

Retail Version 1.0 – A Concept Android App

Introduction

In the same way that the car industry builds concept cars to test ideas so this app has been built to test ideas within the Android app framework. It has been built around a retail companies web site but this app in no way represents or has been made by that company.

It has been designed with certain features that have been explored within the app. The first of these is the use of fragments to enable screens to be designed for both mobile phones and for tablets. Each device has its own set of screens that are optimised for that device. This is described in the section “Fragments”.

Next the app has been used to explore the SQLite database that sits on all Android devices. We have used it within this app to save details of items and to create Lists that sit on the device and not on an internet server. This allows the selected items to be displayed whenever the user requires whether or not they have access to the internet at a specific point in time. This idea is described in more detail in the section “SQLite Database”.

First though to give an idea of the app we will have a walk through.

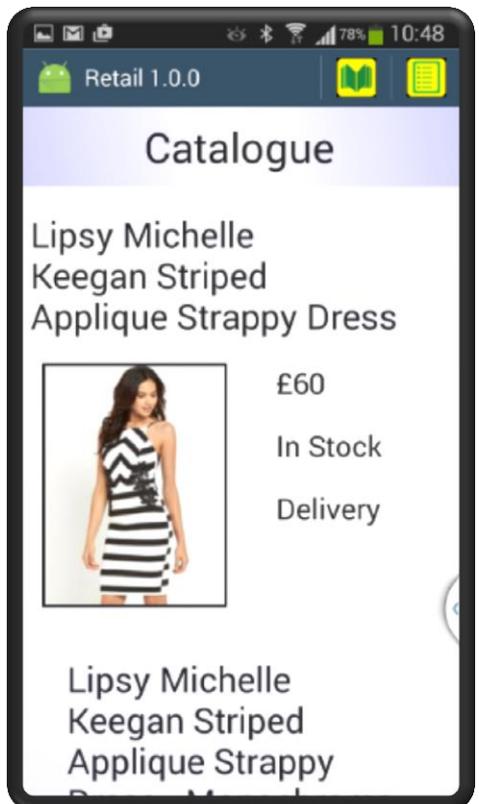
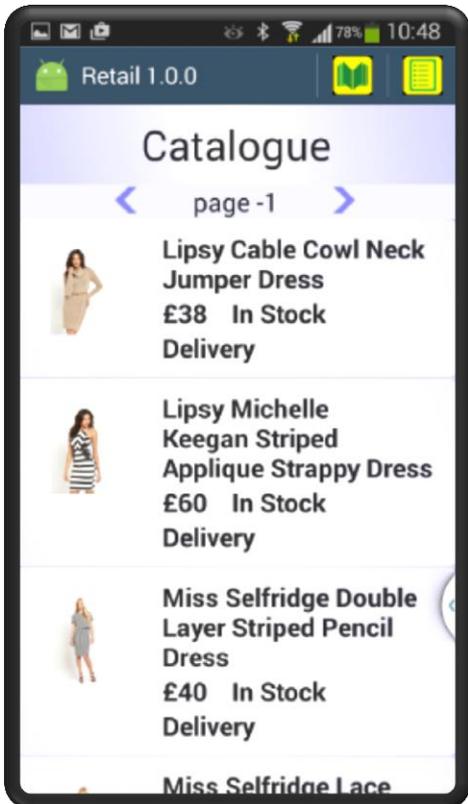
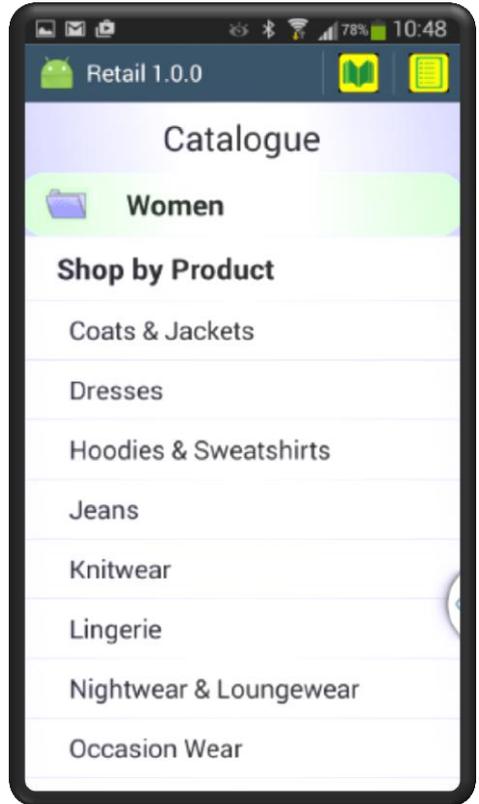
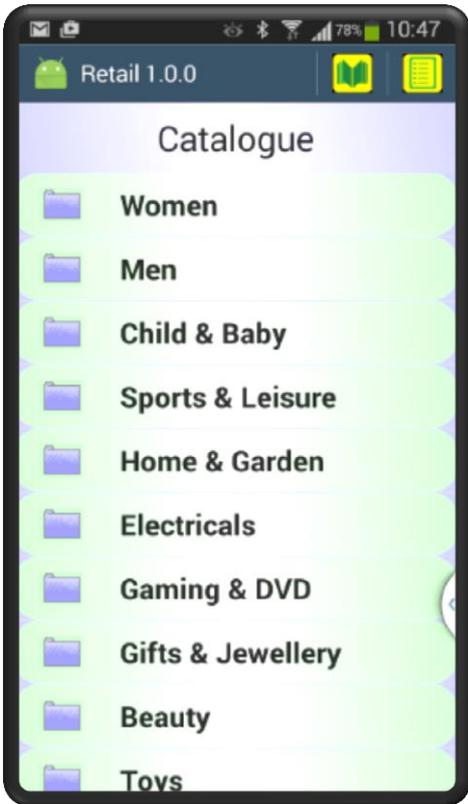
Mobile – First Walk Through

On entering the mobile version of the Retail app the first screen as shown on the following page will be displayed. Displayed are the main catalogue header entries. The user touches one of the entries for the items within that group to be displayed as shown in the next screen display on the following page where the header “Women” has been pressed and thus has opened up.

The screen then displays the sub-headings which include the sub-heading of “Dresses” which is then pressed. There then appears all the items under that sub-heading as shown in the third screen display.

Pressing on one of the entries will result in the detail screen being displayed as shown in the fourth display.

At any point the back button can be pressed returning the user to the previous screen and allowing an alternative choice to be made. On the third screen displaying the list of items the user can page forward to the next twelve items by pressing the forward icon or backwards displaying the previous twelve items by pressing the back icon.



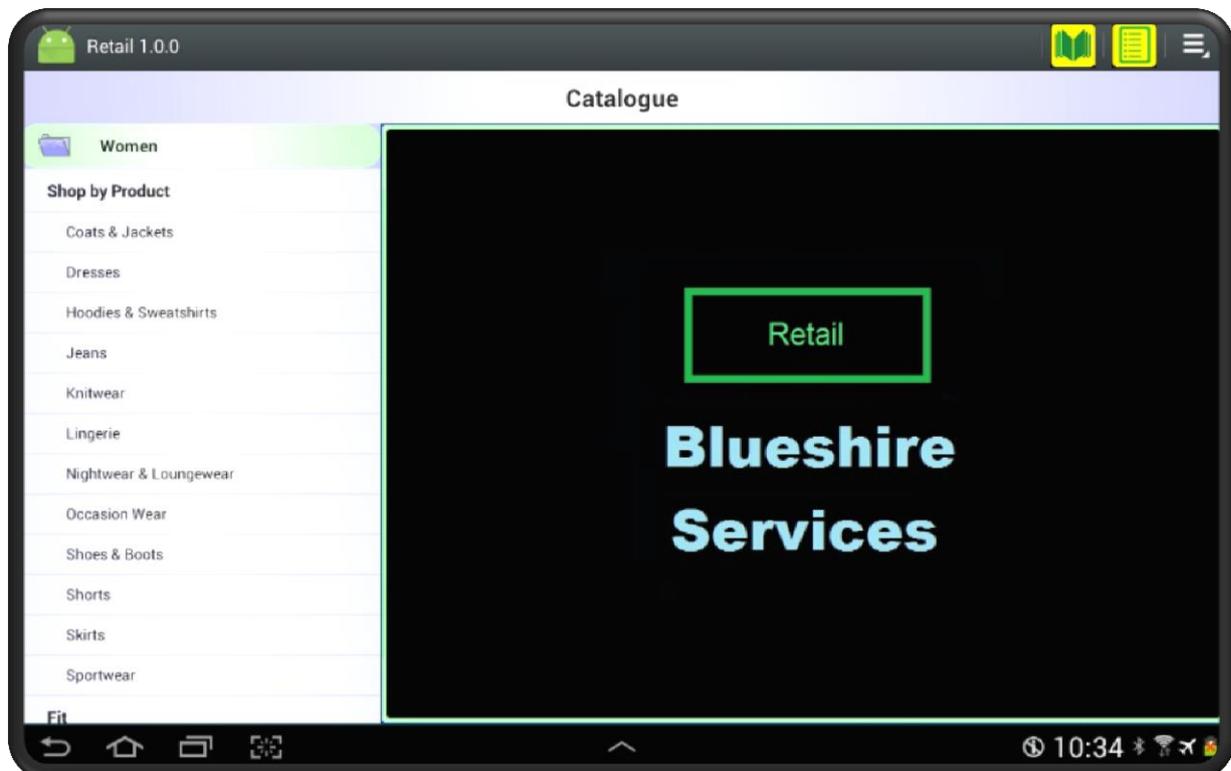
Tablet – First Walk Through

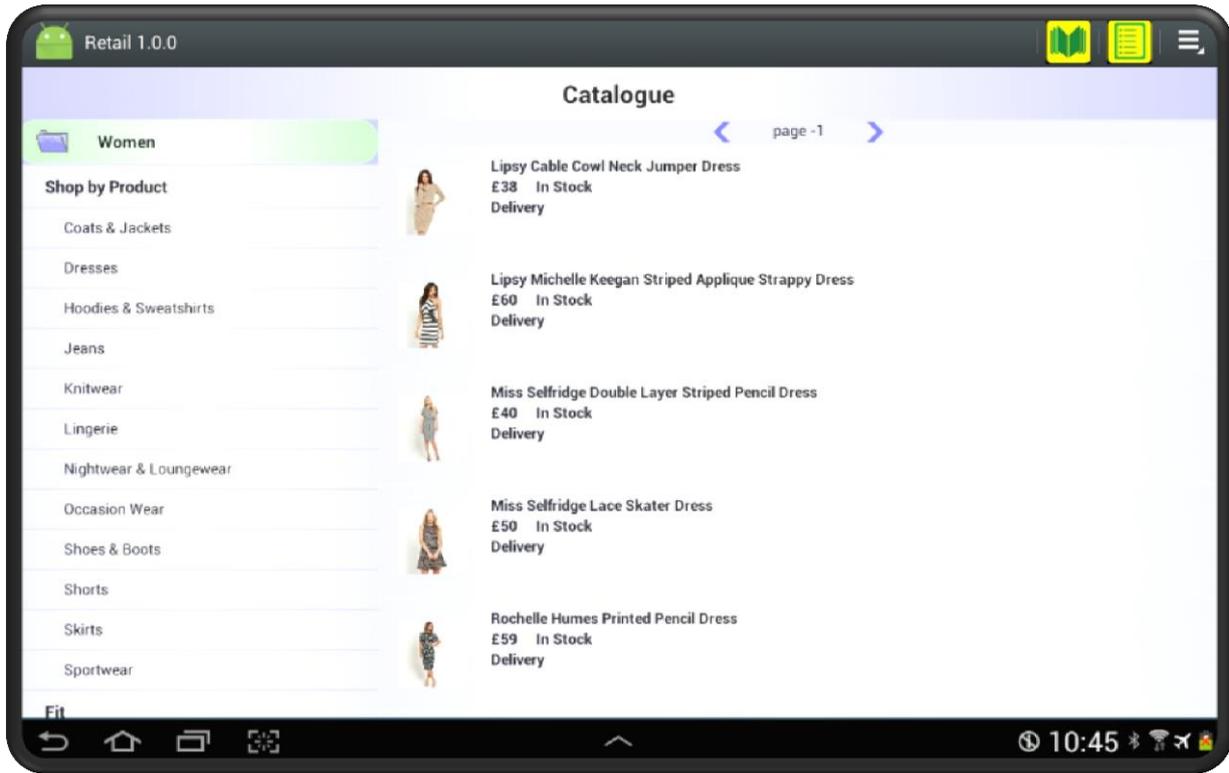
The display for the tablet contains the same information but optimised for the tablets screen dimensions.

On entering the app the main headers are displayed in a list on the left hand side of the screen. The screen design has been arranged with the left side being the navigation display with the right side displaying details. So when one enters the high level entries are displayed on the right and as there are as yet no details to display a banner is displayed instead. It is expected that in a production version this would be replaced by a banner displaying current offers.

On pressing an option in this case the “Women” option, this header opens up to display the sub-headings. The Banner at this point remains. When a sub-heading is selected the entries for that sub-heading are displayed in the right hand side window as shown in the second screen display on the following page.

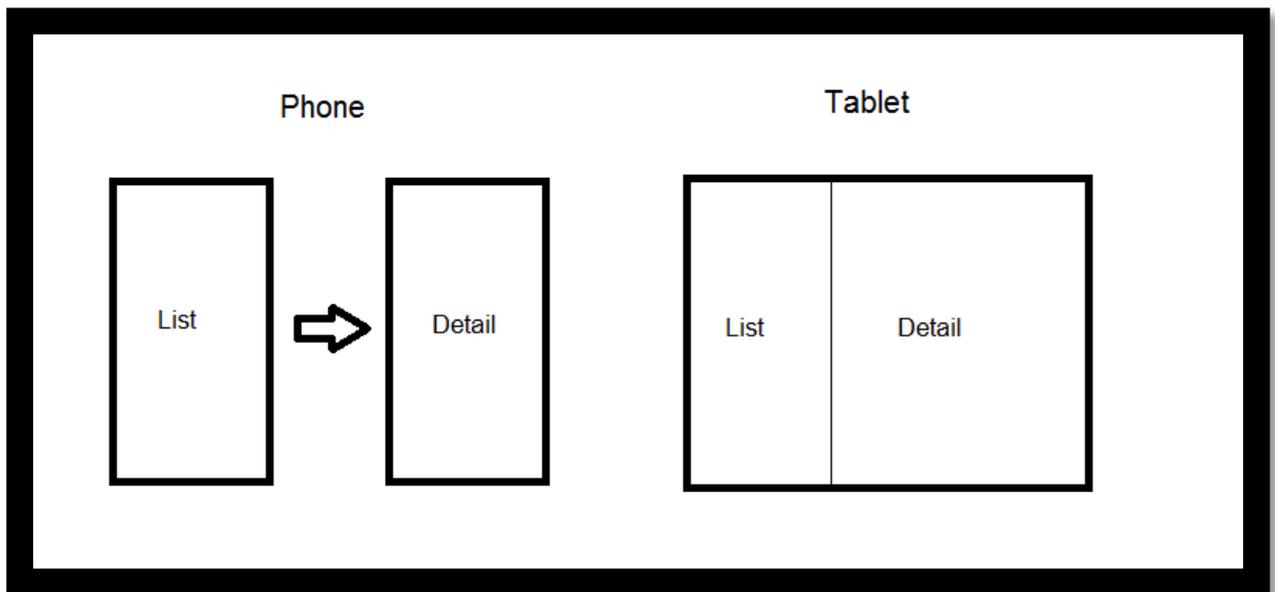
When one of the entries is selected the details of that entry are displayed again in the right hand side. At all times as with the mobile version the user can use the back button to return to the previous screen.





Fragments

The design of the app is based on the design shown in the diagram below and this has been enabled by the use of fragments within the code. In the case of the tablet to have two fragments, one for the list side and one for the detail side each running with a degree of autonomy. The mobile version recognises that it is a mobile and has the detail as a separate activity.

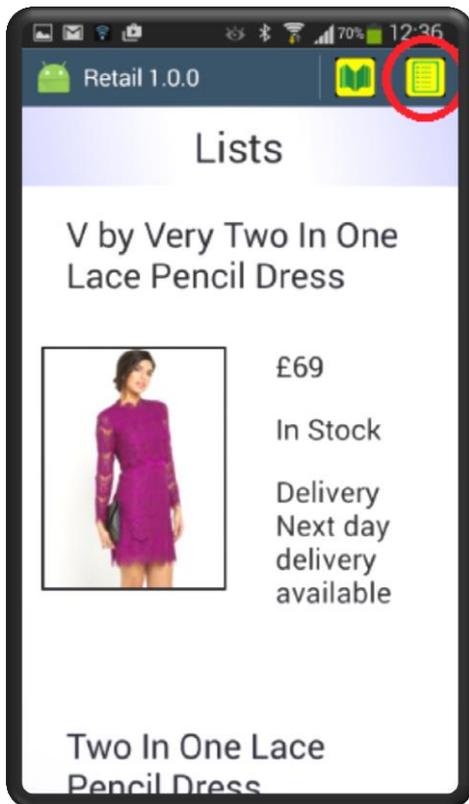
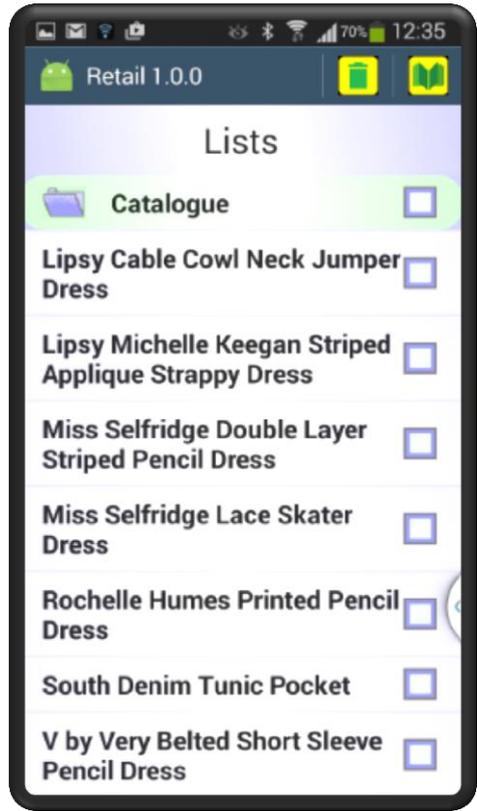
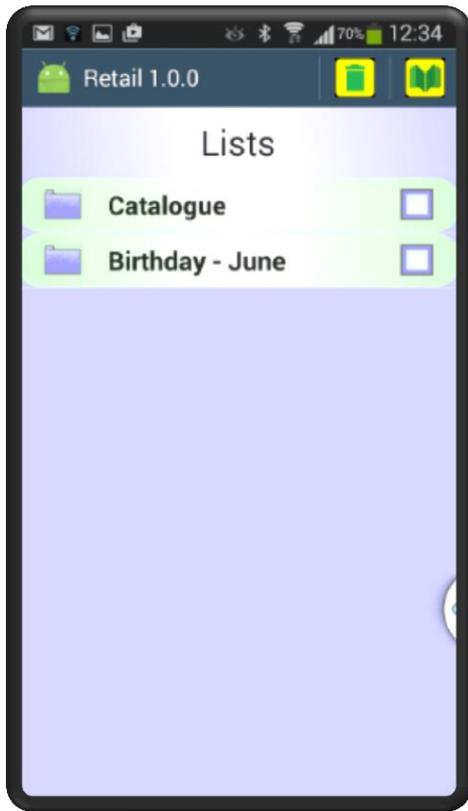


The use of fragments is supplemented by the use of alternate screen displays. Within Android the app can recognise when the device is held in portrait mode and when in landscape mode and assign screen displays accordingly.

Mobile – Second Walk Through

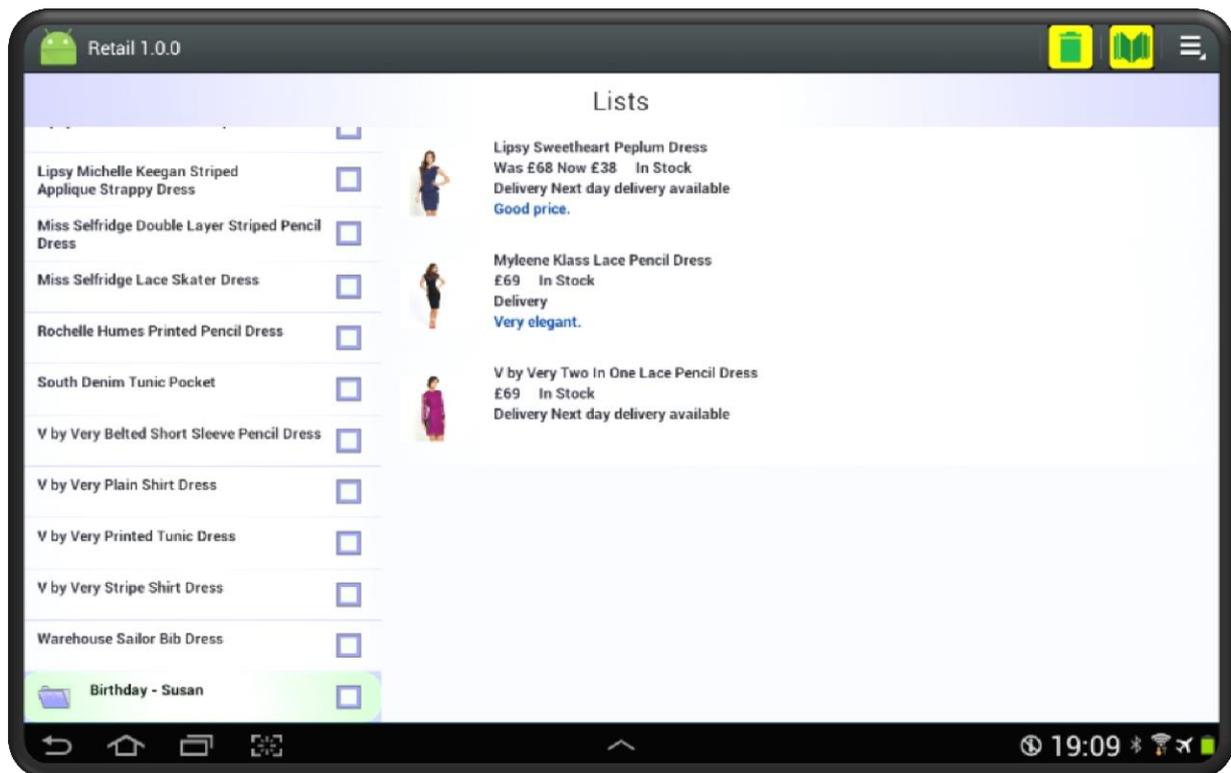
A feature of the Retail app is that it stores its lists within the device itself and not on a server. This feature allows the user to view items within the lists without having an internet connection. When a user wants to for instance show items to friends or family then the item will be available for display and not subject to the internet connection.

To access this feature press the list icon highlighted with a red circle in the third display on the following page. This will bring the user to the screen display shown in the first display on the following page. When the user first loads the app there will be only one list and that is the “Catalogue” list. The catalogue list displays the items that were displayed in the catalogue section. Every time the user displays a new set of catalogue items the list catalogue will be refreshed.



To create a new list the user uses the menu item “Add List” which when pressed will result in the display shown in the fourth display on the previous page. When the OK button is pressed the new list will be created.

To populate the new list use the drag and drop functionality. Press on any item in the “Catalogue” list until a grey shadow appears and then move the grey shadow to the list which you want to put the item. Once this has been carried out the item will be saved until the user decides to remove it. This is achieved by clicking in the check box that is displayed to the right of every item. Then press the delete icon shown at the top of the screen.



Another feature that was been included is the ability to enter comments against items in the list. This is shown in the screen display above where the comment “Good price.” Has been entered against one item and “Very elegant.” against another. These comments are saved on the device and can be entered against any item in ones lists.

Appendix A – Technical Information

Retail has been developed in Android Java and requires a minimum of Android api of 14. This is Android 4.0, Ice Cream Sandwich.

