Responsive Web Design

Josh Hughes
hughesjd@missouri.edu
What is Responsive Design?

A quick overview
Fluid Grid
Fluid Grid

Page - 960 px

Menu
215 px
20 px left margin

Content Area
685 px
20 px left and right margins

Inset Sidebar
215 px
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><strong>22.39583</strong></td>
<td><strong>71.35416666667%</strong> (685/960)</td>
<td><strong>31.386861</strong> 3139% (215/685)</td>
</tr>
<tr>
<td><strong>33333%</strong></td>
<td></td>
<td><strong>2.919708 0292%</strong> (20/685)</td>
</tr>
<tr>
<td><strong>(215/960)</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>2.083333</strong></td>
<td><strong>2.08333333333%</strong> (20/960)</td>
<td></td>
</tr>
<tr>
<td><strong>3333%</strong></td>
<td>left and right margins</td>
<td></td>
</tr>
<tr>
<td><strong>(20/960)</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>left and right margins</strong></td>
</tr>
</tbody>
</table>

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

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

Without Meta Tag

With Meta Tag
Flexible Images
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.
Flexible Video

Similar situation as images, but maintaining proper proportions is a problem.

Easiest solution is to use Fitvids.
Media Queries
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
Example Sites
Browser Support

Short version:
Pretty much everything except Internet Explorer 8 and below
Browser Support

Long version:

Internet Explorer: 9.0+
Firefox: 3.5+
Chrome: 4.0+
Safari: 3.1+ (a bit buggy until 4.0, however)
Opera: 9.5+
iOS: 3.2+
Android: 2.1+
Windows Phone: 7.5+ (Mango)
Blackberry: 4.7.1+
“Fixing” Internet Explorer

There are Javascript options for filling the hole.

`respond.js` is the one I would recommend.
Native Apps
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
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
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)
Things to Keep in Mind as You Get Started
Set breakpoints wherever you need them

Make your choices based on the design rather than the device
Don’t assume that “mobile” users want less content
Mobile First
Mobile First

• Start your design process with mobile
Mobile First

• Start your design process with mobile
• Make the mobile view the default
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
CSS Background Images

Example:

```html
<div class="replace">Alternate Text</div>
```

```css
.replace
{
  text-indent: 100%;
  white-space: nowrap;
  overflow: hidden;
  background: url(images/replace.png) no-repeat;
  width: 600px;
  height: 200px;
}
```
CSS Background Images

Example:

```
.replace
{
    text-indent: 100%;
    white-space: nowrap;
    overflow: hidden;
    background: url(images/replace.png) no-repeat;
    width: 600px;
    height: 200px;
}

@media screen and (min-width: 600px)
{
    .replace { background-image: url(images/new.png); }
}
```
Inline Images

Pros
• Easy for content editors to add
• Scaling is easy

Cons
• Difficult to do any kind of swapping
• Very likely that the user will end up downloading both images
The Double-Download Problem
Dropdown Menu

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

About General Education

The University of Missouri is committed to preparing students for a lifetime of success through a well-rounded education. Graduates of MU must be able to understand and apply knowledge, skills, and attitudes that go beyond their academic field.

General education courses are the...
Select Menu
Just Stack ‘em

The Content Strategy Conference Is Back (And Sold Out)!

Meet the Speakers

Lou Rosenfeld
Ann Rockley
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.
Thanks!

Questions? Comments?

Josh Hughes
hughesjd@missouri.edu