Enterprise Computing: Responsive Design

Professor Stephen Gilmore
School of Informatics
The University of Edinburgh

January 26, 2015
1. Introduction

2. Responsive Web Design
   - Content: Josh Hughes, University of Missouri

3. Non-functional requirements

4. Closing
Non-functional requirement (#5 of 10)

This image represents non-functional requirement #5. If you were at the lecture then you heard me explain in words what it means.
Enterprise systems are data-sharing systems

- Enterprise computing systems exist to maintain and share data, making the information which is important to the enterprise available to as many people as possible.
- The people who consume the enterprise data are distributed in different geographical locations, and/or different time zones, software and hardware updates are completely uncoordinated.
- The challenge is to make the enterprise data available to as many users as possible, across as wide a range of devices as possible.
- The most practical delivery mechanism for data would seem to be the browser.
Not all browsers are the same

What are we?  Browsers!  ...  Browsers!  Browsers!

What do we want?  Faster!  ...  Faster!  Faster!

When do we want it?  Now!  ...  Now!  Now!

Browsers!
Standards

The nice thing about standards is that you have so many to choose from. — Andrew S. Tanenbaum

The awful thing about standards is how much people overuse that Andrew S. Tanenbaum quote. — Stephen Gilmore

- The HTML5 language, standardised in October 2014, is the leading choice for cross-platform mobile applications.

- Many language features are designed with low-powered devices such as tablets or smartphones in mind.

- In theory, standards should fix everything, making any cross-platform testing unnecessary. In practice, they don’t.
One of the advantages of being in a team is that you have access to more resources: talent, knowledge, skill sets, ideas, and also different devices.

It should be the case that you have access to more than one OS (Linux, OS X, Windows etc) with more than one browser (Chrome, Firefox, Safari, Explorer, etc).

It is likely to be the case that you have access to devices with different screen sizes (PC, laptop, tablet, phone, etc). (Or you know someone who has such a device, even if they are not in your team, or even on the course.)

You may also have access to a range of different mobile devices (iPhone, Android, etc).

This is the basis of cross-platform testing.
A web connection

"Get me example.com, please!"

"Hey 272.123.23.2, send me your homepage!"

Client  DNS  Host

Credit: http://alistapart.com/article/planning-for-performance
A web connection from a mobile

Credit: http://alistapart.com/article/planning-for-performance
Responsive Web Design

- Content: Josh Hughes, University of Missouri
Responsive Web Design

Josh Hughes
hughesjd@missouri.edu
Fluid Grid

Page - 960 px

Menu
215 px

Content Area
685 px

Inset Sidebar
215 px

20 px left margin

20 px left and right margins

20 px left and right margins
Fluid Grid

target ÷ context × 100 = percentage
Fluid Grid

Menu:
\[
\frac{215}{960} \times 100 = 22.3958333333\%
\]

Content Area:
\[
\frac{685}{960} \times 100 = 71.3541666667\%
\]

Inset Sidebar:
\[
\frac{215}{685} \times 100 = 31.3868613139\%
\]
Fluid Grid

20px margin for the Menu and Content Area:
\[
\frac{20}{960} \times 100 = 2.0833333333\%
\]

20px margin for the Inset Sidebar:
\[
\frac{20}{685} \times 100 = 2.9197080292\%
\]
## Fluid Grid

<table>
<thead>
<tr>
<th>Menu</th>
<th>Content Area</th>
<th>Inset Sidebar</th>
</tr>
</thead>
<tbody>
<tr>
<td>22.39583%</td>
<td>71.35416666667%</td>
<td>31.386861%</td>
</tr>
<tr>
<td>(215/960)</td>
<td>(685/960)</td>
<td>(215/685)</td>
</tr>
<tr>
<td>2.08333%</td>
<td>2.083333333333%</td>
<td>2.919708%</td>
</tr>
<tr>
<td>(20/960)</td>
<td>left and right margins</td>
<td>(20/685)</td>
</tr>
<tr>
<td>left margin</td>
<td></td>
<td>left and right margins</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(20/685)</td>
</tr>
</tbody>
</table>

Page - **90%** (Up to you)
Viewport Fix

```html
<meta
    name="viewport"
    content="width=device-width,
    initial-scale=1.0"
>
```
Viewport Fix

Without Meta Tag

Managing Slide Shows
Published on February 29, 2012 by Josh Nichols | Posts by This Author

In this presentation, we show how Web Communications uses Slideshow Pro Director to manage slide shows. This was presented at the Feb. Web Developers Group, but it’s the presentation as run by Slideshow Pro Director.

PHP includes with Cascade Server
Published on January 19, 2012 by Josh Hughes | Posts by This Author

The University of Missouri makes heavy use of [content management system](http://www.cms.missouri.edu), a MYSQL-based content management system. One of the downsides to Cascade is that it doesn’t build live pages on the fly. Changes to files and templates must be published out before they will be live on the server. From a fairly minor edit, like changing contact information in the footer, may require your entire site to be republished.

With Meta Tag

Managing Slide Shows
Published on February 29, 2012 by Josh Nichols | Posts by This Author

Interface Blog of Web Communications at the University of Missouri
Flexible Images

1. Set `max-width: 100%` on the `img`

2. Do not set `width` or `height` on the `img` in the HTML
   or
   Set `width: auto` and `height: auto` in the CSS
Media Queries

body {
    background: red;
}

@media screen and (min-width: 600px) {
    body {
        background: green;
    }
}
**Common Media Query Conditions**

`min-width` or `min-height`  
Applied if the window is equal to or greater than this value

`max-width` or `max-height`  
Applied if the window is equal to or less than this value
Common Media Query Conditions

`min-device-width` or `min-device-height`
Applied if the device screen is equal to or greater than this value

`max-device-width` or `max-device-height`
Applied if the device screen is equal to or less than this value
Responsive Web Design versus Other Options
Native Apps

Pros

• Can provide a slick user experience
• Can more easily access device features (camera, GPS, etc.)
• Available for offline use
Native Apps

Cons

• Very expensive
• Hard to do well
• Which platforms do you support?
• Your users (probably) don't want a native app
• You still need a website
Mobile-Specific Websites

Pros

• Easier to optimize for speed
• More freedom to create a unique mobile experience
• Can more easily target less advanced devices, like feature phones
Mobile-Specific Websites

Cons

• Have to deal intelligently with redirects
• Ignores tablets for the most part
• Tends to offer an incomplete experience
Responsive Web Design

Pros

• Only have to maintain a single website
• Don't need to deal with mobile-specific URLs
• Addresses a wide multitude of devices: phones, tablets, desktops, etc.
Responsive Web Design

Cons

• More difficult to optimize properly for specific devices (for example, phones might get desktop-sized images)
Dealing with Images
CSS Background Images

Pros
• Easy to setup
• Least likely option to result in both images getting downloaded

Cons
• Content editors probably aren’t going to be able to use this method
• With some caveats, they don’t resize
Navigation Design Patterns
Dropdown Menu

About
- Requirements
- Course List
- Transfers
- For Faculty/Staff
- Hesburgh Scholarship
- Contact Us

About
The University of Missouri is committed to preparing graduates of informed citizens. Graduates of MU must be able to understand complex issues that go beyond their academic field.

General education courses are the
Select Menu

Mizzou Identity Standards
University of Missouri

Logos & Design
Crafting Mizzou’s graphic identity

About Mizzou Identity
Pride Points
Logos & Design
Photos
Editorial
Websites
Just Stack ‘em
Off-Canvas Flyout
Responsive Design

Most things come down to the following options:

• Drop the content down
• Make the content viewable via a toggle
• Hide the content altogether (use sparingly)

You can also use Javascript if you need to rearrange the HTML to fit a design.
Not all websites are responsive
Non-functional requirement (#6 of 10)

https://www.youtube.com/watch?v=Sqz5dbs5zmo

This video clip represents non-functional requirement #6. If you were at the lecture then you heard me explain in words what it means.
When discussing web content and responsive design, there are three terms which you should know.

- Page bloat.
- FOUC.
- Polyfill, or polyfiller.

If you were at the lecture then you heard the discussion of what these terms mean.
Things to do now

Some things to do now

- Visit "A List Apart", the website about websites.
  - http://alistapart.com

- Find out about responsive images.

- Learn about polyfills such as Picturefill.
  - http://scottjehl.github.io/picturefill/

- Learn about sustainable Web design.
  - http://alistapart.com/article/sustainable-web-design
Live long and prosper