Outline

- Mobile browsers:
  - The mobile browsing experience
  - The WebKit engine
  - Preinstalled browser
  - User-installable browsers
- Mobile web application best practices:
  - Security, privacy, and user awareness
  - Conservative use of resources
  - User experience
  - Handling variances in the delivery context
  - Misc.

The Mobile Browsing Experience (1)

- Navigation:
  - Focus navigation
  - Cursor navigation
  - Touch navigation
  - Multitouch navigation

- Zoom Experience:
  - Basic zoom capabilities
  - Smart zoom capabilities

- Reflow Engines:
  - Reflows multicolumn pages to a one-column design

The Mobile Browsing Experience (2)

- Proxied browsers using proxies to:
  - Eliminate mobile incompatible content
  - Compress content
  - Pre-rendering for faster display
  - Convert content
  - Encrypt content
  - Cache content for frequently visited sites

- Multipage experience:
  - One-page support only
  - Multiple windows
  - Window stack
  - Tab navigation

The WebKit Engine

- An open source layout engine for web browsers

- Originally developed by Apple

- Widely used in the mobile world:
  - Many applications using it
  - Similar rendering experience can be expected on different mobile devices
  - but differences do exist between the implementations

Preinstalled Browsers

- Practically every new phone has a browser:
  - Android browser
  - Internet Explorer Mobile
  - Safari
  - Symbian browser
  - …

User-installable Browsers

- Users may install other browsers also:
  - Firefox Mobile
  - Opera Mobile
  - Opera Mini

Outline

- Mobile browsers:
  - The mobile browsing experience
  - The WebKit engine
  - Preinstalled browser
  - User-installable browsers

- Mobile web application best practices:
  - Security, privacy, and user awareness
  - Conservative use of resources
  - User experience
  - Handling variances in the delivery context
  - Misc.

Security, Privacy, User Awareness

- JSON data may represent a security risk:
  - Don't `eval()` JSON data directly, use a JSON parser

- Signing in at every visit can be a hassle:
  - Enable automatic sign-in (and then also sign-out)
  - but don't store unencrypted passwords

Conservative Use of Resources

- Guidelines:
  - Minimize application and data size:
    - Remove whitespace/minify HTML, CSS, and script files
  - Optimize network requests, such as:
    - Batch up multiple requests, if possible
    - Back off during periods of inactivity
  - Minimize external resources:
    - Merge script and CSS files into as few files as possible
  - Minimize sending of cookie information
  - Keep DOM Size Reasonable:
    - Use pagination

User Experience

- Guidelines:
  - Optimize application start-up time, such as:
    - Consider partitioning large scripts
    - Use local storage where appropriate

  - Minimize perceived latency, e.g., by:
    - Enabling incremental rendering
    - Avoid page reloads

  - Make telephone numbers "Click-to-Call":
    - Add a hyperlink with `tel:` or `sms:` as protocol

  - Use Meta Viewport to identify desired screen size

  - Design for multiple interaction methods (Focus based, pointer based, and touch based methods

Variances in the Delivery Context

- Guidelines:
  - Prefer server-side context where possible
    - MIME types, user agent, and WAP profile data
    - Consider WURFL – a device description repository

  - Use client-side capability detection if necessary
    - CSS Media Types/Media Queries, JavaScript

  - Classify devices for simplified adaption
    - E.g., Basic XHTML -> full AJAX -> device APIs

  - Support a non-JavaScript variant if appropriate

  - Offer user a choice:
    - Let the user change UI if multiple exist

And some more …

- Consider use of Canvas elements or SVG

- Inform the user about automatic network access (XMLHttpRequest)

- Provide sufficient means to control automatic network access (XMLHttpRequest)

Conclusion

- A variety of **browsers** running on a variety of **devices** for a variety of **users**

- demand a conscious and rigorous process for designing mobile web applications