# **Shell** Design Guidelines for Mobile and Tablet

Version 4.1

## Contents

1	Introduction	4
	1.1 Philosophy	5
	1.2 Further guidance	6
2	Mobile and tablet context	7
	2.1 Introduction	8
	2.2 Device types	9
	2.3 Operating Systems (OS)	9
	2.4 Native apps	11
	2.5 Mobile sites and web apps	12
	2.6 Assistive technologies	13
3	Vision	14
4	Principles	15
	4.1 Shell engineered	16
	4.2 A delight to use	17
	4.3 Inspires me to do things differently	20
	4.4 Always connected	

5	St	yle	24
	5.1	First impressions and inside impressions	25
	5.2	Pecten	27
	5.3	Grid	31
	5.4	Colour	38
	5.5	Typography	47
	5.6	Iconography	53
	5.7	Imagery	59
	5.8	Animation and video	67
	5.9	Charts, tables and diagrams	70
		, 3	10
6	In	teractions	74
6	In <sup>:</sup> 6.1	teractions	<b>74</b> 75
6	In 6.1 6.2	teractions Gestures Ergonomics	75 77
6	In 6.1 6.2 6.3	teractions Gestures Ergonomics Hit areas	75 77 80
6	In 6.1 6.2 6.3 6.4	teractions Gestures Ergonomics Hit areas Orientation	75 77 80 81
6	In 6.1 6.2 6.3 6.4 6.5	teractions Gestures Ergonomics Hit areas Orientation	74 75 77 80 81 83
6	In 6.1 6.2 6.3 6.4 6.5 6.6	teractions Gestures Ergonomics Hit areas Orientation Zoning Content types	74 75 77 80 81 83 84
6	In 6.1 6.2 6.3 6.4 6.5 6.6 6.7	teractions Gestures Ergonomics Hit areas Orientation Zoning Content types Interaction patterns	75 77 80 81 83 84 88

7	First impressions	107	10
	7.1 Launch icon	. 108	
	7.2 Loading screen	. 112	
	7.3 Home screens	. 116	
8	Components	122	
	8.1 Headers	. 123	
	8.2 Toolbars	. 141	
	8.3 Footers	. 159	
	8.4 Titles and section dividers	. 163	
	8.5 List components	. 168	
	8.6 Form elements	. 173	
	8.7 Help	. 178	
	8.8 Notifications	. 188	
	8.9 Maps	. 196	
	8.10 User comments	. 201	
9	Templates	204	
	9.1 Contents list	. 205	
	9.2 Articles	. 212	
	9.3 Forms	. 225	

0 Common functionality	232
10.1 Customisation	233
10.2 Search	244
10.3 Registration	251
10.4 Media players	255
10.5 Sharing	260
10.6 Linking	263
10.7 Filtering and sorting	272

## **1** Introduction

This document provides principles, patterns and guidelines for the interaction and visual design of native apps and sites for mobile and tablet devices.

Following these guidelines will ensure that your apps and sites represent the Shell brand, and that the user has a coherent experience across Shell products and services.

In this section

1.1	Philosophy	.5
1.2	Further guidance	.6

## 1.1 Philosophy

The aim of the guidelines is to provide a starting point for design. The designs shown in this document are to illustrate our design thinking rather than specific detailed designs.

The guidelines are intended to be:

## Inspirational

A platform to execute great ideas

## Scalable

Guidelines that can grow, flex and change as technology and adoption changes

## Usable

Can be understood and deployed easily

## Efficient

Reuses device or operating system patterns when appropriate to reduce development time and cost and allow greater scalability across platforms

They provide a bridge between Shell's core visual identity principles and design guidelines specific to mobile and tablet operating systems. See Figure 1.

## 1.2 Further guidance

For the core Visual Identity principles, examples and artwork please refer to the <u>Shell VI for Communications Policy version 2</u> available from <u>www.brandcentral.shell.com</u>.

For guidance on designing for Apple iOS please refer to the iOS <u>Human Interface Guidelines</u> available from developer.apple.com.

For guidance on designing for Google Android please refer to the <u>Android Design Principles</u>, <u>Style</u>, <u>Patterns and Building Blocks</u> available from developer.android.com/design.



**Fig 1** Integration of these guidelines within Shell's VI, and operating system guidelines

## 2 Mobile and tablet context

#### In this section

2.1	Introduction	8
2.2	Device types	9
2.3	Operating Systems (OS)	9
2.4	Native apps 1	1
2.5	Mobile sites and web apps1	2
2.6	Assistive technologies 1	3

## 2.1 Introduction

### 2.1.1 Context of use

Mobile and tablet devices are used for a wide variety of purposes and contexts.

They may be used to learn, to chat, to shop, to complete an urgent social or business task, or simply to pass the time. They may be used at home, at work or travelling. While using the device users may be speaking with other people, attending a business or entertainment event, or watching or listening to other media such as television or radio.

### 2.1.2 Attachment

Users have a more emotional attachment to these devices than they do to their desktop and laptop computers. Smartphones in particular become tightly embedded in their users' personal, family, social and business lives.

### 2.1.3 Capabilities

Both mobiles and tablets have touchscreens that users interact with through gestures. This makes the devices feel more responsive, engaging and natural than most other digital devices. Mobile and tablet devices come in a range of physical and screen sizes, and with a range of storage, processing, sensing and communication capabilities.

### 2.1.4 **Apps**

In addition to browsing websites, users download and use dozens of free and purchased apps – although only a few of those apps will hold a user's attention for long. The simplicity of this process, and the low prices for apps, mean that mobile and tablet users download many more apps than users of desktop and laptop computers.

## 2.2 Device types

This document is intended to cover devices with a range of sizes and capabilities. We divide the devices broadly into **mobile** and **tablet**.

### 2.2.1 Mobile

These devices are pocket sized – ranging from 100 to 150mm tall, 50 to 90mm wide, 9 to 12mm thick and 120 to 180g in weight.

Users carry their mobile with them everywhere they go and often use them while standing and walking. They tend to have a strong emotional attachment to their mobile devices, which are personal and rarely shared.

For many users, mobile devices are still primarily used for messaging and voice calls.

### 2.2.2 **Tablet**

These devices are bag sized – ranging from 190 to 240mm tall, 120 to 190mm wide, 8 to 12mm thick and 400 to 620g in weight.

Users often keep their tablet at home or at their office and use it while sitting or lying down. They have a more functional relationship with their tablet device. Tablets are often shared within families and businesses. For many users, tablet devices are like smaller and more portable laptops, and are used in combination with, rather than in place of, a mobile phone.

### 2.2.3 Touchscreen and gestures

While some mobiles and tablets have keypads, the majority have touchscreens that users interact with through gestures.

This makes them feel more responsive, engaging and natural than other digital devices. See 6.1 Gestures

### 2.2.4 Connected

Both mobile and tablet devices keep users connected through voice and video calls, messaging, browsing and data access over 3G and Wi-Fi connections.

### 2.2.5 Sensors

The built-in sensors enable many of the most engaging experiences with mobile and tablet devices.

These may include a still and/or video camera, ambient light sensor, microphone, GPS, compass, accelerometer, gyroscope and proximity.

## 2.3 Operating Systems (OS)

The principles, patterns and guidelines included in this document are intended to apply to apps and sites designed for a wide range of mobile and tablet. However, some guidelines do reference specific mobile and tablet operating systems and their associated design and style guides.

These operating systems are **iOS 5** from Apple Computer and **Android 4** from Google.

The associate design and style guides are:

- iOS Human Interface Guidelines
  <u>developer.apple.com/library/ios/DOCUMENTATION/Us</u>
  <u>erExperience/Conceptual/MobileHIG/MobileHIG.pdf</u>
- Android Design Principles, Style, Patterns and Building Blocks

developer.android.com/design

More systems

Future versions of this document may include specific guidelines for additional mobile and tablet operating systems and design and styles guides such as Windows 8 and the Metro Design Language from Microsoft.

## 2.4 Native apps

One of the most successful features of recent mobile devices has been the concept of the app. The best apps give users rewarding experiences that inspire them to do something new or make an existing service easier and more enjoyable to use – at home, at work or socially.

Native apps are designed and developed for a specific operating system and device category, and downloaded and updated through a built-in storefront such as Apple App Store or Android Market.

Native apps can provide a fluent experience that fits closely with the visual design and interaction design style of an operating systems and device. They can make optimum use of the device capabilities, such as its display, storage, notifications, sensors and communications. And the built-in storefront takes care of purchases, downloads and updates.

But native apps need to be written from scratch for each operating system and device category and each release. They must pass through an approval process to reach the Apple App Store or Android Market, and those stores take a percentage of any revenue. Depending on the design of the app, users may need to download regular updates with new features, functionality as well as content stored within the app.

### 2.4.1 When to create a native app

Native apps are the best approach when you:

- ✓ Have the time and budget to build for multiple operating systems and device categories.
- ✓ Need to exploit native device capabilities camera, communications, sensors, etc.
- ✓ Want to integrate smoothly with built-in data calendar, contacts, etc.
- $\checkmark$  Want to present rich, interactive data graphics.
- ✓ Need to provide a good offline experience.
- Expect users to search for your brand in the Apple App Store or Google Market.

## 2.5 Mobile sites and web apps

One of the most common activities for users on mobile and tablet is to use their browser to access websites. The most effective and engaging mobile websites are 'web apps' designed to look and feel like native apps.

Mobile sites are built using standard web technologies (HTML5, CSS and JavaScript) so the same site can be accessed on a wide range of operating systems and devices. When built as web apps, mobile sites can be tailored to fit quite closely with the visual design and interaction design style of each operating system and device category. And they can be accessed and updated outside of the Apple App Store or Android Market.

But mobile sites have limited access to the full capabilities of a device, including cameras, sensors and user data such as calendar and contacts. They cannot entirely simulate the fluent interactions of native apps. And while they can be built to run offline, they cannot match a native app's use of local state and storage.

Websites and web apps for tablet devices While most existing websites are directly usable on tablet devices, the best sites use flexible web page layouts to ensure that they provide a tailored experience for tablet users.

## 2.5.1 When to create a mobile site or web app

Mobile sites and web apps are the best approach when you:

- ✓ Have strong assets and skills in web technologies
- Must provide a solution for multiple operating systems and device categories with limited time and budget
- ✓ Do not need to exploit native device capabilities camera, communications, sensors, etc.
- Need continuous access to online data whereas native apps can store data on the device
- ✓ Have users who want to share data or content through social media or business collaboration tools
- Expect users to look for mobile and tablet solutions through general web search or through your main website

## 2.6 Assistive technologies

Mobile and tablet devices and operating systems include a wide range of accessibility features. This has allowed many more users to take advantage of these devices. Third-party apps can also turn mobiles and tablets into assistive devices.

Shell sites and apps for mobile and tablet should support users who have enabled the accessibility options on their device.

### 2.6.1 Legibility

Both iOS 5 and Android 4 allow users to increase the default system font size and to zoom the contents of the browser or their entire screen. The iOS White on Black feature also provides a higher contrast display. See section 5.4.2 Using colour for a guide to creating the right level of contrast.

### 2.6.2 Voice input and output

Both iOS 5 and Android provide voice output and audio feedback through VoiceOver, Speak Selection and Explore-By-Touch. Both also support speech-to-text input and voice control through Siri and Voice Actions.

### 2.6.3 **Apps**

Many accessibility apps are available for mobile or tablet.

For example, Proloquo2Go from AssistiveWare can turn an iPhone or iPad into a text-to-speech generator for people who have difficulty speaking.

Magnify from Apps Lab can turn an Android device into a magnifying glass for people who have difficulty reading small text or seeing small objects.

## **3** Vision

Our sites and apps for mobile and tablet provide our customers with an experience authentic to our values.

They are precision engineered, delightful to use and make connecting with Shell better, faster and smarter.

## **4** Principles

The patterns and guidelines in this document are founded on twelve overall experience design principles. These principles describe the essential goals and attributes of Shell sites and apps for mobile and tablet.

The principles are rooted in Shell's brand values and visual identity, the mobile and tablet context and the guidelines promoted by major mobile and tablet device and operating system vendors.

You can use the principles both as a guide for designing new mobile and tablet products and as a yardstick for measuring existing products. Where this document does not provide specific rules or recommendation, please refer to the principles.

There are twelve principles divided into four groups. The principles are expressed as statements that a user might make after experiencing a product.

- 4.1 Shell engineered
  - Feel precise, reliable and authentic
  - Look like the other Shell sites and apps I use
- 4.2 A delight to use
  - Familiar and easy to learn
  - Make the best of my screen
  - Speak my language
  - Focused on just the things I need to do
- 4.3 Inspires me to do things differently
  - Obvious why I want to carry it with me
  - Understand where I am and what I'm doing
  - Help me discover new possibilities
- 4.4 Always connected
  - Use my existing Shell identity and data
  - Combine well with other Shell sites and apps
  - Like a pocket version of the site I use at my desk

## 4.1 Shell engineered

Your sites and apps give me confidence and represent everything I value about Shell. I know they will never let me down.

**Feel precise, reliable and authentic** Your sites and apps always feel reliable and well made.

The information they provide is useful and accurate.

Look like the other Shell sites and apps I use Your sites and apps use the same colours and shapes as everything else I see from Shell.

I know immediately they are from Shell.

They look perfectly finished and show that you have paid attention to all the details.

Shell Visual Identity Communications Policy ver	rsion 2.0   Policy introduction		06
KEY VISUAL ELEM	ents		
Our visual identity system has bee Working together, these mandator	n created to express Shell's cl ry elements create powerful c	aracter. mmunications.	
Energy and the second s	Primary lock-up The primary lock-up positions the Pecten in an impactful way.	Grid The grid provides a vitil, precise and engineered foundation upon which communications built.	Colour Concolor plates leverages yellow and red [deab]y in 0.2 Intoil on white to provide and kit concolor of the Stell band.
<sup>identy</sup> system. Futura Light Futura Book Futura Medium <b>Futura Bold</b>			
Typography Futura is a typeface based on precise angles and circles that is effective both up close and at a distance.	Photography Photography is realistic and flexib to communicate appropriate mes whilst using Shell's yellow and re	Illustration Illustration is useful for expressing rages concepts. It features a technically is colours. accurate style and solid Shell colours.	Graphic language The graphic language is one of squares and nectorgles to complement the grid and express Shell's engineering precision.
CONTENTS WELCOME	POLICY VIS	UAL APPENDIX	CONTACT

They feel honest, professional and relevant to me.



## 4.2 A delight to use

Using Shell sites and apps is a pleasure. They are always easy to read and simple to use. Wherever I am and whatever I'm doing it's effortless.

Speak my language

#### Familiar and easy to learn

Your sites and apps are easy to learn because they work just like the built-in apps I use every day.

The controls always do what I expect them to.

I can try the gestures I know to find out how things work.

Your sites and apps fit lots of useful information on each screen, but

Make the best of my screen

the text and images are always clear and easy to read.

I can glance at the screen to see the most important information.

They respond well if I hold the screen upright or on its side.

Focused on just the things I need to do



Your sites and apps work in one of the languages I know. They follow the language and region settings on my device. They respect my cultural conventions in the way they talk to me. I can try the gestures I know to find out how things work.



Your sites and apps are always focussed on the few things I really want them to do.

Screens always show the most important information and controls first.

They make common chores faster and more efficient.



## 4.3 **Inspires me to do things differently**

I know immediately how your sites or apps can help me. They demonstrate your innovation and encourage me to try new things.

#### Obvious why I want to carry it with me

Your sites and apps have functions that are clearly useful when I'm away from my desk.

When I first see one of them I know immediately when and how I will want to use it.

They are easy to access and download.



#### Understand where I am and what I'm doing

Your sites and apps for mobile and tablet use the features of my device to predict my needs and keep things simple.

They respond well if I'm distracted or interrupted in the middle of doing something.

You store important information so I can keep working offline.



Help me discover new possibilities

Your sites and apps help me do things I never thought I could, in ways I never imagined.

They make my work more productive and my life more enjoyable. The more I use them, the more important they become to me.



## 4.4 Always connected

Your sites and apps keep me plugged in to Shell no matter where I am. You know me and use what you know to make things better across all my devices.

#### Use my existing Shell identity and data

Your sites and apps don't ask me to create a new account or profile; they just use the one I already have.

They remember me and use what they know about me to make things easier and more relevant.

If I set a preference in one sites or app, the others know too.



#### Combine well with other Shell sites and apps

Your sites and apps for mobile and tablet compliment the other Shell websites and applications that I use

They link together well so I can move easily between them to find the tools and information that I want

If you send me to a non-mobile website I can use it on my device



**12. Like a pocket version of the app I use at my desk** Your sites and apps allow me to carry around the essential parts of the tools and information I use at my desk.

They use the same concepts and language so I can switch between the two without getting confused.



## 5 Style

These style guidelines provide additional support for how styles defined in the <u>Shell VI for Communications Policy 2011</u> should be applied to apps and sites for mobile and tablet. They expand on how the policy should be applied, as well as providing modifications on the guidelines to satisfy the mobile and tablet context.

#### In this section

5.1 First impressions and inside impressions	25
5.2 Pecten	27
5.3 Grid	31
5.4 Colour	38
5.5 Typography	47
5.6 Iconography	53
5.7 Imagery	59
5.8 Animation and video	67
5.9 Charts, tables and diagrams	70

# 5.1 First impressions and inside impressions

When using first impressions and inside impressions the rules stated in the <u>Shell VI for Communications Policy 2011</u> and the <u>Shell</u> <u>Social Media Visual Identity Guidelines Version 1.0</u>, must always be followed unless outlined in this document.

## 5.1.1 Using first impressions for mobile and tablet devices

#### **First impressions**

First impressions can be defined as when the user first sees a Shell piece of communication on their device. This communication includes:

- Launch icons see figure 5.1a
- Loading screen see figure 5.1a
- Home screen/ landing screen

It is important that Shell's visual assets, like the Pecten, colour proportions and imagery are strongly enforced in these areas of communication. This will ensure that the user's first impression of the Shell brand is one that is distinctive and recognisable.

The Pecten must be present on all types of first impression communications.



Loading screen

5.1a First impression examples

25

iOS launch icon

#### Inside impressions

Inside impressions can be defined as everything the user sees after the first impression. This communication includes:

• Any pages within a site or app excluding loading and home/landing screens see figure 5.1b

Shell visual identity rules must still be maintained throughout inside impression. However greater flexibility is allowed within the app or site, such as the use of the Shell supporting colour palette.



#### **Article screen**

.III SHELL 3G	17:20		Ē
<b>Â</b> C	ontent	S	
News and spe	eches		>
Shares			+
Quarterly resul	>		
Dividend	>		
Bonds and cre	>		
Our strategy	>		
Publications a	>		
Images	+		
Q ★ Search Favourites	CC Share	Ç Settings	● ● ● More

5.1b Inside impressions

### 5.2 Pecten

The Pecten is one of the worlds most recognised brand symbols and is key to making the experience feel precise and reliable.



5.2a The Pecten

### 5.2.1 Using the Pecten

The use of the Pecten must always follow the rules stated in the Pecten and Primary Lock-up section of the <u>Shell VI for</u> <u>Communications Policy 2011</u> and the <u>Shell Social Media Visual</u> <u>Identity Guidelines Version 1.0</u>, Elements, unless outlined in this document.

#### **First impressions**

- The Pecten should always be shown.
- Where possible it should follow the 50% optimal exclusion rule.

#### **Inside impressions**

- The number of instances of the Pecten should be kept to a minimum.
- The optimal exclusion zone for the Pecten should be used where possible, if this can't be achieved then the alternative exclusion zone should be used, followed by the minimum exclusion zone.



**5.2b** Pecten exclusion zones, with the minimum digital exclusion zone on a mobile device header

### 5.2.2 Design rules

General

 $\checkmark$ 

#### Minimum size of the Pecten is 30 x 32 pixels

#### Website

 Always display the Pecten in the top left hand section of the header area See figure 5.2b.

This provides consistency with the Shell website.

#### Use the Pecten for a short cut to return to the home screen

This maintains consistency with the Shell website navigation.

#### Mobile apps

✓ The Pecten does not need to be displayed on every page The user will always have viewed the Pecten as part of the first impression, therefore it is not required on all screens of an app. Priority must instead be given to key user information because of the limited screen estate on mobile devices.

**Use the 30% Minimum Exclusion Zone for the Pecten** This allows the Pecten to fit comfortably within a standard header area and maintain the minimum Pecten size. See figure 5.2b. For further guidelines see Section 7.1 Headers.



#### **Tablet apps**

✓ The Pecten can be displayed in the top left area of the toolbar. See figure 5.2c.

The additional screen space on a tablet compared to mobile provides room for additional branding elements. This placement is consistent with the Shell website.

✓ The Pecten must follow the rules for the Alternative Exclusion Zone of 30% - 50%

The additional screen space allows for an increase in the size of the Pecten. See figure 5.2c.

Ensure there is a padding area below the header bar This will provide additional white space for the branding colour strip. See figure 5.2d.



**5.2d** Pecten and the alternative digital exclusion zone with the brand colour strip

## 5.3 **Grid**

### 5.3.1 **Purpose of Grid**

The grid is used to align all the components and typographical elements in any layout. By applying the grid a website or app will feel consistent while making the best use of the available screen space.

### 5.3.2 Using the grid

- The grid will form the starting point for all design layouts.
- The grid is made up of equally spaced columns and rows
  - On a mobile screen each column or row is 2.5% of the total screen width
  - On tablet screen each column or row is 1.25% of the screen width

This will provide consistency and uniformity across varying screen dimensions. See figure 5.3a and 5.3b for examples.

- It is used to help align all elements on the screen, including the header and toolbar. See figure 5.3c and 5.3d for examples.
- To maintain the Shell graphic language a perfect set of squares are formed between the columns and rows for 90-degree right angle alignment. See figure 5.3g – 5.3j for examples.
- Creation and alignment of all elements against the grid must always follow the rules stated in the <u>Shell VI for</u> <u>Communications Policy 2011</u> – Graphic language.



Fig 5.3b Grid for tablet device displaying the alignment for columns

Horizontal alignment



**Fig 5.3c** How the grid is used to align elements on a mobile device (portrait orientation)



**Fig 5.3d** How the grid is used to align elements on a tablet device (landscape orientation)

### 5.3.3 **Design rules**

 Align all components to either the horizontal/vertical mid point of a grid, or the top left corner of the screen See figure 5.3g and 5.3j.

This will be the starting point for the alignment of all screen components.

- Each column and row should be a minimum of 2.5% of the total screen width on a mobile screen
- Each column and row should be a minimum of 1.25% of the total screen width on a tablet screen
- If designing for screen sizes under 320px wide then use a minimum of 5% for each column and row
- Use the grid to evenly divide the width of the screen into equal parts, when viewed in its current orientation
- Following the baseline of a row isn't necessary for all typography

Leading will vary; instead it is used to create a consistent design pattern for all elements and components. See figure 5.3e.

All screen elements created should align to grid at a 90degree right angle following the Shell graphic language See figure 5.3f.



Fig 5.3e Variation in typography leading in relation to the grid



**Fig 5.3f** Grid alignment in relation to the graphic language principles

#### **Option 1** Centre alignment



The diagram illustrates how it is used to align elements on the screen outwards from the central axis





Fig 5.3h Applying design components aligned from the central axis
#### Option 2 Left and top alignment



Fig 5.3i The grid on a tablet device. The diagram illustrates how it is used for top left alignment

Fig 5.3j Applying design components aligned from the top left corner alignment

# 5.4 **Colour**

# 5.4.1 **Purpose of colour**

Colour is a powerful quality of the Shell brand. Using the Shell colour palette correctly will help to provide quick identification and drive brand recognition.

# 5.4.2 Using colour

The use of colour must always follow the rules given in the <u>Shell VI</u> for Communications Policy 2011 – Colour and the <u>Shell Social</u> <u>Media Visual Identity Guidelines Version 1.0</u>. Please refer to this for full detail of colour references and values.

The Shell colour palette is broken down into three sections:

#### • Shell primary palette

This will be the most frequent, prominent and visible palette throughout the designs. It allows the user to easily recognise they are in a Shell site or app.

#### • Shell supporting palette

This will be kept to a minimum, unless designing for a subbrand. It will help prevent visual clutter in an environment when screen space is limited.

#### • Shell neutral palette

This will be used as a complementary palette to present and differentiate information. Neutrals should never dominate or be used in large areas unless it is within the navigation elements.

Primary palette	Supporting pale	supporting palette		web safe)
Shell yellow HTML F7D117	Shell dark blue HTML 003882	Shell purple HTML 611759	Black HTML 000000	Grey 5 (~ 15% grey) HTML D5D5D5
Shell red HTML D42E12	Shell mid blue HTML 0099BA	Shell lilac HTML B491BA	Grey 1 HTML 333333	Grey 6 HTML D7D7D7
Shell white	Shell light blue	Shell orange	Grey 2 (~ 80% grey)	Grav 7
HTML FFFFFF	HTML 8CCCD9	HTML DE8703	HTML 666666	HTML FOFOFO
	Shell light green HTML BAD405	Shell sand HTML EDDDC1	Grey 3 (~ 40% grey) HTML 999999	Grey 8 HTML F7F7F7
Fig 5.4a Shell colour palettes	Shell dark green HTML 00824A	Shell brown HTML 70331F	Grey 4 HTML CCCCCC	Grey 9 HTML F8F8F8

#### **First impressions**

- Adhere to the rules given in the Colour section of the Shell Visual Identity Communications Policy, in particular the 2:1 colour ratio of Shell red and yellow See figure 5.4b, scale A. For additional design examples, please see Section 7, Launch icons and Loading.
- The principle colour is Shell white this occupies 50% of space estate. See figure 5.4b, scale A.
- The supplementary colour is Shell yellow this occupies 30% of screen estate. See figure 5.4b, scale A.
- The accent colour is Shell red this occupies 20% of screen estate. See figure 5.4b, scale A.

#### **Inside impressions**

- The principle colour is Shell white this occupies 60% of space estate. See figure 5.4b, scale B. It is important given the context of use and available screen space that information is presented in a clear and visible format, achieved through the use of this colour, generally used as the screen background.
- The supplementary colour is Shell yellow this occupies 30% of screen estate. See figure 5.4b, scale B.



**Fig 5.4b:** Colour re-allocation for the primary palette for **first (A)** and **inside impressions (B)** 

This colour will be used to create contrast and visual interest on key areas and sections.

 The accent colour is Shell red – this occupies 10% of screen estate. See figure 5.4b, scale B. The limited use of this colour will distinguish key elements and the information on the screen.

# • The neutral palette is used for main body text, supporting information and controls

Most written information uses the neutral palette, along with toolbars and icons. It should not be used to present crucial or key call to actions that could easily be over looked. It can be used as a supporting call to action, such as 'cancel'.

# • The supporting palette is used for separating information when it can't be achieved through the primary and neutral palette

A design can require more colours. This is especially true for content rich sites and apps where visually separating content becomes important. The use of these supporting colours should be kept to a minimum, unless the palette colour is a sub-brand of the app or site. This will help prevent visual clutter on screen and ensure the focus is on key call to action information.

#### **Colour contrast**

Colour contrast is vital to ensure that all users fully understand and distinguish information on screen devices. It is recommended by the WCAG that colours must pass a contrast test for acceptable legibility. This has driven the key text and background colour decisions within this guide.

#### Reference: http://www.w3.org/TR/WCAG/.

 All colours developed for Shell sites and apps must, as a minimum, pass the AA standards for an acceptable colour contrast test as outlined by the WCAG

See figure 5.3c for an overview of colour combinations. Colours are also tested for passing the AAA standard. This is an optimal recommendation. A successful pass on a colour contrast test is not dependent on the screen size of a device, the pixel density or the screen resolution.

• A colour combination may still be used if the font size exceeds 18pts

The colour contrast threshold is dependent on font size. The larger the font size the lower the colour threshold value.

# • For a pass both the colour difference and the brightness difference must exceed their threshold

This means the colour is over the acceptable threshold recommended by the WCAG and is safe to use in any on screen designs.

• A fail occurs when neither value exceeds its threshold

This means that the colour combination will not be legible to some users and should be avoided. If the colour combination is still required, try increasing the font size to plus 18pts and check the contrast again.

Reference <u>http://snook.ca/technical/colour\_contrast/colour.html</u>

Colour 1	Colour 2	WCAG 2 AAA Compliant	WCAG 2 AAA Compliant (18pt+)	WCAG 2 AA Compliant	WCAG 2 AA Compliant (18pt+)
Shell yellow	Shell red	FAIL	FAIL	FAIL	PASS
Shell yellow	80% grey	FAIL	PASS	PASS	PASS
Shell yellow	Shell white	FAIL	FAIL	FAIL	FAIL
Shell red	80% grey	FAIL	FAIL	FAIL	FAIL
Shell red	Shell white	FAIL	PASS	PASS	PASS
Shell white	80% grey	PASS	PASS	PASS	PASS

Fig 5.3c Table illustrates the primary palette colours and neutral palette contrast analysis

Alternative Tools for colour checking

<u>Colour Contrast Analyser</u> - A downloadable Windows application

<u>Juicy Studio: CSS Colour Contrast Test</u> - web-based tool to check CSS code for appropriate contrast

• Ensure colour consistency when supplying image files Colour profiles in Photoshop should be set to "Monitor RGB - Display". This means that no colour profile will be attached as the use of colour profiles could cause unwanted colour shifts on mobile devices and web.

### 5.4.3 **Design rules**

- ✓ Use Shell white for the main background colour White provides a good contrast background colour against on-screen elements helping the user to process information quickly.
- Use Shell yellow for header backgrounds and for splitting or sectioning important content

This will help distinguishing important content, and reinforce the brand. See figure 5.4d.

Use Shell grey for main body content and information 80% grey should be used for text and 60% grey for divider lines.



Fig 5.4d Example of how colours are used on a Contents list

# 5.5 **Typography**

# 5.5.1 **Purpose of typography**

Displaying content and communicating information well is key to creating a successful app or site. Users may have different accessibility requirements for the sizing and viewing of text. Type can also be rendered at different resolutions. In addition to this, mobile and tablet devices have limited screen space and may be used in poor light conditions.

This means that good typography is essential to ensure that text is legible and readable.

# 5.5.2 Using typography

Typography in apps and sites for all devices must always follow the rules given in the <u>Shell VI for Communications Policy version 2</u> – Typography.

In addition to this, the default OS font should also be used for mobile and tablet apps.

• Futura See figure 5.5a

This is the Shell brand font. This can be used for titles and call to actions. For web this may require the usage of font replacement technology such as <u>Typekit</u>. If this is not available the font should roll back to Verdana. As this is not a web safe font it should be used sparingly.

- Verdana See figure 5.5b This is the standard Shell web safe font. This should be used as the default font on all Shell sites.
- **Roboto** See figure 5.5c This is the standard Android font. This should be used as the default font on all Android apps.
- Helvetica See figure 5.5d This is the standard iOS font. This should be used as the default font on all iOS apps.

Futura Light Futura Book Futura Medium Futura Bold

Fig 5.5a Futura

# Verdana Verdana Bold

Fig 5.5b Verdana

Roboto Roboto Bold

Fig 5.5c Roboto

Helvetica Helvetica Bold

Fig 5.5d Helvetica

#### **First impressions**

• Futura is required for first impressions This should be used on loading and welcome screens. For web, where a user may arrive on a site at any page this should also be used for page headers.

Inside impressions

- Futura can be used to support inside impressions This should only be used if there is adequate spacing. See figures 5.5e and 5.5f for examples.
- Supporting fonts should be used for all other copy For web this will be Verdana, for iOS this will be Helvetica and for Android this is Roboto. See figures 5.5e and 5.5f for examples.

## 5.5.3 **Design rules**

#### Font sizes

- Minimum font size for the Futura font family is 16pts This is to ensure that the text is legible, as Futura can be hard to read at small sizes.
- Minimum font size for main body content text is 12pts This is to ensure that the text is legible.

#### Minimum font size for additional text is 10pts This should only be used for supporting information, or for labelling on the toolbar.

Use as few font sizes as possible This ensures the page layout is clean and easy to read.

#### Font colours

- Use 80% grey for main body copy
  This should be displayed on a Shell white background.
- ✓ Use Shell primary colours for titles and action areas Use yellow, red and white, black or 80% grey against a contrasting background colour.



Fig 5.5e Example of how fonts can be used on a mobile app

Don't use black text on a Shell yellow background This combination is often seen as a warning, and is not consistent with the <u>Shell VI for Communications Policy</u> <u>version 2</u> – Typography, Incorrect use of typography. Instead use 80% grey on Shell yellow, or Shell red on Shell yellow. See Section 5.4 Colour for further guidance.

# Don't use the Shell supporting colours for text

The exception to this is when supporting colour text may be required for naming a Shell product or sub-brand.

#### Header Roboto Bold, 80% grey

Content title (h1) Futura Bold. ALL CAPS Shell red

Content heading/ introduction Roboto Bold, 80% grey

> Main body copy Roboto, 80% grey

# THIS IS AN **ARTICLE TITLE**

This is the introduction to the article. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Quisque iaculis accumsan porttitor. Sed commodo tortor facilisis justo mattis id convallis metus lacinia. Fusce vulputate ac, fringilla porttitor magna.

Profit margins are tight in the trucking industry. And fuel can account for up to 30% of a company's operating costs. So some companies are using innovative fuel management systems to boost efficiency. One such system, Shell FuelSave Partner, is the first to combine fuel purchase data with details on driver behaviour and truck performance collected via an on-board device.

"The information helps us devise the right training for individual drivers," says Andreas August, Managing Director of August Gschwander Transport in Germany, which has been using the system. The box-like devices are about the same size as a dashboard satellite navigation system. They are footprint and help their customers report on emissions.



Q

Updated 26 July 2013

On the road, the device records such things as how aggressively the driver accelerates or brakes and feeds it into the database. Global positioning system (GPS) technology tracks the location and speed of the truck.





A

Section title

8:43 PM

8

Fig 5.5f Example of how fonts can be used on an Android tablet app

.... 🗇

Heading

Latest story

# 5.6 **Iconography**

# 5.6.1 Purpose of iconography

Due to limited screen space, icons are used extensively on mobile and tablet devices as a means to quickly communicate tools and menu options to the user. These are an essential part of the graphic language on mobile and tablet devices.

# 5.6.2 Using iconography

As users often only glance at the screen for a moment they must be able to immediately recognise the meaning and purpose of the icon.

Iconography should follow the rules given in the <u>Shell VI for</u> <u>Communications Policy version 2</u> – Iconography, with this additional guidance for mobile and tablet use.

#### **First impressions**

Launch & store icons

Iconography is required for app launch and store icons. This is the first impression a user will have of an app; therefore it needs to represent the Shell brand and the apps purpose. For further guidance see Section 7.1 First Impressions.

• Loading screens

Do not use iconography for loading screens.

Home screens

Do not use iconography for main menu content without supporting labels for the home screen.





Fig 5.6b Mobile iOS app with red selected state and five icons

Validated	Error	Information	Play	Audio	Zoom in	Zoom in
Used to indicate form validation	Used to indicate a form error	Used to indicate more information can be viewed	Used as a video play control	Used to indicate an audio file / podcast	Used to indicate the ability to zoom in (Android)	Used to indicate the ability to zoom in (iOS)
	×	i	►	Ţ	Æ	€

Fig 5.6c Example of notification & identification icons

Inside impressions

#### • Headers

Icons in the header should only be used to access tools or navigations. For further guidance see Section 8.1.1 Headers.

#### • Toolbars

Icons are used to help identify the tools included in the toolbar. For further guidance see Section 8.1.2 Toolbars.

#### • Notifications & identifiers

Icons should be used sparingly. They should only be used as an indicator of information or as a control, e.g. media player icons, comments icons, external link icon.

#### • Do not use icons as page content

If a graphic is required to support content then photography or illustrations should be used instead. For further guidance see Section 5.7 Imagery.

### 5.6.3 **Design rules**

#### 5.6.3.1 General

- ✓ Minimum size to display icons is 18x18px
- Maximum size to display icons is 44x44px on a non-retina display (164ppi) and 88x88px on a retina display (326ppi)

# ✓ Use International standard icons when possible

Use International standard icons, statutory icons and symbols, and other organisations' registered and trademarked symbols without modification.

#### Use OS icons where possible

Where appropriate on iOS 5 and Android 4, use the built-in icons and symbols by default.

#### 5.6.3.2 For creating bespoke icons

#### ✓ Icons should be simple

Icons should be as simple as possible to ensure legibility as intricate details will be difficult to see at small sizes.

#### ✓ Icons should be easy to understand

Use clear and obvious icons to represent the tool or subject. Abstract concepts may be more difficult for the user to understand at a quick glance.

	Search Used to indicate ability or activate to search	Refresh For refreshing content on a page or screen	Favourite Used to mark or open favourite items	Share Used to initiate sharing services or tools	Locate Used to locate the user's current position	Delete Used to discard or delete items	Overflow Used to indicate additional conten or menu items
Android	Q,	S	*	<	••		
iOS	Q	Ç	*		θ	圓	•••

Fig 5.6d Sample iOS5 and Android 4 icons

Map pin Used to indicate a location	Reminder Used for motor reminders	Loyalty Used for loyalty card registration	Route Used for route planning	Sign in Used as a call to action to sign in	Sign out Used as a call to action to sign out
Ŷ	Ċ	*	F	Ø	8

Fig 5.6e Sample Shell icons

- ✓ If the icon represents a tool, base it on the function Base the icon design on the function or the action rather than the tool name, e.g. for 'Station finder' the icon should focus on 'finder' rather than 'station'. See figure 5.6f.
- If the icon represents content, base it on that content Base the icon design on the specific content that the icon represents, e.g. a petrol pump could be used to indicate petrol stations.
- Do not use shadows, gradients or apply any transparency to icons. See figure 5.6g.
   By applying these features you are reducing the legibility of any icon.
- Icons should not be thin. See figure 5.6h.
  The weighting of an icon is important to improve legibility.
  Filled shapes are much easier to see than thin strokes.



**Fig 5.6f** Incorrect/correct use of a metaphor for the 'Station finder' icon. **Note: A petrol pump icon could be used to indicate stations.** 



Fig 5.6g Incorrect/correct use of colour



Fig 5.6h Incorrect/correct use of colour fill

5.6.3.3 **For creating toolbar and header icons** For further guidance see Section 8.1 Headers and Section 8.2 Toolbars.

✓ All icons must be visually equal and balanced in size It is important that the user doesn't place a hierarchy to features in a toolbar unnecessarily. The exception to this rule is if a service has a key function that needs to be visually expressed to the user. See figure 5.6i.

Icons should be shown in a neutral colour
 To ensure consistency with the OS iconography, black, white, or grey should be used depending on the toolbar background colour. See figure 5.6i.

5.6.3.4 For creating contextual and notification icons

They can be shown in a colour other than neutral See figure 5.6j.

Use non-neutral colours sparingly
 It is important that the user is not distracted by the use of colour.

**् ★ < ≎ ∶** 

Fig 5.6i Icons on the toolbar, equally spaced



Fig 5.6j Appropriate use of colour for notification icons

# 5.7 **Imagery**

# 5.7.1 **Purpose of imagery**

Imagery is very important in creating visually engaging mobile and tablet experiences. It must also express the Shell brand while supporting any accompanying content.

# 5.7.2 Using imagery



5.7a Example of Shell photography



**5.7b** Example of first impression photography

Real world photography is Shell's preferred imagery style, however, illustration can be used to express particular concepts where photography cannot, such as detailed technical drawings.

These following principles are also more precisely detailed in the <u>Shell VI for Communications Policy version 2</u> – Photography & Illustration.

#### 5.7.2.1 Photography

Shell's library material should be used where possible. Shell divides photography into first impression and supporting photography categories.

#### 5.7.2.2 First impression photography

Photography should be used for any imagery required in loading and home/landing pages. Do not use illustration. In these instances photography is used to communicate the Shell brand in an effective and recognisable manner.

- Large sized and high-resolution photography can increase loading times and should be used with caution and understanding of the app or site service.
- For first impressions single images are preferred. Multiple images should be limited when working with small screen devices to maintain legibility. It is important that all photography is correctly scaled for the device being used and aligned to the grid. Please see 5.7.3 Design rules for further guidance.

First impression photography must follow these key principles in order to be used in any Shell communication see figure 5.7b:

- Use a realistic style
- Feature yellow and red in the image
- Capture an unusual perspective

#### 5.7.2.3 Supporting photography

Supporting photography should be used for 'inside impressions' please see section 5.1 for further reference. Photography here has a practical role, such as communicating a desired function or helping to support written information.

Supporting photography must follow these key principle see figure 5.7c:

• Use a realistic style

The practical use of supporting photography means the following principles should are flexible and should not be used if it compromises the meaning or communication of the photography's use:

- Feature yellow and red in the image
- Capture an unusual perspective

#### 5.7.2.4 Illustration

Illustration is used when the photography cannot communicate an idea or meaning please see 5.7.3 Design rules.



5.7c Example of supporting/inside photography



5.7d Example of Shell illustration

#### 5.7.2.5 Illustration principles

Illustration must follow these key principles see figure 5.7d:

- Be technically accurate
- Feature yellow and red in the image
- Use flat colours from the Shell palette

For full guidelines on imagery see the <u>Shell VI for Communications</u> <u>Policy version 2</u> – Photography & Illustration.

Shell imagery can be accessed from the <u>Shell Brand Central</u> <u>Library</u>.

## 5.7.3 **Design rules**

#### ✓ Simplify images where possible

Complex photographs and illustrations may be hard to understand on a mobile or tablet device, so be careful to ensure that the detailed elements are easy to read in typical mobile use conditions. See figure 5.7c.

- For photography this can be achieved by cropping an image to focus in on the detail.
- On illustration this can be achieved by simplifying the illustrative elements.

#### Optimise imagery

Images should always look good, irrelevant of which device it is being viewed on.

- Images should not be scaled-up or distorted.
- When images are scaled down in size the important elements of the image must remain legible.
- To improve loading times images should be as small in file size as possible, without damaging the quality.

Image dimensions should be consistent Image sizes should be consistent throughout the Shell brand experience, so where possible use square images, or 2:1





**Before:** Image is too detailed and difficult to understand



After: Image is cropped to focus on important details









Before

Before

After

5.7c Cropping and simplification of imagery

After



#### ratio for rectangles.

- Image should be viewed clearly in either orientation When resizing and cropping imagery consider how this will be viewed in different orientations. See figure 5.7d.
- ✓ Thumbnails should be displayed as smaller versions of the full size image See figure 5.7e.

Thumbnails should not be cropped for simplicity in the same way that full size images are. These should be resized to the correct dimensions, then only cropped on the height or width to match the required thumbnail dimensions.



5.7d Image resize for landscape orientation



**5.7e** Main image and thumbnail image after being resized and then cropped. Note how the thumbnail image displays more of the surrounding environment.

# 5.8 Animation and video

# 5.8.1 **The purpose of animation and video**

Animation and video can add value to create a richer experience as well as supporting the content. They are both really powerful tools for making information more engaging and easier to digest.

## 5.8.2 Using animation and video

Animation and video should follow guidelines provided in the <u>Shell</u> <u>VI for Communications Policy version 2</u> – Digital.

#### **First impressions**

#### • Loading screens

The animation on a loading screen should be restricted to the loading activity indicator. This ensures that the loading status is highlighted, preventing confusion. See figure 5.8a.

Inside impressions

• Video content

Videos can be used to support other content or as individual content items.

#### • Supporting animation

Animation can be used to enhance graphical information on charts and diagrams by highlighting data.



Activity indicator

**5.8a** Loading screen with animated loading acivity indicator

### 5.8.3 **Design rules**

#### ✓ Optimise video

Videos should always look good irrelevant of which device it is being viewed on.

- ✓ Videos should not be scaled-up or distorted, and should be displayed at the correct aspect ratio.
- ✓ To improve loading times the video file size should be as small as possible, without damaging the quality.
- Only use animation where it adds value Avoid using animation for decoration. Animation should only be used to highlight information or concepts e.g. on diagrams and charts.
- Avoid using Flash for animations
  Flash is not supported by iOS, so where possible use HTML5 animations.

# 5.9 Charts, tables and diagrams

5.9.1 **Purpose of charts, tables and diagrams** 

Charts, tables and diagrams are very useful tools for conveying information in a clear and simple format, while also reinforcing the Shell brand values.

### 5.9.2 Using Charts, tables and diagrams

The use of charts, tables and diagrams must always follow the rules stated in the <u>Shell VI for Communications Policy 2011</u>, section Charts, tables and diagrams.

#### **First Impressions**

 Only use charts, tables and diagrams on first impressions if it is important to the app or site's purpose

Data rich content can require a lot of detail making it difficult to adhere to Shell first impression guidelines.

#### **Inside Impressions**

• Use charts, tables and diagrams where possible to present data rich content

Information needs to be legible and clear especially on small screens and given the contexts of use for mobile and tablet devices.

### 5.9.3 **Design rules**

- ✓ Tabs, headings and labels may require flexible lengths Consider in advance that the length of text will change when designing tabs, headings and labels. Data represented in charts, tables and diagrams will not always remain static as languages change. See figure 5.9a.
- Apply the same consideration to font and colour as per the rest of the content
- Font readability in a chart, table and diagram is the same as elsewhere on a device
- Only use Futura for headings at a font size of over 16pts This is due to Futura being difficult to read clearly on small device screens.
- ✓ Offer a link to view more or less detail

Given the smaller screen sizes of mobile devices and their context of use it's important that any essential information is presented. The option for the user to add more information must be clear and visible.

Use horizontal scrolling with a fixed row header in a table or diagram

If the user must scroll to see all the items in the table or



Fig 5.9a An example of a chart for the tablet device. Please note that padding is left around key areas of text.
diagram, then headers must always remain in view to allow the user to easily scan the information provided.

- Always include the option to zoom in and zoom out This will help improve legibility and usability in data-rich diagrams.
- Animation can be used to highlight information Animation can help explain or highlight data on charts and diagrams.

# **6 Interactions**

These interactions will help provide a starting point for the native app and mobile website development by informing the process to creating a navigable hierarchy for different types of applications.

Creative, intelligent interpretation of these examples is encouraged within the framework of our design principles. See section 4 Principles for reference.

**Note**: Detailed design examples of many of the screens are shown in Section 9: templates.

In this section

6.1 Gestures	75
6.2 Ergonomics	77
6.3 Hit areas	80
6.4 Orientation	81
6.5 Zoning	
6.6 Content types	84
6.7 Interaction patterns	
6.8 Transitions	

# 6.1 Gestures

## 6.1.1 **Purpose of gestures**

Gestural interactions are inputs performed by the user on mobile devices to complete a specific task using physical inputs through a touch sensitive screen, instead of using a cursor and mouse, for example.

Please refer to the iOS <u>Human Interface Guidelines</u> and <u>Android</u> <u>Design Principles</u>, <u>Style</u>, <u>Patterns and Building Blocks</u> for a full list of interactions, and variances between platforms.

## 6.1.2 Using gestures

Native apps should follow the conventional gestures that the devices designed for can understand.

- The outputs are predictable and match those used on other applications on the device
- Limit the use of them in demanding contextual situations
- The action and complexity is appropriate to the task being performed
- Ensure they are overlaid with responsive feedback.

Gestural interactions can be supplemented with other navigational devices, for example, the actions of previous and next controls in an image gallery can be replicated by supporting the swipe left / right gesture. See figure 6.1a.

# 6.1.3 Gesture examples

The most common gestures are listed below:

Figure	Gesture	Intended action
		Scroll vertically or horizontal. Also
		used to select listed content for
6.1a	Swipe	deletion
6.1b	Single tap	For activation and selection
		Edit text and for secondary controls
6.1c	Tap and hold	(copy / paste)
		Zoom in and out of content, or select
6.1d	Double tap	(copy / paste)
6.1e	Drag	To pan around or move items
6.1f	Pinch	Zoom in
6.1g	Spread	Zoom out
6.1h	Flick	Nudge, move item or scroll quickly

Also consider:

Figure	Motion	Intended action
6.1i	Shake	Undo previous action
6.1j	Rotate	Change orientation



# 6.2 Ergonomics





Fig 6.2c Tablet reach (landscape)

# 6.2.1 Purpose of ergonomics

The physical size of the devices used, their orientation and the dexterity of the average user mean designers should consider the ergonomics of the UI they're creating.

## 6.2.2 Using ergonomics

The positioning of certain tools and links to functionality is partly determined by how quickly and comfortably the user is able to reach them. See figure 6.2a-c for the zones that are easy to access, and where the user will have to reach.

Left-handed users will have different 'easy' zones than righthanded users, so the combined 'easy' zone for both types of user will be at the bottom of a mobile screen.

## 6.2.3 **Design rules**

 Position frequently accessed tools within 'easy' zones See figure 6.2a-c.

✓ Position tools that require a conscious decision within 'reach' zones

For example, send message, back or cancel should be positioned in the reach areas.

Avoid placing buttons at the far right hand side The hardware and OS controls that come as standard with mobile devices mean that UI elements should be positioned away from these areas. For example, buttons placed towards the far right hand side should be situated well inside the screen to avoid unintentional scrolling through the page.

# 6.3 Hit areas

## 6.3.1 **Purpose of hit areas**

Finger widths vary from user to user, and by their nature mobile devices tend to be used on the move and in a hurry. The resulting inaccuracy needs to be accounted for, reducing both the chance of choosing the wrong item or nothing at all.

## 6.3.2 Using hit areas

Active elements need to be designed with a large enough area with sufficient space around them to prevent unintentional input errors. This is particularly true of web sites that need to adapt to use on a mobile device: elements such as links should be spaced more widely to account for this.

### 6.3.3 **Design rules**

The minimum recommended 'hit area' for controls is 7-8mm, with spacing around the 'hit area' of 1-2mm. See figure 6.3a.

This translates to pixel widths as follows:

Display	Minimum hit area
Non-retina display (164ppi)	44x44px
Retina display (326ppi)	88x88px



6.3a 7mm hit area with 2mm padding

# 6.4 **Orientation**

### 6.4.1 **Purpose of orientation**

Devices allow the user to adjust their view between portrait and landscape. This means the layout of a native app or site should be considered in different orientations, and change accordingly. See figure 6.4a.

## 6.4.2 **Using orientation**

#### • Native apps

Native apps remain fairly fixed in their layout, but some may benefit from showing or hiding elements to suit different orientations. For example a magazine-based app viewed on a tablet may show a contents list on the left hand side in landscape mode, but in portrait mode it may be hidden and revealed as an overlay by tapping a 'contents' button in the header area. This gives the reader a more natural, book-like experience.

#### • Websites

Websites should be designed to 'stretch' and fill the wider width available in landscape mode. This is commonly referred to as a 'responsive layout'. Sophisticated responsive layout designs may change the order or visibility of certain UI elements, being almost infinitely scalable. Where possible, this technique should be employed for our site designs, allowing them to automatically optimise their display, to suit the size and proportion of any screen.



**6.4a** Change in orientation for a magazine layout, demonstrating how a contents list is shown/hidden

### 6.4.3 **Design rules**

### ✓ Copy should re-wrap

Text should rearrange to fill varying widths on different orientations.

### Imagery should be resized

Images should size to fit the varying widths. Generally this will mean images can be larger on a landscape orientation, however this can be impacted by the screen layout.

### Modules can reposition or resize

Portrait widths may require modules to stack, while landscape may allow more flexibility for positioning modules, especially on a tablet. As the sizing and arrangement of modules can change this will also affect the elements within them.

### Show/hide relevant content

It may be beneficial to hide elements such as contents menus to provide a more engaging experience. These can be revealed through further interactions as required. See figure 6.4a.

### Do not reposition headers and toolbars

Main UI elements such as headers and toolbars should not reposition as the orientation changes as these cannot scale.

# 6.5 Zoning

## 6.5.1 **Purpose of zoning**

Mobile and tablets have standardised areas of functionality. This allows a consistent user experience to be created across all mobile apps and sites.

## 6.5.2 Using zoning

Mobile device UIs are separated into three main zones: header, content area and toolbar. Due to hit area (see section 6.3 Hit areas) requirements headers and toolbars on mobile devices require a higher percentage of screen space. A guide to the space each zone should occupy in a typical UI layout is shown in figure 6.5a and 6.5b. Examples and guidance for designing within these areas is detailed in Section 9 Templates.



# 6.6 Content types

## 6.6.1 Purpose of content types

We have broadly separated content into **article** based and **tool** based types. Categorising the content in this way will help you to understand how to apply the guidelines and interaction patterns to the content you create. A native app or website might contain a combination of articles or tools, so UI designs need to be flexible enough to cater for this.

## 6.6.2 Article content

#### What is an article?

Article based content presents information via the written word, video, imagery, or sound. They can be collected into chapters, genres, and sections. Articles can be inter-related and indexed for search purposes, using meta-tags and highlighted descriptions. Please see section 10.2 Search for further information. This content can be frequently updated or added over time so archived material may need to be discoverable alongside newer entries.

#### Using articles

Articles can be used for:

✓ News updates

✓ Periodical-style publications



- ✓ Image or video galleries
- ✓ Blog or diary collections
- ✓ Trading statements and industry commentary
- ✓ Presentations, data tables, charts and diagrams

#### **Functional considerations**

The content can be manipulated using tools within the context of the information presented. For example:

- Copied and pasted into other applications
- Shared via email, or social networks
- Searched through for keywords or similar / related content
- Defined with an inbuilt dictionary or thesaurus
- Made larger or smaller to aid legibility
- Bookmarked for easy future access
- Commented on by its readers

# 6.6.3 Tools content

#### What is a tool?

Tools will contain functionality that allows the user to manipulate or create information in a more personalised way, or perform a specific task. Depending on their complexity and focus, tools may be packaged singularly into an app, or bundled together to create a multifunctional suite of tools.

#### Using tools

Examples of what tools can be used for include:

- Finding your nearest Shell station and planning a route to it
- ✓ Calculating your vehicle's fuel efficiency
- ✓ Collecting and redeeming loyalty points
- ✓ Taking or manipulating photographs or video
- ✓ Setting alerts, reminders and list making
- ✓ Collating feeds and network activity into one area



6.6b Example of a station finder app

#### **Functional considerations**

Tools should always exploit the inherent technology of the device that is available, as well as provide a useful function.

#### Native apps

This is where an app is created that is specific to the device, e.g. iOS or Android. Native app tools should take advantage of the device features such as location services or camera. The different OS will determine the functionality of these apps. With iOS devices it is easier to take advantage of their features due to the limited amount of devices available, however for Android there are many models, ranging from low-end to high end. This must be taken into account when designing for Android apps.

#### • Mobile web apps

Mobile web apps should use any back-end systems or tools in place, such as search or account services. These will have the similar technical limitations as desktop web apps, but the mobile device limitations should be considered.

# 6.7 Interaction patterns

# 6.7.1 Purpose of an interaction pattern

An interaction pattern is a reusable solution to define how a user can navigate through a native app or website. It describes the steps back to and from the main types of home screen.

These patterns build on the basic principles of good navigational design. Please refer to the iOS <u>Human Interface Guidelines</u> and <u>Android Design Principles</u>, <u>Style</u>, <u>Patterns and Building Blocks</u> for a full list of interactions, and variances between platforms.

## 6.7.2 Using interaction patterns

These defined patterns provide a starting point for creating a coherent navigation system. They provide solutions to common interaction problems, ensuring the user understands the interface, and can accomplish their tasks.

# 6.7.3 **Design rules**

We have specified the following interaction patterns that can be used to design how the user will access content and tools through an app or site experience. These patterns will be suitable for different uses, which will be described within this section:

	Pattern	Usage
Α	Out and back	Mobile and tablet
В	Nested detail	Mobile and tablet
С	Hub and spoke	Mobile and tablet
D	Tabbed view	Mobile
Е	Magazine	Tablet
F	Dashboard	Tablet

#### A. Out and back – Mobile and tablet

Mobile and tablet devices have limited screen space available for navigation, and users often interact with those devices when they have limited attention spans, this means that navigation should be simple and easy to use.

Therefore the following patterns all use this simple 'out and back' navigation method, providing the user with one route to reach the content they want, and one way back. See figure 6.7a.

- Step 1. Out to next screen
- Step 2. Back to previous screen

050 101.	U	se	for:
----------	---	----	------

✓ Navigating in and back from content or tools



6.7a Out and back

B. Nested detail – Mobile and tablet

In its basic form this pattern consists of a home screen that shows a list of categories.

- **Step 1.** The user taps a category on the home screen to drill down to a list of objects.
- Step 2. User then taps an item in the list to see the details of that object.
- Step 3. The user taps a back (or 'home', or 'contents') button to return.

More complex apps or sites may contain further levels of information listed either on another screen, or by 'opening' a sub menu below the selected option.

#### Use for:

- ✓ Article based content
- Apps and sites focused on one type of business object For example for apps or tools around orders, products or messages.
- Apps or sites with closely related content or tool types Where the app or sites has a small number of closely related features, functionality or content then this pattern could be used.





#### C. Hub and spoke – Mobile and tablet In its basic form this pattern consists of a menu screen (the hub),

and then one screen for each of the available features or content categories (the spokes).

- **Step 1.** The user taps an item on the menu screen to navigate to that feature or content category.
- Step 2. The user taps the back button to return from the spoke screen.

For tablets see the F. Dashboard pattern for further guidance on multi-functional tool based content.

#### Use for:

## ✓ Multi-functional tool based content

The spoke screens may be articles, tools, lists, maps or any other type of information.

### ✓ Complex apps and sites

This pattern combines well with the nested detail pattern. In this case the user taps an element on a spoke screen to drill into a more detailed spoke screen, and taps the back button to return.





#### D. Tabbed view – Mobile

In its basic form this pattern consists of a collection of screens each showing a different view. The app or site has a tab bar that is consistent and visible on all screens. If this view is coded with accessibility in mind there should be no search engine optimisation (SEO) issues.

Rather than having a main menu or dashboard, the most commonly used view (in this example, a map) becomes the home page. The pattern combines well with the nested detail pattern. See B Nested detail pattern.

- Step 1. The user taps the tab bar to move between views
- **Step 2.** The user interacts with content. If a nested detail screen is used, the user can tap an item within this view and down into another level of detail about that item.
- **Step 3.** This detail screen should remove the tab bar or replace it with a context specific toolbar.
- **Step 4.** The user can tap a back button to return to the original view.

#### Use for:

### Single tool based apps and sites on mobile

Only use for a single tool based app or sites on mobile. Tablets would usually be used for multifunctional tool apps, so the dashboard pattern should be employed instead.



#### 6.7d Tabbed view

E. Magazine – Tablet



Shell Design Guidelines for Mobile and Tablet

6.7e Magazine

This pattern uses the structural semantics of a printed magazine, where written articles are usually accompanied by imagery or video. Tablets are the most appropriate medium for magazinebased apps or sites. Use B. Nested detail pattern for magazines on a mobile.

Content is likely to be manipulated by the reader by using tools such as search, copy, share and so on: a toolbar providing access to this functionality should be included.

- Step 1. The cover summarises the content contained within. Articles can be listed by 'most recent' or 'most important' grouped into sections, usually by subject matter. See Section 10.7 Filtering and sorting for further guidance.
- **Step 2.** The user can move through all articles one by one by swiping from right to left across the screen.
- **Step 3.** To skip to a specific article, the contents list should be available at any time. Access should be consistently situated in the app or site.
- Step 4. A carousel can also be included. This would show thumbnails of sections or individual articles within a section, which the user can quickly 'thumb' through by flicking left or right. This can be accessed through an overlay, which pops up from the bottom of the screen, or down from the top beneath the header.

Use for:

# ✓ Magazine based content

This pattern is most suited to apps or sites where articles are the main focus.

#### F. Dashboard – Tablet

Similar to the hub and spoke pattern for mobile, the larger screen on a tablet allows room for content areas to be shown in more detail.

The dashboard provides an overview of the status of multiple article and/or tool based content in a single view, known as modules. These modules are snapshots of information within a section, e.g. a list of most urgent tasks to be performed, latest announcements, a map of your nearest station or latest share price information.

Personalisation of modules in the dashboard view is desirable when an app or site is visited frequently, for example a B2B productivity tool. This may occur passively, by using cookies to remember what tools or content were used in the last session, or actively by allowing user to adjust settings and preferences, such as creating a new item in a section, setting bookmarks, rearranging modules, or changing the status of the items shown in a section. Please see the Shell cookie guidelines for further guidance on personalisation.

- **Step 1.** To drill down, tap a content section heading e.g. station search to access that section's landing screen, or a specific station location listed to view that detail.
- Step 2. The dashboard pattern leads into simpler interaction patterns, such as nested detail or tabbed views. These patterns exist within a wider navigational hierarchy, allowing users to skip from tool to tool, or article to article.



Use for:

### Multi-functional tool and content based apps Use for apps and sites on tablet devices where multifunctional tools are combined with article content, particularly when rich data and media content is included.

# 6.8 **Transitions**

## 6.8.1 **Purpose of transitions**

A transition is an animation that occurs as the screen changes from one to another. This section defines the standard transitions that should be used as a user interacts with content and tools throughout Shell branded apps and sites.

## 6.8.2 Using transitions

Transitions help content flow smoothly from one screen to another. It is important that a consistent set of transitions is used. This helps create coherent navigation and interaction patterns that support the user's understanding of the app or site. Transitions between content and tools help keep the user focused on what they are viewing, while preventing them from getting lost.

It is important that the number of transitions is balanced against the content being displayed. Too many transitions can cause confusion, along with too few.



6.8a A- Quick change



6.8b B- Slide along (Right to left)



6.8c C - Slide over (Left to right)

	Transition	Description	General usage
A	Quick change	See figure 6.8a.The view changes without any animation.	When changing between two different types of tool or content.
В	Horizontal slide along	See figure 6.8b.The view slides left or right, pushing the previous view out of the screen	When moving forward to view new and related content, or back to a previous screen.
С	Horizontal slide over	See figure 6.8c.The view slides left or right, over the previous view	When viewing supporting or additional content.
D	Open to full screen	See figure 6.8d. An item on the screen opens/expands to fill the full screen	When opening or viewing an item
E	Expand	See figure 6.8e. An item on the screen expands pushing content along or down	When expanding a list of content within a screen.
F	Flip	See figure 6.8f. The view flips over as if turning around from back to forwards	When viewing a screen specifically related to the previous screen, e.g. additional settings or information.

### These are the types of transitions that can be used.



#### Do not use these transitions

	Transition	Description	General usage
×	Fade	The view fades away to reveal another view behind in	n/a
×	Vertical slide along	The view slides up or down, pushing the previous view out of the screen	n/a
×	Vertical slide over	The view slides up or down, over the previous view	n/a

### 6.8.3 **Design rules**

The following are specific rules for the transitions between certain types of content.

### 6.8.3.1 Loading screens to homepage

### ✓ Use a Quick Change transition

To reduce any user frustration at loading times the app should immediately take the user to content or tools as quickly as possible. See figure 6.8a and 6.8g.

### 6.8.3.2 Using the toolbar to change view

✓ Use a Quick Change transition As the user navigates using the toolbar the view should change quickly between these different items. See figure 6.8a and 6.8h.



**6.8g** Quick change transition from loading screen to homepage



#### 6.8.3.3 Interaction patterns

The section defines transitions when navigating specific interaction patterns. See Section 6.7 Interaction patterns for additional information on these patterns.

#### 6.8.3.3.1 Out and back

✓ Use a Slide along (Right to left) transition for Out This gives the impression that the user is going forward. See figure 6.8b and 6.8i.

### Use a Slide along (Left to right) transition for Back This gives the impression that the user is returning. See figure 6.8b

#### 6.8.3.3.2 Nested detail

 Use a Slide along (Right to left) transition for opening a separate sub menu

This gives the impression that the user is going forward. See figure 6.8b.

Use a Slide along (Left to right) transition for returning from separate sub menus to the home screen

This gives the impression that the user is returning. See figure 6.8b.

**Use an Expand transition to open nested sub menus** See figure 6.8e and 6.8j.



**6.8i** Slide along transition to show Out stage from Out and Back interaction pattern



6.8j Example of a transitions in a Nested detail journey

 Use a Slide along (Right to left) transition for opening content from the home screen or sub menu This gives the impression that the user is going forward. See figure 6.8b and 6.8j.

Use a Slide along (Left to right) transition for returning from content to sub menus

This gives the impression that the user is returning.

6.8.3.3.3 Hub & spoke

 Use a Slide along (Right to left) transition for opening hub items

This gives the impression that the user is going forward. See figure 6.8b and 6.8k.

Use a Slide along (Left to right) transition for returning to the home screen

This gives the impression that the user is returning.

#### 6.8.3.3.4 Tabbed view

 Use a Quick Change transition for switching between tabs

As the user navigates using tabs the view should change quickly between these different items. See figure 6.8a.



**6.8k** Slide along transition to show out stage from Hub and Spoke interaction pattern

#### 6.8.3.3.5 Magazine

 Magazine interaction pattern should follow content transition guidelines See Section 6.8.3.4 Content

#### 6.8.3.3.6 Dashboard

✓ Use a Slide along (Right to left) transition for opening dashboard tools

This gives the impression that the user is going forward. See figure 6.8b and 6.8l.

Use a Slide along (Left to right) transition for returning to the home screen

This gives the impression that the user is returning.

Dashboard component interactions should follow content transition guidelines See Section 6.8.3.4 Content.

6.8.3.4 Content

The section defines transitions when navigating through content within an app.

#### 6.8.3.4.1 Contents list

A contents list shows an index of content within an app or site. See Section 9.1 Contents list for additional information.



**6.8I** Slide along transition to show out stage from Dashboard interaction pattern

#### Home screen (Dashboard)

**Contents list open** 





6.8m Slide along transition to show contents list open

- Use a Slide along (Right to left) transition for opening a tablet landscape contents list See figure 6.8b and 6.8m. The contents list Slide along transition will not completely change the screen view, but only push it so far as is required to show the contents list.
  - Use a Slide along (Left to right) transition for closing a tablet landscape contents list See figure 6.8b.
- Use a Quick Change transition when opening an item on the contents list See figure 6.8a.
- ✓ Use an Expand transition to open sub menus within the contents list See figure 6.8e.

#### 6.8.3.4.2 List or article link item

A list item or content link will contain an arrow indicating to the user that they can view further content.

 Use a Slide along (Right to left) transition for opening an item of content

This gives the impression that the user is going forward.

Use a Slide along (Left to right) transition for returning to the previous screen

This gives the impression that the user is returning.



6.8n Slide along transition to show article opening



**6.80** Open to full screen transition to show an image opening from a grid view

#### 6.8.3.4.3 Grid item

A grid screen will generally contain thumbnails of images or videos.

### Use an Open to full screen transition for opening an item of content within a grid

For example, this would be opening an image or video from a gallery. See figure 6.8d and 6.8o.

Use a reverse Open to full screen transition for closing an item of content within a grid See figure 6.8d.

#### 6.8.3.4.4 In context image or video

An 'In context image or video' will be an image thumbnail, video player thumbnail or smaller version of an image that is displayed within an article or tool. This can then be opened to view in full screen or allow a video to be watched.

### Use an Open to full screen transition for opening an in context content item

For example, this would be opening an image or video from an article. This will work in the same way as an item opening from a grid view. See figure 6.8d and 6.8o.

### Use a reverse Open to full screen transition for closing an in context item See figure 6.8d.

# **7 First impressions**

First impressions are defined as what the user first sees in a Shell app or site. In these instances, the core visual identity should be represented rigorously to ensure that the distinctive Shell look is delivered.

The first impressions that a user will have of a native app are the launch icon and the loading screen, however first impression concepts can also be extended to the native app home screens.

For the web, home screens are the main first impression, although as the user may arrive at any page on the site first impressions are important across the full website. See Section 8.1 Headers for further guidance.

In this section

7.1 Launch icon	108
7.2 Loading screens	112
7.3 Home screens	116

# 7.1 Launch icon

# 7.1.1 The purpose of a Launch icon

The launch icon (also known as the application icon) is the first touch point and impression the user has of the Shell brand. This is the same icon used for the App Store or Android Market.

Once the app is downloaded it will permanently live on the user's device home screen.

For further detailed reference, please see:

For first impressions design guidelines Shell VI for Communications Policy 2011,

OS Human Interface Guidelines for designing launch icons <u>developer.apple.com/icons</u>

Android Design for designing launch icons <u>developer.android.com/icons</u>



Fig 7.1a Launch icons on an iPhone
## 7.1.2 Using Launch icons

 It is important that the Shell brand and the purpose of the app is clearly recognised

This is achieved through effective use of Shell visual assets such as the Pecten and primary brand colours.

- All application icons must visually connect as group It is important that any app icon produced is visually related to the Shell brand and any other existing Shell icons.
- Design for a variety of backgrounds
   As users can change their device's screen backgrounds it is
   important to consider how the icon would look on different
   contrasts, tones and patterns.

### 7.1.3 Elements

A launch or store icon must contain:

- **The Pecten** As the launch icon is the first impression of the app, the Pecten must always be shown.
- Icon for app service

An additional icon should be added to signify the purpose of the app. This allows the user to distinguish it from other Shell apps.

# 7.1.4 **Design rules**

#### ✓ Consider label length

Ensure the app title is short enough so it does not get shortened on the user's device.

### ✓ Always show the Pecten. See figure 7.1b

This is a key symbol of the Shell brand and will help to provide user recognition.

#### ✓ Always encase the app icon in a square

This is a standard requirement for any app on iOS. By ensuring that this is consistent for all OS platforms it will help to provide a consistent experience for users.

### ✓ The app icon should be 80% grey

✓ The app icon should have a 75% opacity white highlight

✓ Show the Pecten on a Shell white background Shell white is used for the top section for the Pecten. See figure 7.1b

✓ Show the app on Shell yellow or 15% grey background See figure 7.1b Use Shell yellow for the bottom section for external apps and 15% grey for internal apps.



Fig 7.1b Launch icon examples

### X Don't crop icons

This will help improve legibility and convey the applications purpose.

# Don't include app titles in the icon

Labelling the app outside of the icon is a standard feature on iOS and Android; therefore it doesn't need to be repeated in the icon.

# 7.2 Loading screen

# 7.2.1 The purpose of a loading screen

Loading screens are full screen images used to indicate that an application is starting up. They are typically used for large, data rich applications that require loading time, and are used for apps. Loading screens also reinforce the name of the app, helping to provide an understanding of its purpose.

For first impressions design guidelines, please refer to the <u>Shell VI for Communications Policy 2011</u>,

## 7.2.2 Using loading screens

- Loading screens are important first impressions, and therefore must follow and complement the Shell brand.
- Consider the file size of the loading screen so as not to delay the loading of the main app. For example minimise imagery and do not use animation (except for the activity indicator).



**Fig 7.2a** Loading screen containing the key elements

## 7.2.3 Elements

A loading screen contains:

- Heading/title
- Progress feedback
- Pecten

# 7.2.4 **Design rules**

✓	They must communicate to the user that the app is loading This can be achieved through the use of an animated indicator, or text. See figure 7.2a, activity indicator.
✓	<b>Animation should only occur on an activity indicator</b> This will help prevent increasing the existing loading time.
✓	It must confirm that the correct app has been launched Achieved by showing the title of the app. See figure 7.2a, App heading.
~	Must include room for any potential disclaimers or legal information

need to be displayed to the user. See figure 7.2a, Disclaimer.



**Fig 7.2b** Examples of first impression photography and design principles being applied to loading mobile screen designs

**NOTE:** The aim of these examples is to provide a starting point for design rather than specific detailed designs.

# 7.3 Home screens

## 7.3.1 What are home screens?

Home screens are a key part of an app or site's first impression, and sets the tone for the user's experience of it. These give the user the first point of contact with the content or tools within an app.

The home screen should reflect:

- The Shell brand attributes (layout, colour, typography)
- The site or app's purpose (i.e. is this a tool or magazine?)
- The target audience (B2B, B2C, B2E)
- What content or functionality is available

**NOTE:** The aim of the following examples is to provide a starting point for design rather than specific detailed designs.



Fig 7.3a iOS content

# 7.3.2 Using home screens

Apps and sites should broadly use the same design layout and information structure as each other wherever possible. Note that for websites in particular, users may access the site through an article and never see a home screen. The home screens are the starting point for the key interaction patterns shown in Section 6.7 Interaction patterns:

#### Nested detail

See figure 7.3b and 7.3c.

Used when native app or site has multiple categories and sub categories, particularly where it is focused on content. This can also be used as secondary screens where another home screen pattern is used.

#### • Hub and spoke

See figure 7.3b and 7.3d.

This is used to allow apps and sites to be created that do not require a header or toolbar. This pattern will lead out to other interaction patterns, which can stand alone from the hub and spoke screen.

#### • Tabbed view

See figure 7.3b and 7.3e.

The tabbed view will be most useful for tool-based apps and sites that have a small amount of options. This allows the user to quickly flick between screens.







Nested detail

Hub and spoke

**Tabbed view** 



Dashboard

Magazine







#### • Dashboard

See figure 7.3b and 7.3f.

Dashboards are useful where there is a combination of tools and content, especially when there is a high volume of information.

#### • Magazine

#### See figure 7.3b and 7.3g.

A magazine home screen should be utilised where the app or site is mainly focused on content, especially where there are high levels of visual stimulation.

Examples included are a starting point for designers: the guidance outlined should inform future design iterations, providing a framework for them to develop within.

## 7.3.3 Design rules

#### Make the best use of the screen space

Critically, home screens need to consider the screen estate available without scrolling. If a user doesn't see a hook (either through content or functionality) that encourages engagement, they're unlikely to interact further. This is particularly true of websites.

**Ensure that key information is shown prominently** It should be clearly communicated what the site or app is for.

**Ensure easy access to all key functions and articles** Home screens should provide access to functions and articles as shown in the Section 6.7 Interaction patterns. Links to individual articles or tools should allow for 'deep-diving' into content, along with links back to the high level groups they belong to.

✓ The layout should be uncluttered and spaced precisely Use the grid principles defined in Section 5.3 Grids. Similarly, where modules are used, they should be spaced sufficiently to allow for differentiation between functions and content types.

Highlight latest content where appropriate Home screens, particularly for article based content, should highlight frequently changing or updated material.



Fig 7.3c Article focused home screen using **nested detail** pattern

Fig 7.3d Tools focused home screen using hub and spoke pattern

#### Create a robust navigational convention

Navigational conventions created for the home screen should be flexible enough to be reflected consistently in later screens, providing the user with a degree of familiarity as they move through the content.

#### Optimise layout as the orientation changes

All home screen layouts should adapt to being viewed in different orientations, to ensure the first impression is at its optimum on both portrait and landscape. In a dashboard or magazine view on a tablet, the size of different modules is likely to change and therefore the arrangement may also affect the elements within them.

# The Pecten does not always have to be shown on native apps home screen

On native apps and sites the user's first impression will be through the launch icon and the loading screen, therefore it is not essential to show the Pecten in the layout of the home screen. See Section 5.1 Pecten.

# The Pecten should always be shown on a website home screen

A website home screen will likely be the user's first impression. See Section 5.1 Pecten and Section 8.1 Headers.

# The colour balance of the home screen should reflect the first impression colour requirements

See Section 5.4 Colour.



Fig 7.3e Tool focused home screen using a tabbed view pattern

#### Create a clear typographic hierarchy Headings, subheadings, introductory paragraphs and body text use size, colour or weight to distinguish them, even when the type size is altered by the user.

(i) If scrolling is required, ensure the layout encourages this For example by cutting off elements that suggest there is more content further down the screen.

# (i) Only use Futura on home screens if there are no legibility issues

Futura can help support the first impression on the home screen, however should be used sparingly, and only if the font can be displayed at over 16pts. This is due to legibility issues with Futura at small font sizes.



Fig 7.3f Mixed article and tool home screen using dashboard pattern



Fig 7.3g Article focused home screen using magazine pattern

# 8 Components

Apps and sites are made up of components that are brought together within a single layout, zoned in the manner described in Section 6.5 Zoning. See Section 9 Templates for some examples of how screens are developed using components within the zoning areas. These common components should be used as a basis for creating all apps and sites.

In this section

8.1 Headers	123
8.2 Toolbars	141
8.3 Footers	159
8.4 Titles and section dividers	163
8.5 List components	168
8.6 Form elements	173
8.7 Help	178
8.8 Notifications	188
8.9 Maps	196
8.10 User comments	201

# 8.1 Headers

## 8.1.1 What is a header?

The header sits at the top of the screen, just below the device's status bar. The specific OS will control the functionality and layout of this.

The header's purpose is to help the user to identify the screen they're on, access key functions, navigate through content or jump back to a common starting point e.g. home.

The different OS's define these as follows:

Figure	OS	Header name
8.1a	iOS	Navigation bar
8.1b	Android	Action bar

"The action bar is arguably the most structurally important element of an Android app." - Android developer guidelines



Fig 8.1a iOS header - Navigation bar

# 8.1.2 Using headers

The header bar provides an opportunity to 'brand' each screen



Fig 8.1b Android header – Action bar

consistently. It forms a first impression (see Section 7 First impressions for further guidance) that should be implemented in line with the <u>Shell VI for Communications Policy version 2</u>, and designed consistently within the context of the app or site created. There will be variations driven by the device and the OS.

The functionality shown on iOS headers can vary from portrait to landscape. For example, the contents area changes from an omnipresent list view in landscape, which gets hidden in portrait view and re-accessed by tapping the 'contents' button. See Section 6.4 Orientation for further details.

For Android tablets back, home, contents lists and bookmarking controls are provided by the hardware or operating system, therefore these controls do not need to be repeated in the header.

# 8.1.3 Component elements

The header component is made up of these elements. See figure



#### 8.1c:

#### Pecten

The Pecten is required on the header on mobile websites as any page within the site could be a user's first impression. On mobile and tablet apps the user will have already seen the Pecten through the launch app and loading screen, therefore it is not needed. See Section 5.1 Pecten for further guidance.

#### • Heading title

The short title for the app should be used on the home screen. Subsequent screen titles should reflect the section being explored in the navigation. Wherever possible, ensure that the character length of the title fits without truncation. Where truncation is unavoidable, use an ellipsis at the end of the title to reflect this.

#### • Controls

The header should be used for commonly accessed functions such as back, home or search. The functions shown on iOS apps and Android will vary. Android has standardised controls for back and home built into the hardware, therefore these do not need to be replicated on the UI. The controls may change depending on the screen shown and its placement in the information hierarchy. For example, the home screen may contain different controls to other screens in the app. These controls can take the format of icons (used on mobile and tablet), and buttons (only used on tablet). See figure 8.1d for control button example.

#### • Branding strip

A supporting branding strip can be used for apps, which are specifically for a distinct brand or service within the Shell group, to help differentiation if needed. See figure 8.1e.

#### • Background

Header bars in apps should follow the common heights used by native apps for that platform and device.



**Fig 8.1e** Mobile iOS grey on yellow, with supporting branding strip shown below main header area

iPad ᅙ	4:20 AM			32 % 🗈
Back Contents	Heading	search		A 🜌
Tablet iOS app header				
.ul 🔅			~ 🔜 🥢	8:43 PM
Heading	۹	A	< 8	≡∡

Tablet Android app header

Fig 8.1d Tablet header showing icon and button controls

### 8.1.4 **Design rules**

8.1.4.1 Heading

✓ Use the Futura Bold or the default OS font Use Futura for websites, Helvetica Bold for iOS apps, Roboto Bold for Android apps.

Heading text colour should be Shell red or 80% Grey The heading title should be coloured in either Shell red or 80% grey. Do not mix red headings with grey buttons, or vice versa. See Section 5.3 Colour for further guidance.

✓ Heading title should be in sentence case

✓ Heading title should be sized consistently throughout This ensures that apps feel part of the users device.



Fig 8.1f Android header and secondary bar

#### 8.1.4.2 Controls

#### Control colours should be Shell red or 80% Grey The heading title should be coloured in either Shell red or 80% grey. Do not mix red headings with grey buttons, or vice versa. See Section 5.3 Colour for further guidance.

#### Use icons without labels

Iconography should only be used in the header bar for mobile apps, to provide access to tools or navigation. This is because of the variations in label length that will occur when written in different languages. It also allows for greater consistency when developing for both iOS and Android applications.

#### Only frequently accessed tools should be used

Only frequently accessed tools should be shown in the header bar. The exact nature of these tools will be decided by the app. Examples include search, share, login, pinpoint location or full screen.



Fig 8.1g Branding strip shown with the Pecten

#### 8.1.4.3 Branding strip

Only used this where there is a distinct sub brand Only use this device for this purpose; do not use for Shell only branded apps. The colour of this strip will reflect the core brand colour of the supporting brand, if different from Shell yellow. See figure 8.1e.

#### ✓ Use the Shell supporting colour palette Only the supporting colour palette defined in Section 5.3 Colour should be used.

Do not show the branding strip under the Pecten If the Pecten is shown the strip should run up to it, but not display under it. See figure 8.1g.



#### 8.1.4.4 Background

### ✓ Only use Shell yellow as a background

See Sections 8.1.5 to 8.1.8 for variations.

#### **X** Do not change the height of the header

Do not increase the height to include more controls. Instead, use a secondary bar, distinct from the header, to provide additional options if required. See figure 8.1f. For example, an input field may be added below for keyword search functionality, or a carousel showing sections in a magazine app.

# 8.1.5 **Design variations**

This component will have additional design requirements for different OS's and devices. These are as follows:

Header type	Usage		
Native apps	Mobile		
Native apps	Tablet		
Mobile websites	Mobile and tablet		

#### 8.1.5.1 Native apps - Mobile See figure 8.1h

### ✓ Use the default OS font

Helvetica Bold (for iOS apps), Roboto Bold (for Android apps).

#### ✓ User inner bevel for type and icons

This gives our apps a sense of authenticity within the context of the device being used further outlined in Section 5.7 Iconography.

#### ✓ Use the reflective gloss style on the background For native apps use this reflective gloss style on the Shell yellow background.

#### **X** Do not use the Pecten

Mobile apps do not require the use of the Pecten in the header area on any other screen than the home page. The icon used to launch the app from the user's device, and the loading screen will 'brand' the app sufficiently.

#### 8.1.5.2 Native apps - Tablet See figure 8.1i

### ✓ Use the default OS font

Helvetica Bold (for iOS apps), Roboto Bold (for Android apps).

#### ✓ Use inner bevel for type and icons

This gives our apps a sense of authenticity within the context of the device being used.

# ✓ Use the reflective gloss style on the background

For native apps use this reflective gloss style on the Shell yellow background.

# ✓ The Pecten can appear in header using the Alternative exclusion rule

Tablet apps will have sufficient space available for the Pecten to be used in the header; it should be consistently positioned throughout. See Section 5.1 Pecten for further guidelines.

### ✓ Use icons to represent tools

These should be positioned on the right hand side.

#### **Use buttons with labels for access to content sections** For example, home, back and so on. These should be positioned on the left hand side.

iPad 🔶		4:20 AM	32	2% 📭
Back	Contents	Heading Q search	<b>A</b>	2

Tablet iOS header - Shell red on Shell yellow, with search box in situ and contents selection active

.ul 🛜		•	÷ 💻	×	8:43 PM
Heading	۹	А	<	-22	=_

Tablet Android header – 80% grey on Shell yellow, with search box in situ

.ul 📚		•	~ 💻	N	8:43 PM
Heading	۹	AA	<	×	=

Tablet Android header – 80% grey on Shell yellow, with Shell red branding strip shown

Fig 8.1i Tablet native app header examples

#### 8.1.5.3 Mobile websites - Mobile and tablet See figure 8.1j

### ✓ Use Futura for headings

Where Futura is not embedded into the site or when it is not available on the user's device, Verdana should be used instead.

# ✓ Show the title of the website on every screen in the header

#### ✓ Use the Pecten on all screens

The Pecten must be shown on every page of a mobile website as all screens may be accessed through search engines or third party sites. This means the first impression experience is difficult to control. Individual web screens must, therefore, feel immediately like part of Shell's ecosystem.

✓ Use Iconography in the header to provide access to tools If tools are required then these can be accessed through an icon on the right hand side of the header. This is due to variations in label length that will occur when written in different languages.

Do not display sub headings in the header If a subheading is required, use the main content area to show this. See Section 8.9 Titles and section dividers for further guidance.



There is less reliance on web sites to house navigational controls such as back, home, bookmark and so on, as these controls are built into the browser app e.g. Safari.



Mobile Android website header – Shell red on Shell yellow



Tablet iOS website header – 80% grey on Shell yellow

Fig 8.1j Mobile website header examples

# 8.2 **Toolbars**

# 8.2.1 What is a toolbar?

The toolbar provides consistent access to functions that are commonly used for the app or site created. Navigation should be restricted to the header area, but links to home or contents list can be shown.

The different OS's defines these as follows:

Figure	OS	Toolbar name
8.2a	iOS	Tab bar
8.2b	Android	Top bar or bottom bar
		(Dependent on position)



Fig 8.2a iOS toolbar or Tab bar

# 8.2.2 Using toolbars

Nearly every mobile app will have a toolbar fixed to the bottom of



Fig 8.2b Android toolbar, Top bar or Bottom bar

the screen, so these controls are accessible at all times. Tablet apps, because of the greater space available in the header area, may not always require a separate toolbar.

App style toolbars can be used for websites. The positioning and content can vary but duplication of native browser functionality should be avoided.

Please note that toolbars are not a compulsory requirement within an app or site, and should only be used if necessary.
# 8.2.3 Component elements

The toolbar component is made up of these elements. See figure 8.2c:

• Controls

The toolbar can be used for access to tools, which perform specific tasks (search, share, settings) or link to key areas of content (e.g. home, contents, gallery, video, map). These toolbar items should be specifically relevant to the apps' purpose. Only useful, frequently accessed items should be shown in the toolbar, although the most frequent should be shown in the header. See Section 8.1 Headers for further guidance.

Android mobiles have standardised controls for navigation built into the hardware, therefore these do not need to be replicated on the toolbar. For Android tablets, additional controls may be available, such as contents, bookmarking and full screen controls. See <u>Android Design Principles</u>, <u>Style, Patterns and Building Blocks</u> for further guidance.

#### Icons

Iconography should be used to identify the tools included in the toolbar. Use accepted conventions where possible, such as Search, Settings, Share etc. These may differ in iOS and Android devices. See figure 8.2c. Refer to Section 5.6 Iconography for further guidance.



**Fig 8.2c** Toolbar elements, showing Search, Favourites, Share, Settings and More

#### • Labels

Short titles will further identify the toolbar items, and should always be used along with an icon. This is only relevant to iOS, as labels are not used on Android.

### • Active tab

The active tab shows the selected toolbar item.

#### • Background

Toolbars in apps should follow the common heights used by native apps for that platform and device.



#### Android tablet top bar

Fig 8.2d Toolbar elements, showing Search, Favourites, Share, Settings and More

# 8.2.4 **Design rules**

#### 8.2.4.1 Controls

# ✓ Maximum number of toolbar items is five

As spacing between these tools should be sufficient to allow for easy selection. The number of tools shown in the default view can be between two and five.

# ✓ If one item is required use a button

If the toolbar area is used for a single tool on a mobile app, for example taking a picture, use an OS styled button and icon, anchored to the bottom of screen. See figure 8.2e.

## The position of the toolbar should remain consistent throughout the application regardless of orientation For iOS mobile this will always be at the bottom on the app, however on iOS tablets and on all Android devices allow for the toolbar to be positioned vertically left or directly beneath the header. See Section 8.2.5 Design variations for further guidance.

# (i) Where possible avoid mixing the display of tools and content in the toolbar

This should be avoided where possible to reduce any possible confusion and keep a clear distinction within the app or site between functionality and content.



Fig 8.2e Toolbar showing one menu button item

#### 8.2.4.2 Icons

## ✓ Use Shell white for icons

### ✓ Follow iconography rules These are outlined in Section 5.6 Iconography.

## Do not use the Pecten as a toolbar icon

Brand consistency will largely be achieved by using colour for selected states and other UI elements such as the header bar.

#### 8.2.4.3 Labels

## ✓ Use white for labels

✓ Use Helvetica (for iOS)

As labels are not used in Android the label typeface is not required.

- Labels should be in sentence case
- ✓ Labels should be sized consistently throughout
- ✓ Label text should be centre aligned to the icon
- Labels should be short enough to sit together with their respective icons, without wrapping or truncation



#### iOS toolbar – White on Shell red (selected state)



### iOS toolbar - 80% grey on Shell yellow (selected state)



Android toolbar - Shell red on Shell yellow (selected state)

Fig 8.2f Toolbar active tab colour variations

Use this for both landscape and portrait orientations. If this is not possible:

- Use a shorter label
- Reduce the number of tools available
- Add a 'More' function to show remaining tools
- Remove labels and use iconography only. Note that the user is then solely reliant on these icons to identify functionality.

# **×** Do not use Futura for labels

Futura should not be used for labels, as this typeface isn't legible at the smaller size required.

# 🗴 Do not use a label on its own

An icon should always be shown along with the label.

#### 8.2.4.4 Active tool

- Use white for labels and icons on a Shell red background, or 80% grey on a Shell yellow background These are the only allowed colour combinations on iOS. Shell red labels on a Shell yellow background do not pass colour contrast rules at this smaller text size. See figure 8.2f. See Section 5.3 Colour for further guidance.
- Android toolbars can use Shell red on Shell yellow As no labels are required, legibility of text in Shell red on Shell yellow is not an issue.

Use of a reflective gloss style on the tab background This gives our apps a sense of authenticity within the context of the device being used.

(1) Consider using a drop shadow on white labels and icons, and a bevel on 80% grey labels and icons This is consistent with iOS and Android styles. Light labels and icons on a darker background require a shadow to provide depth, while dark labels and icons require a bevel to provide depth. Adding this gives our apps a sense of authenticity within the context of the device being used.

## Do not change the colours used for toolbars throughout the app

#### 8.2.4.5 Background

# ✓ Background colour for toolbars is 80% grey

## ✓ Use a reflective gloss style on the background This gives our apps a sense of authenticity within the context of the device being used, as it is consistent with iOS and Android styles.

# Do not change the height of the toolbar

Toolbars in apps should follow the common heights used by native apps for that platform and device. Do not increase the height to include more tools – these should all sit along one row. Instead, use a 'more' button, which take the user to a new screen with a full list of options. See Section 6.5 Zoning.

# 8.2.5 **Design variations**

This component will have additional design requirements for different OS's and devices. These are as follows:

Header type	OS	Usage
Native apps	Android	Mobile
Native apps	iOS	Tablet
Native apps	Android	Tablet
Mobile websites	All	Mobile and tablet

8.2.5.1 Native apps – Android, Mobile

✓ Toolbar can also be displayed under the header On Android mobile apps the toolbar can also be displayed under the header.



Fig 8.2f Android mobile app showing toolbar under the header

#### 8.2.5.2 Native apps – iOS, Tablet

# ✓ Toolbar can be displayed under the header

On iOS tablet apps the toolbar can also be displayed under the header. See figure 8.2g.

# Additional data feeds can be displayed along with the tools

The extra screen estate for tablet headers allows for additional key tools to be housed here, e.g. Twitter feeds, news feeds. See figure 8.2g. This will allow content to be displayed throughout the app without the user needing to navigate.



Fig 8.2g iOS tablet app showing top toolbar with additional data feed

155

#### 8.2.5.3 Native apps – Android, Tablet

 Toolbar can be displayed under the header
On Android tablet apps the toolbar can also be displayed under the header. See figure 8.2h.

✓ Toolbar can be displayed on the left of the screen The toolbar can be anchored to the left hand side of the UI. This allows sufficient differentiation from the default controls that may be shown along the bottom of the screen. See figure 8.2i.

Additional data feeds can be displayed along with the tools

The extra screen estate for tablet headers allows for additional key tools to be housed here, e.g. Twitter feeds, news feeds. See figure 8.2g. This will allow content to be displayed throughout the app without the user needing to navigate.



Fig 8.2h Android tablet app showing top toolbar with Shell yellow tab shell Design Guidelines for Mobile and Tablet



#### 8.2.5.4 Mobile website – Mobile and tablet

# ✓ The use of toolbars on mobile websites is optional

A more conventional, menu based navigation system may be more appropriate to the mobile website.

# ✓ Use Verdana for labels

If labelling is required then Verdana should be used. If the space available for the labels is not large enough then please refer to Section 8.2.4.3 Labels for further guidance.

# ✓ Position the toolbar under the header

The toolbar should be positioned at the top on the header on mobile websites to mimic the navigational menu conventions that are a standard on websites.

# 8.3 Footers

# 8.3.1 What are footers?

Footers are found at the bottom of a page and contain useful supporting content or links content for an app or website.

# 8.3.2 Using Footers

Footers can balance a website page and provide complementary support to the header or the article.

- They can be used to display important content on a site that doesn't fit into top bar navigation Items such as contact information and help are important to users, but won't always warrant a spot within a header navigation menu.
- They are used to navigate between content areas Footers can be used as links to other sections to allow the user to quickly access different parts of the app or site.

# 8.3.3 Component elements

The footer contains these elements:

- A = Major content categories
- B = Search bar
- C = Social Media Buttons
- D = Site links. This can contain:
  - $\circ$   $\;$  Links to sections on the site
  - o Partner sites
  - o Help/support
  - Contacts
  - o Latest news
- **E** = **Copyright watermark**

Α	 In this site		
	Search the sit	te <del>Q</del>	В
	Follow us: <b>You</b> Tube	= 🗗 🚥 🔊 ·	С
	Media releases	Go to desktop site	
D	 Shares	Contact us	
	Results	Terms & conditions	
	Images	Privacy policy	
	© 20:	12 Shell	Е

Fig 8.3a Page footer example for mobile web

# 8.3.4 **Design Rules**

- ✓ Use a strong background colour to clearly separate the main content and the footer. See figure 8.3b.
- Use headers, dividers, and generous white space to separate content headings. See figure 8.3a.

The separation between headings is important for linking and distinguishing information.

For mobile use a maximum of two columns in portrait orientation

This will ensure all information is legible and suitable hit areas are maintained. See section 6.3 Hit areas.

For a tablet use a maximum of four columns in portrait orientation. See figure 8.3d.

More content can be displayed due to an increase in screen space. Equal column heights should be used where possible, e.g. all columns should have a similar number of menu items.

Increase the number of columns for horizontal viewing Content will spread across the whole page in horizontal viewing, providing more space for additional information.

liquefied natura	al gas (LNG).	~ <del>.</del>	
Read full media n	elease on Shell.co	m 🗗	
Comments			
Inceptos erat, auctor viverra John Waterw	in pulvinar pellentes eu, tincidunt. <b>orth, 2 days ago</b>	que	
Inceptos erat, auctor viverra Jane Smith, 4	in pulvinar pellentes eu, tincidunt. <b>4 days ago</b>	que	
Add your comm	ent		
Your name			
Your messag	je		
	comment		
Leave a			
Leave o			
Leave of Related articles Royal Dutch Shel quarter 2011 eur	ll pic fourth ro and GBP	>	
Leave of Related articles Royal Dutch Shel quarter 2011 eur Royal Dutch Shel new shares	I plc fourth o and GBP I plc: issuance of	>	
Leave of Related articles Royal Dutch Shel quarter 2011 eur Royal Dutch Shel new shares Joint industry co support	l plc fourth o and GBP l plc: issuance of ntribution to	> > >	
Leave of Related articles Royal Dutch Shel quarter 2011 eur Royal Dutch Shel new shares Joint industry co support In this site	I plc fourth o and GBP I plc: issuance of ntribution to	> > >	
Leave of Related articles Royal Dutch Shel quarter 2011 eur Royal Dutch Shel new shares Joint industry co support In this site Search the s	I plc fourth o and GBP I plc: issuance of ntribution to ite	> > >	
Leave a Related articles Royal Dutch Shel quarter 2011 eur Royal Dutch Shel new shares Joint industry co support In this site Search the s Follow us: You	I plc fourth o and GBP I plc: issuance of ntribution to ite	> > >	
Leave a Related articles Royal Dutch Shel quarter 2011 eur Royal Dutch Shel new shares Joint industry co Joint industry co Support In this site Search the s Follow us: You Tube	I plc fourth o and GBP I plc: issuance of ntribution to ite () Go to desktop	> > >	Contrasting
Leave a Related articles Royal Dutch Shel quarter 2011 eur Royal Dutch Shel new shares Joint industry co Support In this site Search the s Follow us: You Tube Media releases	I plc fourth o and GBP I plc: issuance of ntribution to ite () Go to desktop Contact us	> > >	Contrasting background
Leave a Related articles Royal Dutch Shel quarter 2011 eur Royal Dutch Shel new shares Joint industry co Joint industry co Support In this site Search the s Follow us: You Media releases Shares Results	I plc fourth o and GBP I plc: issuance of ntribution to ite Go to desktop Contact us Terms & condition	> > > Q a site	Contrasting background colour
Leave a Related articles Quarter 2011 eur Royal Dutch Shel new shares Joint industry co Joint industry co In this site Search the s Follow us: You Nedia releases Shares Results Images	I plc fourth o and GBP I plc: issuance of ntribution to ite () Go to desktop Contact us () Terms & condition Privacy policy	> > > a site	Contrasting background colour

Fig 8.3b Page footer in context

✓ For information over two columns group content logically and provide section headings

When increasing the amount of content it is important that it is displayed in a coherent format.

			In this site			
In this site			Search the site	Q	Follow us: Tube	E F 🚥 🔊
Search the s	ite Q	Section headings	News	Support	Company	Learn
Follow us: <b>You</b> Tube	E f 💀 🔊		Media releases	FQA	About us	Careers
Media releases	Go to desktop site	Divider	Shares	Customer service	Privacy policy	Shell corporate
Shares	Contact us	lines	Results	Business support	Shell partners	Shell suppliers
Results	Terms & conditions				· · · · · · · · · · · · · · · · · · ·	
Images	Privacy policy		Images	Contact us	Terms & conditions	Shell employees
© 20	012 Shell			© 20	12 Shell	
Fig 8.3c Pac	ge footer on a					

Fig 8.3d Page footer on a tablet (portrait orientation)

mobile (portrait orientation)

# 8.4 Titles and section dividers

# 8.4.1 What are titles and section dividers?

These are components that introduce or separate content. They are applicable to content items such as articles and forms, but can also be used in tool screens.

# 8.4.2 Using titles and section dividers

Titles and section dividers should be used to create structured hierarchy within the screen. This gives the content a more logical flow and aids comprehension.



Fig 8.4a Articles on a tablet and mobile using titles and section dividers

# 8.4.3 Component types

Titles and section dividers include these types. See figure 8.4b and figure 8.4c.

- **Content title (h1)** This is the main title of the content or tool.
- Sub title (h2)

This is a secondary title to lead into another paragraph of content.

### • Section dividers

These can be used to separate different categories of content on a page, different sections within an item of content or be used as category titles.



Fig 8.4b Content titles, sub titles and section dividers on iOS mobile

# 8.4.4 **Design rules**

#### 8.4.4.1 Content title

## ✓ Use Shell red for titles

If the supporting colour palette has been used within the app, it may be relevant to use that sub brand colour for titles. See figure 8.4b.

# Generally, use the default OS font

This is Helvetica for iOS, Roboto for Android and Verdana for websites. See figure 8.4b.

Futura Bold may be used sparing and only above 16pts Futura bold may be used sparingly for headings, where it can be used at a large enough size to provide visual impact. See figure 8.4c.

# ✓ Titles should be in sentence case

If additional impact is required ALL CAPS titles could be used in only if there is enough space to provide a good visual impact and if the font size used is over 16pts. There is no strict character limits on the title; this should be specified within each app or site. See figure 8.4c.

# For websites there should only one h1 title

This is required to follow standard web guidelines (<u>www.w3.org/</u>) and to support Search Engine Optimisation (SEO).



Fig 8.4c Content titles and section dividers on Android tablet

#### 8.4.4.2 Sub titles

### Use Shell red for titles

If the supporting colour palette has been used within the app, it may be relevant to use the sub brand colour. See figure 8.4b.

### ✓ Use the default OS font

This is Helvetica for iOS, Roboto for Android and Verdana for websites. See figure 8.4b

#### 8.4.4.3 Section dividers

## ✓ Use Shell white for text

Use 80% grey backgrounds for related content Use this to separate sections of content that are contained within one subject, e.g. separating main article copy from comments section relating to an article, or to label imagery/video thumbnails. See figure 8.4d.

## Use Shell red for backgrounds to separate types of content

Use this to separate different categories or subjects of content within one screen, e.g. related articles list from main content item. This can be used as a full width of the screen, or at the same width of the screen content. See figure 8.4e.

#### Latest story Updated 20th Jan 2012

#### Royal Dutch Shell sets out new growth agenda



Shell today updated shareholders on progress against its strategic plan to generate profitable growth.

In today's volatile economic environment, the company's strategic aim remains to drive

**Fig 8.4d** Section dividers used to show publish date and label the video thumbnail

Leave a comment	
Related articles	
Royal Dutch Shell plc fourth quarter 2011 euro and GBP	>
Royal Dutch Shell plc: issuance of new shares	>
Joint industry contribution to support	>
In this site	
Search the site	۶.

**Fig 8.4e** Section dividers used to show 'leave a comment' title and 'related articles' list



Fig 8.4f Section dividers using supporting colour palette

## Shell supporting colours can be used If the app or site uses a sub brand colour then a supporting palette colour can be used for a section divider. See figure 8.4f.

# 8.5 List components

# 8.5.1 What are list components?

List components display a group of items to direct a user to another screen, via a short title or introduction to the content.

# 8.5.2 Using list components

Lists can be used to display articles, search results, or menu items. See Section 9.1 Contents list, Section 10.1.3 Settings and preferences and Section 10.2 Search for further guidance on how these lists can be used.

# Depending on the purpose of the lists they could also be filtered or sorted. See Section 10.7 Filtering and sorting for further guidance.

# 8.5.3 **Component types**

List components include the following types. See figure 8.5b and figure 8.5c.

## Contents list

A full list of sections within an app or site, accessed through





the header or toolbar. See Section 9.1 Contents list for further guidance.

### Settings list

Contains a list of items that a user may interact with to change settings. Section 10.1.3 Settings and preferences

## • Search results list

This is a list of results that are returned when the user has actioned a search. Section 10.2 Search for further guidance.

## • Article list

This is a list of content or article items. These can contain both titles and thumbnail images. See Section 9.2 Articles for further guidance.

## Contacts list

A list of contact details providing the user with information on how to contact Shell, e.g. telephone numbers, addresses, and emails. See Section 10.6 Linking for further guidance.

# 8.5.4 Component elements

Label

This title indicates the screen that the content item will direct the user to, or is the name of a group of content items that can be expanded.



Single list item with image

Fig 8.5b List item variations

## Select icons

This is the call to action indicator.

## • Thumbnail

This is a small image relating to the article or item of content within the list.

# 8.5.5 **Design rules**

#### 8.5.5.1 General

✓ Use white for the background

✓ Use 60% grey for dividers

 Use Shell red arrows on the right hand side of the list item
Red is to be used for the icons that describe the call to action

associated with each list item.

#### 8.5.5.2 Label

- Use the default OS font This is Helvetica for iOS, Roboto for Android and Verdana for the web.
- ✓ Use 80% grey for the label
- ✓ Labels should be in sentence case
- ✓ Labels should always be left aligned
- ✓ Label font size should be no smaller than 16pts



Fig 8.5c List item elements

## Do not use Futura for labels

Futura should not be used for labels, as this typeface isn't legible at the smaller sizes required for labels.

#### 8.5.5.3 Icons

✓ Use Shell red arrows or plus icons on the right hand side of the list item

 Other identification icons may also be used
For specific lists such as contact details identification icons can be added to the left of the label. These must be in 80% grey. See figure 8.5d.

#### 8.5.5.4 Thumbnails

- ✓ Thumbnails should be square
- ✓ Thumbnails should represent the content that the list item will take the user to

See Section 5.7 Imagery for further guidance.

- Thumbnails should be place to the left hand side of the label
- ✓ Thumbnails should only be separated by a divider, with no additional padding

### The Hague, The Netherlands Tel +31 70 377 4540

- Tel +44 20 7934 3856
- ir-europe@shell.com

Fig 8.5d Contact list item with identification icons

# 8.6 Form elements

# 8.6.1 What are form elements?

Forms allow for users to submit information, provide content or set preferences in an app or site. Form elements are standard inputs and interaction items that should be used to design and build forms across all OS's and devices.

# 8.6.2 Using form elements

Forms elements can be used in a variety of ways across native apps and websites. As well as registration forms, these elements can also be used as part of tools, such as location search inputs, and manipulating content, e.g. filtering and sorting.

See Section 8.7 Help, Section 8.8 Filtering and sorting and Section 10.2 Search for further guidance.

**NOTE:** The aim of the following examples is to provide a starting point for design rather than specific detailed designs.



**Fig 8.7a** Loyalty registration using a variety of elements



**Fig 8.7b** Station finder tool using search input field

# 8.6.3 Component types

The contents list component is made up of these elements. See figure 8.7c. See Section 8.7 Help for further guidance on inline validation.

• Textfields & text areas

These are text inputs for information such as search keywords, names, email addresses and comments

Check boxes

These allow the user to make either singular (e.g. confirming terms and conditions) or multiple choices

• Switches & toggles

These allow the user to select between two options.

• Drop down menu

This allows the user to select from a list of grouped options.

• Date selector

This allows the user to select a date.

Submit buttons

This is will submit the form.





Fig 8.7c Full list of component elements on a form

# 8.6.4 **Design rules**

#### 8.6.4.1 General

## ✓ Use 80% grey for labels

- Use the default OS font for internal and external labels This is Helvetica for iOS, Roboto for Android and Verdana for websites.
- ✓ Field labels should always be in sentence case
- ✓ Active fields should use a Shell yellow border
- ✓ Fields with an error should use a Shell red border and Shell red internal text

(i) Avoid wrapping field labels on to two lines Use contextual help to provide instructions or improve the clarity of a label if necessary.

#### 8.6.4.2 Text fields & text areas See figure 8.7d.

- ✓ Use default OS styles for text fields
- Align field labels left and above the form element This makes a longer form, but is much faster for users to complete. If space is an issue the field label can be contained



-		
	i	
Textfield, with help		
Textfield, active with help		
	i	
•		
Textfield - inline validation		
Textfield - inline validation Text to be validated		
Textfield - inline validation Text to be validated Textfield - Error		

Fig 8.7d Text fields



#### Fig 8.7e Search text field



Shell Design Guidelines for Mobile and Tablet

within the form field; this should disappear once the user begins typing.

### 8.6.4.3 Radio lists & check boxes See figure 8.7f.

# ✓ Use OS controls with Shell yellow colouring

Align field labels right from the form element This makes a longer form, but is much faster for users to complete. If space is an issue the field label can be contained inside the form input field. See figure 8.7a.

### 8.6.4.4 Switches & toggles See figure 8.7f.

✓ Use OS controls with Shell yellow colouring

 $\checkmark$  Align field labels left or above the form element

8.6.4.5 Drop down menus See figure 8.7f.

✓ Use OS controls

✓ Use a 40% grey outline

✓ Use white with a slight gradient as a background

#### 8.6.4.6 Date selector See figure 8.7h.

## ✓ Use a 40% grey outline

✓ Date selector should open an OS picker wheel on mobile This is easier for the user on a mobile device, rather than inputting dd/mm/yyyy. Tablet devices should use a pop up calendar.

#### 8.6.4.7 Call to actions See figure 8.7i.

- ✓ Buttons should have a Shell yellow background with key line highlights to add depth
- ✓ Buttons can have Shell red or 80% grey labels

## ✓ Consider hit areas

Every form element is likely to require a gestural interaction, and spacing should be sufficient to allow for clear selections to be made. See Section 6.3 Hit areas.

## ✓ There should be one clear submission button

All forms should have a single submission button, which is distinct from any other buttons or calls to action on the screen.

## Lesser actions take a lower priority

Other buttons should be avoided on a form. Links are



Fig 8.7g Drop down menu and date selector with picker wheel pop up



#### Fig 8.7h Submit button

preferable. E.g. forgotten password links. An exception to this is the 'Find an address' button. See figure 9.3c.

# 8.7 **Help**

# 8.7.1 What is help?

It is important for a user to be comfortable within a website or app. This is particularly relevant if the user is interacting with the service or feature for the first time. Guidance on functionality will help establish initial behaviour.

For apps this can be achieved by taking the user through a series of steps, known as demos and walkthroughs, supported by a dedicated help section.

For websites a centralised help area is required, e.g. FAQs.

Please refer to the iOS <u>Human Interface Guidelines</u> and <u>Android</u> <u>Design Principles</u>, <u>Style</u>, <u>Patterns and Building Blocks</u> for detailed descriptions of help within their OS.

**NOTE:** The aim of the following examples is to provide a starting point for design rather than specific detailed designs.

# 8.7.2 Help patterns

# 8.7.2.1 Native app patterns

Pattern	Description
Demo (also known as walkthrough)	A demo normally occurs on first launch of an app. It animates a series of screens showing the prime functions of the application.
Tutorial	Tutorials allow the user to interact and control the animation of the presented information by tapping, scrolling, and swiping. The user is normally presented with small screen shots of the screen they would be seeing in the app. See figure 8.9a.
Contextual help (Single screen overlays)	Single screen overlays serve to point out key interface elements in context of viewing a specific screen. See figure 8.9b.

8.7.2.2 Website help patterns

Pattern	Description
Centralised help area (FAQs)	An area within a website
	dedicated to helping users by
	providing potential options for
	solving key problems.
Contextual help	This is help that is provided
(Single screen overlays)	within the screen the user has
	accessed. This will be
	displayed as a pop up, either
	automatically or user initiated.
## 8.7.3 **Demos and tutorials**

#### 8.7.3.1 Using demos and tutorials for apps

### They should be used to demonstrate the key features for both online and offline actions related to the app

The benefit within a demo or tutorial is that any type of functionality can be displayed.

#### The user should be allowed to quit a demo or tutorial at any time in the sequence

The functionality being presented may not be relevant to a user.

Only use for first launch but provide the user with option to access the information at a later date

Repeating this information every time an application is launched provides no benefit to the majority of users.

## The number of steps should be kept to a maximum of 6 with the length of time being under 30 seconds Anytime longer the user will either lose interest or forget the functionality being shown.

Don't present too much functionality up-front It can be difficult to really understand the content functionality without seeing and using it in context.



Feedback to user

Fig 8.7a Example showing the start of a turtorial for Shell station finder

- 8.7.3.2 **Design rules for demos and tutorials**
- Use it as an opportunity to further express the brand and any key messaging. Please see figure 8.7a.
   Please refer to the Section 5 Style for further information.
- Always provide the user with feedback to where they are in the process. Please see figure 8.7a.
- ✓ The option to exist and leave should be in an obvious and consistent place throughout. Please see figure 8.7a.

## 8.7.4 Contextual help

 $\checkmark$ 

#### 8.7.4.1 Using contextual for apps

 Explanations should be kept short and concise The limited screen estate means that detailed explanations into particular features are not possible. Please see figure 8.7b.

# They are used to explained a key feature or action that occurs in one single screen

This will help the user absorb and related the information to the environment they are in.

**Use them throughout the app (not just at first launch)** They can be used to support more detailed services and tasks that can't always be explained in a demo or tutorial.

Allow the overlay to be dismissed by the user along with providing them with the option to bring it back Accidental tapping can occur on touchscreens therefore providing a way to recover from this is important. See figure 8.97b.



Fig 8.7b Example showing contextual help

#### 8.7.4.2 **Design rules for overlays for apps**

✓ The overlay must be distinguishable from the rest of the activity on screen

This can be achieved by using a background colour that provides a strong contrast.

- Always provide direction cues to the area being discussed on screen. Please see figure 8.7b.
- Allow the user to dismiss overlays in a clear and intuitive way

This should be achieved by tapping anywhere on screen or by providing a close button. Please see figure 8.7b.

## 8.7.5 Centralised help

 $\checkmark$ 

#### 8.7.5.1 Using centralised help area for websites

 Always provide an obvious way for accessing help content on every page of a site

This will prevent the users frustration being escalated further.

✓ Provide a clear and logical way to search and access specific problem solutions

Please see Section 10.7 Filtering and sorting for further information.

Provide another method of contact if the solution to the problem cannot be fixed through the dedicated help area This should be in the form of contact details with email being the first method of access.

## 8.7.6 Contextual help

#### 8.7.6.1 Using contextual help for websites

 Offer quick links for solutions next to areas that might be troublesome for users

See figure 8.7c item 1. This allows users the opportunity to quickly solve the problems.

# Provide users with a range of suitable options for solution

This will allow them to decide on what is the best method for solving the problem themselves. See figure 8.7c item 2.

#### Increase and decrease the potential solution options depending on the viewing device

Be aware that a mobile web service will be viewed on a smaller screen limiting the number of options for suitable viewing.

Do not provide solutions for every potential problem area

Too many can cause confusion.



**Fig 8.7c** An example of contextual help within a mobile website

## 8.9.1 Design rules for contextual help

✓ **Descriptions should be kept short and concise** This will help the user to quickly understand if the proposed option can be of any help to them. Please see figure 8.7c.

 Text should always be accompanied with a call to action and easily identified as a link
 Please see figure 8.7c, item 1.

## 8.8 Notifications

## 8.8.1 What are notifications?

Notifications allow apps to inform the user about important information, or when a time related event occurs. This can be in the foreground while the app is open, or as a push notification that is sent to a user's device even if the app is closed.

The notification could result from a user action, be related to significant system event, or could offer potentially useful information.

Please refer to the iOS <u>Human Interface Guidelines</u> and <u>Android</u> <u>Design Principles</u>, <u>Style</u>, <u>Patterns and Building Blocks</u> for further detail on their use of notifications.

## 8.8.2 Using notifications

It is important to carefully balance the urgency of a notification message against the potential harm from distracting or annoying a user.

## 8.8.3 Notification patterns

Notifications are used to inform the user about specific information:

Pattern	When to notify
Success	When the user initiated action completes successfully.
Failure	User initiated action fails
Non-critical event	Notifies users of significant system events or status that can be safely ignored, at least temporarily.
Optional task	Notifies the user of an optional, alternative or required task that can be safely postponed.
For your information	Notifies the user of potentially useful, relevant information.

#### Unprompted notification

## 8.8.4 Using notifications

#### To inform the user of an event or item that is specifically related or useful to their task See figure 8.10a The exception to this rule is when an unprompted notification is needed, like arrival at a destination. See figure 8.10b.

#### Only use it is really needed

By displaying a notification, you are potentially interrupting users, which can cause annoyance. It is important that the information is relevant and useful to them.

#### Notifications should always be optional

Provide notification settings inside application settings. The user should be in control of notifications and able to modify to their preference.

Pop up notifications can be used for non-critical events that don't require immediate actions

**For critical events a dialogue box should be used** This notification will interrupt the user's actions, so should only be used for critical events. This should provide the user with information or actions to guide them on what they are required to do next. See figure 8.8c.

Use standard OS push notification patterns for notifying the user of messages or updates See figure 8.8d



#### Notification

**Fig 8.8a** An example of a notification within the service. For this to occur the user notification settings don't need to be turned on.



	Net 2012	capital inve	stment of	\$30 billion	n in Nche
l	<b>Q</b> Search	★ Favourites	CAA Share	Settings	••• More

**Fig 8.8b** An example of a location based notification. The user is currently using a different Shell service. For this to occur their notification settings must be turned on.

- When information is not directed specifically to them, their current task or is time sensitive
- If the information in the notification is a duplication of the current information on screen
- If it is to inform the user of errors, when the error can be recovered by the app or site
- Do not use to inform the user of low-level operations if the error can be resolved quickly Instead allow the system to do this without involving the user.

Do not use notifications if the user cannot manually action, recover or stop
 It is important that the user is able to respond to any notification.

Page Title http://www.shell.com C Google Heading C Google Heading C Google Heading C Google Heading C Google Cover St 2 Cover St 2

**Fig 8.8c** An example of an iOS styled dialogue box. The information presented to the user is deemed critial to their task and is related to the OS service.



**Fig 8.8d** A push notification on a launch icon showing there are updates waiting to be viewed.

## 8.8.5 **Components elements**

Notifications are made up of these elements

• Title and/or main body text

The title indicates the key information or action and the main body text provides further details or explanations.

• Background

A strong contrasting background is used.

• Icons

A selection of iconography and call to action symbols are used. Please see Section 5.6 Iconography for further details.



Validated

Error

Used to indicate form validation Used to indicate a form error



Fig 8.8f Standard iconogrpahy used for user notifications

## 8.8.6 **Design rules**

#### 8.8.6.1 Style

✓ Use the default OS font

If the notification is not using an OS alert use Shell white on a Shell red background

Notifications must be displayed in strong contrasting colours to their environment see figure 8.8c and 8.8e. This will help the user distinguish the relevant information for the other activity on screen.

Websites notifications and those that are not directly related to a service or function within an application should follow the standard OS pattern and execution. See figure 8.8c. This will help the user distinguish if information is specifically related to the app or site's features or functionality or if it is related to a service or feature within their device, e.g. it should follow a standard OS pattern if the app is informing the user a service isn't enabled, e.g. WiFi or location services.

Provide directional cues if the information is related to specific location on screen. See figure 8.8e.

#### 8.8.6.2 Interaction

- ✓ If the notification has a call to action the user can press the link to take them to the specified action or screen
- ✓ The user should be able to ignore the notification by pressing a close symbol
- ✓ If the notification requires no call to action the user can press on the notification area to remove the notification from the screen
- ✓ The duration and frequency must match the content and circumstances of the notifications

#### 8.8.6.3 Iconography

 Use standard icons that reflect the notification and the brand

See figure 8.8f. Green is used as a standard indication of a positive action along with the use of Shell graphic language for the square.

# Don't use information symbol for accompanying notifications

Notifications are classed as a form of information. Showing this would just be a duplication of the communication.

#### 8.8.6.4 Text and Language

 $\checkmark$ 

 Always include a title if the information requires a certain amount of detail or explanation
 See figure 8.8e. Users should be able to quickly understand the purpose of the notification with minimal effort.

The title must be abort and consise, summarising t

The title must be short and concise, summarising the most important information to the user

Provide a small section of the notification if it is too long to inform the user in one go

This will allow the user to make the decision themselves as to whether they want to action it.

#### Avoid using imperative, forceful language Describing that the user 'must' do something is not relevant to a patification. Notifications are for non-critical information.

to a notification. Notifications are for non-critical information that the user has the option to ignore.

## 8.9 Maps

## 8.9.1 What are GPS based maps?

GPS navigation is used for planning routes when traveling by foot, car, bike or public transport.

## 8.9.2 Using GPS based maps

#### 8.9.2.1 Using maps for native apps and sites

GPS maps should be used where locations are relevant to the functionality, e.g. Station finder. The default location should be considered, e.g. where location access is allowed the user's current location should be shown.

#### 8.9.2.2 Map API

Google maps should be the default service used to by Shell apps and websites. For further information please see http://www.google.co.uk/mobile.

- Google Maps Navigation is free
- Google maps is available on the following services
  - Android
  - Blackberry
  - iOS 5
  - Nokia/Symbian
  - Symbian OS
  - Windows mobile (including windows 7)
  - Phones with Java-Platform (MIDP 2.0 and up)



Fig 8.9a Satellite image powered by Google maps on an Android mobile

- Palm OS (Centro and newer)
- Palm webOS (Palm pre and Palm Pixi)

Please note: Apple has announced that the Google Maps API will not be available when developing apps for iOS 6. This will need to be considered when developing mobile and tablet apps with map capability in the future.

## 8.9.3 Component elements

The map component is made up of these elements. See figure 8.7a:

#### • Map image

A Street and/or satellite view of a location is always shown.

#### • Search bar

This allows users to enter details or search for key locations and facilities.

#### • Iconography

Icons such as pinpoints and markers are used to inform a user of a location or area. See Section 5.6 Iconography for further guidance.



Fig 8.9b Street view powered by Google maps on a Andoid device

## 8.9.4 **Design rules**

#### 8.9.4.1 General

✓ The default map for all Shell service is Google maps A consistent map service must be used to provide a coherent experience for users across Shell websites and apps.

#### 8.9.4.2 Satellite and street view maps

#### ✓ Provide users with choices

A selection of filtered results should always be shown to allow the user the final decision. See figure 8.9b.

# Always include the option to view results in list as well as in context against a map

Large results in close proximity to one-another can be difficult to view in the context of a map. See figure 8.9b, list icon. On iOS a back icon will appear here to let the user return. On Android the default tools will provide this functionality



Fig 8.9c Directions list and selected state

#### 8.9.4.3 Directions

✓ For directions ensure that items are listed in order of importance

Key information such as time and distance should be towards the top of the screen. See figure 8.9c

Make sure the users current stage is clear and prominent against all other on screen information

It is highly likely that the user's attention is distracted at this point making it even more important that they can process information quickly. See figure 8.9c

#### 8.9.4.4 Iconography

#### Google map icons

Items should follow the standard conventions used by Google maps and the device's OS. See figure 8.9d.

#### ✓ Shell icons

Variation to the standard Google map and OS icons occur when there isn't any existing iconography for the feature or service. For example figure 8.9e is used to distinguish Shell petrol stations from other stations on a map. For further reference please refer to the Section 5.6 Iconography.

# Description box Pin point Shell station ()

Fig 8.9d An example of a Google map icons

#### Shell station



Fig 8.9e An example of a Shell icon

8.9.4.5 Interaction

✓ The ability to zoom in and zoom out should be available on every map

## 8.10 User comments

## 8.10.1 What are user comments?

User comments allow readers to share their opinion on an article. See Section 10.5 Sharing for further guidance.

## 8.10.2 Using user comments

Use where possible to encourage engagement and discourse. Creating a conversation with users will help build a relationship with them and increase their likelihood to share content and engage further. User comments are most appropriate to article type content, but could be used as feature with a tool. Consider how the comments should be moderated.

## 8.10.3 Component elements

The user comments component is made up of these elements.

#### • Comment item

These are the items that comprise of past user comments, author and post date. Consider the character length limitations. These shouldn't be too low to ensure that the user will not get frustrated when leaving comments.

#### Add a comment form

This is a simple form that allows the user to add a message, along with some personal details such as name and email address. This will also include a call to action button. See figure 8.10a.

mments
Inceptos erat, in pulvinar pellentesque auctor viverra eu, tincidunt. John Waterworth, 2 days ago
Inceptos erat, in pulvinar pellentesque auctor viverra eu, tincidunt. Jane Smith, 4 days ago
d your comment
our name
ur message

Fig 8.10a User comments component at the end of an article

## 8.10.4 Design rules

# (i) These design rules are specific to user comments within articles and other content items

If you are using user comment components within a tool based app or site then styling may change to suit its position and layout within it.

#### 8.10.4.1 Comment item

#### ✓ Use 80% grey for all text and icons

- ✓ Use a comment icon to indicate each comment
- ✓ Use section dividers to separate past comments and add a comment form See Section 8.4 Titles and section dividers.

Highlight the author and the comment post date in bold

#### 8.10.4.2 Add a comment form

- Ask the minimum amount of information required This will help reduce frustration when submitting a form on a mobile device.
- **Follow form element design rules** See Section 8.6 Form elements.

## **9 Templates**

A template is a common screen layout that is made up of components. See Section 8 Components. Different apps or websites will require different templates. These templates are closely related to the patterns defined in Section 6.7 Interaction patterns.

In this section

9.1	Contents list	205
9.2	Articles	.212
9.3	Forms	.225

## 9.1 Contents list

## 9.1.1 What is this template?

A contents list is used for indexing information within an app or site. This list should be accessible from either the header or toolbar. The guidelines outlined are suitable for both app and site. Sites that are designed for use on mobile or tablet devices would have their core navigation accessed from this contents screen.

Content lists are likely to exceed the length of the screen, particularly for mobile devices, so scrolling is inevitable.

These screens are generally used for any app or site which is article focused, where information or sub levels can be drilled down to from the list. Patterns that include a contents list are dashboard, magazine and nested detail. See Section 6.7 Interaction patterns.



Fig 9.1a Contents list shown on a tablet and mobile

## 9.1.2 Template layout

The design of the list should balance the need for clarity, by providing some white space between list items, and text that is large enough to read, and efficient use of space as a whole to minimise scrolling and keep key content items in view by default.

See figure 9.1b for grid layout and figure 9.1c for layout and module guidelines.



Fig 9.1b Content list grid layout

## 9.1.3 Components list

The contents list component is made up of these elements. See figure 9.1c and figure 9.1d.

#### Content item

This lists all the content sections within an app or site.

#### • Sub level nested item

These list items are a sub level to a main content list item.

#### • Label

This title indicates the screen that the content item will direct the user to, or is the name of a sub level that can be expanded.

• Icons

This is the call to action indicator.

#### • List container

On a tablet the container for the contents list will change along with the orientation. If the tablet is portrait this will be a pop up. See figure 9.1e. Landscape orientation will show this as a side panel. See figure 9.1f. Mobile devices will not require a container, as they will be viewed as a full screen. See figure 9.1a.



Fig 9.1c iOS mobile contents list

## 9.1.4 **Design rules**

See Section 8.5 List components for further guidance on component list design rules.

#### **Content item**

✓ Use Shell white for the background

✓ Only show one column of items

- Do not add extra columns when the orientation changes Only ever show one column of contents list items. The landscape view of a contents screen on mobile will merely increase the line length.
- Do not increase the size of list elements, change their spacing or alignment when the orientation changes

Sub level nested item

 $\mathbf{V}$ 

- **Use 15% grey for the background** Subsections within a nested list have a light grey background to subtly distinguish them from the main list.
- ✓ Use white for dividers

✓ Always provide a back button to return to previous level



Shares

÷

Closed expandable nested item



Opened expandable sub level item

Fig 9.1d iOS contents list elements

✗ Do not use nesting items down to more than two levels More than two levels of nested items may create confusion.

lcons

- ✓ Use a Shell red arrow for a single list item
- ✓ Use a smaller Shell red arrow for sub level list items
- ✓ Use a Shell red plus icon for a closed nested item
- ✓ Use a 80% grey minus icon for the open nested icon

Do not use any other iconography The use of any other kind of iconography should be avoided in the contents list view.

Label

#### ✓ Use the default OS font

This is Helvetica for iOS, Roboto for Android and Verdana for the web.

#### ✓ Use the default OS font in bold on expandable title This is Helvetica Bold for iOS, Roboto Bold for Android and Verdana Bold for the web.

#### ✓ Use 80% grey for the label

- Use Shell red for the label on selected expandable title for nested items
- ✓ Labels should be in sentence case
- ✓ Labels should always be left aligned
- Labels should be short enough to fit on one line Wherever possible, ensure that the character length of the title fits without truncation. Where truncation is unavoidable, use an ellipsis at the end of the title to reflect this.
- ✓ Sub level nested item labels should be indented Subsections in a nested list are indented to distinguish them from the main list.

Label font size should be no smaller than 16pts

Do not use Futura for labels on native apps Futura should not be used for labels, as this typeface isn't legible at the smaller sizes required for labels.

# Show the contents list alongside content in landscape orientation

Tablet devices in landscape orientation can show the list and the article or section associated with it, rather than having contents on a separate screen.

✓ Show the contents list as a pop up in portrait orientation When a tablet is in portrait orientation, it is beneficial to hide the contents to provide a more 'book-like' experience. It can be revealed either by an overlay or underlay revealed on the left hand side of the article area.

Use a Shell yellow reflective gloss for the pop up background

Contents	
News and speeches	>
Shares	- 4
Quarterly results	>
Dividend	>
Bonds and credit rating	>
Our strategy	)
Publications and presentati	>
Images	- 4
Videos	)
Calender	)
My Library	3



Fig 9.1e iOS tablet contents list in portrait view

Fig 9.1f iOS tablet contents list landscape view

## 9.2 Articles

## 9.2.1 What is this template?

An article is a piece of content made up of a combination of written text, imagery, video, tables and graphs. It can be used for things such as news items, research statistics and company updates.

Articles are likely to exceed the length of the screen, particularly for mobile devices, so scrolling is inevitable.

**NOTE:** The aim of the following examples is to provide a starting point for design rather than specific detailed designs.



Fig 9.2a Different articles shown on a tablet and mobile

## 9.2.2 Template layout

The content of the article is the focus of the layout, especially on a mobile where space is limited. See figure 9.2b for grid layout, and figure 9.2c for layout and module guidelines.

If the app or site is more focused toward magazine or news content, then there is flexibility to re-style the standard components, and although the layout can change, this should still follow the design rules and considerations. See figure 9.2e for how you can adapt the layout and component styles.



Fig 9.2b Grid layout on the article template

## 9.2.3 Components list

See figure 9.2c.



Fig 9.2c Component elements on an example article layout on iOS mobile

Shell Design Guidelines for Mobile and Tablet

The template must contain these components

#### • Content title

See Section 8.4 Titles and section dividers This is the title of the article. A short or truncated version will be required for items such as related links. See Section 10.6 Linking.

#### • Content heading / introduction

See Section 5.5 Typography This is an introductory paragraph to the article.

#### Main body content See Section 5.5 Typography and Section 8.6 List components This is the main content of the article.

The template may also contain these components

#### Time / date stamp See Section 5.5 Typography This is the publish date of the article. It can allow the user to sort and filter article content by date. See Section 8.8 Filtering and sorting.
• Image / Video thumbnails





Fig 9.2d Example article layout on iOS website See Section 5.7 Photography and Section 5.8 Animation and video

This allows images or video thumbnails to be displayed that are relevant to the content.

#### • Section divider

See Section 8.4 Titles and section dividers This allows elements of the article to be broken up into sections to increase the readability of the article.

#### • Sub title

See Section 8.4 Titles and section dividers This allows the article to be easily scanned by the user and improves the readability of the article.

#### • Hyperlinks

#### See Section 10.6 Linking

These are text links within the article that link to other content items.

#### • User comments

See Section 8.12 User comments

This allows the user to read comments and submit their own comment on an article.

#### • Related articles

#### See Section 10.6 List components

This is a list of related content that encourages the user to continue to engage with the app.

#### • Footer

See Section 8.3 Footers

This is generic content that is placed at the bottom of the page. A content footer will be required on all mobile sites, but is not necessary on native apps.



#### Fig 9.2e Example of an article layout on an Android tablet



Fig 9.2f Example of an article in situ showing important content first

### 9.2.4 **Design rules**

### ✓ Show important content first

Important content should be visible without scrolling in portrait view on a mobile, and landscape on a tablet. See figure 9.2f.

### Divide content with sections dividers

Sections of content should be divided with section dividers to improve article readability. See Section 8.4 Titles and section dividers.

✓ The background to an article should always be white

# ✓ Colours and typography should follow guidelines

See Section 5.4 Colour and Section 5.5 Typography.

#### (i) Consider the impact font size

Article text is likely to be rendered in different sizes and typefaces, depending on the platform, user settings (for example, increasing the font size to improve legibility), and could be displayed in a variety of languages, so ensure the layout will cater for this.

#### (i) Ensure the flexibility of content

Ensure that flexibility is created in the layout to allow for text wrapping onto multiple lines, around images and for different orientations. See figure 9.2g.

#### (i) Use the space well

The layout of the article should make optimum use of the screen whilst still providing 'breathing space' around it. Similarly, modules should be spaced sufficiently to allow for differentiation between functions and content types. Keep the space other elements consume, such as navigation, to a minimum.

#### (i) Encourage user engagement

To help discoverability, relevant, related articles should be shown together with functionality like commenting and sharing (via email or social networks) to allow interaction with the content other than merely reading it.



Fig 9.2g Article shown in mobile portrait and landscape views

### 9.2.5 Functional considerations

### ✓ The user should be able to manipulate the content

- Increase or decrease font sizes
- Zoom in and out of images
- o Reading or viewing full screen
- o Searching for keywords or related articles within it

#### Consider whether this additional functionality is relevant to the app or site's purpose

- Allow for commenting and the inclusion of user generated content. See Section 8.12 User comments.
- Copying content into other applications (avoid this facility for photography due to copyright restrictions)
- Bookmarking or adding as a 'favourite' for future reference
- Sharing the article via email or social networks. See Section 10.5 Sharing.
- Rating articles
- Tool tips for additional information

# 9.3 **Forms**

### 9.3.1 What is this template?

This template brings together the form elements described in Section 8.6 Form elements to create forms that should be easy to use. Forms can be frustrating on small devices so should be used sparingly Form design should be assessed stringently to avoid confusion, the inclusion of unnecessary questions or settings, and provide clear calls to action and user feedback.

**NOTE:** The aim of the following examples is to provide a starting point for design rather than specific detailed designs.



Fig 9.3a Different forms displayed on an iOS mobile

### 9.3.2 Template layout

By using established patterns designed for specific operating systems or platforms, the guidance outlined below should be inherited by default. Designing custom form elements, or deviating from convention should be considered carefully.

For guidance on designing forms for Apple iOS please refer to the iOS <u>Human Interface Guidelines</u> available from developer.apple.com.

For guidance on designing forms for Google Android please refer to the <u>Android Design Principles</u>, <u>Style</u>, <u>Patterns and Building Blocks</u> available from developer.android.com/design.

For sites, form elements should reflect the conventional patterns established for web.

See figure 9.3b for grid layout and figure 9.3c and 9.3d for layout and module guidelines.



Fig 9.3b Grid layout on a registration form

### 9.3.3 Components list

These components make up a form

See figure 9.3c for an overview of the form component types.

#### • Form elements

See Section 8.6 Form elements

These are the elements that are used to gather information from the user.

- Text fields
- Text areas
- o Radio lists
- $\circ$  Check boxes
- Drop down menus
- Date selector
- Address finder

#### • Feedback elements

See Section 8.6 Form elements and Section 8.7 Help Forms should be designed so they can be completed as quickly, and as accurately as possible. This is particularly important when on the move. Feedback is crucial in creating a positive experience.

- o Contextual help
- $\circ \quad \text{Inline validation} \quad$
- $\circ$  Error message
- Call to actions



Fig 9.3c Component types on a form

#### See Section 8.6 Form elements

These include buttons, icons and links to allow users to access contextual help, secondary form functionality and submission of the form as a whole.

These components may also be used on a form

#### • Introduction

See Section 5.5 Typography This is an introductory paragraph to the form.

#### • Section dividers

See Section 8.4 Titles and section dividers This allows elements of the form to be broken up into sections to increase the readability of the form.







### 9.3.4 **Design rules**

See Section 8.6 Form elements for design guides on form components.

#### 9.3.4.1 Form design

#### ✓ Make forms easy to use

They should be designed to allow for the quick and easy input of data, using recognisable patterns to communicate the kind of response required.

The way form elements are designed (affordance) should help the user format their responses correctly, for example the use of three separate input fields for date entries that require dd / mm / yyyy, rather than one large input box. See figure 9.3d.

#### Make forms as simple as possible

Forms should be kept as short as possible, and avoid excessive scrolling. The design of the form should balance the need for clarity, by providing some white space between items, and text that is large enough to read, and efficient use of space as a whole to minimise scrolling.

#### Divide content with sections dividers

Sections of content should be dividers with section dividers to improve article readability. See Section 8.5 Titles and section dividers.



Fig 9.3e Date selector using iOS picker wheel

### ✓ The background should always be white

# Colours and typography should follow standard style guidelines

See Section 5.4 Colour and Section 5.5 Typography.

### Use progress indictors where appropriate

When a lot of information is required, split the form over multiple screens, and use a progress indicator to show the form length.

### ✓ Use OS controls where possible

If a complex interaction is required in a mobile app or site, use conventions such as the date picker to facilitate a simpler interaction. See figure 9.3e.

### 9.3.5 Functional considerations

#### 9.3.5.1 Form design

### ✓ Use forms sparingly

Forms should be used sparingly on mobile devices, as they can be tricky to use on the move.

#### Calls to action should be easy to see and press They should be large enough to be both discoverable, and allow the user to tap them easily and accurately. Space

between these calls to action should be great enough to avoid selecting the wrong item. See Section 6.3 Hit areas.

- Mandatory or optional fields should be clearly marked so users know which fields must be filled in for successful completion
- Display a confirmation screen on successful submission Successful submission confirmation is usually shown as a separate page listing next steps, reference numbers and so on. An email can be sent if required as further confirmation.

#### 9.3.5.2 Feedback elements

#### ✓ Use inline validation

This ensures the user clearly understands if there are any errors with the form before they submit it.

#### ✓ Use contextual help

Contextual help provides more detailed information to ensure the correct completion of a form element in the right format. This opens a pop up message to provide additional information on the form element. See figure 9.3f.

#### Use error handling

Error messaging should clearly indicate which elements need correcting to facilitate the eventual submission of the form.



Fig 9.3f Contextual help pop up message

# **10 Common functionality**

It is important to create a consistent user experience across all Shell apps and sites. The following section provides guidelines for standardising common functionality that may be used in the development of native apps and mobile websites.

#### In this section

10.1	Customisation	233
10.2	Search	244
10.3	Registration	251
10.4	Media players	255
10.5	Sharing	260
10.6	Linking	263

# 10.1 Customisation

### 10.1.1 The purpose of customisation

Apps and sites will allow for different levels of customisation. Native apps can save preferences and content to the app itself, while websites and web apps will require a secure area for things like managing preferences, saving content and downloading specific files.

### 10.1.2 Using customisation

Customisation settings will be specific to the app or site. Devicewide preferences may override these settings, e.g. sound or push notifications.

Consider what level of customisation will be appropriate for the app or site. You should think about what preferences will be relevant for an app, and also how the app or site needs to manipulate the user's data. Also consider how the user may want to save and bookmark content within the app or site. See Section 10.1.5 Bookmarks and favourites for further guidelines.

There are two types of customisation:

#### • Active customisation

This is where the user consciously controls the settings or preferences of an app or site, changing default settings or the storage of personalised data.

• Settings and preferences

- Saved profile data
- $\circ$  Bookmarks and favourites

#### • Passive customisation

Passive customisation is where the app or site adapts as the user is interacting with it, usually without the user's knowledge: it generally happens in the background.

#### $\circ$ Session data

### 10.1.3 Settings or preferences

#### 10.1.3.1 Using settings and preferences

The settings and preferences of an app can be used to personalise an application specific to that user, e.g. language, sounds or units used (miles or km), or could be used to control how the app will interact with other networks, e.g. push notifications. Once these changes are set, the app retains these choices until the user changes them.

Additional preferences could be required for specific modules, e.g. within the home screen – see Section 7.3 Home screens - Dashboard view Magazine view. The app or site can allow the user to have enhanced control over these modules. The settings for these would be controlled within the preferences screen, allowing the user to modify the following:

- Which modules are shown
- The order in which the modules appear
- The content preferences for each module.

10.1.3.2 **Settings process** See figure 10.1a.

Step 1. Selecting settings The user can select a top-level settings screen from either the header, toolbar or contents list.

Step 2. Amend setting The user should amend their settings using standard OS controls. Step 3. Confirming settings

After the user has updated their settings they will need to consciously confirm their changes. The user should also be able to cancel the changes. There should be a clear distinction between these controls.

II SHELL 3G	17:20	Ē
<	Settings	=
Language		English 🗲
Distance		Mi Km
Sounds		>
Sharing		>
Push notifi	cations	ON
Q Search	Č Settings	CC Share

Step 1. Top level settings screen

Fig 10.1a Settings screen process

.III SHELL 3G	17:20	Ē
<	Language	=
English		✓
Français		
Deutsch		
Nederland	s	
Italiano		
Español		
Português		
Dansk		
Suomi Q <sub>Search</sub>	Settings	Share

Step 2 - Second level settings screen – user has selected 'English'

II SHELL 3G 17:20 📼
<ul><li>≺ Language Ξ</li></ul>
English
Français 🗸
Deutsch
Nederlands
Italiano
Español
Bestumike
Confirm
Canaal
Cancel
Norsk (bokmal)

Step 3. Confirmation pop up

#### 10.1.3.3 Functionality rules

#### ✓ Use standard iOS controls for settings

Settings and preferences screens should use standard OS input controls, e.g. picker wheel, switches/toggles. Please refer to the iOS <u>Human Interface Guidelines</u> and <u>Android</u> <u>Design Principles</u>, <u>Style</u>, <u>Patterns and Building Blocks</u> for a full list of interactions, and variances between platforms. These should use Shell colours to enhance the brand experience.

✓ The user will be required to confirm their changes

#### An undo option should be available if settings are potentially harmful

If the settings changed are potentially harmful, or the settings changes are complicated, then consider including an undo option after the confirm action has been selected. This will allow the user to return to the previous setting.

#### Do not include additional content

Content such as terms and conditions, versioning or corporate information should not appear within this section. These should have a separate information screen.

### 10.1.4 Saved profile data

#### 10.1.4.1 Using saved profile data

To access the saved profile data, users may be required to sign in to a secure area, which would be password protected. See Section 10.3 Registration. Apps and websites will require different levels of security.

#### Native apps

Native apps tend to require less strict security rules, as data is stored on the phone. Most likely, the user will only be required to sign in once. However, the security level is also dependent on what information is being sent. If so, the user may be required to sign in for additional security details every time they open the app.

#### • Websites and web apps

Depending on the level of security required, the user may be required to sign in for every session (high security) or their details would be stored by a cookie or the app so signing in isn't required unless the user signs out (low security).

Saved profile data could include name and contact details, saved vehicle information, previous searches and reminders or alerts. See Section 9.3 Forms for further guidance.



Fig 10.1b Registration screen allowing the user to save their loyalty card details to a native app

#### 10.1.4.2 Functionality rules

Make the sign in process as simple as possible Ensure that only the minimum information required is requested to allow the user to save their profile data. This helps reduce any user frustration. See Section 10.3 Registration for further guidelines.

Do not add a sign in process unless it is essential Using forms can be frustrating on mobile devices, therefore only add registration and sign in processes if they are essential to the site or app. See Section 10.3 Registration for further guidance.

### 10.1.5 Bookmarks and favourites

10.1.5.1 **Using bookmarks and favourites** If the app or site has a high level of written articles and content is may be appropriate to allow the user to bookmark and favourite content items.

#### 10.1.5.2 Bookmarks and favourite process

#### Step 1. Selecting an item

There are two ways for the user to bookmark or favourite content depending on they key purpose and functionality of the app.

#### • Main tool feature

If content is the key purpose of the app then bookmarks or favourites should appear on the toolbar. In this instance on each content item a 'bookmark' or 'favourite' icon would appear in the header to allow the user to bookmark or mark the item as a favourite. See figure 10.1c.

#### • Content interaction

If content is additional to the key purpose of the app, but still relevant then at the end of each content article or item additional buttons should appear to allow the user to 'bookmark' or 'favourite' the content. See figure 10.1d.



**Fig 10.1c** Main tool feature - user can 'favourite' content from the header, and view 'favourites through the toolbar

#### Step 2. Viewing bookmarks and favourites

The user must be able to find and view any items they have bookmarked or marked as a favourite. This must be achieved through main navigation, e.g. the toolbar, or on a contents list.

#### 10.1.5.3 Functionality rules

 Ensure that the user can easily find content they have bookmarked or marked as a favourite

This may be via the toolbar, from the more list or contents list.

Do not create additional bookmarking tools for websites Do not replicate bookmarking functionality within a mobile website, as the browser already utilises these tools.

#### ...II SHELL 3G 17:20

for 2012 targets established.

- Shell expects to grow the dividend in 2012, reflecting an improving financial position.
- Net capital investment of \$30 billion in 2012 as Shell invests for a new tranche of growth.
- Measured increase in spending and payout some 30-50% higher than the 2008-11 total.
- Growth outlook driven by over 60 new projects and options including liquefied natural gas (LNG), deep water, tight gas, liquids-rich shales and traditional plays.

Read full media release on Shell.com



**Fig 10.1d** Content interaction - user can select to 'favourite' content at the end of an article

### 10.1.6 Session data

#### 10.1.6.1 Using session data

A session is where an app or site is still active in the background, although not being currently used. On the web, cookies will save this data to the user's device. The session data could include font sizes, screen magnification, current location, target location (e.g. within route planner), and current search values.

#### 10.1.6.2 Functionality rules

Native apps

✓ Data will be remembered if the user leaves the app running in the background

✓ Data will be reset if the app is properly closed down rather than put to the background

#### Web apps or websites

Data will be stored in cookies until the cookie has been cleared or deleted

See Shell's cookie guidelines for further details.

## 10.2 Search

### 10.2.1 The purpose of search

Search is used to allow the user to easily locate specific content by entering a search parameter matched against a set of data to produce a set of relevant results.

To allow the search functionality to work successfully a content strategy needs to be in place, ensuring that all content is tagged and categorised and cross-referenced correctly. This will ensure that when a user inputs a search term, the correct and relevant results are returned.

Wherever possible, retain the characteristics and features of inbuilt search functionality within the device OS or web browser to aid familiarity and consistency with other apps and sites.

### 10.2.2 Using search

Search is a great way of cutting through information guickly, and is desirable functionality to include in a mobile app or site.

#### • Use search when:

There is extensive content that the user might want to explore. This also requires that the content is curated effectively, e.g. the correct meta data and categorisation is being added to the content, so that the user can find relevant content when they search.

#### Don't use search when:

There is a small amount of content or tools that the user can easily navigate through.

#### 10.2.2.1 Types of search

The search process is made up of two parts: the search term (what type of information is the user looking for) and the results (a list or map of all items that match the search term).

The different types of information the user can search will define how the search functionality works, as well as the format of the results produced e.g. a list, points on a map, or a gallery.

**Content search** See figure 10.2a. • The user enters a keyword. This triggers a search through

.ıl 牙 10:08 AM	I 🖅 10:08 AM	
Search	Search	
Enter a keyword	Shell	Q
No recent searches	Offer by Shell for Cove	>
	Royal Dutch Shell plc first quarter 2012	>
	Shell Sustainability Report	>
	CNPC and Shell sign first shale gas	>
	Royal Dutch Shell plc fourth quarter 2011	>
	Royal Dutch Shell plc Board announces	>
< ★ < ♥ :		

User selects search toolbar item User enters a keyword and a list and is shown the search input

of matching items is returned with the keyword highlighted

Fig 10.2a Content search

Se

content to find matches for that keyword either through the title, tags, category or content. This can produce a results list, or if the content being searched are images or videos the results may be presented as a gallery.

#### Location search See figure 10.2b.

The user enters a location, e.g. postcode or place name. An API will be used to search Google Maps and produce results that can be displayed in either a map or list format.

#### Data list search

The user enters the name of an item, e.g. car type. This item will then be checked and matched to a specific item list within a table or database of results. The user will be presented with a list of results, or if appropriate, a gallery. See figure 10.2e.

#### 10.2.2.2 Storing data

This is where the searchable data is stored. For native apps it is preferable that the data is stored within the app, for quicker access, as well as availability offline. Data stored online may also be accessed by an app, e.g. Google Maps location data. For sites, all data would be stored online.



User enters a search location

Fig 10.2b Location search

Results are returned on a map

Ð

### 10.2.3 Search process

#### Step 1 – Accessing search

Search can be accessed through an icon in the header, which would open a new screen, or a search panel within the app or web screen. This can be placed at the top of the screen, under the header, within the header on a tablet, at the bottom of the screen, and also at the top of a content list. For web this may appear within the content of the screen. The search functionality will operate in the same way on an app and website. See figure 10.2c.

#### Step 2 – Entering a search term See figure 10.2a.

When a user activates the search input box the keyboard should appear, allowing them to input their search term. The keyboard will include the search call to action so the user does not need to leave the field to activate the search. The type of keyboard presented (e.g. letters, numbers, URL specific) should be relevant to what the user is entering.

#### Step 3 – Search results returned

The results can be displayed in different ways depending on the type of search carried out.

#### List results

This will display a list of results. The user can select an item to view it in more detail. This view can also support additional functionality depending on the type of search and data location. See figure 10.2a.





- **Live results** As the user inputs a keyword, results can be shown instantly. See figure 10.2a.
- Predictive search As the user inputs a parameter, suggested items that match the input can be displayed, e.g. if the user starts inputting the name of car, then the app can predict and suggest the complete names. This will require a table of information for the app or site to read from. Relevant previous search terms should be shown first. These types of functionality may be limited if the data location is outside of the app, and the user is offline. Predictive search pop ups will function the same on iOS and Android displayed using the standard OS styles. See figure 10.2d.
- Map results See figure 10.2d.

Google maps should be used to display location results as pins on a map page. The user can select a pin to view pop up information, then carry out an additional action, e.g. get directions to that location. The option to switch between a map and list view should be available in this instance to give the user the flexibility to view results as they desire.

#### • Gallery

Image or video search results can be displayed as a gallery. The user can select a thumbnail to view the item in more detail. See figure 10.2e. See Section 10.4 Media Players for further guidance.



Station 'pins' showing location search results

**Fig 10.2d** App has remembered user's previous searches

#### Step 4 – Manipulating search results

Complexity and a high number of search results can mean filtering or sorting is appropriate. This will depend on how the content has been tagged and categorised. This can be performed before the search has been actioned, or when the results have been returned.

#### • Filtering

Allows the user to change the way results are presented, e.g. when searching for nearby stations results can be filtered by fuel types – diesel, unleaded, LPG. See Section 10.7 Filtering and sorting for further guidance.

#### • Faceted search

Allows the user to add multiple filters to browse content. Use this conservatively as the display may look confusing on small screens.

#### • Sorting

Allows the user to re-order the search results prioritising elements of the content, e.g. by nearest location. See Section 10.7 Filtering and sorting.

Consider the best way to allow the user to control filtering and sorting. This could be handled using standard controls such as a drop down menu for simplicity. See Section 9.3 Forms.



Fig 10.2e Gallery results

### 10.2.4 Functionality rules

✓ Ensure the user can easily access search

- ✓ Use filtering, faceted search and sorting if the user will be presented with many search results
- ✓ Use the relevant search results type, and provide the user with different formats where appropriate

# 10.3 Registration

### 10.3.1 The purpose of registration

The sign in and register processes should be standardised to ensure the user has a consistent experience using secure areas and signing up for services. See Section 9.3 Forms for further guidance.

### 10.3.2 Using registration

Form filling can be a frustrating process on a mobile device, and should only be used if required, and kept as simple as possible.

Registration may involve creating a profile or account to sign up for updates or services, which will involve sending information such as an email address to receive newsletters. This may involve further verification by the user.

It is important that the security of the app or site is clearly communicated to the user. On the web this will be achieved through the security icon used in the browser header, while on an app this can be achieved through the use of icons in the header.

If a secure area is required in the application, consider how much content should be available before and after sign in to clearly communicate the benefits of registration; for example, a tour of the app functionality once registration has been made.

### 10.3.3 Registration process

Step 1 – Accessing registration or sign in There could be three entry points to a sign in / registration process:

#### 1. Landing screen

If the app or site requires the user to sign in for full access, then the initial screen will contain a sign in form with an additional link for the new users to register.

#### 2. Throughout the app or site

If the app or site allows the user to view content pre sign-in, then throughout the app or site there may be links to the sign in / registration process. See figure 10.3a.

#### 3. Header icon

If appropriate, an icon on the header will prompt the user to sign in or register. See figure 10.3a. The user will be taken to a sign in screen, with a link to register if they do not have an account.

#### Step 2 – Registration

The user should be able to register for any service. If appropriate Facebook connect and Twitter can also be used to allow the user to register for an account. This is only appropriate for B2C users. Once the user has registered for a service they should be able to sign in immediately. See Section 9.3 Forms for further guidance.



Fig 10.3a Sign in icon in header



Fig 10.3b Sign out icon in header
#### Step 3 – Sign in See figure 10.3c.

For an app the user should only need to sign in once. They can then delete their details within a settings screen. See Section 10.1 Customisation. On the web this will be if they sign out or delete their cookies. The user can manage the sign in details from a settings screen. A sign up form may also appear within context, e.g. at the bottom of an article screen.

Once the user has signed in then the app or site will remember their details, and the header icon will change to allow the user to sign out. See figures 10.3a - b.

If icons cannot be used to display the account status a sub title within the header bar should be shown. Once the user has signed in or registered a confirmation screen will be shown.

#### Step 4 – Sign out

The settings screen should have an option for the user to sign out and sign in as a different user (see Section 10.1 Customisation). If the app or site has a sign in icon in the header, once signed in this will change to a sign out icon, allowing the user to sign out from here.



Fig 10.3c Sign in screen

### 10.3.4 Functionality rules

### ✓ Make it as simple as possible

The registration form should ask for the minimum amount of information that is required to complete the process.

### Provide confirmation

After submitting a form the user should be taken to a confirmation screen. This ensures that the user can be confident the form submission has been successful.

### Add contextual help where appropriate

Supporting information should be applied to form fields where the user might require some additional help in filling out a form, e.g. explaining why a mobile number is required, or where to find a loyalty card account number.

### Use validation to make the process easier

If the user has entered incorrect details into any of the sign in or registration forms they will be shown an alert message and the incorrect form items will be highlighted with an 'X' icon and a red outline. Inline validation should also be used where possible to give the user an understanding of any errors in their inputs before submitting the form. See figure 10.3d.

### Use password masking

Passwords should be masked by default and should follow the operating system standards, i.e. on iOS the last letter typed is shown, on Android the option to show password is offered.

See Section 9.3 Forms for further guidance.

Form title		Q
n dolor sit amet ut		
asellus, quisque.	et. An	iet
th		~
ail*com		×
git card number		i
	▼	
Submit		
	th ail*com git card number Submit	th ail*com git card number

Fig 10.3d Form with validation

## 10.4 Media players

### 10.4.1 The purpose of media players

Media players are components that are used to display videos, images and audio. These components should look and function in a consistent way, while following the specific device platform variations. See Section 5.8 Animation and video for further guidance.

### 10.4.2 Using media players

A gallery should be used to display multiple media items, e.g. images, video and audio. A media player will also be required to display and control the video or audio content. This should use standard controls.

### 10.4.3 Media types

### 10.4.3.1 Images

If a single image needs to be shown in full size rather than in context as a supporting image, a zoom icon should be added to indicate the image can be viewed in full screen size. See figure 10.4a. A gallery module can also be used to display multiple images. See figure 10.4b.

#### 10.4.3.2 Video

Video content will always be represented with a play icon on the thumbnail. Videos will only play if the user has pressed the thumbnail to open the video rather than automatically. See figure



Fig 10.4a Image with zoom icon

10.4b. This will be displayed in a proprietary player, or if the video is sourced outside of the app or site, e.g. on YouTube or Vimeo, it will be displayed using their players.

Consider how videos will be optimised and streamed to ensure that the user can easily view the video, given the variation in signal strength when on the move. If the video uses up a high amount of data allowance then consider warning the user if they are not on a wireless connection.

### 10.4.3.3 Audio files / podcasts

Audio files will be represented using a standard thumbnail with an audio icon. Imagery should only be shown if it is specifically relevant to the audio file, e.g. a profile picture of the person speaking. See figure 10.4b. Audio files will only play if the user has pressed a thumbnail rather than automatically. This should be integrated into the app functionally where possible. Downloadable podcasts should be played in the OS standard player, such as iTunes for iOS or Google Listen for Android. See figure 10.4f.

### 10.4.4 Media player process

### Step 1. Accessing the media

The user can access a media item in two ways, either through the context of another content item, such as an article See figure 10.4c, or by using the header, toolbar or contents list to view a gallery of multiple media items. The user will not be able to play a video within the context of an article. Pressing the play button will take the user to a media player. See Step 2. Exploring media and Step 3. Viewing media, below.



### Fig 10.4b Audio file thumbnails amongst video thumbnails



### Fig 10.4c Video in context of an article

#### Step 2. Exploring media

When multiple media items are contained within an app or site they should be displayed together within a gallery screen. The gallery has different display formats that will allow the user to select, control and view media.

#### • Grid view

The grid view is a screen of thumbnails which preview the content and indicate the media type, e.g. image, video or audio. When the user presses a thumbnail they will be able to view the media in more detail. See Step 3. Viewing the media, below. On a tablet the grid view can show additional information, such as title, video length and views. As the device orientation changes the grid view will reposition to fill the screen. See figure 10.4d.

#### • Carousel view

There will be a main item displayed (if the user has arrived here from the grid view then this will be the item selected by the user). There will also be a mini carousel underneath the main item that will allow the user to explore the other gallery content. As the device orientation changes the carousel controller will move to the bottom of the screen. See figure 10.4e.



Fig 10.4d Videos displayed within a gallery, grid view



Fig 10.4e Videos displayed within a gallery carousel view

Step 3. Viewing media

The user can view a media item in closer detail.

### • Player view

The player will show a full size view of the item with the file information and scrubber bar. The user can access the full screen player by pressing the full screen icon, e.g. **S**. This is the default view. As the device orientation changes the view will re-orientate. See figure 10.5f.

### • Full screen view

This will show a full screen of the item, e.g. a full size video, image or audio player. The user can press the screen to show the header and scrubber bar, and from here go back to the previous screen. The full screen view will not change orientation until the user changes the devices orientation.



Fig 10.4f Video presented in a YouTube player

### 10.4.5 Functionality rules

 Consider what media players are appropriate to the app or website

If gallery content is a key purpose of the app then this should be a main navigation option, otherwise it is only accessed through the content in context.

✓ All media types can be displayed within a single gallery Different media types can be displayed together. However, icons should be used to signify the media type before the user presses the thumbnail. See figure 10.4b.

✓ Media should be played within the app where possible It is preferable to keep the user within an app or website, rather than send them to a third party site, e.g. YouTube.

# 10.5 Sharing

### 10.5.1 The purpose of sharing

It is important to allow for sharing of content (sections of content, pictures, videos or whole articles) through other networks or editing software. The user can share by posting through their social media accounts, printing, emailing, and copying or by sharing opinions, e.g. leaving a comment on an article. There should be a standardised method for the user to achieve this on both apps and sites. The level of copyright that applies to the content should influence sharing options, e.g. copyrighted images can not be copied to the user's device.

### 10.5.2 Using sharing

Firstly determine what should be shared and how. This should be appropriate to the apps purpose and the type of content (article, image or video). Consider whether sharing is integral to the app or site's key functionality, or if it is a way for the user to share content that interests them.

### 10.5.3 Sharing process

### Step 1. Choosing to share

The user can select share throughout the app or site using a standard share icon ( $\Box < i$  iOS and Android). There are two ways for the user to share content depending on the key purpose and functionality of the app.



Fig 10.5a Main tool feature - user can 'share' content from the header, and toolbar

#### • Main tool feature

If sharing content is the key purpose of the app then share should appear on the toolbar. In this instance on each content item a 'share' icon would appear in the header to allow the user to share the item. See figure 10.5a.

### • Content interaction

If the content is additional to the key purpose of the app, but still relevant then at the end of each content article or item additional buttons should appear to allow the user to 'bookmark' or 'favourite' the content. See figure 10.5b.

### Step 2. Selecting share method

Selecting an item of content to share will open a pop up menu prompting the user to select how they want to share the app. See figure 10.5c.

### • Social media

Sites such as Facebook, Twitter and Linked In have APIs that can be used to allow the user to quickly sign in to their account and post the item that they want to share. These APIs can be integrated so the user does not need to leave the app or site.

### • Email

The email share link can connect to the users default mail account and allow them to email content without leaving the app or site. Once the email has been sent, the user should be returned to their previous screen.



Fig 10.5b Content interaction - user can select to 'share' content at the end of an article

#### • Print

The print option should only be displayed if it is appropriate to the content. If the user is connected to a printer it will allow them to print the content and return to the previous screen.

#### Step 3. Sharing opinions

At the footer of a content item, such as an article, the user can input and submit a comment. This will display a confirmation and a link to return the user to the content. See Section 8.10 User comments for further details.

### 10.5.4 Functionality rules

### ✓ Use tool APIs where possible

Integrating sharing tools, such as Facebook and Twitter will create a better user experience as the user will not have to leave the app or site.



Fig 10.5c Share pop up menu using standard OS controls with Shell colouring

# 10.6 Linking

### 10.6.1 The purpose of linking

It is important to include hyperlinks throughout the app or site to ensure that the user can access information or tools of interest to them, as well as increasing user engagement. For web this is also important for SEO.

### 10.6.2 Using linking

There are three different types of linking that will be used in a native app or mobile website.

### • Hyperlinks

Text links that will take a user to another item of content.

### Contact information

Contact details may also be displayed, therefore it is important to utilise these device features, and allow the user to quickly use the contact information, e.g. link straight to calling or emailing services.

### Related content

Related content links are list items that relate to the content the user is currently viewing, and prompts the user to explore content further.

### • Linking between apps and sites

This is where content is pulled in from a website into an app

to ensure that the user does not leave the app, and therefore improves their experience.

### 10.6.3 Hyperlinks

### 10.6.3.1 Using hyperlinks

Hyperlinks are text items that will link the user to a different screen, or external content. These will tend to be used within article content.

### 10.6.3.2 Functionality rules

# ✓ It should always be visually obvious to the user that an item or text is a link

These should always have an underline and be shown in a different colour. See figure 10.6a.

### Highlight items that are external hyperlinks

If the link is taking the user to another location, e.g. an external site, the hyperlink should include an external link icon to denote this. See figure 10.6a.

### Provide distinct active and visited states

- Multiple links together, these should be displayed as a bullet list
- ✓ Calls to action should appear under the main content
- Contextual links should appear within the body copy as a part of the sentence.



Fig 10.6a External hyperlink.

### (i) Avoid the use of generic call to actions

Avoid the use of generic call to actions like "read more" or "find out more", and instead use more descriptive sentences. These are shown to have higher click rates.

### **X** Do not use single words for hyperlinks

Avoid using a single word as a hyperlink to make sure the hit area is large enough. See Section 6.3 Hit areas for more guidance.

### 10.6.4 Contact information

### 10.6.4.1 Using contact information

This information is made up of contact methods, such as telephone, email and websites (see Section 10.6.6 Linking between apps and sites). The user should be able to select an item on these contact details and be directed to the relevant services, e.g. calling or emailing.

### 10.6.4.2 Functionality rules

- ✓ Where possible, use conventions deployed by the OS or browser to access contact functionality
- Selecting the title of a contact should provide a pop up menu for the user

The specific content on this menu will depend on what contact information is associated with the contact, e.g. if there is an address, phone number and email it would have the following items: show on a map, call the number, send an email and add to contacts (with all the contact details attached – address, phone number, email address).

Selecting an address should provide a pop up menu for the user to show a location on a Google map, add to their contacts or copy the address.



- Selecting phone numbers should provide a pop up menu for the user to call the number, send a text message, add to contacts or copy the number.
- ✓ Where relevant, opening hours should be displayed, e.g.
  9am to 5pm, Mon-Fri.
- Pressing email addresses should provide a pop up menu for the user to send an email, add to contacts or copy the address.

### 10.6.5 Related links

### 10.6.5.1 Using related links

Throughout the app or site relevant links should be included where possible. This allows the user to continue to engage with content on the app or site, and supports SEO. See figure 10.6c.

### 10.6.5.2 Functionality rules



This should include defining content subject, category, tagging and date.

Royal Dutch Shell plc fourth guarter 2011 euro and GBP equivalent dividend payments > Royal Dutch Shell plc: issuance of > new shares Joint industry contribution to support > community projects in Somalia ō. Q A ... Search Favourites Share Settings More

Fig 10.6c Related links

### 10.6.6 Linking between apps and sites

10.6.6.1 **Using linking between apps and sites** Where possible the user should not leave the app or site. If the user is in an app which links to a Shell website, then that content should be pulled into the app, allowing the user to easily return to the previous app screen. See figure 10.6d.

### 10.6.6.2 Functionality rules

### ✓ Pull web content into the app

If an app needs to use web content then this should be pulled into the app rather than sending the user away from an app.

### ✓ Hide all website header and footer elements

The user should not be aware that website content has been pulled into the app. This ensures a smooth user experience.

### ✓ Take the user to the appropriate store

When websites linking to an app the link should always take the user to the appropriate page on the App store or Android marketplace.

(i) Where this isn't possible, use a browser within the app See figure 10.6e.



### Save up to 10p per litre off Shell fuels

We are offering customers the opportunity to save up to 10p per litre off Shell fuels when purchasing a selected range of shop products.

Customers can qualify for an instant discount of 2p per litre off Shell fuel purchased, up to a 50 litre fill, when purchasing one of a selected range in Shell shops.



Original article in a mobile website

II SHELL	3G	17:20	Ē
<	N	ledia releases	=

# Save up to 10p per litre off Shell fuels

We are offering customers the opportunity to save up to 10p per litre off Shell fuels when purchasing a selected range of shop products.

Customers can qualify for an instant discount of 2p per litre off Shell fuel purchased, up to a 50 litre fill, when purchasing one of a selected range in Shell shops.

The discount can be increased by buying more than one of the selected products, and customers could receive up to 5 deals in total to receive a 10p per litre discount!

Products	s included	in	the	offer:	Coca	Cola	2

Q	$\star$		¢	•••
Search	Favourites	Share	Settings	More

Article being displayed within the app

Fig 10.6d Website article content being displayed in an app



**Fig 10.6e** Browser displaying article content within an app

# 10.7 Filtering and sorting

### 10.7.1 The purpose of filtering and sorting

When a user is exploring content or search results they may want to refine the content to more easily find content that is relevant to them. They key ways to do this is through further filtering or sorting.

### • Filtering

Allows the user to choose from pre-defined items to reduce the amount of results or items of content shown.

### • Sorting

Allows the user to order the content or results by their preference, e.g. alphabetically or nearest location.

### 10.7.2 Using filtering and sorting

Filtering and sorting should only be used if there is a high level of content that the user must explore through. This is especially relevant when the user is searching for content items. See Section 10.2 Search for further guidance.



al 🖅 '	10:08 AM
Filter by	*
Shell V-Power	~
Shell V-Power Diesel	~
Shell FuelSave Unleaded	~
Shell FuelSave Diesel	~
Unleaded	~
Diesel	
Autogas (LPG)	
♀ ≡	•

Search results with filtering and sorting as part of search tool Selecting a filter or sort icon take the user to a settings screen. User can go back to view updated results

Fig 10.7a Station finder example, filtering on an Android mobile

### 10.7.3 Filtering and sorting process

### Step 1. Exploring content

When the user is either viewing content through a search result, or browsing articles they should be provided with the option to filter or sort content. This can be provided in the header area as an icon or button see figure 10.7b, or under the header as part of a tool. See figure 10.7a.

Step 2. Choosing filtering or sorting settings Selecting a filter or sort icon will take the user to a settings screen where they can select their preferences. For mobile this will be a new screen, while on a tablet this will be slide over or a pop up. For filtering the user will be able to select multiple options, while for sorting only one option can be selected. See figure 10.7a.

### Step 3. Update content

The user should press back to update the content. This will be part of the Android action bar, while for iOS the back icon will appear in the header.



Filter appears in top left of the header as a button



Filter menu slides in from the left

Fig 10.7b Inside Energy example, showing filtering on an iOS tablet

### 10.7.4 Functionality rules

- ✓ Use when there is a lot of content for the user to explore If the content is not tagged properly then consider whether to exclude this functionality or update the content.
- Ensure the filtering and sorting options are easy to find If filtering and sorting is a high priority for the content then these options should be clearly displayed on the header or above the content.
- ✓ Allow for multiple filtering options where appropriate
- $\checkmark$  The user should only be able to select one sort option
- ✓ A tablet will show the filtering or sorting preferences screen as a slide over or a pop up
- ✓ Mobile takes the user to a new settings screen

(i) Only provide sorting and filtering options where required If there is a very minimal amount of content then filtering and sorting may not be required. Also, consider if you only need to sort OR filter rather than both, e.g. there may not be any filtering options that are relevant to the content.