**Mobile Learning Content Apps Development Workshop For Mobile Learning Association of Malaysia**

**Asas HTML**

**HTML5 Declaration**

<!DOCTYPE html>

**Basic Needed HTML Tagging**

Link

<a href=”#page\_tuju”>Next Page</a>

Image

<img src=”image.jpg” />

Paragraph

<p>This is paragraph</p>

Division

<div>Division</div>

Header

<h3>We got six header</h3>

**Basic Configuration**

Apache Directory will : C:\Web

Document for web directory will : C:\Web\htdocs

**File/Template Given For Development**

**Jquerymobile.zip contain**

css/jquery.mobile.min.css  
js/jquery.min.js  
js/jquery.mobile.min.js  
index.html

**HTML Format**

<!DOCTYPE html>

<html>

<head>

</head>

<body>

</body>

</html>

**jQuery Mobile framework**

**Mobile page structure**

<!DOCTYPE html>

<html>

<head>

<title>Page Title</title>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<link href="css/jquery.mobile.min.css" rel="stylesheet" type="text/css">

<script src="js/jquery.min.js"></script>

<script src="js/jquery.mobile.min.js"></script>

</head>

<body>

...content and pages goes here...

</body>

</html>

**Page**

Inside the <body> tag, each view or "page" on the mobile device is identified with an element (usually a div) with the data-role="page" attribute.

<div data-role="page">

...

</div>

Within the "page" container, any valid HTML markup can be used, but for typical pages in jQuery Mobile, the immediate children of a "page" are divs with data-role="header", class="ui-content", and data-role="footer".

<div data-role="page">

<div data-role="header">...</div>

<div role="main" class="ui-content">...</div>

<div data-role="footer">...</div>

</div>

**Multi-page template structure**

<!-- Start of first page -->

<div data-role="page" id="foo">

<div data-role="header">

<h1>Foo</h1>

</div><!-- /header -->

<div role="main" class="ui-content">

<p>I'm first in the source order so I'm shown as the page.</p>

<p>View internal page called <a href="#bar">bar</a></p>

</div><!-- /content -->

<div data-role="footer">

<h4>Page Footer</h4>

</div><!-- /footer -->

</div>

<!-- /page -->

<!-- Start of second page -->

<div data-role="page" id="bar">

<div data-role="header">

<h1>Bar</h1>

</div><!-- /header -->

<div role="main" class="ui-content">

<p>I'm the second in the source order so I'm hidden when the page loads. I'm just shown if a link that references my id is beeing clicked.</p>

<p><a href="#foo">Back to foo</a></p>

</div><!-- /content -->

<div data-role="footer">

<h4>Page Footer</h4>

</div><!-- /footer -->

</div>

<!-- /page -->

**Button markup**

Add classes to style a and button elements.

<a href="#" class="ui-btn">Anchor</a>

<button class="ui-btn">Button</button>

**Grids**

The jQuery Mobile framework provides a simple way to build CSS-based columns that can also be responsive.

**Buttons in grids**

<fieldset class="ui-grid-a">

<div class="ui-block-a" ><input type="submit" value="Submit" data-theme="a"></div>

<div class="ui-block-b" ><input type="reset" value="Reset" data-theme="b"></div>

</fieldset>

**Responsive**

It's not recommended to have many buttons with text on one row at small screens, because the text might get truncated. You can use responsive grids to stack the buttons at small screens. Here we use the framework preset breakpoint by adding class ui-responsive to the container.

<div class="ui-grid-b ui-responsive">

<div class="ui-block-a"><a href="#" class="ui-shadow ui-btn ui-corner-all ui-btn-icon-left ui-icon-arrow-l">Back to order</a></div>

<div class="ui-block-b"><input type="button" data-icon="plus" value="Add products"></div>

<div class="ui-block-c"><input type="button" data-icon="check" data-iconpos="right" value="Confirm order"></div>

</div>

**Icon**

In widgets where you set the icon with a data-icon attribute you use the name of the icon as value. For example: data-icon="arrow-r".

To add an icon to link buttons and button elements, use the name prefixed with ui-icon- as class. For example: ui-icon-arrow-r.

**List of Icons**

home action arrow-d-l arrow-d-r arrow-d arrow-l arrow-r arrow-u-l arrow-u-r arrow-u audio calendar camera carat-d carat-l carat-r carat-u check clock cloud grid mail eye gear heart home info bullets bars navigation lock search location minus forbidden edit user phone plus power recycle forward refresh shop comment star tag back video alert delete

**Transition**

By default, the framework applies a fade transition. To set a custom transition effect, add the data-transition attribute to the link.

<a href="index.html" data-transition="pop">I'll pop</a>

When the Back button is pressed, the framework will automatically apply the reverse version of the transition that was used to show the page. To specify that the reverse version of a transition should be used, add the data-direction="reverse" attribute to a link.

**Tansition available:-**

Fade pop flip turn flow slidefade slide slideup slidedown

**Listview**

A listview is coded as a simple unordered list (ul) or ordered list (ol) with a data-role="listview" attribute and has a wide range of features.

<ul data-role="listview">

<li>Acura</li>

<li>Audi</li>

<li>BMW</li>

<li>Cadillac</li>

<li>Ferrari</li>

</ul>

Another features with filterable and reveal

<ul data-role="listview" data-filter="true" data-filter-reveal="true" data-filter-placeholder="Search fruits..." data-inset="true">

<li><a href="#">Apple</a></li>

<li><a href="#">Banana</a></li>

<li><a href="#">Cherry</a></li>

<li><a href="#">Cranberry</a></li>

<li><a href="#">Grape</a></li>

<li><a href="#">Orange</a></li>

</ul>

**Listview with thumbnail**

To add thumbnails to the left of a list item, simply add an image inside a list item as the first child element. The framework will scale the image to 80 pixels square.

<ul data-role="listview" data-filter="true" data-filter-placeholder="Taipkan Nama">

<li>

<img src="gambar/syam.jpg" width="150" />

<h3>En. Norhisyam Dasuki</h3>

<p>CIAST</p></a>

</li>

<li>

<img src="gambar/supyanphoto.jpg" width="150" />

<h3>Prof. Madya Dr. Supyan Hussin</h3>

<p>Mobile Learning Association of Malaysia / UKM</p>

</li>

<li>

<img src="gambar/razali.bmp" width="150" />

<h3>Ir. Razali Hamzah</h3>

<p>Terato Tech Sdn Bhd</p>

</li>

</ul>

**Collapsible**

To create a collapsible block of content, create a container and add the data-role="collapsible" attribute. Directly inside this container, add any header (H1-H6) or legend element. The framework will style the header to look like a clickable button and add a "+" icon to the left to indicate it's expandable. After the header, add any HTML markup you want to be collapsible. The framework will wrap this markup in a container that will be hidden/shown when the heading is clicked.

<div data-role="collapsible">

<h4>Heading</h4>

<p>I'm the collapsible content. By default I'm closed, but you can click the header to open me.</p>

</div>

To expand the content when the page loads, add the data-collapsed="false" attribute to the wrapper

The default icons of collapsible headings can be overridden by using the data-collapsed-icon and data-expanded-icon attributes. In the example below, data-collapsed-icon="carat-d" and data-expanded-icon="carat-u".

<div data-role="collapsible" data-collapsed-icon="carat-d" data-expanded-icon="carat-u">

<h4>Heading</h4>

<ul data-role="listview" data-inset="false">

<li>Read-only list item 1</li>

<li>Read-only list item 2</li>

<li>Read-only list item 3</li>

</ul>

</div>

**Navbar**

A navbar is coded as an unordered list of links wrapped in a container element that has the data-role="navbar" attribute.

<div data-role="navbar">

<ul>

<li><a href="#" data-icon="grid">Summary</a></li>

<li><a href="#" data-icon="star" class="ui-btn-active">Favs</a></li>

<li><a href="#" data-icon="gear">Setup</a></li>

</ul>

</div>

**Popup and Dialog**

To create a popup, add the data-role="popup" attribute to a div with the popup contents. Then create a link with the href set to the id of the popup div, and add the attribute data-rel="popup" to tell the framework to open the popup when the link is tapped. A popup div has to be nested inside the same page as the link.

<a href="#popupBasic" data-rel="popup" class="ui-btn ui-corner-all ui-shadow ui-btn-inline" data-transition="pop">Basic Popup</a>

<div data-role="popup" id="popupBasic">

<p>This is a completely basic popup, no options set.</p>

</div>

Standard dialog markup can be placed into a popup. To create a modal style dialog, add the data-dismissible="false" attribute to the popup to prevent the click-outside-to-close behavior so people need to interact with popup buttons to close it.

<a href="#popupDialog" data-rel="popup" data-position-to="window" data-transition="pop" class="ui-btn ui-corner-all ui-shadow ui-btn-inline ui-icon-delete ui-btn-icon-left ui-btn-b">Delete page...</a>

<div data-role="popup" id="popupDialog" data-overlay-theme="b" data-theme="b" data-dismissible="false" style="max-width:400px;">

<div data-role="header" data-theme="a">

<h1>Delete Page?</h1>

</div>

<div role="main" class="ui-content">

<h3 class="ui-title">Are you sure you want to delete this page?</h3>

<p>This action cannot be undone.</p>

<a href="#" class="ui-btn ui-corner-all ui-shadow ui-btn-inline ui-btn-b" data-rel="back">Cancel</a>

<a href="#" class="ui-btn ui-corner-all ui-shadow ui-btn-inline ui-btn-b" data-rel="back" data-transition="flow">Delete</a>

</div>

</div>

**Panel**

Flexible by design, panels can be used for navigation, forms, inspectors and more.

Here is an example of the panel before the header, content and footer in the source order:

<div data-role="page">

<div data-role="panel" id="mypanel">

<!-- panel content goes here -->

<h3>Default panel options</h3>

<p>This panel has all the default options: positioned on the left with the reveal display mode. The panel markup is <em>before</em> the header, content and footer in the source order.</p>

<p>To close, click off the panel, swipe left or right, hit the Esc key, or use the button below:</p>

<a href="#demo-links" data-rel="close" class="ui-btn ui-shadow ui-corner-all ui-icon-delete ui-btn-icon-left ui-btn-inline">Close panel</a>

</div><!-- /panel -->

<!-- header -->

<!-- content -->

<!-- footer -->

</div><!-- page -->

Opening a panel

To control a panel from a link, set the href to the ID of the panel you want to toggle (mypanel in the example above). This instructs the framework to bind the link to the panel. This link will toggle the visibility of the panel so tapping it will open the panel, and tapping it again will close it.

<a href="#mypanel" class="ui-btn ui-shadow ui-corner-all ui-btn-inline ui-btn-icon-left ui-icon-bars">Default panel</a>

Closing a panel

Clicking the link that opened the panel, swiping left or right, or tapping the Esc key will close the panel. It's common to also add a close button inside the panel. To add the link that will close the panel, add the data-rel="close" attribute to tell the framework to close that panel when clicked.

<a href="#my-header" data-rel="close">Close panel</a>

**Table: Column Toggle**

The column toggle table mode selectively hides columns at narrower widths as a sensible default but also offers a menu to let users manually control which columns they want to see.

<table data-role="table" id="table-custom-2" data-mode="columntoggle" class="ui-body-d ui-shadow table-stripe ui-responsive" data-column-btn-theme="b" data-column-btn-text="Columns to display..." data-column-popup-theme="a">

<thead>

<tr class="ui-bar-d">

<th data-priority="2">Rank</th>

<th>Movie Title</th>

<th data-priority="3">Year</th>

<th data-priority="1"><abbr title="Rotten Tomato Rating">Rating</abbr></th>

<th data-priority="5">Reviews</th>

</tr>

</thead>

<tbody>

<tr>

<th>1</th>

<td><a href="http://en.wikipedia.org/wiki/Citizen\_Kane" data-rel="external">Citizen Kane</a></td>

<td>1941</td>

<td>100%</td>

<td>74</td>

</tr>

<tr>

<th>2</th>

<td><a href="http://en.wikipedia.org/wiki/Casablanca\_(film)" data-rel="external">Casablanca</a></td>

<td>1942</td>

<td>97%</td>

<td>64</td>

</tr>

<tr>

<th>3</th>

<td><a href="http://en.wikipedia.org/wiki/The\_Godfather" data-rel="external">The Godfather</a></td>

<td>1972</td>

<td>97%</td>

<td>87</td>

</tr>

</tbody>

</table>

**Table: Reflow**

The reflow table mode works by collapsing the table columns into a stacked presentation that looks like blocks of label/data pairs for each row.

<table data-role="table" id="movie-table" data-mode="reflow" class="ui-responsive">

<thead>

<tr>

<th data-priority="1">Rank</th>

<th data-priority="persist">Movie Title</th>

<th data-priority="2">Year</th>

<th data-priority="3"><abbr title="Rotten Tomato Rating">Rating</abbr></th>

<th data-priority="4">Reviews</th>

</tr>

</thead>

<tbody>

<tr>

<th>1</th>

<td><a href="http://en.wikipedia.org/wiki/Citizen\_Kane" data-rel="external">Citizen Kane</a></td>

<td>1941</td>

<td>100%</td>

<td>74</td>

</tr>

<tr>

<th>2</th>

<td><a href="http://en.wikipedia.org/wiki/Casablanca\_(film)" data-rel="external">Casablanca</a></td>

<td>1942</td>

<td>97%</td>

<td>64</td>

</tr>

<tr>

<th>3</th>

<td><a href="http://en.wikipedia.org/wiki/The\_Godfather" data-rel="external">The Godfather</a></td>

<td>1972</td>

<td>97%</td>

<td>87</td>

</tr>

<tr>

<th>4</th>

<td><a href="http://en.wikipedia.org/wiki/Gone\_with\_the\_Wind\_(film)" data-rel="external">Gone with the Wind</a></td>

<td>1939</td>

<td>96%</td>

<td>87</td>

</tr>

</tbody>

</table>

**Communication**

<h2>Senarai Penceramah</h2>

<div data-role="popup" id="d1">

<a href="tel:+60197347511" class="ui-btn ui-icon-phone ui-btn-icon-notext ui-corner-all ui-btn-inline"> Telefon</a>

<a href="sms://+60197347511" class="ui-btn ui-icon-action ui-btn-icon-notext ui-corner-all ui-btn-inline">Hantar SMS</a>

<a href="mailto:ir.syam@gmail.com" class="ui-btn ui-icon-mail ui-btn-icon-notext ui-corner-all ui-btn-inline">Pesanan Emel</a>

</div>

<ul data-role="listview" data-filter="true" data-filter-placeholder="Taipkan Nama">

<li> <a href="#d1" data-rel="popup" data-transition="slideup">

<img src="gambar/syam.jpg" width="150" />

<h3>En. Norhisyam Dasuki</h3>

<p>CIAST</p></a>

</li>

<li>

<img src="gambar/supyanphoto.jpg" width="150" />

<h3>Prof. Madya Dr. Supyan Hussin</h3>

<p>Mobile Learning Association of Malaysia / UKM</p>

</li>

<li>

<img src="gambar/razali.bmp" width="150" />

<h3>Ir. Razali Hamzah</h3>

<p>Terato Tech Sdn Bhd</p>

</li>

<li>

<img src="gambar/yahaya.jpg" width="150" />

<h3>Dr. Yahaya Abdullah</h3>

<p>KRU Academy</p>

</li>

</ul>

**Deploy – Create Native App**

Setup Git - https://github.com/

Register/Create account

Setup Phonegap Builder - https://build.phonegap.com/

Register/Create account

Link Accaunt to Git Hub

Download and Install Git

At the heart of GitHub is an open source version control system (VCS) called Git\*. Created by the same team that created Linux, Git is responsible for everything GitHub related that happens locally on your computer.

https://github-windows.s3.amazonaws.com/GitHubSetup.exe

**Using Git Hub and Git Shell**

Step:-

1. Open web browser, https://github.com/your\_username, Create 1st Repositories with readme file.
2. Open desktop/windows Git Hub, Clone form web
3. Copy all files created before into new repositories folder (cloned from Git Hub Server)
4. Open Git Shell,
   1. cd foler\_or\_repositories
   2. git add \*
   3. git coommit -m 'mesej'
   4. git push
5. If any changes made, open Git Hub on Windows – click sync

Sumber : demo.jquerymobile.com

Disediakan oleh

Norhisyam Dasuki