Mobile 2
&
Graphics
Agenda: Implementing Mobile

- Geolocation
- Orientation
- Gestures
- Canvas
- Caching applications
jQuery Mobile

• One of oldest and previously most used mobile frameworks.
• Recently has been less-well supported and has become more risky to build new apps on.
• Remaining benefits
  – Supports a unified UI framework across mobile platforms
  – Supports touch events
• Touch events and orientation are still browser-dependent, so jQM is still useful for this.
• Another option: hammerjs.github.io
  – Seems to be broadly respected, I have not used it.
Geolocation and Orientation

• Review
  – Geolocation example
    • https://geolocation.67328.is
    • Code at https://geolocation.67328.is/_src
  – Orientation example code
    • https://orientation.67328.is
    • Code at https://orientation.67328.is/_src
    • I'm using a method from jQuery Mobile because there is no standard that all browsers have adopted.
    • jQM deals with the idiosyncrasies.
Standard touch events

• touchstart
  – finger first touches the screen
• touchmove
  – finger moves in contact with the screen
• touchend
  – finger leaves the screen
• touchcancel
  – device launches another app
jQuery Mobile touch events:

- **tap**
  - quick touch and release on an HTML element
- **taphold**
  - touch and hold on an HTML element
- **swipe**
  - swipe gesture in any direction
- **swipeleft**
  - swipe gesture to the left
- **swiperright**
  - swipe gesture to the right
Mobile Graphics

- **SVG** – Scalable Vector Graphics
  - similar to Illustrator
- **Canvas** – Bitmapped graphics
  - similar to Photoshop

<table>
<thead>
<tr>
<th>Canvas</th>
<th>SVG</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pixel based (Dynamic .png)</td>
<td>Shape based</td>
</tr>
<tr>
<td>Single HTML element</td>
<td>Multiple graphical elements, which become part of the DOM</td>
</tr>
<tr>
<td>Modified through script only</td>
<td>Modified through script and CSS</td>
</tr>
<tr>
<td>Event model/user interaction is granular (x,y)</td>
<td>Event model/user interaction is abstracted (rect, path)</td>
</tr>
<tr>
<td>Performance is better with smaller surface, a larger number of objects (&gt;10k), or both</td>
<td>Performance is better with smaller number of objects (&lt;10k), a larger surface, or both</td>
</tr>
</tbody>
</table>

Table source and good discussion of when to use which:
Examples / Text

• These examples are adapted from Richard Roger's *Beginning Mobile Application Development in the Cloud*

• I'll provide chapter excerpts that discuss these examples
  – (before my edits)

• I recommend you read them.

• He does a good job of explaining code.
Canvas

• Review the code

• How to:
  – Change the background color to blue
  – Change the line color to magenta
  – Have parallel lines drawn with every line 15px to the right and down
Off-line web apps

- Web apps can be designed to work when off-network
  - e.g. poor reception, on airplanes, tablets with wifi only
1. Use application caching to store the code
2. Use localStorage to store data
3. Implement sync code to sync data to server when network is available.

- Good discussion and example in section 20.4.3 of JavaScript: The Definitive Guide.
Application Caching

• Application caching works for desktop browsers as well as mobile

• When you web app is launched (or visited)
  – The browser first gets the manifest
  – If it did not change:
    • use cached files
  – If it did change:
    • get files over the network and replace cache
  – If the network is down (or off)
    • use the cached files

• The manifest can also indicate what files should not be cached
Count app

- Code walkthrough
  - localStorage
Application Caching

• If multiple html pages, then each should point to the same cache manifest

• You need to include *all* of the files that much be cached.
  
  – E.g. I originally was missing http://code.jquery.com/ ... /icons-36-white.png which is required by jQuery Mobile
Web App stored to phone vs. in browser

• Once you save a web app icon to the iPhone
  – its localStorage is different than the browser's local storage
  – you need to visit it the web app once to cache it before you can go into airplane mode and test it
Work for Next Tuesday

• Mobile Canvas Homework