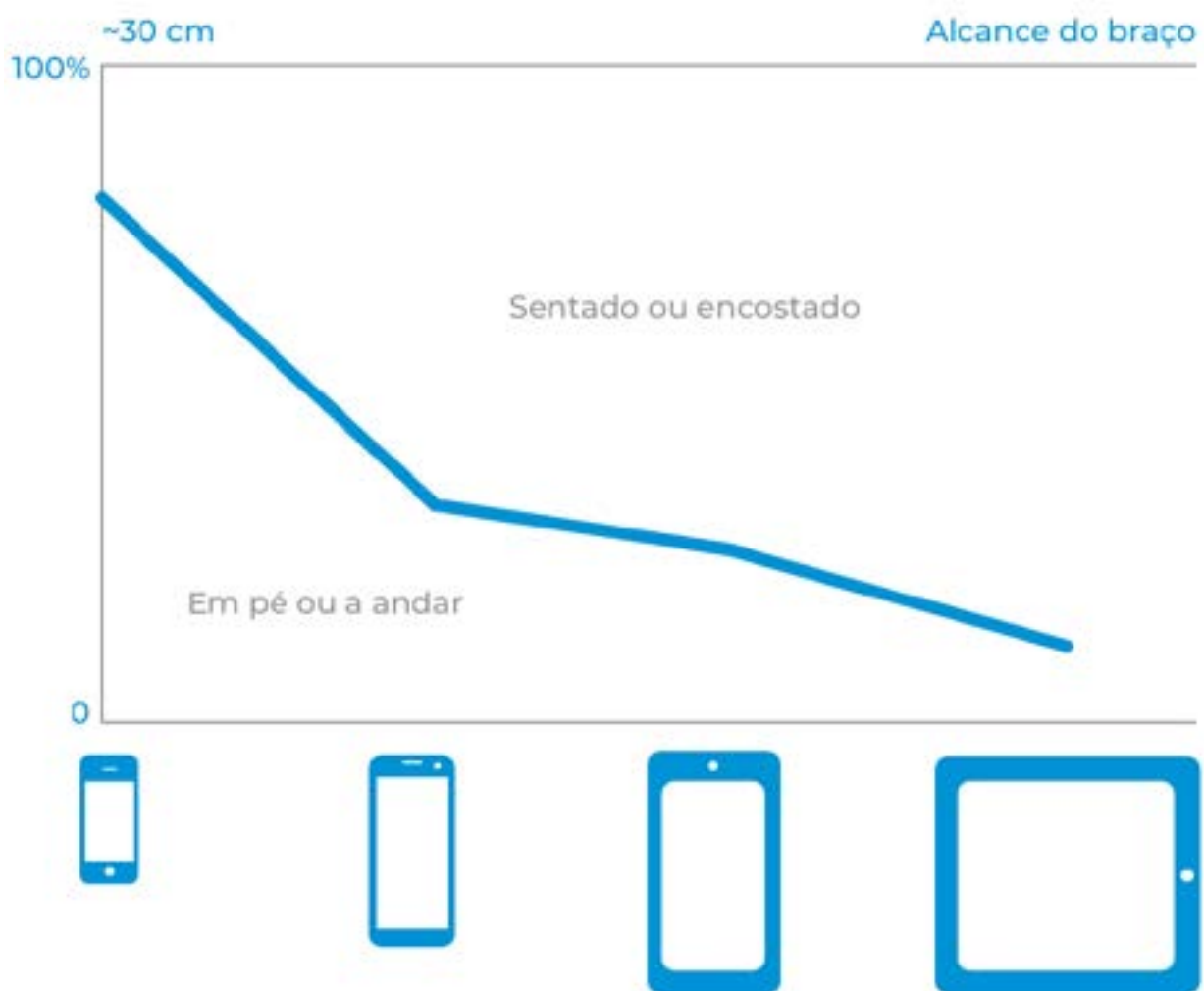


Figura 1 – Distâncias do ecrã dependente no dispositivo

Além disso, quando se trata de dispositivos móveis maiores como tablets, os utilizadores habitualmente usam-no com algum tipo de suporte:

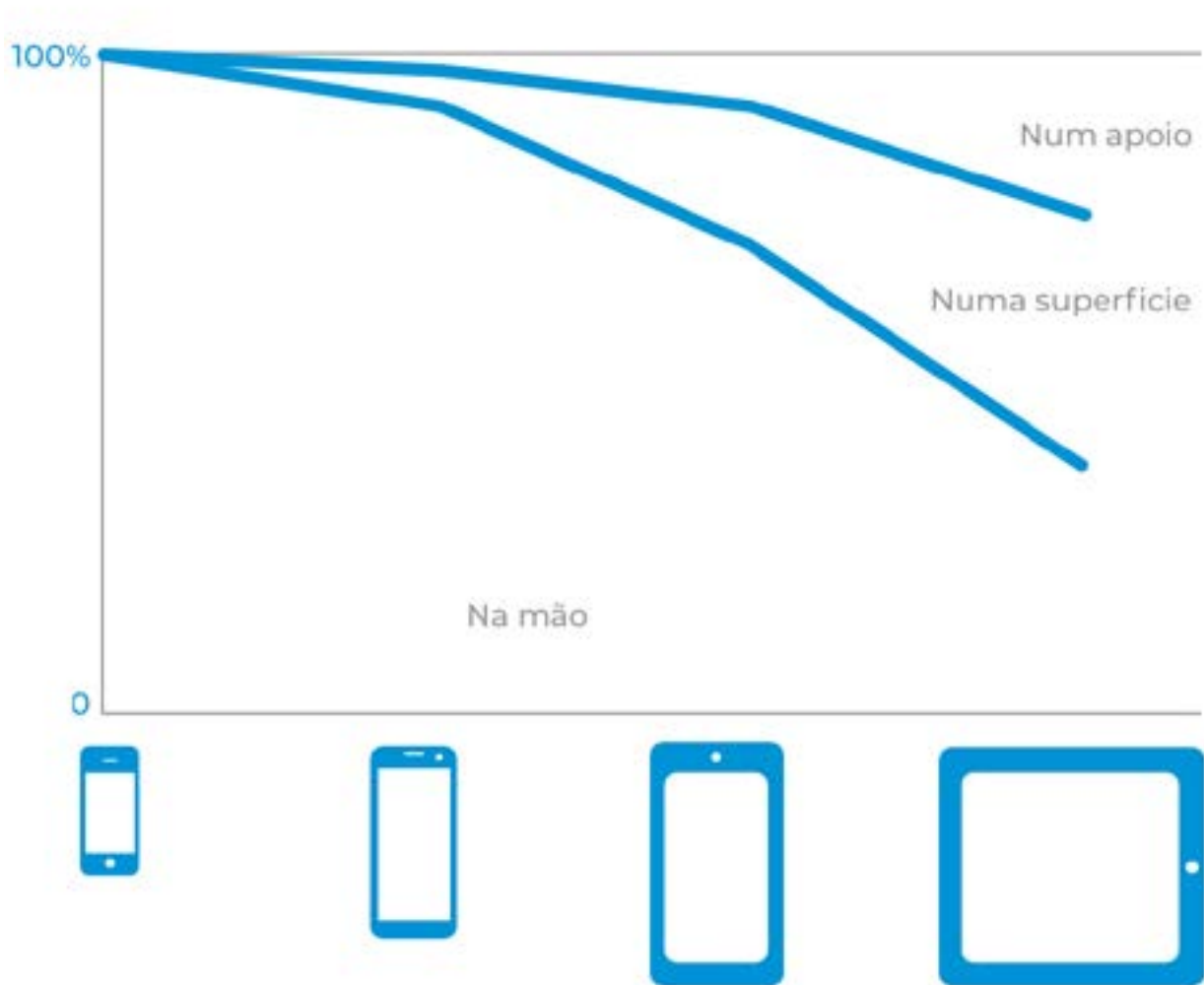


Figura 2 – Forma de uso dependente do tamanho do dispositivo

Os dispositivos mais pequenos são mais utilizados quando o utilizador está em movimento, tanto na rua, como em casa. Os dispositivos maiores normalmente são utilizados à distância de um portátil.

Abaixo encontra-se uma tabela que apresenta as melhores práticas para usar tipografia em dispositivos móveis baseado no contexto, tamanho e distância de utilização:

Tamanho dos conteúdos e subtítulos, em pontos tipográficos						
Classe do Dispositivo	Tamanho Mínimo	Conteúdo Básico	Conteúdo Melhorado	H3	H2	H1
Telemóvel Pequeno	4	5.5	7.2	8.5	10.8	14.4
Telemóvel Grande	6	8.5	10.8	12.6	16.2	21.6
Phablet	7	9.8	12.6	14.7	18.9	25.2
Tablet Pequeno	8	11.2	14.4	18.8	21.6	28.8
Tablet Grande/ Desktop	10	14	18	21	27	36

Tabela 1 - guia para tipografia

---

#### Fontes e Recursos Adicionais:

<https://www.uxmatters.com/mt/archives/2017/05/design-for-fingers-touch-and-people-part-2.php>



## 3.3. NAVEGAÇÃO

---

As boas práticas de usabilidade indicam que a navegação deverá ser evidente, uma vez que disponibilizar a melhor funcionalidade ou conteúdo não tem qualquer utilidade se o utilizador não conseguir encontrar essa funcionalidade ou conteúdo.

Algumas regras básicas para a componente de navegação numa aplicação móvel são:

- **Não esconder elementos de navegação** – devem ser evitados elementos de navegação oculta, como por exemplo, navegação por gestos, porque os utilizadores irão ter dificuldade em identificar essa forma de navegação.
- **Navegação consistente** - muitas vezes, quem desenvolve aplicações esconde a navegação em páginas individuais. Esta prática deve ser evitada porque pode confundir ou desorientar o utilizador.
- **Comunicar a localização atual** - uma das maiores falhas no desenvolvimento de aplicações móveis é a falta de indicadores da localização onde o utilizador se encontra na aplicação. Para um utilizador entender para onde quer navegar, ele precisa de perceber onde está.

Em geral, é boa prática apresentar ao utilizador apenas um caminho para cada ecrã. Se o utilizador precisa de ver um ecrã em vários contextos, considere o uso de um alerta, *pop over* ou exibição modal.

**Dica:** A melhor solução é utilizar padrões de navegação estabelecidos – como o [Tab Bar](#) (para iOS) e [Navigation Drawer](#) (para Android). A grande maioria dos utilizadores de aplicações móveis estão familiarizados com os dois padrões de navegação. Quando há uma solução simples, não é preciso repensar tudo.

---

#### Fontes e Recursos Adicionais:

<https://www.uxpin.com/studio/blog/guide-mobile-app-design-best-practices-2018-beyond/>

<http://babich.biz/basic-patterns-for-mobile-navigation/>

<https://developer.apple.com/design/human-interface-guidelines/ios/app-architecture/navigation/>

<https://material.io/design/navigation/understanding-navigation.html>

### 3.3.1. Deslizar na horizontal

---

O deslizar vertical como forma de interação já é considerado normal para utilizadores de dispositivos eletrónicos. Com a introdução de dispositivos móveis, com espaço de ecrã limitado, o deslizar na horizontal, que antes era considerado confuso, neste momento é aceite como solução bastante útil.

#### Quando usar deslizar na horizontal

O deslizar na horizontal funciona bem quando se pretende mostrar um subconjunto de uma categoria. Alguns exemplos:

1. Um catálogo de produtos ou itens, com muitos elementos:

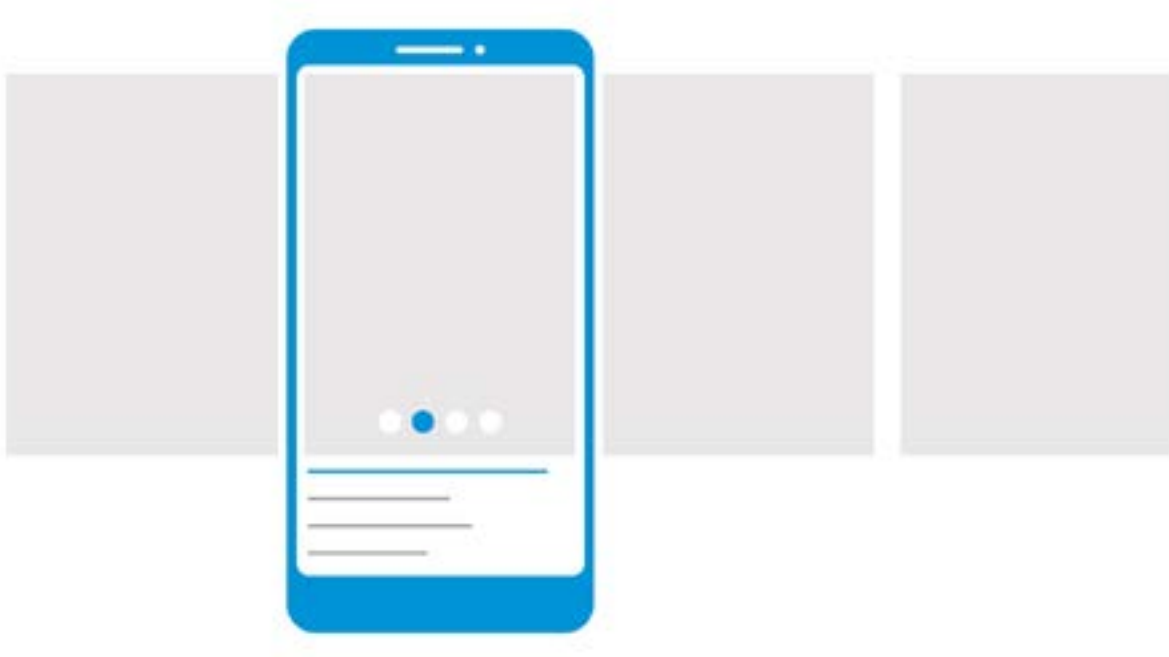


Figura 3 - Exemplo de apresentação de vários itens horizontalmente

### 3. Componentes da Interface

2. Apresentar informação numa área visual muito grande que não é fácil de ver de relance como por exemplo num mapa.



Figura 4 - Exemplo de deslizar horizontal usado num mapa

3. Mostrar secções discretas ou partes de informação em aplicações

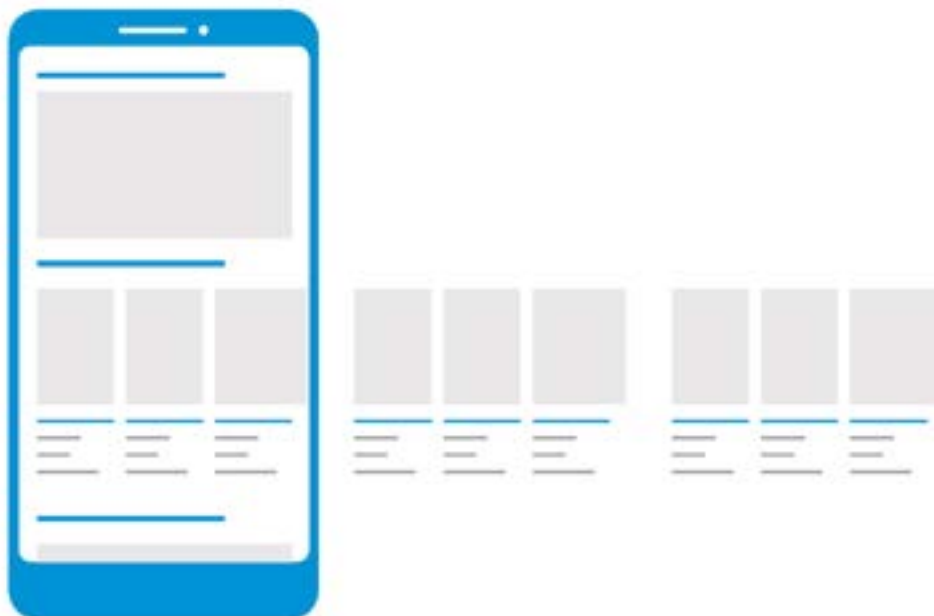


Figura 5- Exemplo de mostrar informações extra de uma secção

### Como usar o deslizar na horizontal

Embora muito útil, o deslizar na horizontal pode ser pouco intuitivo se não for apresentado da forma correta. É necessário indicar a possibilidade de poder efetuar um deslizar horizontal usando dicas visuais.



Figura 6 - Exemplo de como mostrar a possibilidade de deslizar horizontal

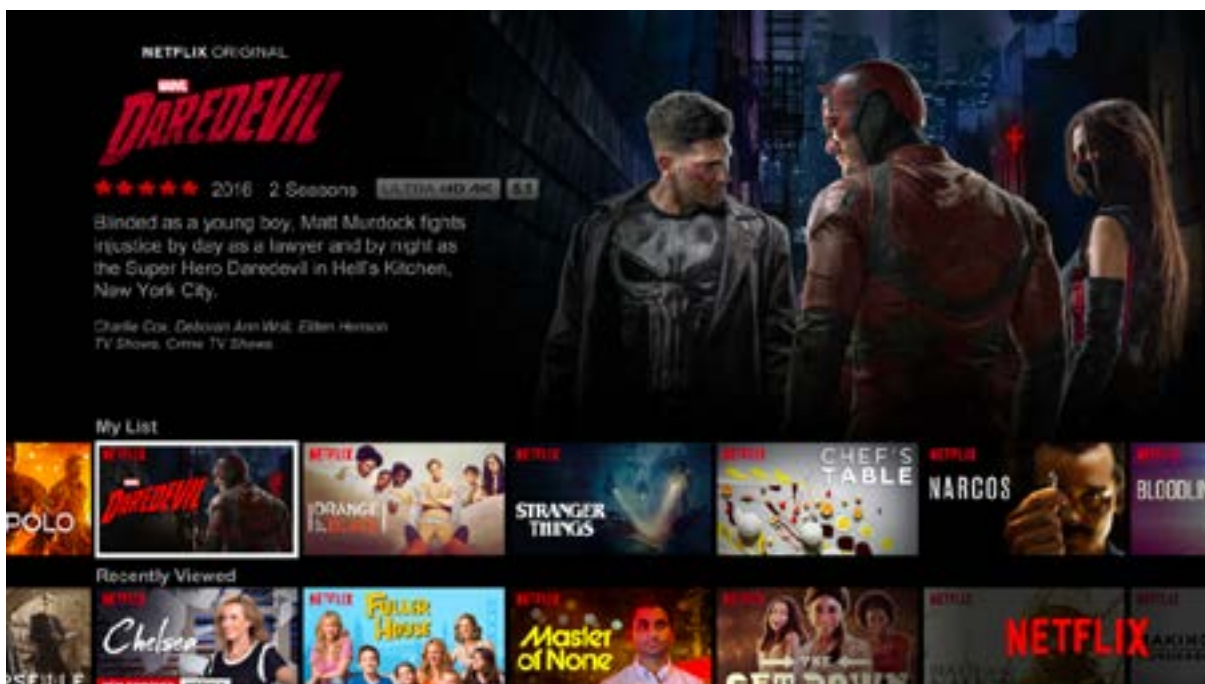


Figura 7 - Exemplo do uso de deslizar horizontal no Netflix

## Indicar o fim de uma lista

Além de perceber que o conteúdo pode ser deslizado horizontalmente, também é importante indicar quando o conteúdo termina e não poder continuar a deslizar.

A forma mais recomendada para indicar o final do conteúdo horizontal é inserindo um espaço maior no final. Assim o utilizador entende que não mais conteúdo para o lado dentro das dimensões expectáveis.

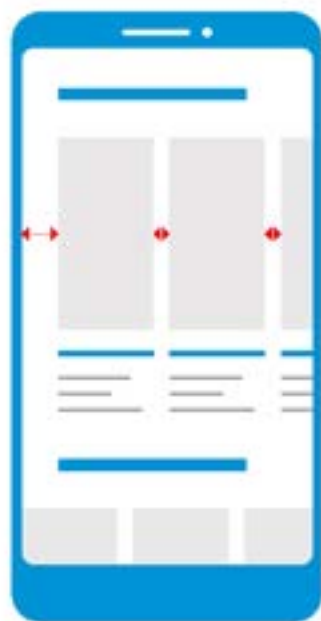


Figura 8 - Exemplo de espaço extra para indicar o final do conteúdo

---

### Fontes e Recursos Adicionais:

<https://uxplanet.org/horizontal-scrolling-in-mobile-643c81901af3>

<https://uxplanet.org/bidirectional-scrolling-is-here-to-save-responsive-design-be1afe53206d>

<https://uxdesign.cc/scrolling-interactions-techniques-d6dafbfa4716>

<https://www.smashingmagazine.com/2018/08/best-practices-for-mobile-form-design/>

## 3.4. TABELAS

Habitualmente as tabelas são usadas para apresentar conteúdos complexos e com diversas categorias. A apresentação desta informação torna-se uma tarefa quase impossível quando lidamos com dispositivos móveis com limitações de espaço devido a ecrãs de pequena dimensão e resolução.

Algumas regras básicas para apresentar tabelas em dispositivos móveis:

- **Rodar o dispositivo deveria ser um último recurso** - quando se roda o telefone para uma vista horizontal, o espaço ganho em largura é diretamente inverso ao espaço perdido a nível vertical.



Figura 9 - Exemplo de visualização horizontal vs. vertical

- **Indicar claramente se o *scroll* horizontal é necessário** - existem muitos casos onde será necessário disponibilizar um *scroll* horizontal porque há mais conteúdo do que é possível apresentar. Nestes casos, para se conseguir que seja óbvio para o utilizador a utilização do *scroll* horizontal, deverá ser apresentado o conteúdo cortado ou em alternativa apresentar setas que indiquem que há mais conteúdo na horizontal.



Figura 10 - Exemplo de indicação visual de mais conteúdo horizontal



Figura 11 - Exemplo de indicação de mais conteúdo horizontal com o uso de setas

- **Deixar o utilizador escolher a informação a visualizar** - quando o espaço disponível não possibilita apresentar toda a informação, deveremos dar ao utilizador a possibilidade de filtrar a informação, para que possa seleccionar a informação útil.



Figura 12 - Exemplo de como apresentar filtros

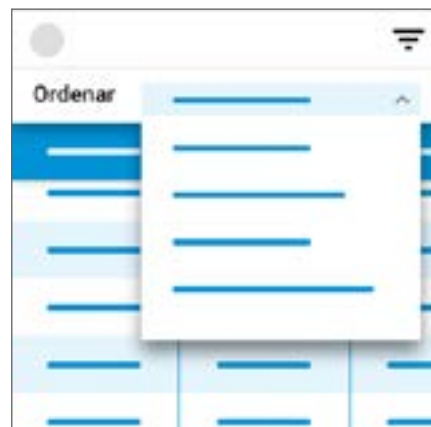


Figura 13 - Exemplo de como apresentar ordenação

- **Utilizar acordeões para grupos de dados** - se o conteúdo inclui dados que conseguem ser agrupados em categorias lógicas, então poderá fazer sentido a utilização de um acordeão. Assim o utilizador consegue ter uma ideia geral do conteúdo, observando as categorias; e ter um acesso mais direto a informação que considera útil.



Figura 14 - Exemplo de agrupamento de conteúdos e uso de acordeões

- **Utilizar ícones e cores** - com o uso correto de cores e ícones, o utilizador consegue distinguir os conteúdos da tabela de uma forma mais fácil e comparar dados de uma forma mais eficaz.

A imagem mostra uma interface de usuário com uma barra de navegação azul no topo. Abaixo, há uma tabela com seis linhas. Cada linha contém um ícone de seta verde para cima ou vermelha para baixo, seguido de um valor percentual. Os ícones de seta verde indicam valores positivos, e os ícones de seta vermelha indicam valores negativos. Os valores são: +2,35%, +1,15%, -0,18%, +0,11%, -0,49% e -0,21%.

Figura 15 - Exemplo de como usar cores e ícones

#### Fontes e Recursos Adicionais:

<https://www.nngroup.com/articles/mobile-tables/>

<https://www.nngroup.com/articles/visual-indicators-differentiators/>

<https://www.uxpin.com/studio/blog/guide-mobile-app-design-best-practices-2018-beyond/>

<https://www.nngroup.com/articles/mobile-tables/>

<https://www.nngroup.com/articles/visual-indicators-differentiators/>



## 3.5. FORMULÁRIOS

---

### 3.5.1. Facilidade de Interação

---

Os utilizadores devem conseguir preencher um formulário de forma rápida e simples.

Há dois fatores principais que influenciam a facilidade de se completar ou não o preenchimento de um formulário:

- **Perceção de complexidade** - quanto mais complexo um formulário parece, menores serão as probabilidades de um utilizador completar o preenchimento de um formulário.
- **Esforço de interação** - o nível de esforço na interação com o formulário tem um grande impacto no seu preenchimento. Se o utilizador, por exemplo, tem dificuldade em conseguir introduzir a informação, entender a pergunta ou em entender mensagens de erro, maior será a taxa de abandono de preenchimento do formulário.

### 3.5.2. Número de campos

---

Sempre que possível deveremos reduzir o número de campos a preencher para o mínimo possível. Não deveremos pedir ao utilizador maior esforço do que o estritamente necessário.

Por vezes há alguma informação que só vamos precisar quando o utilizador quiser efetuar uma tarefa específica. Pelo que poderá só fazer sentido pedir essa informação apenas quando o utilizador inicia essa tarefa.

### 3.5.3. Apresentar teclado apropriado

O preenchimento de um campo num dispositivo móvel é uma tarefa complexa. Sempre que possível, deve ser apresentado ao utilizador um teclado adequado ao conteúdo a preencher:



Figura 16 - Exemplo de apresentação do teclado apropriado iOS

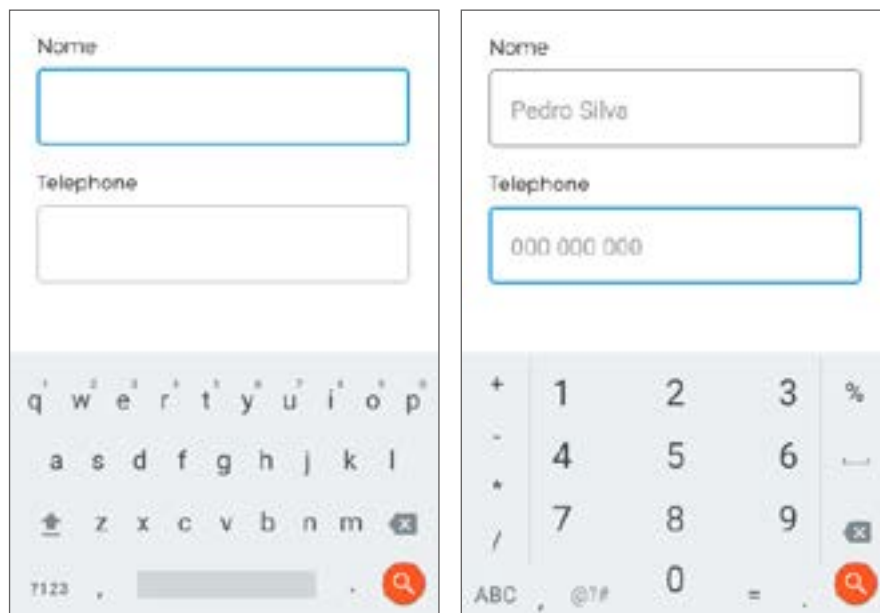


Figura 17 - Exemplo de apresentação do teclado apropriado Android

### 3.5.4. Serviços de localização

No preenchimento da morada do utilizador, uma boa prática deve ser, permitir que o utilizador pré-seleccione a sua localização com base na sua georreferenciação, o que permitirá o preenchimento automático desse campo. Usando, por exemplo, a API do Google Places, é possível apresentar uma solução híbrida que ajuda a resolver questões de georreferenciação imprecisas.



Um formulário de morada com os seguintes campos:

- Morada: Rua dos Açores, 23
- Andar: (campo vazio)
- Localidade: Lisboa
- Código Postal: 1000 - 001 Lisboa
- Botão Submeter

Figura 18 - Exemplo do uso de serviço de localização para facilitar preenchimento da morada

### 3.5.5. Autenticação biométrica

Um dos maiores problemas do utilizador é tentar lembrar-se da sua palavra passe. Hoje em dia, já é possível realizar a autenticação usando a impressão digital ou reconhecimento facial.

Esta opção deverá apenas ser usada com autorização do utilizador, uma vez que o utilizador poderá não querer utilizar este sistema de autenticação.



Figura 19 - Autenticação biométrica

### 3.5.6. Câmera fotográfica

Hoje em dia já é possível utilizar a câmera fotográfica do dispositivo móvel de forma a digitalizar os dados do utilizador e assim preencher os dados automaticamente.



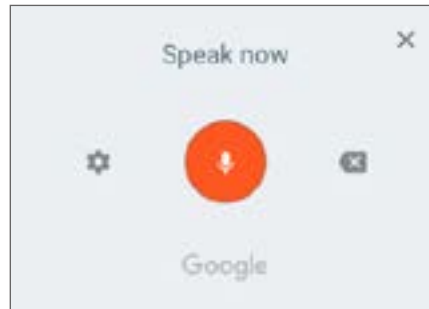
*Figura 20 - Exemplo do uso da câmara fotográfica*

Sempre que possível, esta opção deve ser explorada para ajudar a reduzir o tempo e o esforço necessário por parte do utilizador para preencher um formulário. Esta solução, embora muito prática, ainda pode apresentar alguma taxa de erro pelo que o utilizador deverá sempre ter a opção de conseguir editar os campos, para que possa corrigir alguma informação que esteja errada.

### 3.5.7. Voz

---

Com o desenvolvimento de reconhecimento de voz em dispositivos móveis, cada vez mais é comum haver a possibilidade de completar tarefas utilizando comandos de voz. Esta opção poderá ser uma grande ajuda para um cidadão que tenha dificuldades de visão ou mesmo de literacia digital reduzida.



*Figura 21 - Exemplo de uso de voz em sistema Android*



*Figura 22 - Exemplo de uso de voz em sistema iOS*

É preciso ter atenção que esta opção não irá funcionar em todos os casos, porque o reconhecimento de voz ainda é limitado, principalmente em línguas menos comuns, como é o caso do Português de Portugal.

---

**Fontes e Recursos Adicionais:**

<https://www.nngroup.com/articles/mobile-input-checklist/>

<https://www.nngroup.com/articles/gui-slider-controls/>

<https://www.smashingmagazine.com/2018/08/best-practices-for-mobile-form-design/>

## 3.6. NOTIFICAÇÕES

Todos os dias, os utilizadores são confrontados com inúmeras notificações inúteis que os interrompem ou distraem das suas tarefas diárias, o que pode tornar a aplicação muito incomodativa.

Num estudo efetuado em 2014 pela *App Iterate*, conclui-se que a razão principal pela qual as pessoas desinstalam aplicações são notificações incomodativas (71% dos utilizadores entrevistados).

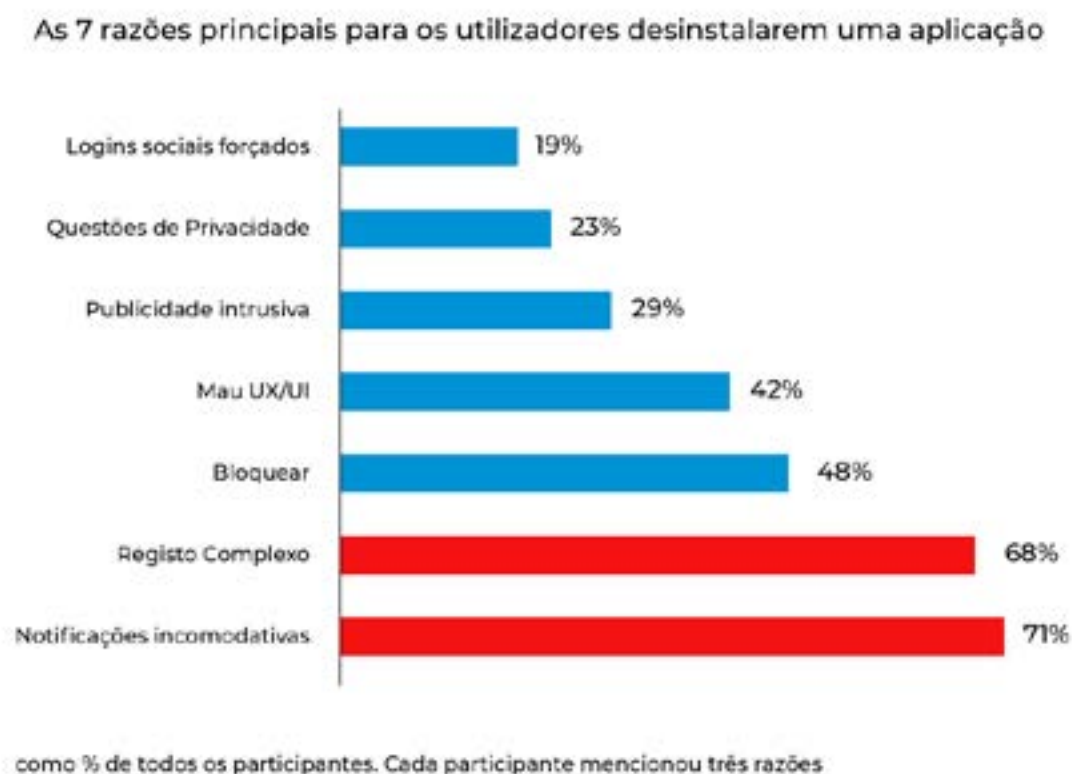


Figura 23 – Razões principais para desinstalar uma aplicação

Envie notificações apenas se achar que são úteis para o utilizador. Algumas boas práticas quando se trata de notificações são:

- Não devem ser enviadas notificações apenas com o propósito de atrair o utilizador à aplicação.
- Validar que a mensagem é clara e perceptível. Independentemente do conteúdo da notificação, a linguagem escrita e visual deve abordar o utilizador de forma que ele entenda.

- Apresentar conteúdo de acordo com os gostos pessoais do utilizador ou com o seu perfil de utilização. Independentemente da frequência com que utilizam a aplicação, os utilizadores apreciam conteúdo que é diretamente relacionado com os seus gostos pessoais.
- Evitar enviar notificações aos utilizadores em alturas impróprias. Nenhum utilizador gosta de ser acordado com uma notificação a meio da noite. Sempre que possível, enviar as notificações no fuso horário atual do utilizador.

## 3.7. TOQUE

O uso de dispositivos móveis é feito maioritariamente através do toque com os dedos da mão. Esta forma de interagir com os dispositivos apresenta desafios complexos e oferece muito mais opções de interação do que o uso do rato nos computadores desktop.

### 3.7.1. Posições mais comuns

Ao observar os hábitos comuns de utilização de dispositivos móveis, foram identificadas 6 posições principais:

Um ponto importante é que menos de 50% dos utilizadores



Figura 24 – 6 posições mais comuns de utilização dos smartphones – fonte: UXMatters

#### Fontes e Recursos Adicionais:

<https://uxplanet.org/mobile-design-best-practices-2d16d37ecfe>

<https://uxplanet.org/how-to-craft-mobile-notifications-that-users-actually-want-7b585e0e1fa1>

usam o telemóvel utilizando apenas uma mão e a posição poderá variar muito de acordo com o dispositivo, as suas necessidades ou o contexto de uso.

Abaixo há um exemplo de como os utilizadores mudam de posição dependendo da atividade a realizar:

O ponto importante a ter em mente ao desenvolver as aplicações é manter a estrutura

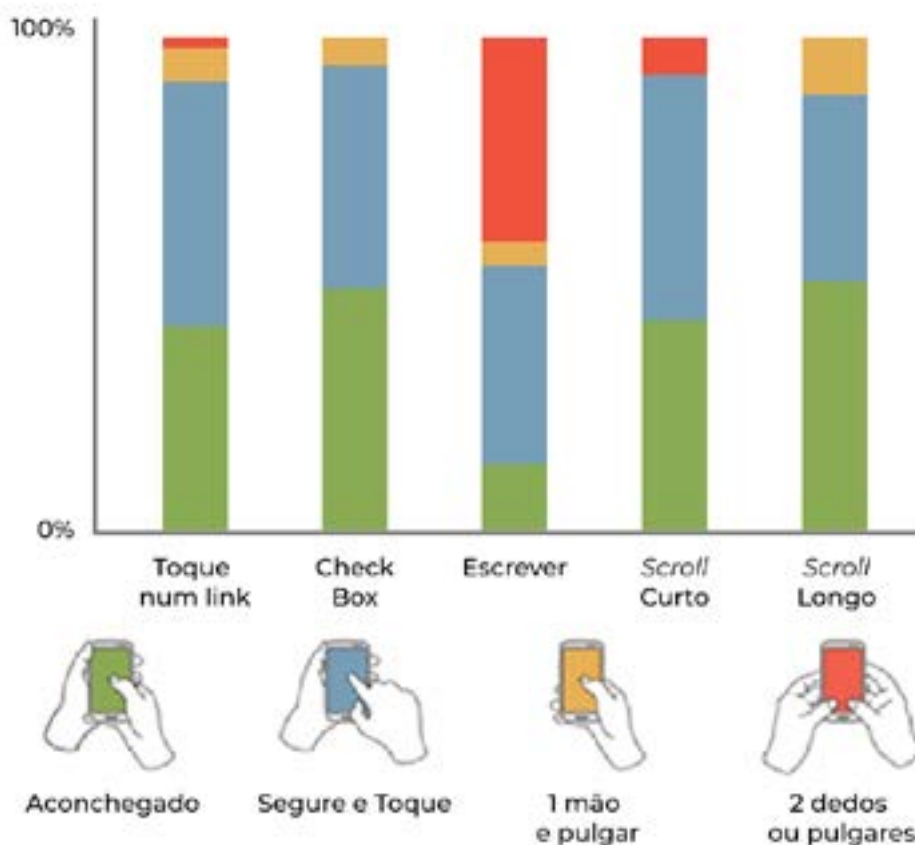


Figura 25 – Posições mais usadas vs. atividade a realizar – fonte: UXMatters

consistente para não obrigar o utilizador a mudar de posição de mãos ou a posição do dispositivo continuamente.



### 3.7.2. Espaço no ecrã

O espaço mais utilizado em dispositivos móveis é o espaço central do ecrã. Esta interação também coincide com a posição no ecrã onde o utilizador é mais preciso com o toque.



Figura 26 – Precisão do toque relativo à zona do ecrã – fonte: UXMatters

Os utilizadores habitualmente efetuam scroll para colocar a informação mais importante no centro do ecrã. É por isso que as estruturas de páginas em formato de listagens e grelhas funcionam em grande parte das situações.

A melhor prática de desenvolvimento de conteúdo para dispositivos móveis é colocar a informação primária no centro, a informação secundária na parte inferior da página e a informação menos relevante em cima, mais longe dos dedos como o exemplo abaixo.

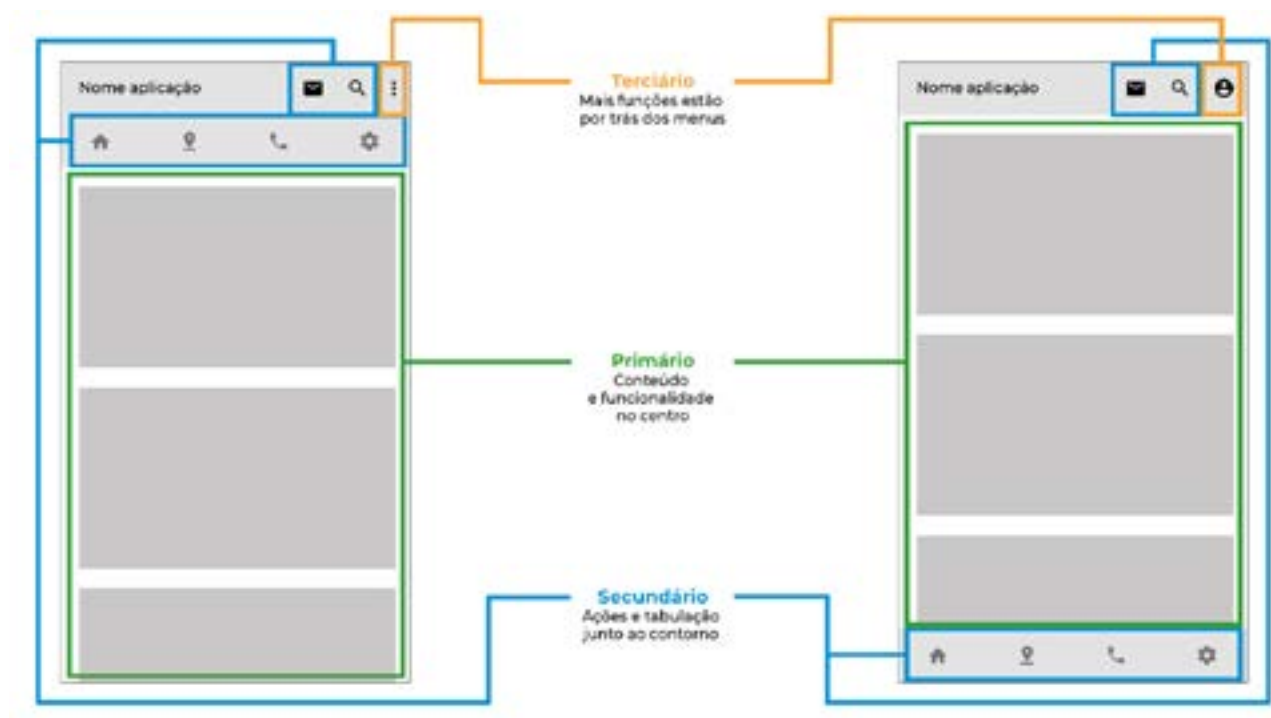


Figura 27 – Melhor estrutura base para os conteúdos principais.

### 3.7.3. Tamanho da área de toque

O tamanho da área de toque num elemento é fundamental para a boa utilização de uma aplicação num dispositivo móvel. É muito mais difícil pressionar uma área muito pequena do que uma área de toque maior. Igualmente, grupos de elementos com áreas de toque com pouco espaçamento entre os elementos, pode causar muitas frustrações, porque utilizador pode pressionar o elemento errado.

#### Tamanhos recomendados

Os fabricantes de dispositivos móveis apresentam algumas guias do tamanho recomendável para os seus dispositivos.

- **Apple iOS:** Tamanho mínimo de 44 pixéis por 44 pixéis ([Device Screen Sizes and Orientations](#)).
- **Google Android:** Tamanho mínimo 48 pixéis por 48 pixéis ([Hierarchy and placement](#)).

Fontes e Recursos Adicionais:

<https://www.uxmatters.com/mt/archives/2017/03/design-for-fingers-touch-and-people-part-1.php>

No entanto, a densidade de pixels pode variar dependendo do dispositivo, e por isso recomenda-se medir o tamanho real no ecrã físico. Neste contexto, o tamanho recomendado é de **7-10mm**.

Um estudo da [MIT Touch Lab](#), chegou à conclusão que o tamanho médio do dedo indicador de um adulto mede 16-22 mm, ou seja, acima do tamanho recomendado por fabricantes. Além disso, chegou-se à conclusão o tamanho médio de um polegar adulto é de 25 mm de largura.

Para que as aplicações possam ser utilizadas pelo maior número possível de utilizadores, recomendamos o uso das medidas apresentadas pela MIT Touch Lab:

- 25x25mm (Recomendado) para interações com polegar (72x72 pixels)
- 16x16mm para interações com dedo indicador (57x57 pixels)
- 7-10mm para casos excecionais (48x48 pixels)

#### 3.7.4. Gestos

---

Inicialmente, as formas de interagir com dispositivos eletrónicos limitavam-se a teclados, rato e comandos. No entanto, com a introdução do iPhone e mais tarde dos tablets, foram introduzidas novas formas de interação através do toque e gestos.

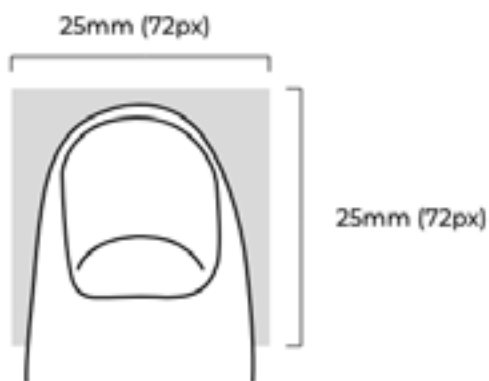


Figura 28 - Dimensões recomendadas para uso com polegares

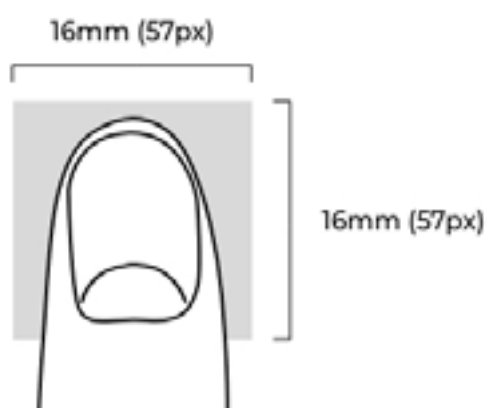


Figura 29 - Dimensões recomendadas para uso com dedo indicador

---

#### Fontes e Recursos Adicionais:

[Finger-Friendly Design: Ideal Mobile Touchscreen Target Sizes - Smashing Magazine](#)

<https://medium.com/@zacdicko/size-matters-accessibility-and-touch-targets-56e942adc0cc>

<https://www.bbc.co.uk/guidelines/futuremedia/accessibility/mobile/design/touch-target-size>

Os gestos apresentam uma forma de interagir que é considerada mais natural e intuitiva de utilizar, por assemelhar-se à interação com objetos reais. Há habitualmente duas razões principais para usar gestos em vez de botões na interação do utilizador com as aplicações:

#### **1. Simplificação da Interação**

Quanto mais uma aplicação usar o controlo por gestos, menos será necessário o uso de botões no ecrã, disponibilizando dessa forma mais espaço para conteúdo. Isto torna a aplicação mais focada no conteúdo permitindo que o utilizador interaja com o que é mais importante sem obstruções ou distrações.

#### **2. Facilidade de Uso**

Um gesto, uma vez descoberto e percebido por um utilizador, pode-se tornar numa parte agradável da experiência do utilizador e pode melhorar a interação ao reduzir o número de passos no fluxo de utilização. Um exemplo, é quando o utilizador pretende apagar itens numa lista. Pressionar um item de cada vez demora muito mais tempo do que um simples deslizar para o lado.

### **Cuidados no uso de gestos**

Se um gesto parece intuitivo para nós não significa que seja intuitivo para outro utilizador e um dos grandes obstáculos de controlo por gestos é a curva de aprendizagem de utilização. Um outro aspeto a considerar é que cada vez que removemos um elemento de interface, como por exemplo um botão, isso implica um aumento da curva de aprendizagem de utilização da aplicação.

Isto acontece porque os gestos não são visíveis, como no caso dos botões. Os utilizadores precisam de descobrir a existência desta opção. Quanto mais dependermos do uso de gestos para interações, mas aumenta a possibilidade de existir confusão.



Figura 30 – Gestos mais comuns

Outros aspetos a manter em mente quando desenvolvemos interfaces baseadas em gestos:

- **Esforço do utilizador aumentado:** nem todos os gestos são naturais, nem fáceis de aprender ou lembrar. Por exemplo, em muitas aplicações um gesto com um dedo tem um resultado diferente do que o mesmo gesto com dois dedos e ainda outro com três dedos ou quatro.
- **Falta de resposta:** em muitos casos, os gestos não deixam nenhum registo da sua trajetória. Isto quer dizer, que se o utilizador usar um gesto poderá não obter uma resposta ou indicação de erro.
- **Falta de consistência:** grande parte dos gestos ainda não são normalizados e consistentes através de todas as aplicações e nem sempre são óbvios para o utilizador. Um gesto simples como deslizar sobre uma mensagem email pode funcionar de forma diferente nas várias aplicações de email.
- **Zoom de imagens.** como já referido, os dispositivos móveis têm um limite de espaço o que torna difícil o utilizador visualizar os detalhes das imagens. Deveremos sempre disponibilizar a possibilidade de *Zoom* nas imagens através dos gestos *Spread* e *Pinch*. Também será necessário indicar que esses gestos estão disponíveis e garantir que a imagem tem a resolução suficiente para se conseguir visualizar o seu detalhe.

## Que gesto usar e quando?

Quando se usa uma interação baseada em gestos na interface das aplicações, é sempre uma boa prática indicar a existência dos gestos, como usar e como se vão comportar. Assim o utilizador terá uma curva de aprendizagem mais pequena e a interface apresentada será mais intuitiva.

Com regra base, sugerimos sempre seguir as indicações de uso de gestos indicadas pelo fabricante:

- Apple iOS - [Gestures](#)
- Google Android - [Types of Gestures](#)

---

## 3.8. PERMISSÕES

---

Devido às diferentes funcionalidades disponíveis nos dispositivos móveis, muitas vezes é necessário solicitar a permissão do utilizador para poder aceder a determinado tipo de conteúdos e funcionalidades. A forma de como abordamos e quando abordarmos o utilizador é muito importante e pode fazer a diferença entre ser intrusivo ou realçar a experiência da aplicação.

### 3.8.1. Conteúdo

---

O conteúdo da mensagem ao pedir permissões é fundamental para o utilizador entender o que estamos a pedir e porquê. Eis algumas boas práticas de como o fazer:

- Explicar o porquê do que estamos a solicitar. Quais são os benefícios para o utilizador? Porque faz sentido partilharem esta informação com a aplicação/entidade?
- Não utilizar frases como “Para personalizar o serviço” ou “melhorar a experiência do utilizador”
- Ser claro e transparente. A que informações a aplicação irá aceder e porquê?

## 3.8.2. Momento de contato

---

O momento em que solicitamos as permissões ao utilizador deve ser oportuno. Se for no momento errado pode gerar confusão ou diminuir o nível de confiança na aplicação.

Solicite as permissões críticas num momento inicial. Os utilizadores já esperam a necessidade de permissões de acesso, por isso se for algo importante, solicite logo.

Evite solicitar permissões totais. Especialmente se não forem imprescindíveis para a experiência inicial.

Solicite as permissões menos importante apenas na altura em que sejam necessárias e não antes. Assim, o utilizador não precisa de disponibilizar as permissões desnecessariamente.

Evite pedidos de permissões no meio de uma tarefa, especialmente se não for relacionado com a tarefa. A reação automática nestes casos é não aceitar, o que pode afetar negativamente a experiência mais tarde.

## 3.8.3. Reversão de Decisão

---

Muitas vezes os utilizadores podem não conceder certas permissões inicialmente. No entanto, mais tarde poderá ser necessário conseguir rever, ou alterar essa permissão para que o utilizador possa conseguir realizar alguma tarefa na aplicação.

De forma a evitar uma experiência negativa, não deve apresentar uma mensagem de erro, mas sim seguir as indicações abaixo:

- Lembrar o utilizador da sua decisão inicial e indicar como pode alterar a permissão.
- Os utilizadores podem ter dificuldade em encontrar a localização correta para alterar as permissões nas definições do dispositivo. Para facilitar esta operação deverá ser apresentado um botão que os conduza à funcionalidade necessária.

---

### Fontes e Recursos Adicionais:

[Designing Effective App Permission Requests - Nielsen Norman Group](#)

---

## 3.9. CACHE OU CONTEÚDOS OFFLINE

---

Devido à mobilidade dos utilizadores na utilização dispositivos móveis, há muitas ocasiões em que a velocidade de ligação a dados móveis pode variar ou mesmo desaparecer. Nestas situações poderá ser muito frustrante para utilizador, que esteja a meio de uma tarefa, não conseguir completá-la por não ter ligação de dados móveis (por exemplo: a preencher um formulário).

Sempre que possível é recomendada a disponibilização de algumas funcionalidades base em modo offline. Abaixo apresentamos alguns casos onde poderá fazer sentido:

### 1. Formulários

Sempre que seja necessário um utilizador preencher um formulário e desde que não haja conflitos sobre questões de segurança (por exemplo: sessões), é recomendável guardar os dados preenchidos para que o utilizador consiga terminar o preenchimento do formulário, mesmo se falhar a ligação de dados móveis. Uma outra alternativa é guardar os dados para mais tarde, quando o utilizador tiver novamente ligação a dados móveis, conseguir continuar o preenchimento do formulário no mesmo ponto.

### 2. Exploração da aplicação

Em alguns casos, poderá fazer sentido o utilizador conseguir navegar pelos conteúdos da aplicação, mesmo quando está sem dados móveis. Quando for caso disso, recomenda-se reduzir as funcionalidades disponíveis e indicar sempre visualmente através de botões que se encontra em estado offline (isto é, sem dados móveis) ou apresentando uma notificação de funcionalidades reduzidas.

### 3. Documentação

Se a aplicação desenvolvida disponibilizar documentação ao utilizador, essa documentação deverá poder ser consultada mesmo sem dados móveis. Nestes casos deverá ser apresentada, ao utilizador, a opção de poder escolher quais os documentos que pretende “descarregar” para consulta offline.

### 4. Utilidades

Caso a aplicação tenha funcionalidades que façam sentido continuar disponíveis sem dados móveis (por exemplo, tirar fotografias ou digitalizar documentos), recomenda-se disponibilizar apenas as funcionalidades necessárias e não todas as funcionalidades da aplicação. A necessidade de armazenamento interno do dispositivo para os dados irá crescer à medida que se pretendam disponibilizar mais funcionalidades em modo offline.



## 3.10. PAGAMENTOS

---

Nos campos de seleção dos formulários, é boa prática apresentar apenas as opções que estão disponíveis para utilização. Por exemplo, caso só seja possível pagar com cartão de crédito ou Paypal, não faz sentido apresentar a opção para pagar com *MBWay* ou cartão de débito. A apresentação de opções indisponíveis poderá confundir o utilizador sobre a opção a seleccionar.

---

## 3.11 PESQUISA

---

A opção de pesquisa dentro de uma aplicação móvel é fundamental e nos casos de aplicações muito complexas, são a melhor forma para o utilizador encontrar o que procura.

### 3.11.1. Barra de Pesquisa

---

Esta barra deve ser claramente visível e o seu funcionamento deve ser fácil de entender. A barra, sempre que possível, deverá ser um campo de input completo e deve-se evitar o uso de apenas um ícone de pesquisa.

Esta recomendação, surge pelo facto de que, quando se usa apenas o ícone de pesquisa para indicar a existência desta opção, ao seleccionarmos o ícone estaremos a esconder o contexto obrigando o utilizador a efetuar mais interações para conseguir realizar a pesquisa:

1. Clicar no ícone
2. Esperar que apareça o campo
3. Clicar no campo de input
4. Preencher o conteúdo da pesquisa.

Ao apresentar o campo completo, o utilizador precisa de apenas duas interações:

1. Clicar no campo de input
2. Preencher o conteúdo da pesquisa

Se a pesquisa for a forma predominante de navegação, deverá ser apresentada em primeiro plano ao entrar na aplicação:



*Figura 31 - Exemplo campo de pesquisa numa posição predominante*

#### 3.11.2. Usar ícone de lupa

---

Uma barra de pesquisa deve sempre ser acompanhada por um ícone de lupa. Os utilizadores já estão acostumados a este padrão e ajuda a identificar mais facilmente o campo. O ícone deverá ser o mais simples possível. Quanto mais detalhe apresentar o desenho do ícone, mais complicada será a interpretação do utilizador.



*Figura 32 - Exemplo de ícone simples de lupa*

### 3.11.3. Escolher um bom padrão

---

O uso de um bom padrão da funcionalidade de pesquisa pode reduzir o esforço do utilizador, especialmente quando o utilizador necessita de realizar ações repetitivas, ou em situações onde precisa de usar precisão.

Os dados pré-preenchidos automaticamente num formulário com dados sugeridos, podem em muitos casos ser precisos o suficiente para o utilizador não necessitar de efetuar qualquer preenchimento. Um bom exemplo é o uso de georreferenciação para preencher a localização atual, quando se pretende obter direções para outra localização.

### 3.11.4. Sugestões Automáticas

---

Provavelmente, o padrão de pesquisa mais útil que surgiu na Web 2.0 foi a apresentação de sugestões automáticas. Estas sugestões são usadas para reduzir a entrada de dados e fornecer resultados imediatos.

Deverá sempre fornecer sugestões automáticas da forma mais rápida possível, como após introduzir um terceiro carácter para fornecer valor imediato e reduzir o esforço de entrada de dados no campo de pesquisa.



Figura 32- Exemplo do uso de sugestões automáticas

### 3.11.5. Pesquisa Recentes e Guardadas

Mesmo quando os utilizadores estão familiarizados com o uso de pesquisa, a pesquisa exige que eles recuperem informações da sua memória. É por isso que as aplicações devem armazenar todas as pesquisas recentes para fornecer esses dados ao utilizador na sua próxima pesquisa. Assim reduzimos o tempo e o esforço que utilizador precisa para procurar novamente o mesmo item.



Figura 33 - Exemplo de pesquisas guardadas

Esta opção funciona particularmente bem em aplicações onde o utilizador precisa de repetir a mesma pesquisa.

### 3.11.6. Pesquisa por voz

---

Por vezes surgem situações em que, para o utilizador, se torna difícil escrever no telefone e é propenso cometer erros. Em alguns casos, para utilizadores com alguma limitação momentânea ou mais definitiva, a escrita pode limitar a acessibilidade da aplicação. Nestes casos a opção de poder realizar a pesquisa por voz apresenta-se como uma boa alternativa.

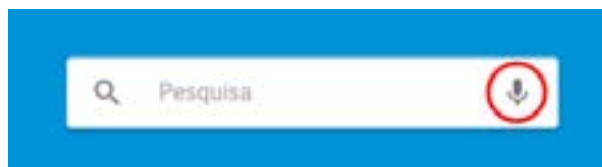


Figura 34 - Exemplo do uso de pesquisa por voz

Basicamente, em qualquer aplicação que tenha um campo de pesquisa, pode ser adicionada a opção para efetuar a pesquisa por voz. Este padrão é uma boa opção para aplicações que vão ser usadas em movimento.

#### Alerta

Embora a opção de pesquisa por voz seja uma boa opção, ainda há limitações e está dependente da qualidade do programa de reconhecimento de voz disponível no equipamento.

---

## 3.12. BOTÕES PRIMÁRIOS DE CALL-TO-ACTION

---

A melhor forma de ajudar o utilizador a conseguir completar uma determinada tarefa é apresentar indicadores de como interagir com a aplicação. Os botões primários conhecidos como *call-to-action* são considerados a melhor opção. No entanto, para uma *call-to-action* conseguir funcionar bem como ajuda para o utilizador, deverão ser seguidas as seguintes boas práticas.

---

#### Fontes e Recursos Adicionais:

<https://uxplanet.org/mobile-ux-design-user-friendly-search-51e5f78f5a1e>

<https://blog.appsee.com/the-basics-of-creating-the-best-search-ux-on-mobile/>

<https://www.raywenderlich.com/726-20-best-practices-for-mobile-app-search>

### 3.12.1. Tamanho e Cor

---

O botão de ação principal deverá ter sempre destaque relativo aos restantes elementos da página. Um tamanho maior indica que é o elemento mais relevante e uma cor que contrasta com os restantes elementos da página permite chamar mais a atenção do utilizador. (Figura 35)

O *call-to-action* principal é normalmente associado à opção de avançar para o próximo passo numa tarefa ou então terminar uma tarefa, por isso é aconselhável a sua apresentação no final do conteúdo da página conforme a figura abaixo. (Figura 36)



*Figura 35 – Exemplo de call-to-action com destaque relativo aos restantes elementos*



*Figura 36- Exemplo de call-to-action que está no fundo da página e mantém-se fixo*

Nos casos em que o conteúdo da página exige scroll, o *call-to-action* deverá ser apresentado no fundo da página de forma sobreposta ao seu conteúdo. O botão deverá manter-se sempre na mesma posição, mesmo quando efetuar scroll. Desta forma, a ação principal estará sempre disponível para o utilizador.

---

**Fontes e Recursos Adicionais:**

<https://tubikstudio.com/ux-practices-8-solid-tips-on-cta-button-design/>

<https://uxmovement.com/mobile/5-techniques-to-make-mobile-call-to-action-buttons-intuitive/>

<https://usabilla.com/blog/floating-action-button-bad-ux-design/>

# 4.

## Referências



- <https://www.gov.uk/service-manual/>
- <https://www.nngroup.com/>
- <https://uxplanet.org/>
- <https://www.uxmatters.com/>
- <https://www.uxpin.com/studio/blog/>
- <https://uxdesign.cc/>
- <https://www.smashingmagazine.com>