Client side web programming

Responsive Design & HTML5

Jaana Holvikivi
School of ICT

Helsinki Metropolia University of Applied Sciences

Contents

- Responsive design
- Accessibility
- Audio and video
- Geolocation
- JQuery

Helsinki Metropolia University of Applied Sciences

Responsive design: Some rules

- A good mobile experience requires a different design than the desktop.
- Start from mobile,
  - create **a mobile first design**,
  - then take performance seriously,
  - use Javascript to add in additional content, and
  - always use fluid layouts.
- This isn't mobile Web design or desktop Web design, it's responsive Web design.

http://www.uie.com/articles/strategy_for_responsive_design

Helsinki Metropolia University of Applied Sciences

Devising a Strategy for Responsive Design

The Core Tactics: 1. Discovering the breakpoints.

- These are the page widths that will cause design elements to re-order.
  In between breakpoints, items will change their size or flow.
- A responsive design can have multiple breakpoints:
  - for a small-screen phone,
  - a large-screen phone,
  - a tablet,
  - a laptop/desktop.
- By letting the content and navigation drive the breakpoints, teams find they can often get away with fewer screen configurations. (high-resolution Retina iPad & laptop display, lower resolution tablets might just need a little adjustment to that same configuration).

http://www.uie.com/articles/strategy_for_responsive_design

Helsinki Metropolia University of Applied Sciences

The Core Tactics 2 & 3

2. Keeping page load times low.
   - Smart placement of media queries and progressive enhancement can dramatically reduce the footprint of the CSS file on smaller, slower devices.
3. Image size optimization.
   - Right now, this is the hardest core tactic to get under control, because there are no solid best practices to follow.
   - Retina and other high-resolution technologies create a problem for teams, because they need large images to look good, but those same images are slow to load on lower resolution screens.

Helsinki Metropolia University of Applied Sciences

Mobile First

- Luke Wroblewski: "*What's the minimum amount of content and navigation that we need to make our design useful?*
- The guiding principle of Mobile First is that it's easier to add to a design than to take away. By starting with a minimum configuration, you can then add in more as you gain more space from larger screens and resolutions.
- It's possible that there are things in your current design that don't need to be in any responsive configuration, because they really aren't useful to the user.

Helsinki Metropolia University of Applied Sciences

Research First

- Does the team truly understand who their users are? Do they know what those users will need from their design?
- Field research, analytics, and other study methods to understand
  which functions are important and which are nice-to-have.
- A set of scenarios

Helsinki Metropolia University of Applied Sciences

Shifting Information to Interaction

- Having a lot of screen real estate gives the designer leeway, especially when it comes to flatting out the information displayed.
- A strategy for creating interactions where information once laid flat:
  - Using an interactive interface, users can directly manipulate filters to drill into the data they need.
- Site navigation:
  - from mega-menus
  - to thoughtfully interactive.
- Smaller screens beg for data to be a starting point of interactions,
  - with obvious touch actions for the most important functions
  - subtle gestures for shortcuts.

Helsinki Metropolia University of Applied Sciences

WURFL, the Wireless Universal Resource FiLe

- WURFL is a Device Description Repository (DDR),
- a software component that maps HTTP Request headers to the profile of the HTTP client (Desktop, Mobile Device, Tablet, etc.) that issued the request
- wurfl.xml file (i.e. the repository) contains that definition of thousands of devices
- WURFL Cloud is available at varying prices, also a free Cloud
- WURFL APIs for all major platforms (PHP, Java, .net, C++ and more)

Helsinki Metropolia University of Applied Sciences

CSS and media query

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

@media (query)
        {
        /* CSS Rules used when query matches */
        }
@media (min-width: 500px) and (max-width:
600px)
```

Helsinki Metropolia University of Applied Sciences

Web Accessibility Initiative WAI

- http://www.w3.org/WAI/
- Web Content Accessibility Guidelines (WCAG) 2.0
- Designing for Inclusion design for all
- legislation requirements, see i.e. http://www.intermin.fi/fi/kehittamishankkeet/maahanmuutto_2020

Helsinki Metropolia University of Applied Sciences

Web Content Accessibility Guidelines 2.0

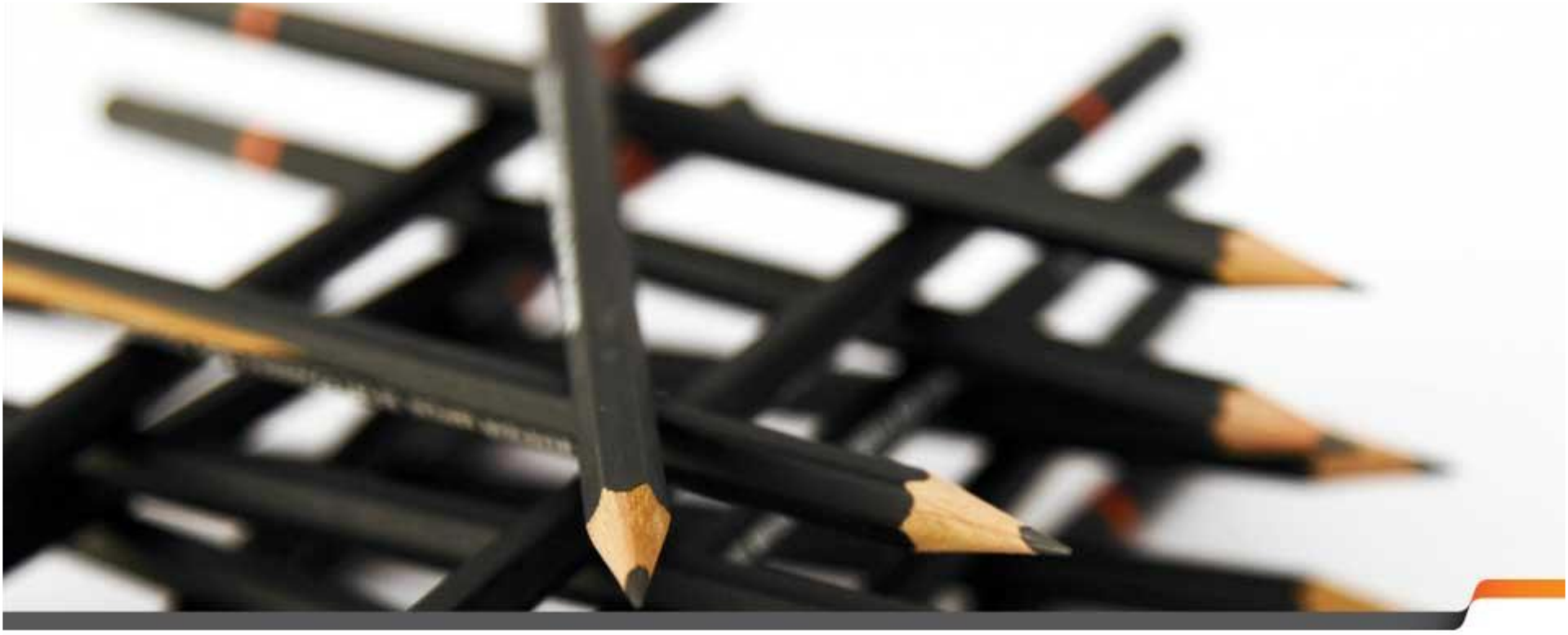
- avoid Flash and PDF
- Canvas might not show for screen readers
- Provide text alternatives for any non-text content
- Time-based Media: Provide alternatives for time-based media; audio and video
- Adaptable: Create content that can be presented in different ways (for example simpler layout) without losing information or structure.

Helsinki Metropolia University of Applied Sciences

Web Content Accessibility Guidelines

- Distinguishable: Make it easier for users to see and hear content including separating foreground from background.
- Keyboard Accessible
- Enough Time: Provide users enough time to read and use content.
- Seizures: Do not design content in a way that is known to cause seizures.
- Navigable
- Make text content **readable** and understandable.
- Make Web pages appear and operate in **predictable** ways.
- Input Assistance: Help users avoid and correct mistakes.
- Compatible: Maximize compatibility with current and future user agents, including assistive technologies

Helsinki Metropolia University of Applied Sciences



HTML5 features

Helsinki Metropolia University of Applied Sciences

Audio and video

- Video formats:
  - H.264 or MPEG-4 in Apple products (incl. Safari)
  - Ogg: open source, free, see xiph.org
    - Firefogg.org converts into ogg
  - WebM
- Only Chrome supports all three

```
<div style="text-align:center" id="video1">
<video controls width="420" height="150">
    <source src="fish.mp4" type="video/mp4">
    <source src="fish.ogg" type="video/ogg">
    <source src="fish.webm" type="video/webm">
    Your browser does not support HTML5 video.
  </video>
</div>
```

Helsinki Metropolia University of Applied Sciences

Audio and video

- Audio formats:
  - mp3
  - Ogg or oga
  - wav

```
<div style="text-align:center">
<audio controls id="audio1">
    <source src="bird1.mp3" type="audio/mpeg">
    <source src="bird1.oga" type="audio/ogg">
    <source src="bird1.wav" type="video/wav">
    Your browser does not support the audio element.
  </video>
</div>
```

Helsinki Metropolia University of Applied Sciences

Geolocation

The geolocation object

- The geolocation API is published through a geolocation child object within the navigator object. If the object exists, geolocation services are available

```
if ("geolocation" in navigator) {
  /* geolocation is available */
} else {
  alert("I'm sorry, but geolocation services are not supported by your browser.");
}
```

- https://developer.mozilla.org/en-US/docs/Using_geolocation
- http://dev.w3.org/geo/api/spec-source.html

Helsinki Metropolia University of Applied Sciences

Getting the current position

To obtain the user's current location:

- call the getCurrentPosition() method.
- This initiates an asynchronous request to detect the user's position, and queries the positioning hardware to get up-to-date information.
- When the position is determined, a specified callback routine is executed.
- You can optionally provide a second callback to be executed if an error occurs. A third, optional, parameter is an options interface where you can set the maximum age of the position returned and the time to wait for a request.

```
navigator.geolocation.getCurrentPosition(function(position) {
  do_something(position.coords.latitude, position.coords.longitude);
});
```

The above example will cause the do_something() function to execute when the location is obtained.

Helsinki Metropolia University of Applied Sciences

A location-aware web page 1

```
<!DOCTYPE html>
<html>
<body>
<p id="demo">Click the button to get your coordinates:</p>
<button onclick="getLocation()">Try It</button>
<script>
var x=document.getElementById("demo");
function getLocation()
  {
  if (navigator.geolocation)
    {
    navigator.geolocation.getCurrentPosition(showPosition);
    }
  else{x.innerHTML="Geolocation is not supported by this browser.";}
  }
function showPosition(position)
  {
  x.innerHTML="Latitude: " + position.coords.latitude +
  "<br>Longitude: " + position.coords.longitude;
  }
</script></body></html>
```

Helsinki Metropolia University of Applied Sciences

A location-aware web page 2

```
<!DOCTYPE html>
<html><head><title>Location</title>
<script>
function findYou(){
            navigator.geolocation.getCurrentPosition(showPosition,
            noLocation, {maximumAge:1100000, timeout:30000});
   }
function showPosition(position){
  var latitude = position.coords.latitude;
  var longitude = position.coords.longitude;
  var accuracy = position.coords.accuracy;
  document.getElementById("lat").innerHTML="your latitude is "+ latitude;
  document.getElementById("lon").innerHTML="your longitude is "+ longitude;
  document.getElementById("acc").innerHTML="accurate within "+ accuracy +
"meters";
 }
function noLocation(locationError){
            document.write("Request failed");}

</script> </head>
<body>on the next page
 </body></html>
```

```
<body>
<h2>Your location</h2>
<script>
findYou();
</script>
<p id="lat">Here</p>
<p id="lon">Here</p>
<p id="acc">Here</p>
</body>
</html>
```

Helsinki Metropolia University of Applied Sciences

Google map call

Documentation at:
https://developers.google.com/maps/

```
<script src="http://maps.googleapis.com/maps/api/js?sensor=false">
</script>

var position = new google.maps.LatLng(latitude, longitude);

var map=new
google.maps.Map(document.getElementById("map1"),mapOpt);

<body>
<div id="map1" style="width:500px;height:450px;"></div>
</body>
```

Helsinki Metropolia University of Applied Sciences

http://jquery.org/ - Javascript library

Adding jQuery to Your Web Pages

- Download the jQuery library from jQuery.com;
  & Include it in your document

```
<head>
<script src="jquery-1.8.3.min.js"></script>
</head>
```

- Include jQuery from a CDN (Content Delivery Network), like Google

```
<head>
<script src="//ajax.googleapis.com/ajax/libs/jquery/1.8.3/jquery.min.js">
</script>
</head>
```

- or Microsoft

```
<head>
<script src="//ajax.aspnetcdn.com/ajax/jQuery/jquery-1.8.3.min.js">
</script></head>
```

Helsinki Metropolia University of Applied Sciences

jQuery Syntax

- The jQuery syntax is tailor made for **selecting** HTML elements and perform some **action** on the element(s).
- **$(*selector*).*action*()**
- A $ sign to define/access jQuery
- A (*selector*) to "query (or find)" HTML elements
- Similar to CSS selectors
- A jQuery *action*() to be performed on the element(s)
- Examples:
  $(this).hide() - hides the current element.
  $("p").hide() - hides all <p> elements.
  $(".test").hide() - hides all elements with class="test".
  $("#test").hide() - hides the element with id="test".

Helsinki Metropolia University of Applied Sciences

Jquery actions

- Fired by events, such as
  moving a mouse over an element;
  selecting a radio button;
  clicking on an element
- Effects: hide/ show, animate
- Changing contents of the page (DOM)
- jQuery provides several methods for AJAX functionality (server communication)

Helsinki Metropolia University of Applied Sciences