

# Mobile Design

Responsive & Adaptive Websites

# Brochure-ware

- Web mimicked hard copy design
- TBL tech, not designed for that
- Use set heights and widths to "nail" elements
- A desktop was a desktop was a desktop - the fact of life
  - 1024 x 768 display resolution
- Web evolution: users don't like to scroll
  - Horizontal and vertical
- "Above the fold"
- page "chunking"
- small fonts
- boxes, densely packed

# June 29, 2007



- No mobile-friendly websites
- Screen: 480 x 320px (portrait)
- New fact of life: pinch-and-zoom
  - Default: every website, zoomed out

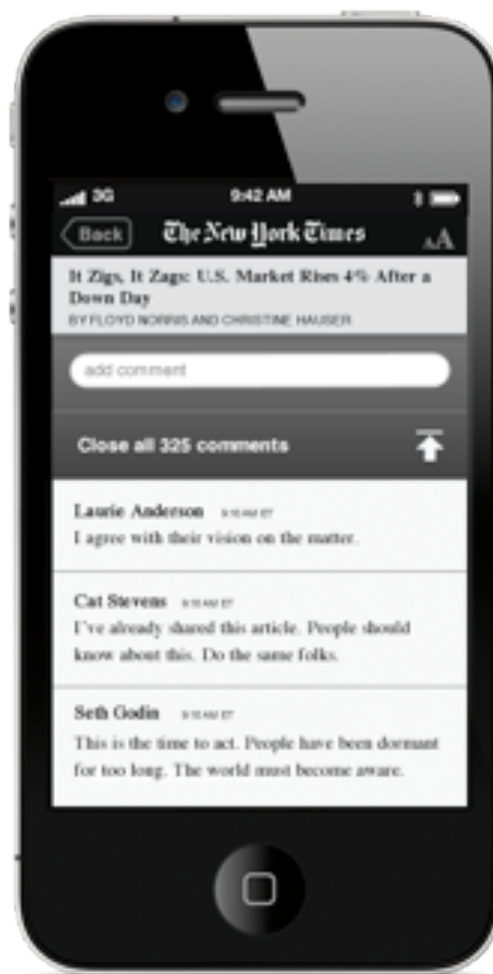




"Responsive"

"Fluid"

"Adaptive"

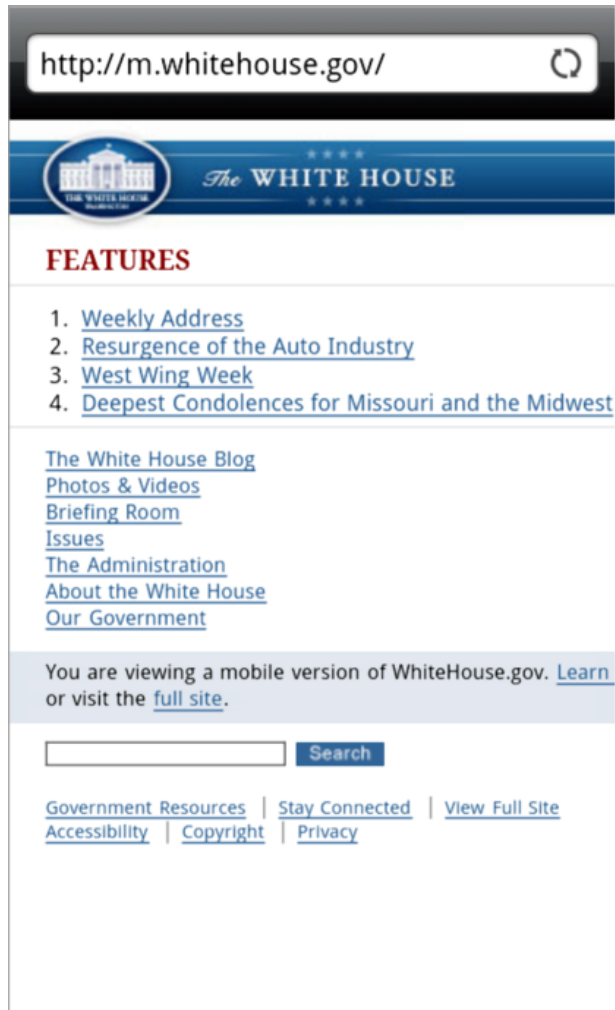


# Mobile Web Limitations

- Small Screen Size
- Low bandwidth
- Awkward controls
- Limited processor and memory



# Mobile Web Design Best Practices



- Assuming: the same website makes sense on a mobile device, then...
- Need to re-factor:
  1. Layout
  2. Navigation
  3. Graphics
  4. Text

# Best Practices

Layout, Navigation, Graphics, Text

# Dealing with Mobile Devices

## Goals

- **No horizontal scroll bars**
- **All elements fit their containers**
- Text must not be too small to read comfortably; there must be a lot of contrast between text color and its background
- Images must still be recognizable for what they are, i.e. not too small
- The navigation elements look like buttons and they are finger-sized (easily tap-able)

## Techniques:

1. Add **viewport meta tag**
2. **Scale images** as necessary
3. Use **media queries** to UNDO desktop layouts
  - Return side-by-side content to the natural document flow
  - Remove floats
4. **Reconfigure or move** wide elements like HTML Tables

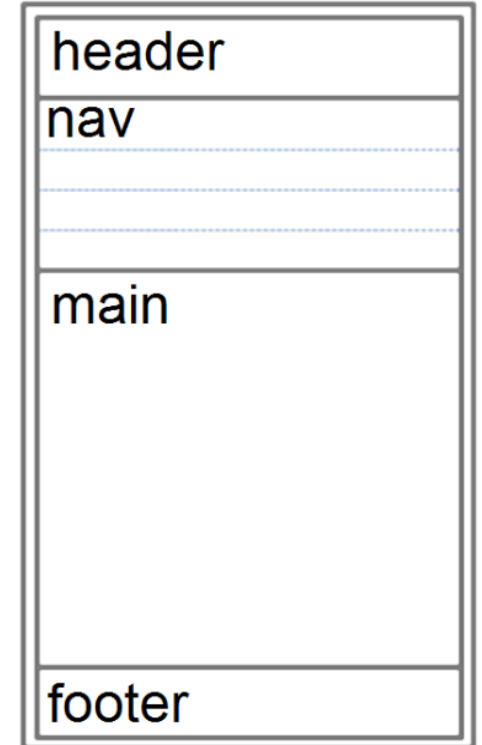


# Layout

- Single column design
- Limit scrolling to one direction
- Use clear, short heading elements
- Use simple lists with short phrases instead of heavy textual content
- Avoid using HTML tables
- Avoid using pixel units in style sheets
- Avoid absolute positioning
- Hide content that is not essential for mobile use.



wrapper



# Navigation

- Provide minimal navigation near the top or bottom of the page
- Toggle lesser navigation until needed
  - Use micro-interactions to show sub-menus as needed (sliding in and out when tapped)
  - hamburger menus?
- Avoid hyperlinks that open files in new windows or pop-up windows

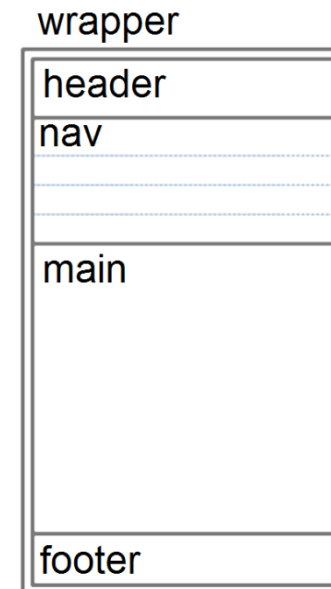
# Images

- Avoid using floats
- Avoid displaying images that are wider than the screen width
- Configure alternate, small optimized background images or no background images at all
- Avoid using images that contain text
- Avoid the use of large graphic images (large file sizes)

# Text

*Remember: images can scale; text stays the same size*

- Configure good contrast between text and background colors
- Use common font typefaces
- Increase text sizes (sometimes)
- Use a short, descriptive headings



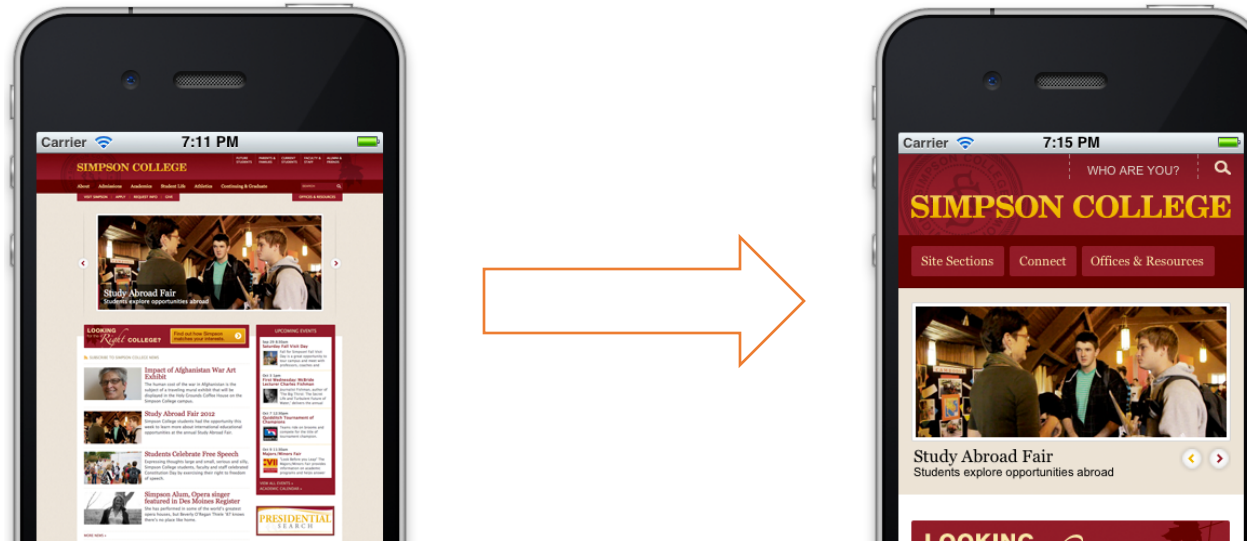
# Viewport Meta Tag

*The default action for most mobile devices is to zoom out and scale the web page*

The fix...

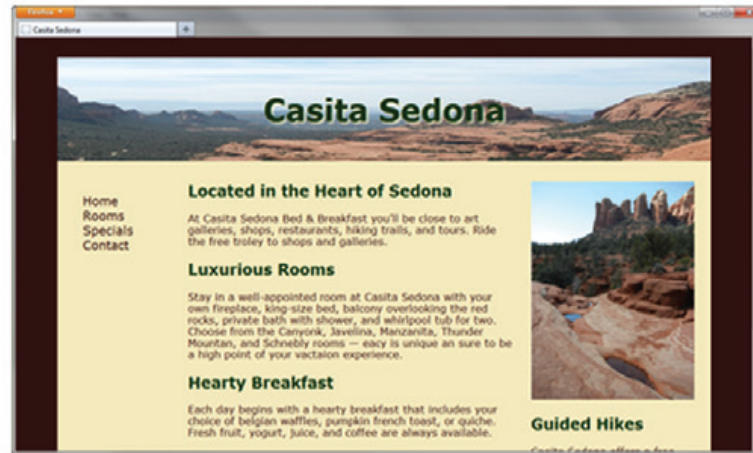
- Viewport Meta Tag
- Created as an Apple extension to configure display on mobile devices
- Configures width and initial scale of browser viewport
- Insert, in the <head>...

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

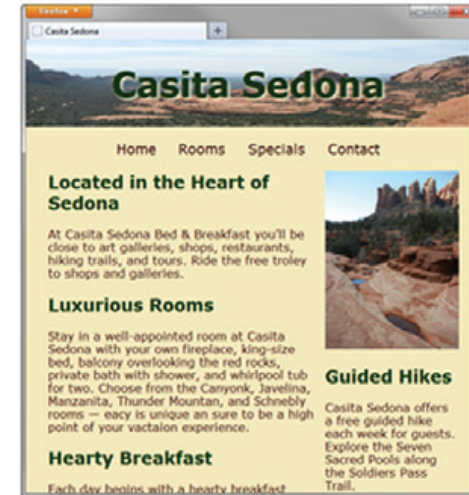


# Flexible Images in HTML

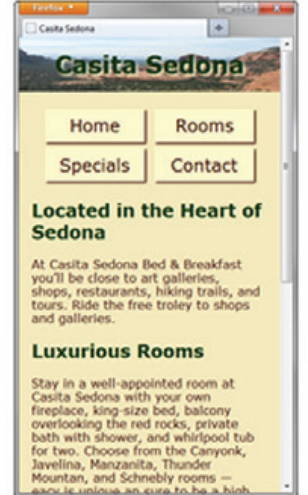
Desktop Browser



Tablet  
Display Width



Smartphone  
Display Width



- In the HTML:
  - remove height and width attributes
  - wrap the image in a container that has a controlled width
- In the CSS:

```
.scalable { width: 100%; }
```

# Media Queries

- Media Query
  - Determines the capability of the mobile device, such as screen resolution
  - Directs the browser to styles configured specifically for those capabilities

- In CSS

```
@media (max-width: 768px) {  
    //CSS here  
}  
  
@media (max-width: 480px) {  
    //CSS here  
}
```

Reference:

[https://developer.mozilla.org/en-US/docs/Web/Guide/CSS/Media\\_queries](https://developer.mozilla.org/en-US/docs/Web/Guide/CSS/Media_queries)



# Reconfigure HTML Tables, Floats ...anything that doesn't fit

- HTML Tables
  - Change the *font-size*
  - Move the table to a different area of the webpage
  - Use *overflow-x:auto* on a container element
  - Reconfigure the table into some other element (like an unordered list) using JavaScript
  - Google: **responsive html tables** ...or something like that
- Floats
  - Remove floats – use *text-align: center* instead