

# Mobile Application Development

## Session 4 – Activities

The final product(s) for this session can be found at:

<http://www.dcs.bbk.ac.uk/lo/mad/madexamples/session4/classactivities/listviewnav/zedland-hotels.html>

<http://www.dcs.bbk.ac.uk/lo/mad/madexamples/session4/classactivities/panelnav/exland-tourist.html>

### 1. Creating Primary and Secondary Navigation with Listview

Open your *zedland-hotels.html* page from session 3. Find the primary navigation links to the five star, four star and three star hotels in the content section of the page. Put these links into a `<ul>` element with a *data-role* of *listview*. Add *slidefade* transitions.

```
<ul data-role="listview">
  <li><a href="#five" data-transition="slidefade">Five Star Hotels</a></li>
  <li><a href="#four" data-transition="slidefade">Four Star Hotels</a></li>
  <li><a href="#three" data-transition="slidefade">Three Star Hotels</a></li>
</ul>
```

Save your *zedland-hotels.html* page and view it in an emulator, or upload it to the titan server and view it on your mobile device.

Repeat this process for the secondary links on the five, four and three star pages of the Zedland Hotels app.

### Insetting a Listview

Add a *data-inset="true"* property and value to each of the `<ul>` listviews you created in task 1.

Save your *zedland-hotels.html* page and view it in an emulator, or upload it to the titan server and view it on your mobile device. Note the differences to the appearance of the navigation.

## Making a Listview Searchable

In a real world scenario, the list of hotels would be likely to get long, so we would need some way of making the list more accessible. To do this, we will make the listview searchable using the *data-filter* property. We will also add placeholder text to the search box using the *data-filter-placeholder* property.

```
<ul data-role="listview" data-filter="true" →  
data-filter-placeholder="Enter a Hotel Name">
```

Save your *zedland-hotels.html* page and view it in an emulator, or upload it to the titan server and view it on your mobile device. Test the search feature.

## 2. Creating a Footer Navigation Menu

We will add three new pages to our app: *about.html*, *contact.html* and *login.html*. These will be external pages, but they will be pre-fetched in order to speed up loading.

Create the pages using a basic jQM template. Give the jQM pages ids of *about*, *contact* and *login* respectively.

```
<div data-role="page" id="about">  
  . . .  
</div>
```

Save *about.html*, *contact.html* and *login.html*.

Now, in the footer of *zedland-hotels.html*, add a `<div>` with a *data-role* of *navbar*. Inside this `<div>` add a simple unordered list with links to *about.html*, *contact.html* and *login.html*. Make sure the pages the links point to are pre-fetched.

```
<div data-role="footer">
  <div data-role="navbar">
    <ul>
      <li><a href="about.html" data-prefetch>About</a></li>
      <li><a href="contact.html" data-prefetch>Contact</a></li>
      <li><a href="login.html" data-prefetch>Login</a></li>
    </ul>
  </div>
</div>
```

### Make the Menu Fixed

To fix the menu so that it is not subject to scrolling, add *data-position="fixed"* to the footer.

```
<div data-role="footer" data-position="fixed">
```

### Make the Menu Persist

Now add the navbar menu to the footer of ALL pages in the application.

In order to make the menu persist across transitions, we need to make sure that all instances of the navbar menu have the same id. To do this we use the *data-id* property. Add this property and the value *footernav* to the navbar on each page of the application.

```
<div data-role="navbar" data-id="footernav">
```

The final task is to highlight the active page and make sure the menu persists across pages. We do this by adding a special jQM class to the `<a>` element in EACH menu page. For example, on the contact page, we would write.

```
<li><a href="contact.html" class="ui-state-persist →
  ui-btn-active">Contact</a></li>
```

## 3. Creating Panel Navigation

### Create the Page

Create a new HTML file, and save it as *exland-tourist.html*.

Create a basic jQuery Mobile page, including links to jQuery and jQuery Mobile CDNs, viewport declaration, and a single basic page

with header, content and footer sections.

Add the following content.

header <h1>	Visit Exland
content <p>	Find hotels, camping, car hire, restaurants and much more in the Principality of Exland.
footer <h4>	Footer Navigation ( <i>nav bar to be added later</i> )

### Create the Panel

Before the header of the page, create a new <div> element with a *data-role* of *panel*. Give the panel an *id* of *navpanel*.

```
<div data-role="panel" id="navpanel">
</div>
```

### Create the Icon

Now, just before the page header, add a link to the panel. Note that the link needs no text, as it will be displayed as a button.

```
<a href="#navpanel"></a>
```

To make the link display as a button, and to display correctly, we need to link it to a jQM CSS class that specifies (a) the link should be displayed as a button, (b) all corners should be rounded, (c) the icon used should be the bars icon, (d) any button text will be suppressed and only the icon image will show.

```
<a href="#navpanel" class="ui-btn ui-corner-all →
ui-icon-bars ui-btn-icon-notext"></a>
```

Now save your page and view it in an emulator, or in-situ on your device.

You should now have a basic panel.

### Customizing the Panel

The panel currently opens from the left, shunts the content to the left on opening, and scrolls with the content. We will customize the panel by getting it to open from the right, overlay the content and remain fixed whether content is scrolled or not. To do this we need

three data properties: *data-position*; *data-display*, and *data-position-fixed*.

```
<div data-role="panel" id="navpanel" data-position="right"  
➔ data-display="overlay" data-position-fixed="true">
```

## Adding the Navigation to the Panel

### HTML

Add a `<div>` element with the id *navcontainer* inside the panel. Inside the new `<div>` add a nested list as shown below.

```
<div id="navcontainer">  
  <ul>  
    <li><a href="#">Hotels</a>  
    <ul>  
      <li><a href="#">Five Star</a></li>  
      <li><a href="#">Four Star</a></li>  
      <li><a href="#">Three Star</a></li>  
    </ul>  
    </li>  
    <li><a href="#">Camping</a></li>  
    <li><a href="#">Hire Cars</a></li>  
    <li><a href="#">Restaurants</a></li>  
  </ul>  
</div>
```

Save your page and view it in an emulator, or in-situ on your device.

### CSS

Finally, add CSS styles for the navigation as shown in the table below.

#navcontainer ul	margin	0
	padding	0
	list-style-type	none
	width	100%
#navcontainer a	display	block
	color	#000
	width	100%
	padding	3px 12px 3px 8px
	text-decoration	none
	border-bottom	1px solid #000
#navcontainer li li a	display	block
	width	100%
	padding	3px 3px 3px 17px
	text-decoration	none
	border-bottom	1px solid #000
	font-weight	lighter
#navcontainer a:hover	background-color	#E8E8E8

Save *exland-tourist.html* and view it in an emulator, or in-situ on your device.