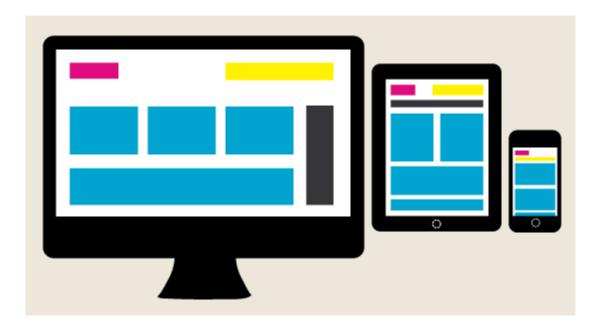# Responsive Web Design

INF5750/9750 - Lecture 6 (Part III)

# Problem Area

• Different types of devices are used to access the WWW, but we want to use the same HTML code

• Flexibly adjust layout, content, scale images etc

• Devices also have different capabilities

• You also want to use the same content for different purposes i.e. web viewing, printing, presentations

• Instead of needs for a desktop Web version with most common screen resolutions, and a particular mobile version, the idea is to approach the issue the other way around: use flexible and fluid layouts that adapt to almost any screen.

# Responsive Web Design (RWD)

- Three key technical features are the heart of responsive Web design:
  - Media queries and media query listeners
  - A flexible grid-based layout that uses relative sizing
  - Flexible images and media, through dynamic resizing or CSS
- Responsive websites *respond* to their environment

# Media Queries

- There are three ways to implement media queries:
    - Use @import to import style rules from other style sheets

    ```
    @import url(style600min.css) screen and (min-width: 600px);
    ```

    - Put media queries directly in the style sheet

    ```
    #nav      { float: right; }
    #nav ul  { list-style: none; }
    @media screen and (min-width: 400px) and (orientation: portrait) {
        #nav li   { float: right; margin: 0 0 0 .5em; border:1px solid #000000; }
    }
    @media screen and (min-width: 800px) {
        #nav  { width: 200px; }
        #nav li  { float: left; margin: 0 0 0 .5em; border: none; }
    }
    ```

    - Include a query in a linked style sheet's media attribute

    ```
    <link rel="stylesheet" type="text/css" media="screen and (max-device-width: 800px)"
    href="style800.css" />
    ```

    - How to test? To get the real resolution use viewports

    ```
    <meta name="viewport" content="width=device-width">
    ```

# CSS + JS = Media Query Listeners

• Because of the (cascading) nature of CSS, default styles are defined at the top with the media query matching rules and styles below

• Styles defined at the top will be cascaded to the matching styles in the rule, or even completely overwritten

• Listeners are useful to monitor changes and respond

```
// media query event handler
if (matchMedia) {
    var mq = window.matchMedia("(min-width: 500px)");
    mq.addListener(WidthChange); WidthChange(mq);
}

// media query change
function WidthChange(mq) {
    if (mq.matches) {
            // Download normal images since window width is at least 500px
    } else {
            // Download scaled smaller images since window width is less than 500px
    }
}
```

# Flexible Grids (Lots of math)

• A flexible grid-based layout is one of the cornerstones of responsive design

• Using CSS for positioning and for laying out margins and spacing

• Stop using pixel-based layouts and start using percentages or the em for sizing, the formula for calculating the em:

- target ÷ context = result

• Normal context for the body font size is 16 pixels

• If the designer specifies that the h1 should be 24 pixels, you can calculate the following: 24 ÷ 16 = 1.5

This results in the following CSS style:

h1{  font-size: 1.5em;  }

# CSS Grid (Grid Alignment)

- start by setting the display block to 'grid'

- column layouts using standard CSS3

```
<div class="wrapper">
    <article class="main">My awesome content here</article>
    <footer class="footer">Some informations here</footer>
    <header class="header">My site title goes here</header>
    <aside class="sidebar">Here is my side content</aside>
    <aside class="annexe">Some more side content</aside>
</div>

.wrapper {  display: grid; grid-columns: 200px 15px 1fr 15px 100px;  grid-rows: (auto 15px)[2] auto; }
.header, .footer { grid-column-span: 5; }
.sidebar, .main, .annexe { grid-row: 3; }
.header { grid-row: 1; }
.footer { grid-row: 5; }
.sidebar { grid-column: 1; }
.main { grid-column: 3; }
.annexe { grid-column: 5; }
```

**On small screen width**

**On large screen width**

# Resources

- A Book Apart – Responsive Web Design
  - http://www.abookapart.com/products/responsive-web-design

- Media Queries, W3C
  - http://www.w3.org/TR/css3-mediaqueries/

- Responsive Web Design, MSDN Magazine
  - http://msdn.microsoft.com/en-us/magazine/hh653584.aspx