The Lively Sketchbook
• Ubiquity
• Intimacy
• Embeddedness
The Feed App – On The Server and the Phone
Installing Wordpress

  - Unzip into MAMP's htdocs folder;
  - Create a database in MAMP;
  - Go to http://localhost:8888/wordpress/ to create your WordPress site
    - Use the database you've created and "root" as your user
- Login and go to the Plugins folder, then add:
  - WPtouch iPhone Theme
  - FeedWordPress
The Feed App On The Phone

• Launch Dashcode

• Choose the RSS template for Mobile Safari

• Add a feed URL

• Add a title and offline viewing capability

• Add a web clip icon via drag-and-dropping an image
  • At least 57 pixels square, no shine or gloss
  • Note: a bug in Dashcode means that you may need to copy this icon into the Images folder by hand

• Save the project to disk, and test it.
My RSS Feed
WEDNESDAY, JUNE 9, 2010

Media Fluency?: NITLE Summit
In late March of this year I was privileged to speak...

Assessment in a Web 2.0 Environment
I agree in principle that we who work in education ...

Gratitude and clarifications
First, my thanks to everyone who responded to my...

Changing lives
I keep running into the same wall from different di...

Another bootstrapping experience
Today's the second day we'll meet. Already the acti...

A happy birthday
Today the father of interactive computing, the thin...

"In Our Time" podcast series on...
There's a new set of four episodes from BBC Radio...
A Bit Of User Interface Design
The Wisdom of Apple

• Expect to spend about 60% of your total time designing the app

• Read the Human Interface Guidelines

• Three key steps:
  • Define your focus - i.e., define the solution you're creating
  • Refine the feature set
  • Determine your users' mental model
Some Key Elements of the HIG

• Minimum target: 22 pixels vertically, 44 pixels square
• Navigation follows a tree model, emphasized by visual animation - no home button or breadcrumbs
• Upper left corner navigates up the tree, upper right corner modifies/adds items
• Lists:
  • Ungrouped: textual – all items at a given level
  • Grouped: text + icons – classify and group items
• Pickers:
  • 12 items optimal
• Toolbars:
  • Icons only – correspond to tools
• Tab bars:
  • Text and icons – switch between different modes, groupings
• Buttons:
  • Have rounded corners to encourage tapping
• Animation:
  • Use to provide feedback
• Aesthetics:
  • Keep the look and feel clean and uncluttered
  • Emphasize tactile design
  • Prioritize
The Handheld Database
Creating the Database App

- Select the Browser template, and complete the basic fields
- Test the app

Edit your datasource:

- Add an image field \textit{(imagelocation)} to \textit{sampleData.js}
- Add the images (e.g., in a folder labeled \textit{Photos})
- Refresh the datasource

- Add an image part \textit{(parkImage)} to the \textit{detailLevel}
- In the "list" Data Source view, bind the \textit{imagelocation} to the src of \textit{parkImage}
- Add a Map Button (parkMap) to the \textit{detailLevel}, and bind:
  - \textit{location} to the \textit{state} of \textit{parkMap}
  - \textit{name} to the \textit{address} of \textit{parkMap}
Acadia
Maine, USA

The scenery in Acadia is amazing this time of year!
Getting Input
Building a Calculator

- Select the Custom Project template
- Add two Text Field parts (*numberOne*, *numberTwo*) to the content area
- Add a Text part (*resultSum*) below them
- Add a Lozenge Button (*calcSum*) below them
- Add an onclick event to *calcSum*
- Add the following code to the myClickHandler:

```javascript
var firstnum = parseFloat(document.getElementById("numberOne").value);
var secondnum = parseFloat(document.getElementById("numberTwo").value);
var thesum = firstnum + secondnum;
document.getElementById("resultSum").value = thesum.toFixed(4);
```
Adding Memory

• Add to the load() function:

```javascript
var oldfirstnum = localStorage.getItem("first_num");
var oldsecondnum = localStorage.getItem("second_num");
document.getElementById("numberOne").value = oldfirstnum;
document.getElementById("numberTwo").value = oldsecondnum;
```

• Add to myClickHandler:

```javascript
localStorage.setItem("first_num", firstnum);
localStorage.setItem("second_num", secondnum);
```
Visualizing Data and Getting Images
Ingredients for Visualization

- Download the jqPlot library at http://www.jqplot.com/
- Drag the three key components into the Files window:
  - On a Custom template, add:
    - A Lozenge Button (visualizeIt)
    - A Text part (tableSetup)
    - An empty div (chartdiv):
      ```html
      <div id="chartdiv" style="height:400px; width:300px;"></div>
      ```
      - This last component will need to be typed in by hand
  - Make the changes to main.js and main.css shown in the next page:
function load()
{
    dashcode.setupParts();
}

function myClickHandler(event)
{
    var tablearray = [[1,1], [2,4], [3,9], [4,16]];
    var tablebegin = "<table id='myTable'><tr><th scope='col'>x</th><th scope='col'>y</th></tr>";
    var tableend = "</table>";
    var tablecontent = "";
    for (var i = 0; i < tablearray.length; i++)
    {
        xelement = "<td>" + tablearray[i][0] + "</td>";
        yelement = "<td>" + tablearray[i][1] + "</td>";
        tablerow = "<tr>" + xelement + yelement + "</tr>";
        tablecontent = tablecontent + tablerow;
    }
    document.getElementById("tableSetup").innerHTML = tablebegin + tablecontent + tableend;
    $.jqplot('chartdiv', [[[1, 2], [3.5,12],[5,13.1],[7,33.6],[9,85.9],[11,219.9]]]);
}

table {
    border-collapse: collapse;
    border-style: solid;
    border-width: 1px;
}

td {
    border-style: solid;
    border-width: 1px;
}
Accessing Native Functions: the Photo Library

- Decompress PhoneGap (*phonegap-iphone.zip*)
- Install it by double-clicking on *PhoneGapLibInstaller.pkg*
- Should this not work, full instructions for an alternative install are available at: http://phonegap.pbworks.com/Getting-Started-with-PhoneGap-(iPhone)
  - This alternative install will require that you install Git first: http://code.google.com/p/git-osx-installer/downloads/list
- Launch XCode, and create a PhoneGap project
- Modify the index.html file (inside the www folder) as shown in the next page
- When running the finished code under iPhone Simulator 3.1.3, make sure to choose an image that is not one of the photos included with the simulator – due to a bug in the simulator, these photos will not work
<html>
<head>
    <meta name="viewport" content="width=320; user-scalable=no" />
    <meta http-equiv="Content-type" content="text/html; charset=utf-8">
    <title>PhoneGap Photo Library</title>
    <link rel="stylesheet" href="master.css" type="text/css" media="screen" title="no title" charset="utf-8">
    <script type="text/javascript" charset="utf-8" src="phonegap.js"></script>
    <script type="text/javascript" charset="utf-8">
        function PictureSourceType() {}
        PictureSourceType.PHOTO_LIBRARY = 0;
        PictureSourceType.CAMERA = 1;

        function getPicture(sourceType)
        {
            var options = { quality: 10 }; 
            if (sourceType != undefined) {
                options["sourceType"] = sourceType;
            }
            navigator.camera.getPicture(getPicture_Success, null, options);
        }

        function getPicture_Success(imageData)
        {
            document.getElementById("test_img").src = "data:image/jpeg;base64," + imageData;
        }
    </script>
</head>
<body id="stage" class="theme">
    <img style="width:60px;height:60px" id="test_img" src=""></img>
    <button onclick="getPicture(PictureSourceType.PHOTO_LIBRARY)" style="width:60px;height:60px">From Photo Library</button>
</body>
</html>
Hippasus

http://hippasus.com/rrpweblog/
rubenrp@hippasus.com

This work is licensed under a Creative Commons Attribution-Noncommercial-Share Alike 3.0 License.