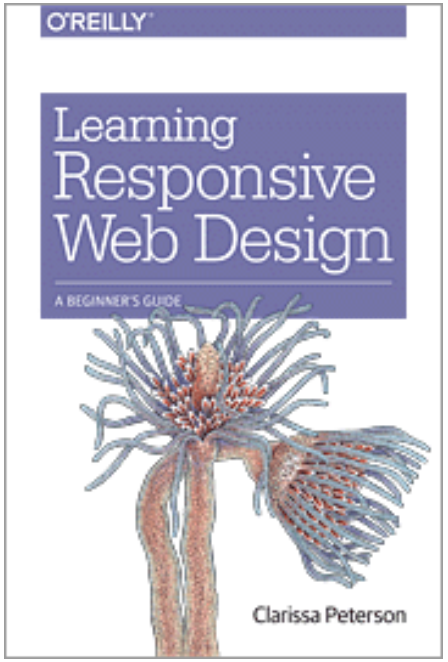
Mobile web development

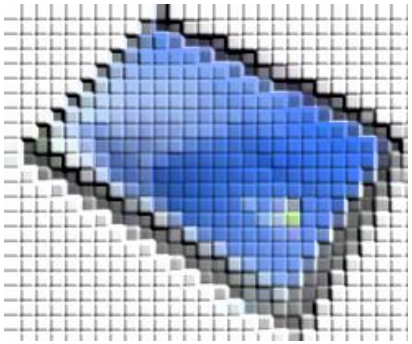
Putting internet technology on phones and tablets

CENTRIQ TRAINING

Course materials

- **Learning Responsive Web Design - A Beginner's Guide** by Clarissa Peterson
- O'Reilly Media, June 2014

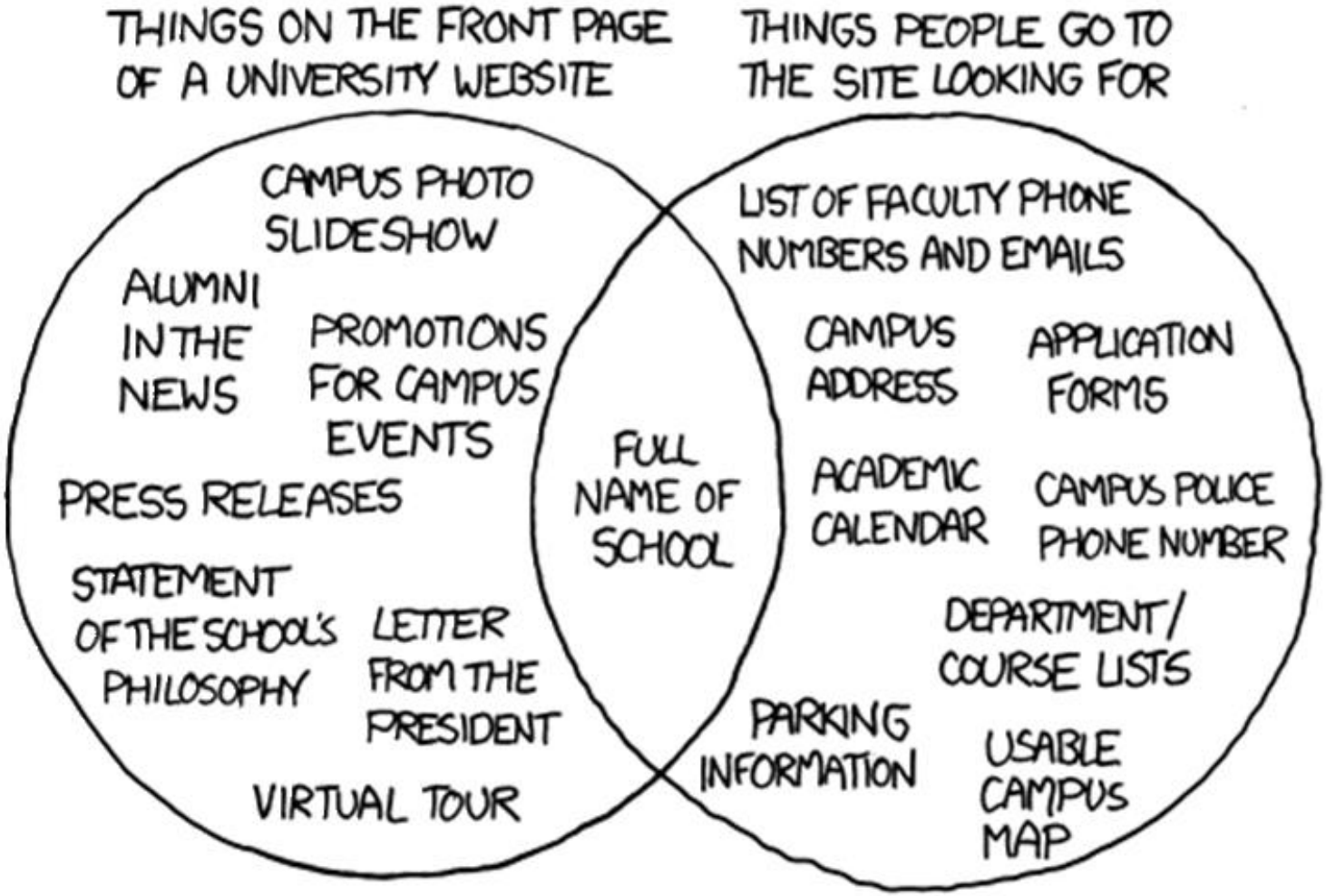




A designer’s choices are a small limited form of tyranny. - Ethan Marcotte

Design Strategy

Extending current web to mobile?



A mobile web experience

- Understand the user and solve their problems
  - Find the capabilities that work
- Not another web site
  - Don’t redirect to other URLs. Better in one site.
- Not just an app
  - What’s the goal for the container?
  - The web experience is just water.

Mobile user behaviors

- Three mobile user behaviors
  - I'm **microtasking**
    - **repetitive** and frequent jobs
    - **urgent** or changing info, updates
  - I'm **bored** – explore, escape, engage
  - I'm **local** – what's happening here, context
- *Tapworthy* by Josh Clark

Meeting expectations

- Reduced logic or simple apps?
  - Clarity
  - Data rich is for desktops
- People don’t want dumbed-down
  - Facebook was initially stripped down. Not happy.
- People might want simple
  - Gmail spun off Inbox

Requirements

- Customers drive mobile behavior
  - business analysis processes create use cases (requirements) for designers/developers
- Users need
  - to communicate something
  - to know something
  - to be entertained
  - to buy something
  - to remember something

Interface

- Users need a way to talk to the app
  - Best UIs are ones that mimic reality
  - Best current solutions
    - content first – when content is implicitly navigable
    - touch enabled – when gestures are understood
    - voice
- App needs to talk back
  - visual, audio, kinesthetic

Design

- Solving the requirements while using

"Constraints shape and **focus problems**, and provide clear challenges to overcome...

– Marissa Mayer

How Constraints Cultivate Growth

@shayhowe

Design vs. user experience

Design goals

- Content takes precedence
  - no site maps / navigation webs
  - put navigation options in other places
    - bottom of scrolled page is best one-handed
    - link at top can connect to bottom navigation
    - use contextual nav options for tasks or deeper nav
- Simple and focused

Developer design goals

- Reuse - architecture
  - Content
    - **Best solution**: component frameworks (Angular, Polymer, Web Components)
  - Code
    - **Best solution**: HTML5, JavaScript frameworks
- Ease of development – mature tools
- Maintainability – mature tools + architecture

Accessibility

- Web Content Accessibility Guidelines 2.0
  - W3C recommendation
- Perceivable
  - Provide text alternatives for any non-text content
  - Provide alternatives for time-based media.
  - Create content that can be presented in different ways without losing information or structure.
  - Make it easier for users to see and hear content.
- Sites
  - https://www.section508.gov/

Accessibility

- Operable
  - Make all functionality available from a keyboard.
  - Provide users enough time to read and use content.
  - Do not design content in a way that is known to cause seizures.
  - Provide ways to help users navigate, find content, and determine where they are.

Accessibility

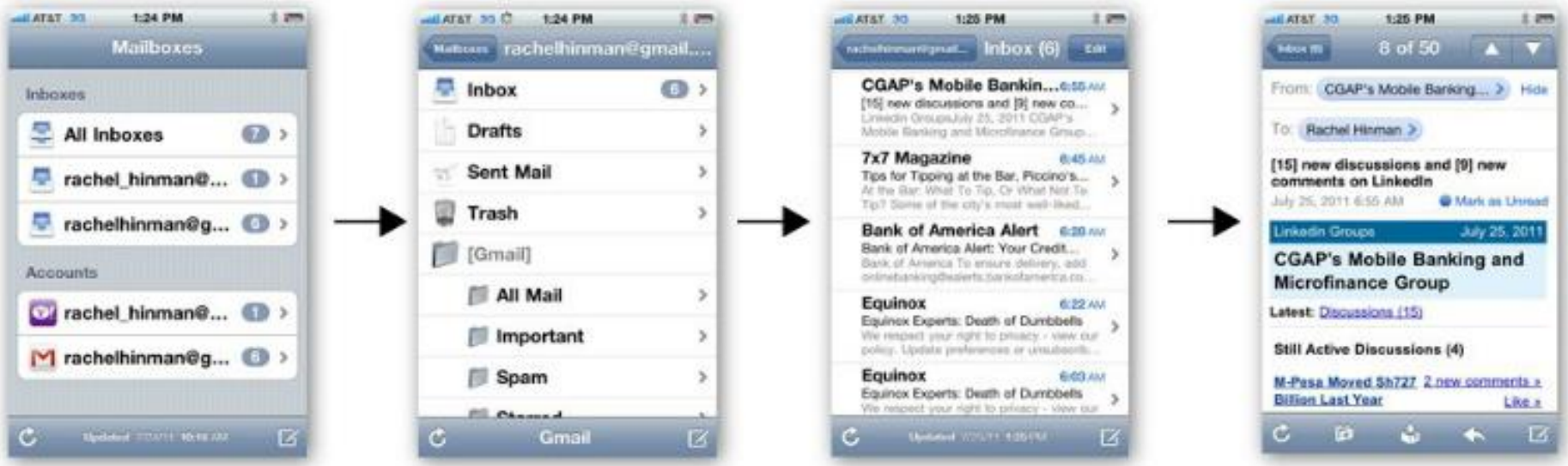
- Understandable
  - Make text content readable and understandable.
  - Make web pages appear and operate in predictable ways.
  - Help users avoid and correct mistakes.
- Robust
  - Maximize compatibility with current and future user agents, including assistive technologies.

GUI vs UX

- GUI design
  - component by component on a page
- UX design
  - workflow for a task
  - more significant when the screen is smaller

UX patterns

- Nested doll – big, medium, small, detail

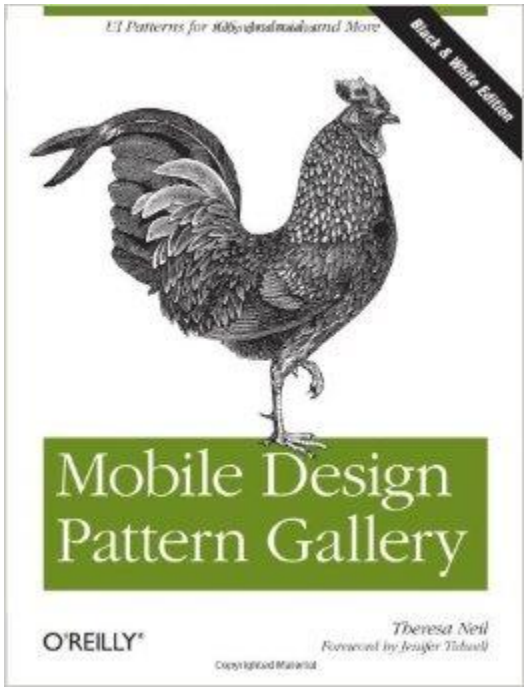


UX patterns - web sites

- http://ui-patterns.com/ - gallery
- http://patternry.com/ - library building
- http://patterntap.com/ - gallery
- http://www.smileycat.com/design_elements/ - gallery
- http://developer.yahoo.com/ypatterns/ - one example each
- http://patternbrowser.org
- http://www.ecommr.com/ - ecommerce patterns

UX patterns

- **Mobile Design Pattern Gallery,** 2nd ed., by Theresa Neil,
  O'Reilly Media, Inc. May 2014
  - Navigation, forms, tables, search, tools, charts, tutorials, social patterns, feedback, help, anti-patterns



Exercise

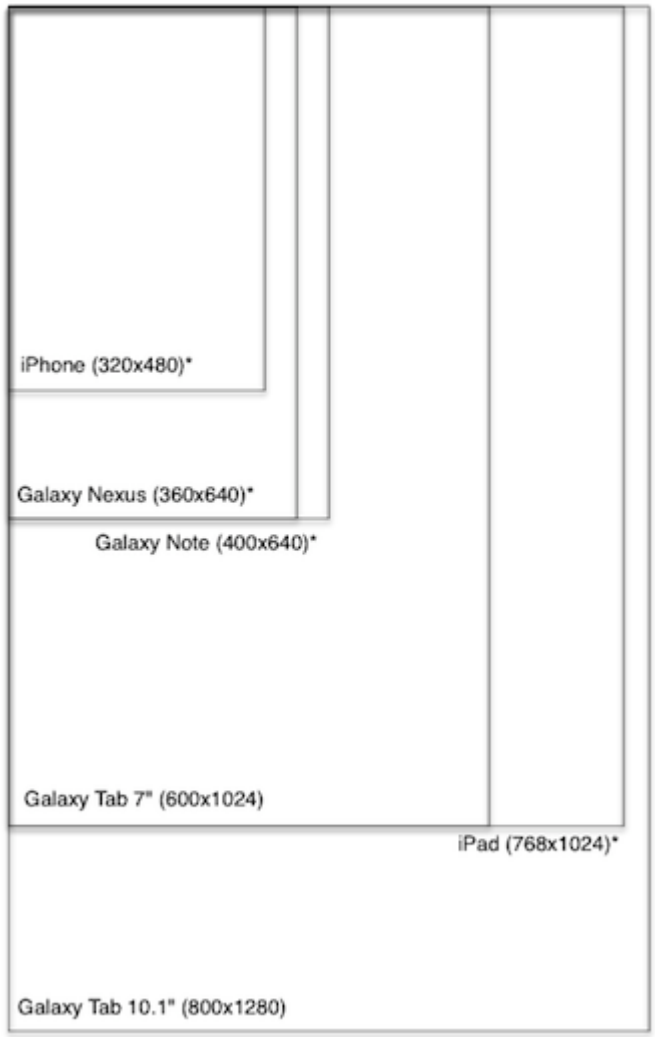
- #1 Explore Visual Studio Code

Basic constraints

- Combination of
  - Average display size
  - Available bandwidth
  - Primary input method
  - Common user posture
- Requires unique optimization

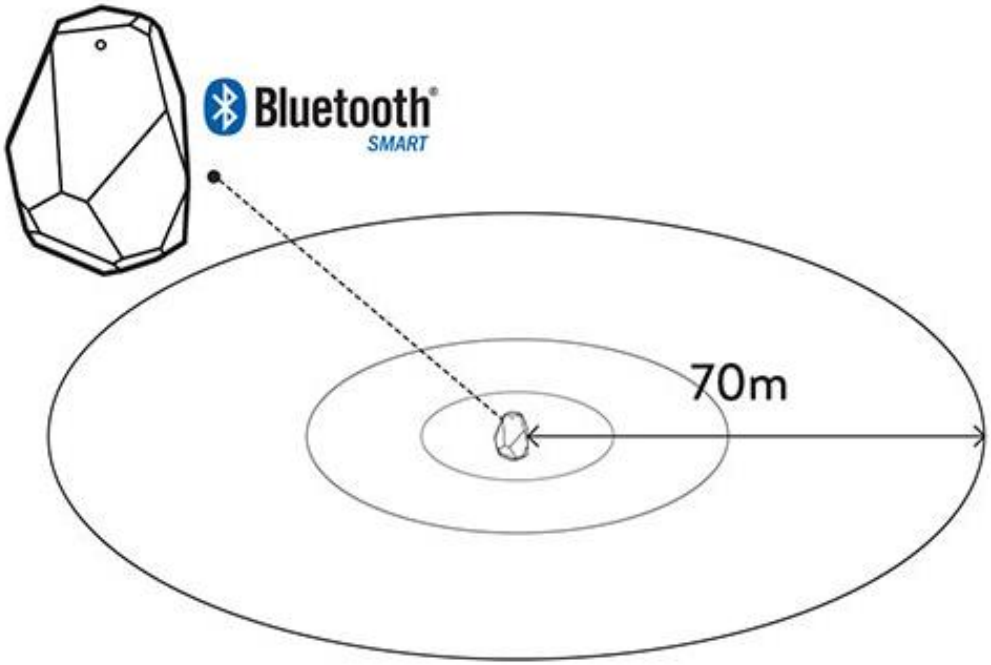
Screen size

- Desktop settled on 1024 x 768
- First smartphones were 320 x 480 (-80%)
  - 1 desktop screen = 5 phone screens
  - requires you to focus, not fluff



Communication protocols

- Web Sockets
  - direct I/O connections (IE10+)
- Bluetooth LE
  - replacing NFC
  - uses beacons
- Messaging API
  - Send text message
  - http://www.w3.org/messaging-api/



Solutions – device, network

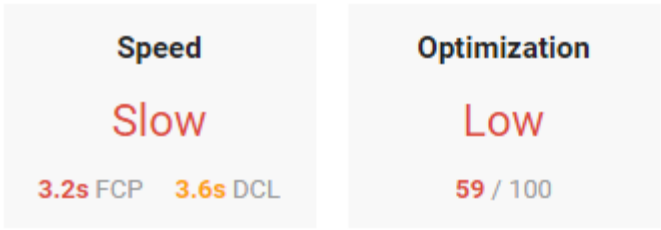
- Device viewport
  - dimensions of browser
    - find with media query or JavaScript
  - **Best solution**: fluid/responsive design
- Device connectivity
  - Mobile requires wireless which often means no internet service especially without cell service
  - **Best solution**: caching (web workers), offline storage

Solutions - network

- Bandwidth = page load time
  - speed of network connection
  - 2-3 seconds max before they go to another site. The expectations have been set by desktop.
  - **Best solutions**: caching, offline storage, AJAX, pre-loading
- Communication
  - Getting and sending **data**
  - **Best solution**: AJAX to RESTful web services, HTTP, Web workers, **WebSockets**

Exercise

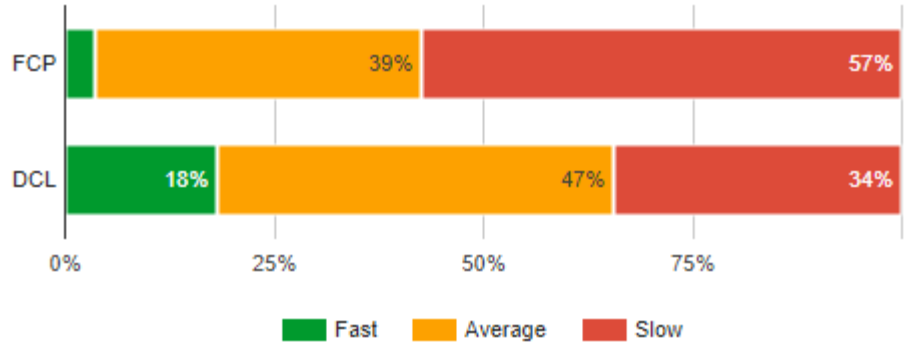
- Test and optimize
  - https://developers.google.com/speed/ - use Analyze with PageSpeed Insights



Data from the Chrome User Experience report indicates this page
This page has a low level of optimization because most of its res

Report for: https://www.centriq.com/

Page Load Distributions



Location detection

- Best supported device API
- Four types
  - IP – about 50% accurate by city
  - Cell towers
    - one – 500 - 2,500m
    - two – 100 - 1400m
  - WiFi beacons – 50m
  - GPS – 10m
    - takes time and more power
    - no laptops/desktops

Content process

- Reusable content starts with structured content
  - content authors write/create
  - content managers structure / markup with metadata
  - content publishers/designers create flexible layouts for metadata
  - content editors prioritize content for layouts and act as release managers / art directors

Text

- Structure your data with the client first
- Use metadata
  - Use microformats (schema.org)
- Web fonts
  - limit to desktop sizes due to files sizes and readability (Ethan Marcotte)

Performance

- Do anything to increase performance
  - send less stuff
    - use HTML5 application cache
    - minify files
    - reduce JS libraries
    - use CSS3 for effects, not images
    - limit CSS grid systems
    - don’t use @import, use link
  - use fastest solutions

Solutions – user context

- Context
  - environment of access, activity of user (bored, waiting)
  - no query
  - **Best solutions**: geolocation

Solutions - development

- Client language
  - Being able to perform client **actions**
  - **Best solution**: JavaScript, transpiler to JS, **WebAssembly**
- Graphics
  - Being able to show client dynamic **images**
  - **Best solution**: **CSS3**, SVG, <canvas>

W3C Working Groups

- Device APIs - http://www.w3.org/2009/dap/
  - battery status
  - contacts in addressbook
  - HTML media capture - camera, microphone
    - Media capture API
  - network info - ethernet, wifi, 2g, 3g, 4g
  - sensor API - temperature, air pressure, humidity, ambient light, ambient noise, magnetic fields, proximity
  - vibration API
  - web intents - service discovery

W3C Working Groups

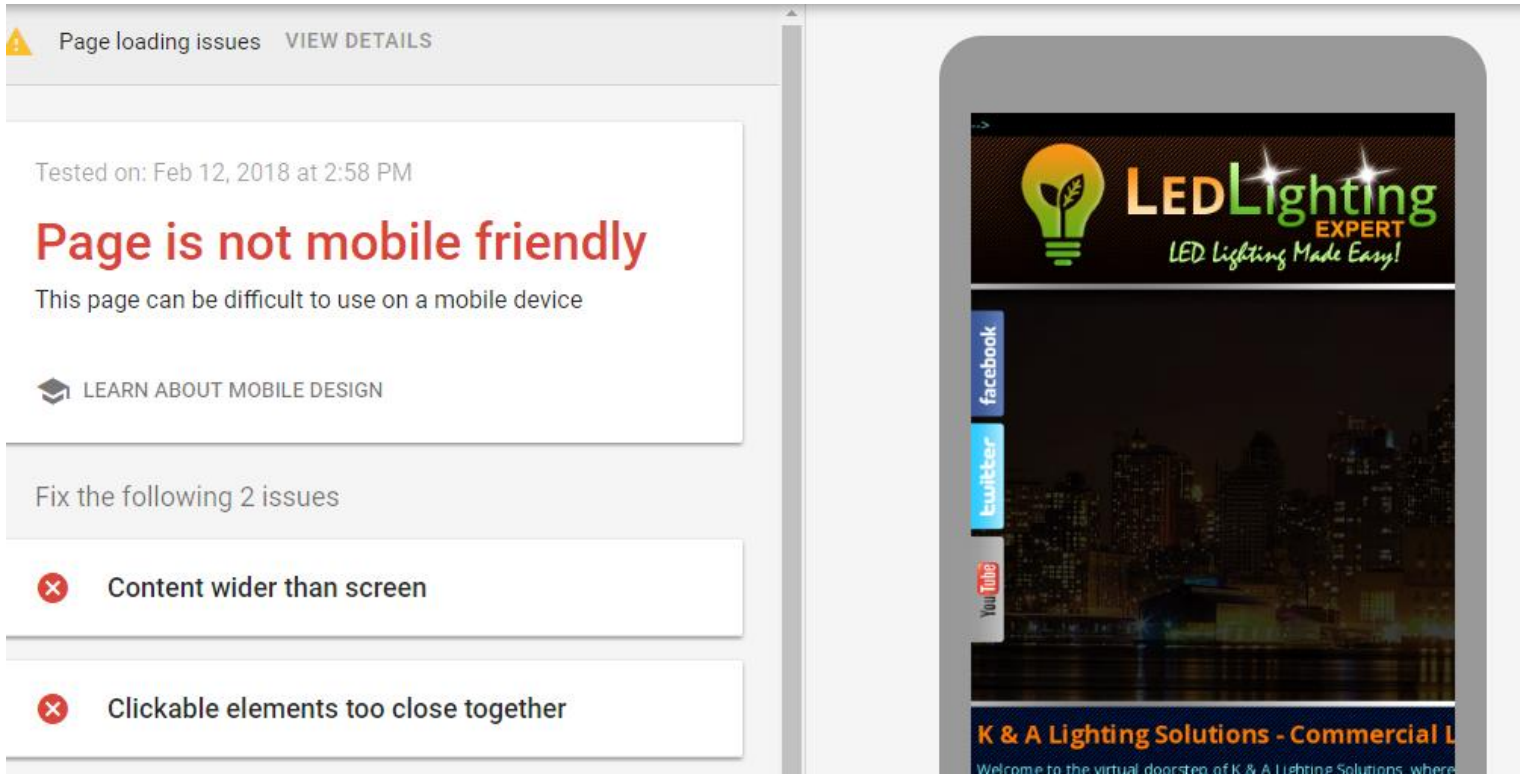
- File Writing - http://dev.w3.org/2009/dap/file-system/file-writer.html
- Filesystems - http://dev.w3.org/2009/dap/file-system/file-dir-sys.html
- Notifications - http://www.w3.org/2010/web-notifications/
- Orientation / acceleration - http://lists.w3.org/Archives/Public/public-device-apis/2009Nov/0026.html

Web sites - Google

- Google Developers Mobile-Friendly Websites
  - https://developers.google.com/webmasters/mobile-sites/
- The Mobile Playbook
  - The Busy Executive's Guide to Winning with Mobile
  - http://www.themobileplaybook.com/en-us/
  - 2012

Exercise

- #5 Google Mobile Friendly Test



Web sites - Google

- Think with Google – emerging technology
  - https://www.thinkwithgoogle.com/advertising-channels/emerging-technology/
- Android experiments
  - https://experiments.withgoogle.com/android

Exercises

- #2 Create a mobile friendly web page template
- #3 Use SCSS

Best practices and standards committees leading the efforts for mobile web apps.

Guidelines and standards

Mobile page templates

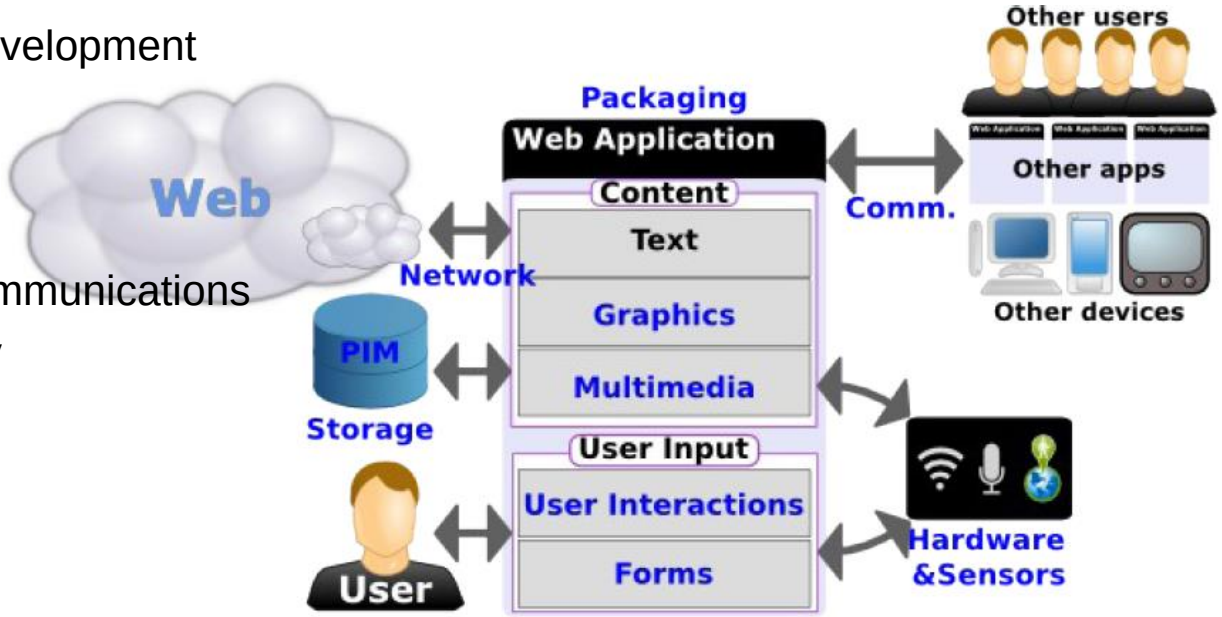
- *HTML5 Boilerplate Mobile
  - http://html5boilerplate.com/mobile
  - **Paul Irish**, Alex Gibson, …
- Initializr
  - http://www.initializr.com
  - quick templates for H5BP including responsive

Mobile Web Application best practices

- http://www.w3.org/TR/mwabp (Dec 2010)
  - The focus of 2008's MWBP was mobile Web *browsing*.
  - The focus of MWABP is development of mobile Web *applications*.
- Cards
  - http://www.w3.org/2010/09/MWABP/

W3C Current state and roadmap

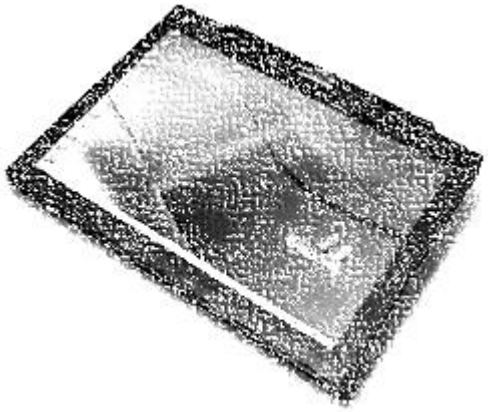
- https://www.w3.org/2015/08/mobile-web-app-state/
- Core Web Design and Development
  - Graphics and Layout
  - Device Adaptation
  - Forms
  - Data storage
- Media and Real-Time Communications
- Usability and Accessibility
- Device Interaction
- Network Integration
- Application Lifecyle
- Payment and Services
- Performance & Tuning
- Security & Privacy



Exercise

- #4 Set up remote host with FTP

server/host	doughoff.com
port	21
Encryption	Only use plain FTP (insecure)
username	mobilewebapps@doughoff.com
password	mobilewebapps395
Advanced - Remote directory (teacher will assign #	/student?
URL	http://squarealarm.com/mobilewebapps/student0?/



Prototyping

Purpose

- A prototype is used to
  - **brainstorm** for requirements from stakeholders
  - **elicit and validate** requirements from stakeholders
  - **understand** requirements domain
  - **communicate** a recommended design to designers, developers, and stakeholders
  - **visualize** solutions by designers

UX sketching

- sketching is efficient
  - no technology to distract
  - focuses you
  - lets you to take risks
  - communicates & helps understand problem

Prototyping tools

- sketching (analysis) vs. drawing (design)
- Fast, simple, limited
  - Photoshop (old school), Balsamiq, Pop (prototype on paper)
- In-between, rich, fairly quick
  - Quartz composer, Origami (iOS), Axure, Sketch, Adobe XD
- Fallbacks
  - Microsoft PowerPoint, Apple's Keynote

Wireframes

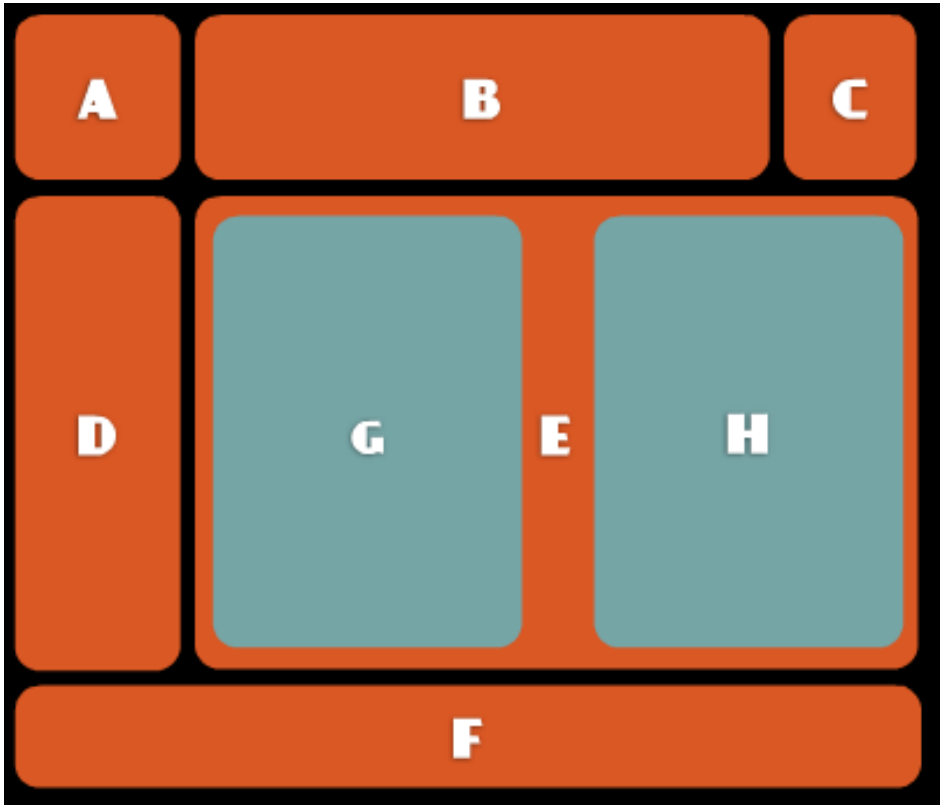
- allows developers to build logic to support it
- allows designers context to work on visuals

Wireframe for Wellstone Action homepage

Wireframes for various Wellstone Action internal pages

Wireframing - content reference

- A - logo
- B - company name
- C - login
- D - nav bar
- E - main content
  - G - posts
  - H - ads
- F - footer

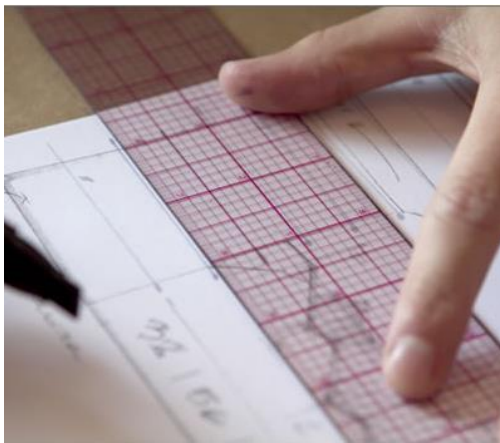


UX sketching process

- 1. Start with a 30% light grey marker
  - dot intersections, add lines
  - outline the basic divisions / sections
- 2. Use ball point pen
  - to add detail
- 3. Use 60% grey marker
  - to reinforce critical areas
- 4. Create refs to details

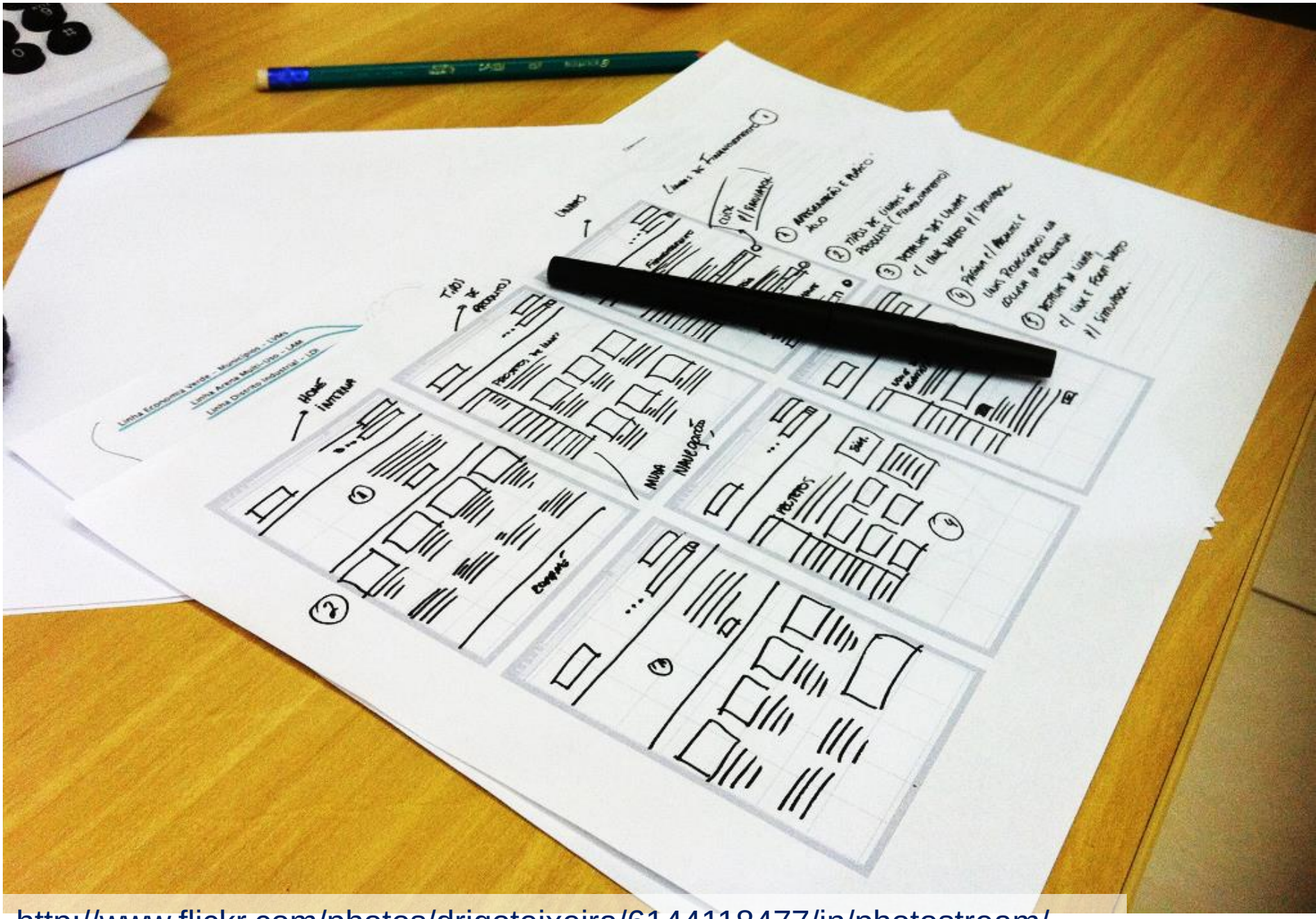
UX sketching techniques

- Use sticky notes for tooltips, dropdowns, …
  - easily added, removed for scanning/copying
  - cut, combine, use different colors for types
- Templates – photocopy basic layout
  - change layouts by taping over with copies
  - reuse sketch as underlay – copy at 20%
- Use a quilting ruler for even spacing, cutting paper, dark lines to make light areas pop.



Sketching tips

- Use context to make it real
  - sketch mobile at full scale
  - draw/picture surrounding environment on a few
- Use 8 ½ x 11" paper
  - hang on wall
  - batch scan, copier friendly, cheaper paper
- Tools
  - Cool Grey Prismacolor 12/set
  - Collins Quilt & Sew Ruler 2“x18"



http://www.flickr.com/photos/drigoteixeira/6144118477/in/photostream/
http://www.flickr.com/groups/ilovewireframes/pool/with/6144118477/

Sketch to mobile

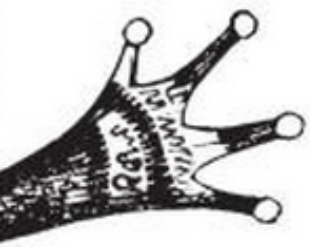
- For a more real experience
  - Hang sketches on wall and photograph
  - Adjust size so they fit your device
  - Keep sequence in order
  - Upload to mobile device and swipe through.

Designing in the browser

- After sketching, wireframing, etc.
- Client feedback on iterations
  - embellish screenshots in Photoshop
- Designers like
  - CSS that is almost complete
  - direct interaction with final medium
  - creating websites, not pictures of websites

Exercise

- #6 Reverse engineer a prototype



CSS patterns

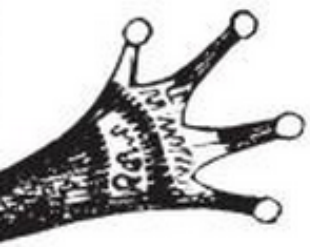
Centering content

- margin: 0 auto;
- width: ##

- or text-align: center

- or ?

Exercise

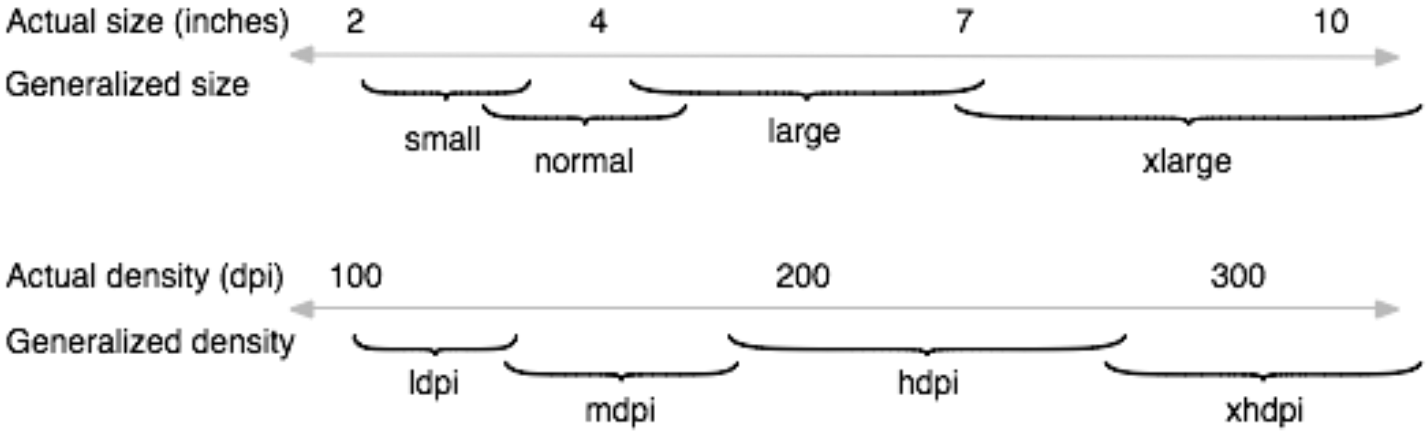


> It was actually the responsive side of things that were easier than I personally anticipated, and I would definitely never do anything else. It seems, in retrospect, completely ridiculous to do anything specific for each platform.

Responsive web design

The problem

- multiple screen sizes
  - measured diagonally
- multiple screen densities
  - **medium**, high (1.5x), extra high (2x - Retina), xxhdpi (3x)
  - ignore hardware resolution (ppi)



History

- The right thinking - Apr 2000
  - John Allsop wrote “A Dao of Web Design” article
  - http://alistapart.com/article/dao
- The right design - May 2010
  - Ethan Marcotte coined term in “Responsive Web Design” article
  - http://alistapart.com/article/responsive-web-design
  - Book

Graceful degradation

- Moving from existing desktop to mobile
- Problems
  - forces you to fit everything into a smaller space

Progressive enhancement

- The Filament Group
- The inverse of graceful degradation
- Build a solid foundation for any browser
- Add stuff on top
  - browsers should only do what they understand
  - use animations and transitions, don’t expect them to work

Responsive

- **Adaptive / fluid / flexible** means designs that adjust to the layout
- **Responsive** is more encompassing
- Multiple components
  - flexible grid
  - flexible images
  - flexible CSS - media queries
  - flexible widgets

Mobile First

- Luke Wroblewski coined term
  - Nov 2009
  - concentrate on the primary goals of the site
  - reverses graceful degradation
  - uses progressive enhancement

Responsive design

- flexible grids
- flexible content
  - font sizes / families
  - images & media
    - foreground
    - background
    - resolution (Retina)
    - progressive JPEGs again?
- media queries for breakpoints

Responsive design breakpoints

- Breakpoints
  - break points are at common device/design sizes
  - CSS based on min-width governs break points
- Traditional breakpoints
  - 320, 480, 640, 800, and 1024 pixels
- It’s not about the break, it’s about in-between behavior
- Ethan Marcotte - I’m a big, big believer of matching breakpoints to the design, not to individual devices.

Reducing content strategies

- Flow
  - content blocks change width and height
  - based on importance
- Collapsing
  - accordions, carousels, scrollers
  - based on activity
- Deleted

User override

- No logic is perfect.
- Give users a choice to pick which version of the site they want.
  - provide a link to another version.
  - Desktop | Mobile

Links

- Responsive Design Is… - https://responsivedesign.is/
  - News, patterns (templates), examples, advice

meta ... viewport

- **<meta name="viewport" content="width=device-width, initial-scale=1”>**
- Must use meta tag to get browser to set layout viewport
  - Opera supports CSS version
  - but as device resolutions increase, sites will start to break
  - expressed in DIPS (px or name)
  - this is the only workable way - now width media query works
  - only use width, not pixels
- <meta name="viewport" content="width=device-width, minimum-scale=1.0, maximum-scale=1.0">
  - Worked a little better on pages that zoom too much

RWD - Google

- https://developers.google.com/web/fundamentals/design-and-ui/responsive/
  - Use the meta viewport tag to control the width and scaling of the browser's viewport.
  - Include width=device-width to match the screen's width in device-independent pixels.
  - Include initial-scale=1 to establish a 1:1 relationship between CSS pixels and device-independent pixels.
  - Ensure your page is accessible by not disabling user scaling.

Exercise

- Test meta viewport four different ways
  - No viewport
  - Viewport
  - Viewport with no scaling
  - Viewport with 2x initial scale

Exercise

- #14 Inspect a responsive web design
- Check out other styleguides at
  - http://blog.hubspot.com/marketing/web-design-style-guide-examples (Apple, Atlassian, Mozilla, Buffer, Yelp, Gov.uk, Deviant Art, Disqus)

Media queries

Intro

- Lets you use conditional CSS based on media feature rules
- Media types
  - all, braille, embossed, handheld, **print**, projection, **screen**, speech, tty, **tv**
- Adaptable layouts for mobile devices, print, …
  - @media screen {
    - /* layout, typography, colors rule sets */ }
  - @media print {
    - /* layout, typography, colors rule sets */ }

Media queries

- Three ways
  - @**media** screen { /* styles */ }
- Or
  - @**import** url(“640px.css”) screen and (max-width=640px);
  - <**link** rel="stylesheet" href="640px.css" media="screen and (max-width=640px)">

Media queries – features

- **width**
- height
- device-width
  - “we don’t care” - Peter-Paul Koch
- device-height
- orientation
- aspect-ratio
- device-aspect-ratio
- color
- color-index
- monochrome
- resolution
- scan
- grid

Media queries – feature prefixes

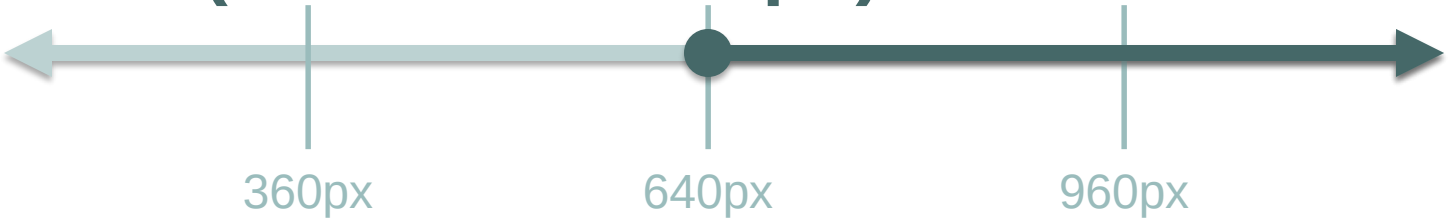
- Prefixes can modify features
  - min-, max-
    - (at least) min-width
    - (no wider than) max-width
- Use any measurement style
  - @media screen and (min-width: 20em)
  - @media screen and (min-width: 20rem)
  - @media screen and (min-width: 320em)

Media queries - feature operators

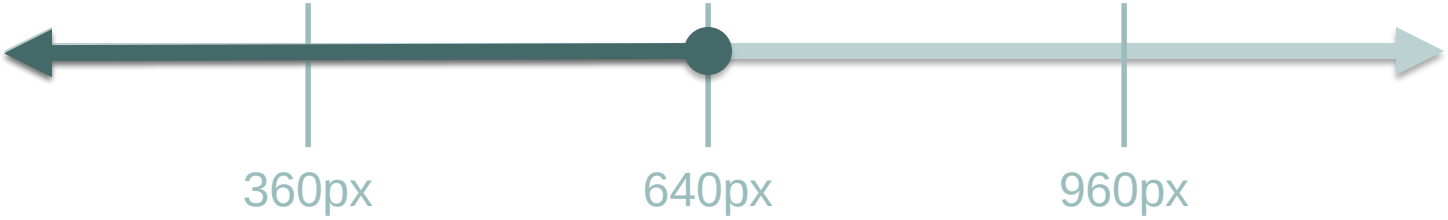
- and
  - @media screen and (min-width: 400px) and (max-width: 700px) { … }
- Minor
  - not
  - , - the union operator
    - @media handheld and (min-width: 20em), screen and (min-width: 20em) { … }
  - only
    - Used to hide style sheets from older browsers (<=IE8)

Setting breakpoints with width

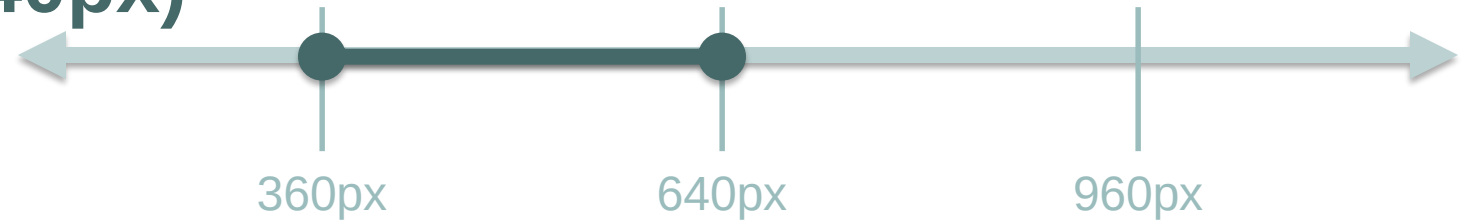
- Base + (min-width 640px) *mobile first



- Base + (max-width: 640px) degrading



- Base + (min-width 360px) and (max-width: 640px)



Exercise

- Follow instructions in Exercises handout for
  - #7 Inspect relative font sizes
  - #8 Detect viewport dimensions
  - #9 Detect device type by media query – basic
  - #10 Detecting device type by media query - advanced

JavaScript/server methods to query client software

Browser detection

Agent vs. feature detection

- Agent detection – **client/server** side logic content and code selection by user-agent in request
- Feature/capability detection - **client** side logic content selection by what it knows it can do

Request Headers

```
GET /page/detecting-sniffing-features-device HTTP/1.1
Host: www.scientiamobile.com
Connection: keep-alive
Cache-Control: max-age=0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 6.3; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/46.0.2490.86 Safari/537.36
Referer: https://www.google.com/
Accept-Encoding: gzip, deflate, sdch
Accept-Language: en-US,en;q=0.8
Cookie: _ga=GA1.2.956937014.1447856347; _gat=1
```

Agent detection

- Capability detection
  - **primary features** - touch?, screen is large or small?
- Two workflow choices - both slow
  - Redirect?
    - window.location.href = '/tablet‘
    - change history?
  - Load resources?
    - complex - needs dynamically loaded CSS / JS

Agent detection issues - client

- throughput - how fast will the network handle it?
  - loading all the content at once
  - doing redirects
- memory of device
  - loading large images
- latency - how long does the request and response take?
  - making smaller AJAX requests
  - making redirect requests

Feature detection - future

- Use @media for conditional loading of CSS with the @supports for browser properties.
- https://drafts.csswg.org/css-conditional/#at-supports

IE	Edge *	Firefox	Chrome	Safari
			49	
			62	
		57	63	
11	16	58	64	11
	17	59	65	11.1
		60	66	TP
		61	67	

Exercise

- Follow instructions in Exercises handout for
  - #12 Inspect user-agent strings
  - #13 Detecting client by user-agent
  - Modernizr class usage – modernizr.html

Responsive CSS

Fluid units

- Use relative/percentage units
- Relative to widths of containers
  - %
  - Pixel heights are OK since we let people scroll if that's what you really want.
- Relative to size of fonts
  - em, rem

Fluid units

- Better than proportions
  - No measurement at all makes blocks inherently fluid
  - All block elements you build
    - should never have widths (columns exception)
    - should never EVER have heights
    - should be constrained only by their parents
- Any time you can avoid setting a measurement, you should.

Relative units of measurement

- **em** - based on % of **parent** element size or default style from browser (16px)
- **%** - based on % of parent element
- **rem** - based on % of **root** element size
  - IE9+

Relative units of measurement

- **62.5**% is Mr. Rutter's magic font-size
- ems
  - body { font-size : **62.5**%; } /* 10px */
  - ul { font-size : 1.6em; } /* 16px */
  - ul p { font-size : 1.6em; } /* 16 * 1.6 px */
- rems (IE9+, Opera 11.6+)
  - body { font-size : **62.5**%; } /* 10px */
  - ul { font-size : 1.6rem; } /* 16px */
  - ul p { font-size : 1.6rem; } /* 16px */

Relative units of measurement

- Viewport units: vw, vh, vmin, vmax
  - Length units representing 1% of the viewport size for viewport width (vw), height (vh), the smaller of the two (vmin), or the larger of the two (vmax).
  - use with calc(x + y) to get subtle font, line height adjustments
- em : rem :: % : v*
- http://caniuse.com/#search=vw

Font spacing

- Also adjust line-height
  - 1.2 – ideal in typography
  - 1.3 - default
  - 1.5 – 2 – airy for better tracking lines / disabilities
  - Use more air for wider blocks
- Paragraph padding/margin
  - At least 1.5 times line-height
- Font size affects line-height
  - https://pearsonified.com/typography/

Fluid units

- Best practice
  - Set structure in percentages (divs, aside, nav, …)
  - Set absolute locations in pixels (images, icons, sprites)
  - Set line-height in no units! (relative)
    - http://meyerweb.com/eric/thoughts/2006/02/08/unitless-line-heights/
  - Recommended but not practical…
    - Set any type related CSS (body, h1, list indents, margins…) in rems (or ems) with a pixel fallback (rule: 10px; samerule: 1rem)

Enhancing with media queries

- Start with no media queries
- Embrace the cascade
  - Don’t just swap style sheets, let it cascade
  - Add styles for next size up
  - More nimble, versatile. Reduces jumpiness during device re-orientation

Hide inline elements

- Pro - easy
- Con - loads element anyway

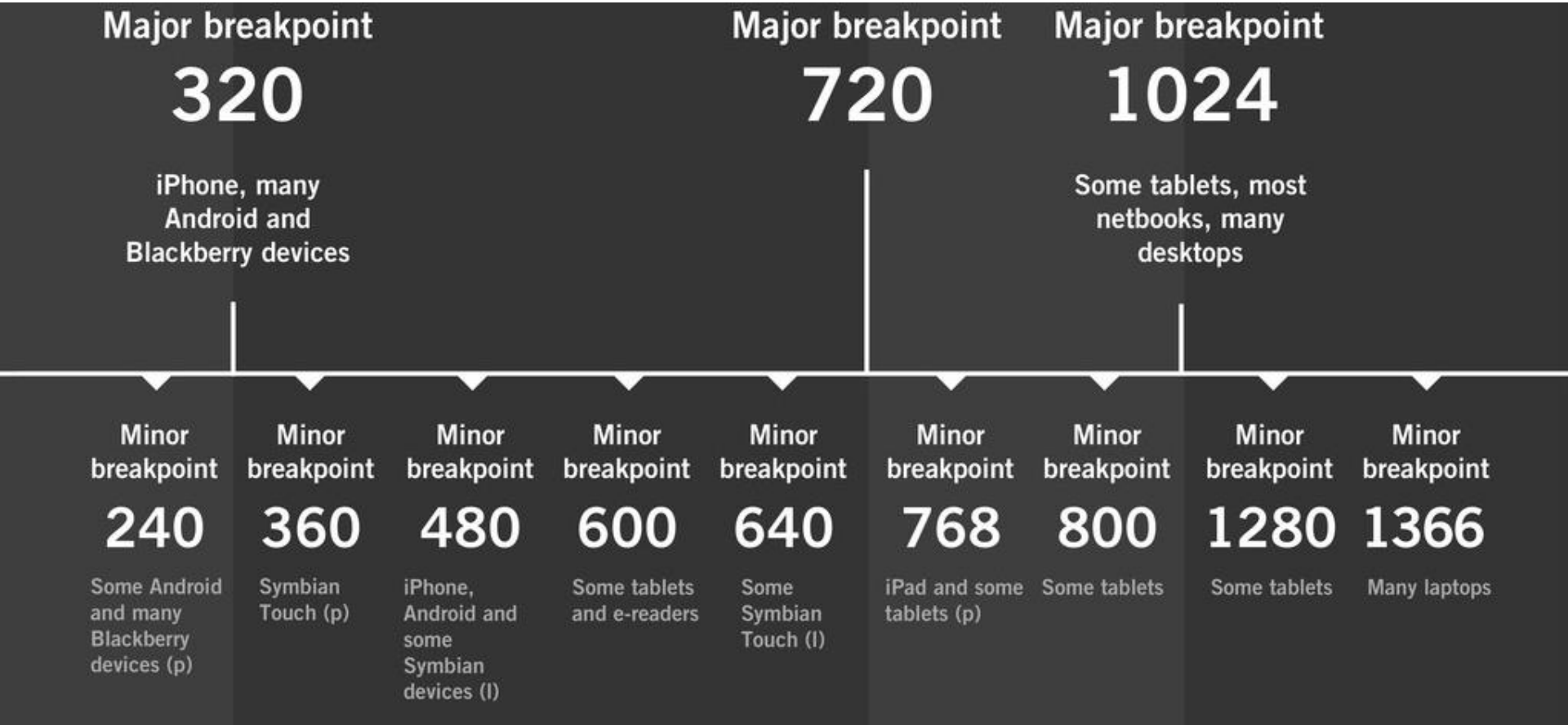
- Useful for printing
  - @media print {
    h1 img { display: none; }
    h1 img:after { content: attr(alt); }
    }
- You may need display: none !important;

Breakpoint strategy

- Create breakpoints when the layout breaks.
  - Shrink the width of the browser window until the design breaks, then fix it either by changing things or by creating a breakpoint and changing the layout at that point.

Breakpoints

- Andy Clarke's version of S. Rieger's system

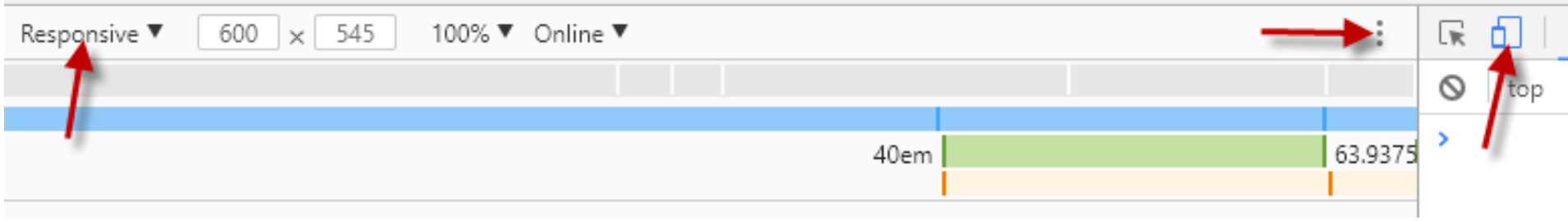


Breakpoint testing

- http://quirktools.com/screenfly/ - fixed widths
  - Google device inspector
- http://mattkersley.com/responsive/ - 5 versions at once
- https://gist.github.com/1685127 - code your own viewer

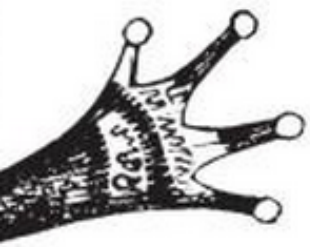
Dev Tools media query inspection

- Open device toolbar
- Select show media queries from stoplight menu
- Leave Responsive as style
- Choose device (grey) or breakpoint bars
- Right-click on breakpoint bar to see code



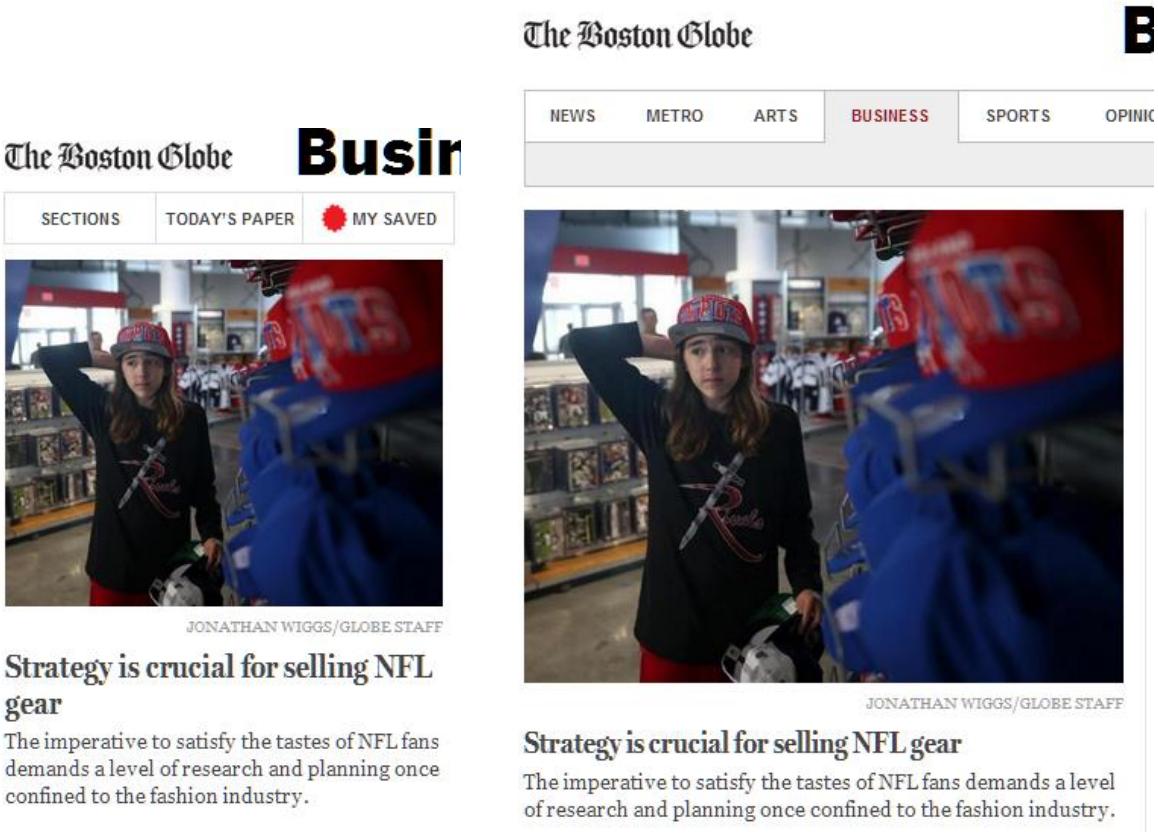
Exercise

- #11 Responsive web design with media queries demo



Exercise

- Use the media-queries worksheet to
  - change font sizes, font families, line-heights
  - change text layout to use multi-columns in 2, 3, and 4 columns
  - adjust image sizes, margins, float…
  - remove/add color
- Leave at least three different sets of CSS in media queries



Responsive images

Optimize images before use

- Reduce file size
  - lower jpg quality, selective quality
  - flatten color areas in gifs
  - export for right display size
- Utilities
  - ImageOptim - http://imageoptim.com/ (Mac)
  - Trimage - http://trimage.org/ (PC)
  - TinyPNG - http://tinypng.org/ (service)

The resource downloading problem

- Bad: Setting an **image** to **display:none** won't stop the image from downloading
- Bad: Setting an **element** to **display:none** won't stop a background from downloading
- OK: Setting a **parent element** to **display:none** DOES stop a background from downloading
  - in most browsers
- **BEST:** Using a media query to (cascade) override or select between options DOES work
- http://timkadlec.com/2012/02/media-query-asset-downloading-tests/

Fluid images

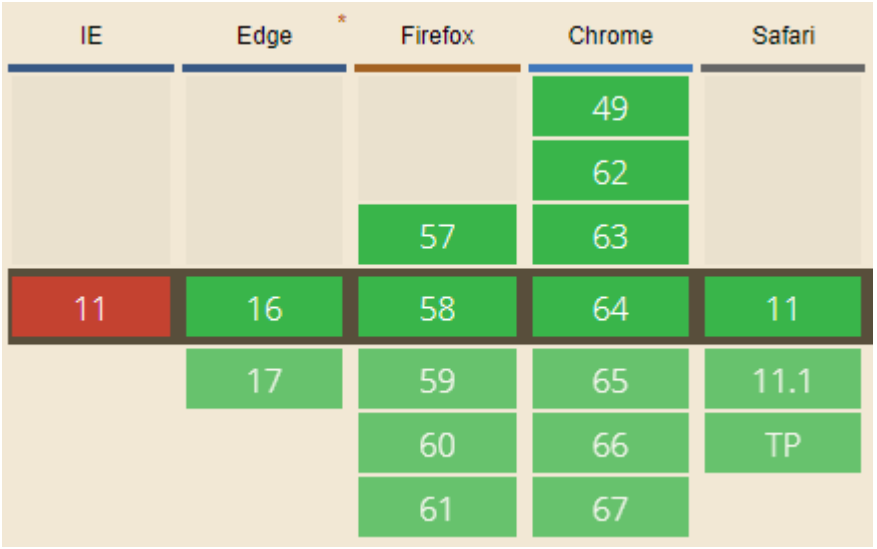
- max-width=100%
  - delete height and width img attributes
    - will override width, but not height
  - prevent this image from ever overflowing its parent
- send biggest possible image?
  - smallest device wastes its energy
  - smallest device usually has less bandwidth
  - best viewed on a Retina display?

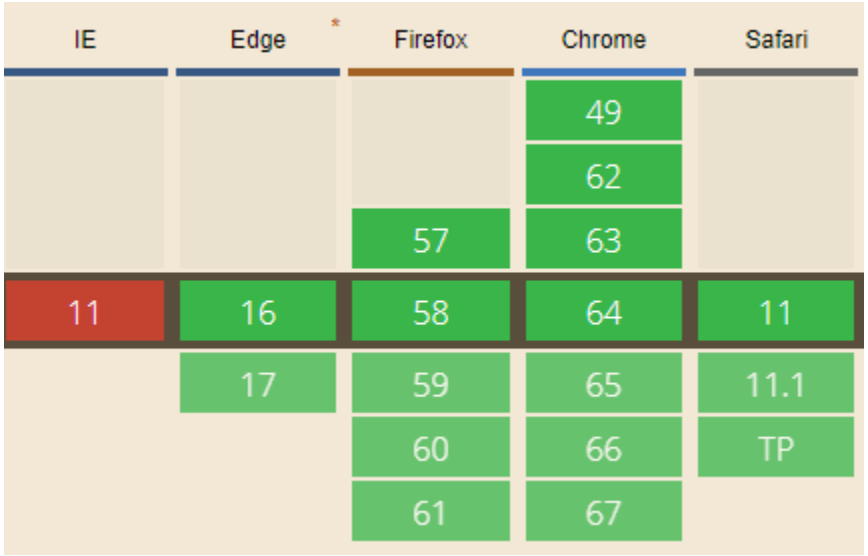
Fluid images

- Easy:
  - `img, object { max-width: 100% }`
- to fit within a parent container
  - problem when container is larger than image
  - problem when server has large number of requests

srcset vs. picture

- Standards track
  - <img **srcset**="a.jpg 2x, b.jpg 100w, c.jpg 100w 2x">
  - uses devicePixelRatio
- https://responsiveimages.org/
- **<picture>**
- uses srcset





Resources - fluid images

- Scott Jehl's **picturefill** polyfill
  - https://github.com/scottjehl/picturefill
- **Backstretch** jQuery plugin -
  - http://srobbin.com/jquery-plugins/jquery-backstretch/
  - to fit the background image to the viewport

2x images

- Always use the same PPI resolution, any will do!
  - 72 ppi is instilled into web culture even though those monitors were last seen in the 1980's
- Save at 2x pixel proportions
  - a normal image - 200 x 200 px
  - the 2x image - 400 x 400 px
- You'll be using vector and Smart Objects (Photoshop vector) images more.
  - Use Photoshop's Image Size / Nearest neighbor option to retain hard edges

2x image swapping - CSS

- backgrounds
  - @media only screen and (-webkit-min-device-pixel-ratio: 2), only screen and (min-device-pixel-ratio: 2) {
    - header h1 a {
      background-image: url(images/swappable@2x.png);
      background-size: 164px 148px; }
  - }
- inline (deal with it)
  -

Resizing to layout 1

- Fluid image
  - No cropping
  - Adjusts with width set to relative size (%, ems, rems)
  - Can be used with min- or max-
- responsive/fluid-image.html

Resizing to layout 2

- background-image
  - use a div instead of an img
  - use `background-size:cover` to size proportionately
  - flexible cropping from one side with background-position
  - IE8 uses polyfills
    - http://louisremi.github.io/background-size-polyfill/
- responsive/cropping.html

Resizing to layout 3

- img with max-width %
  - wrapped in a hidden overflow div with height
  - proportional vertical scaling below a certain height
  - crop from top or bottom
    - uses transform:rotate()
- responsive/cropping-center.html

Resizing to layout 4

- Hybrid of 1 & 2
  - img is invisible to see the background
  - supports a hi-res image
  - responsive/cropping-hybrid.html

Icon fonts

- Scalable, colorable, animatable
- Demo
  - http://css-tricks.com/examples/IconFont/
  - size, color, shadow, transparent knockouts with CS
- Create your own font
  - https://icomoon.io

CSS for icons

- No, not sprites. Pure CSS.
- http://www.noupe.com/css/cikonss-1-0-responsive-icons-in-pure-css-ie-doesnt-fail-77325.html

SVG for icons

- inlined SVG (IE9+)
- https://useiconic.com/open

Font Awesome

- SVG font
- http://fontawesome.io/
- Copy link from http://cdnjs.com
- Use
  -

Responsive video

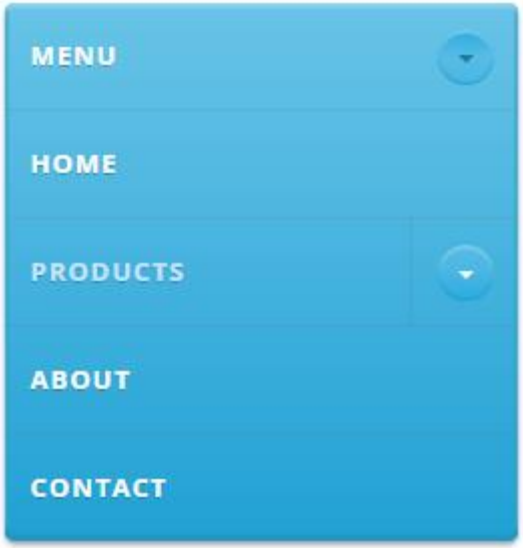
- HTML5 without an <iframe> or <object>
  - video { max-width: 100%; height: auto; }
- FitVids.js - jQuery plug-in
  - http://fitvidsjs.com/

Cloud APIs

- Image/video manipulation & storage
  - https://cloudinary.com/
  - free for 300,000 images/videos

Exercises

- svg-icons.html
  - add your own icon from fonticonic and style
- #17 **srcset & picture**



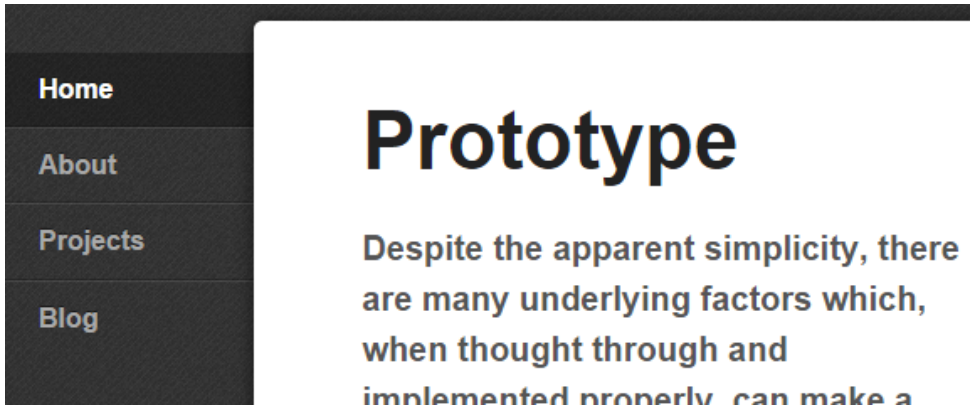
Responsive widgets

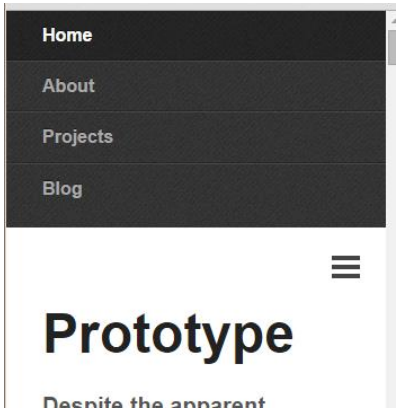
Responsive nav bar

- Top - common tasks, bottom - exploring
- Use only essential links and don’t duplicate
  - hide secondary navigation
  - use lots of space
  - content first
- Transform links into:
  - menu button
  - grid of icons
  - accordion

Responsive nav bar – plug-in

- Responsive Nav
  - http://responsive-nav.com/





Convert menu to dropdown 1

- Hide with media query
- `<nav>`
  - `<ul> <li><a href="/" class="active">Home</a></li> <li><a href="/collections/all">Books</a></li> </ul>`
  - `<select><option value="" selected="selected‟ > Select </option> <option value="/"> Home</option> <option value="/collections/all">Books</option> </select>`
- `</nav>`
- nav select { display: none; }
  - @media (max-width: 460px) {
    nav ul { display: none; }
    nav select { display: inline-block; }
  - }

Convert menu to dropdown 2

- Use #1<nav> only or create on the fly and append <select> version dynamically
- `$("<select />").appendTo("nav");`
- `$("<option />", { "selected": "selected", "value" : "", "text" : "Go to..." }).appendTo("nav select");`
- `$("nav a").each(function() { var el = $(this); $("<option />", { "value" : el.attr("href"), "text" : el.text() }).appendTo("nav select"); });`
- `$("nav select").change(function() { window.location = $(this).find("option:selected").val(); });`

Convert menu to dropdown 3

- jQuery plug-ins
  - Responsive Menu Plugin
    - https://github.com/mattkersley/Responsive-Menu
  - Mean Menu
    - http://www.meanthemes.com/plugins/meanmenu/
  - FlexNav
    - http://jasonweaver.name/lab/flexiblenavigation/

Convert menu to static top table

- http://inspectelement.com/tutorials/pull-down-for-navigation-a-responsive-solution/

Home | Portfolio | Blog | About | Contact | Twitter | Dribbble | A Longer Navigation Item

Pull Down for Navigation Demo

See the tutorial here

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Vestibulum tortor quam, feugiat vitae, ultricies eget, tempor sit amet, ante. Donec eu libero sit amet quam egestas semper. Aenean ultricies mi vitae est. Mauris placerat eleifend leo.

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Vestibulum tortor

Pull Down for Navigation Demo

See the tutorial here

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Vestibulum tortor quam, feugiat vitae, ultricies eget, tempor sit amet, ante. Donec eu libero sit amet quam egestas semper. Aenean ultricies mi vitae est. Mauris placerat eleifend leo.

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Vestibulum tortor quam, feugiat vitae, ultricies eget, tempor sit amet, ante. Donec eu libero sit amet quam

Responsive layout - UX

- Navigation patterns - Brad Frost
  - http://bradfrostweb.com/blog/web/responsive-nav-patterns/
  - Float label pattern
    - http://bradfrostweb.com/blog/post/float-label-pattern/

Tables - strategies

- hide non-essential columns
- convert a row to a two-col table
  - http://css-tricks.com/responsive-data-table-roundup/
- show a chart instead of tabular data
- invert the axes
- let the columns scroll under column 1
  - http://foundation.zurb.com/responsive-tables.html

Tables - Plug-ins

- Stackable - http://johnpolacek.github.io/stacktable.js/
- FooTable - http://css-tricks.com/footable-a-jquery-plugin-for-responsive-data-tables/

Stuff	
Something	
Rate	3.375%
Amount	$123.12
Points	1.125
Number	4,000
Type	Potato
Name	Paul
Something Else	
Rate	2.750%
Amount	$345.23
Points	5
Number	180
Type	Spaceship
Name	Skippy

Stuff	Rate	Amount	Points	Number	Type	Name
Something	3.375%	$123.12	1.125	4,000	Potato	Paul
Something Else	2.750%	$345.23	5	180	Spaceship	Skippy

Exercises

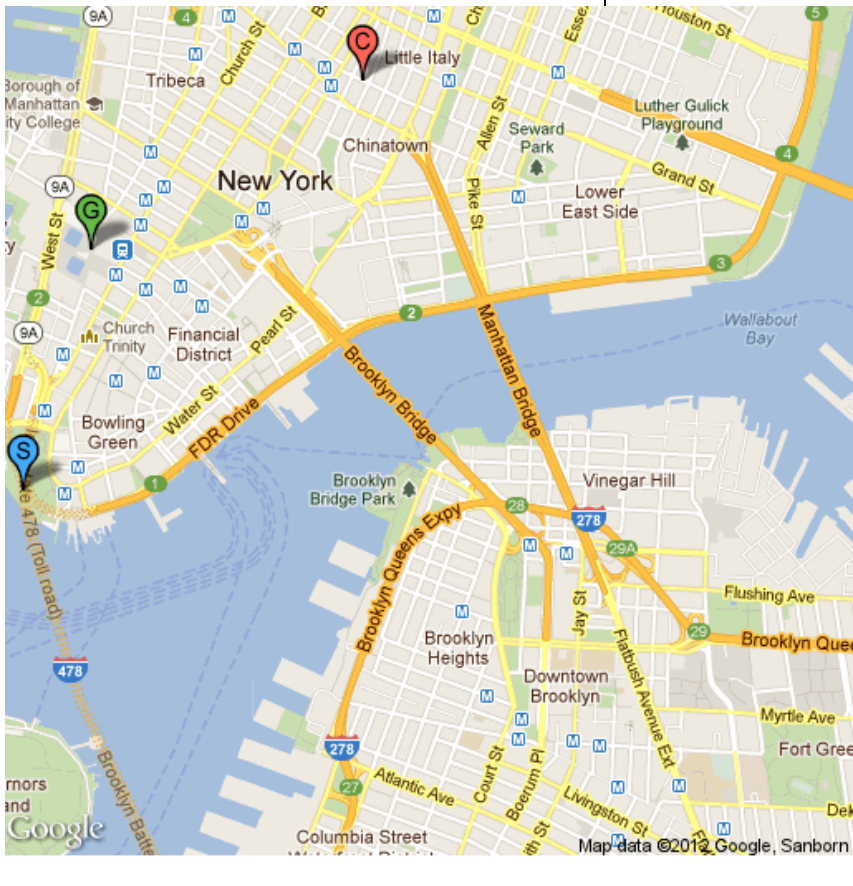
- Navigation – navigation.html
- Stacktable – stacktable.html


Ankeny
Harlan
Blair
Urbandale
Des Moine
Schuyler
Fremont
Columbus
Omaha
Atlantic
Norwalk
Indianola
Winterset
Wahoo
Bellevue
Red Oak
Creston
Chariton
Seward
Lincoln
Shenandoah
Leon
Clarinda
Crete
Auburn
Maryville
Beatrice
Fairbury
Trenton
Falls City
Hiawatha
Marysville
St Joseph
Cameron
Holton
Atchison
Kearney
Excelsior
Springs
Clay Center
Leavenworth
Wamego
Gladstone
Liberty
Manhattan
Topeka
Kansas City
Marshall
Junction
City
Lee's
Summit
Overland
Park
Belton
Sedalia
Warrensburg
Harrisonville
Ottawa
Emporia

Maps

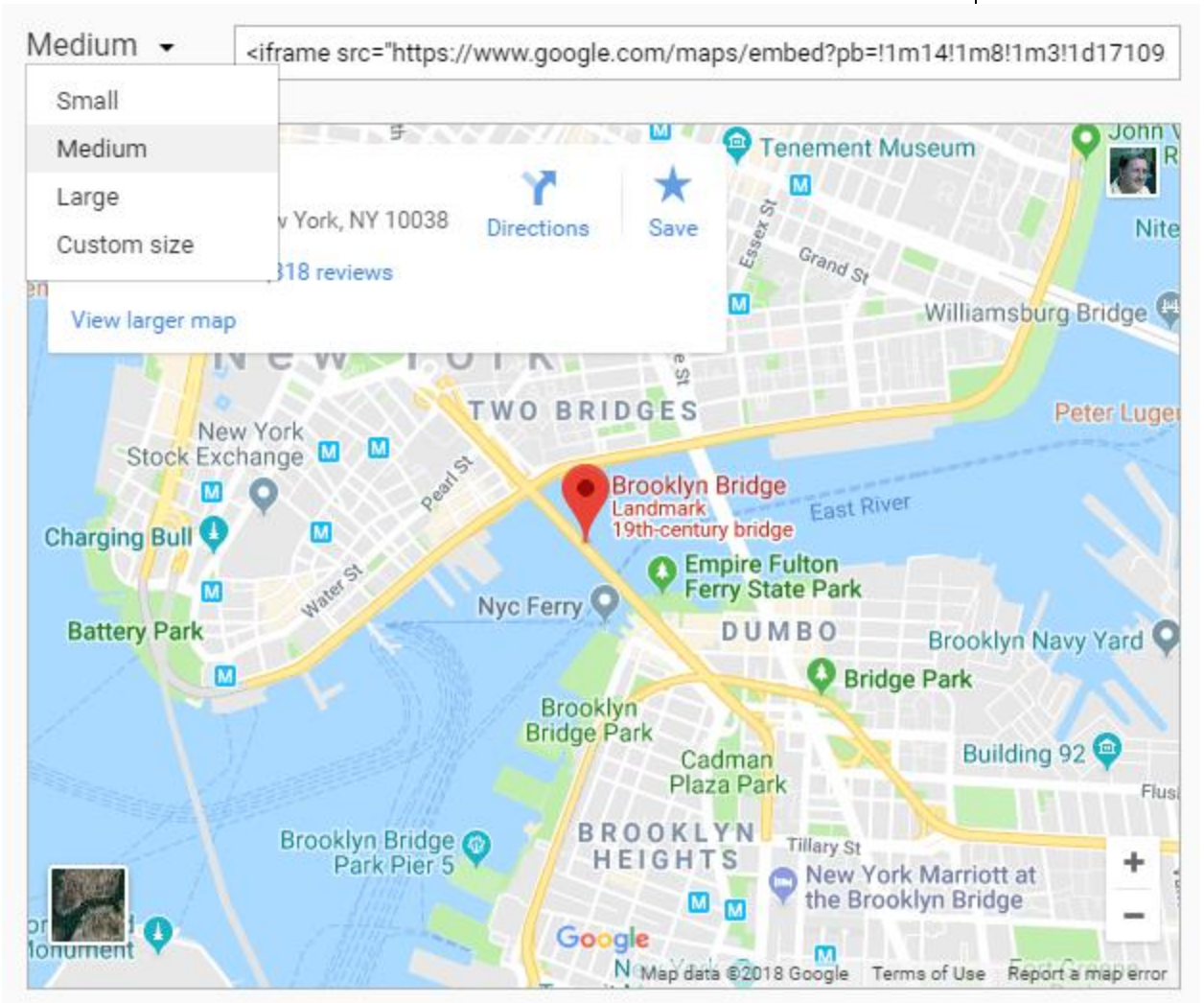
Google Maps – static maps

- <img border="0" src="http://maps.googleapis.com/maps/api/staticmap?**center**=Brooklyn+Bridge,New+York,NY&**zoom**=14&**size**=512x512&**maptype**=roadmap&**markers**=color:blue%7Clabel:S%7C40.702147,-74.015794&markers=color:green%7Clabel:G%7C40.711614,-74.012318&markers=color:red%7Clabel:C%7C40.718217,-73.998284&**sensor**=false" alt="Points of Interest in Lower Manhattan">
- https://developers.google.com/maps/documentation/static-maps/intro



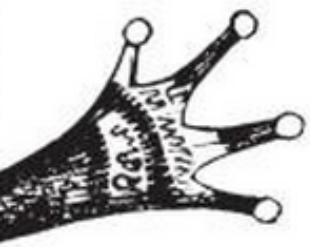
Google Maps – dynamic maps

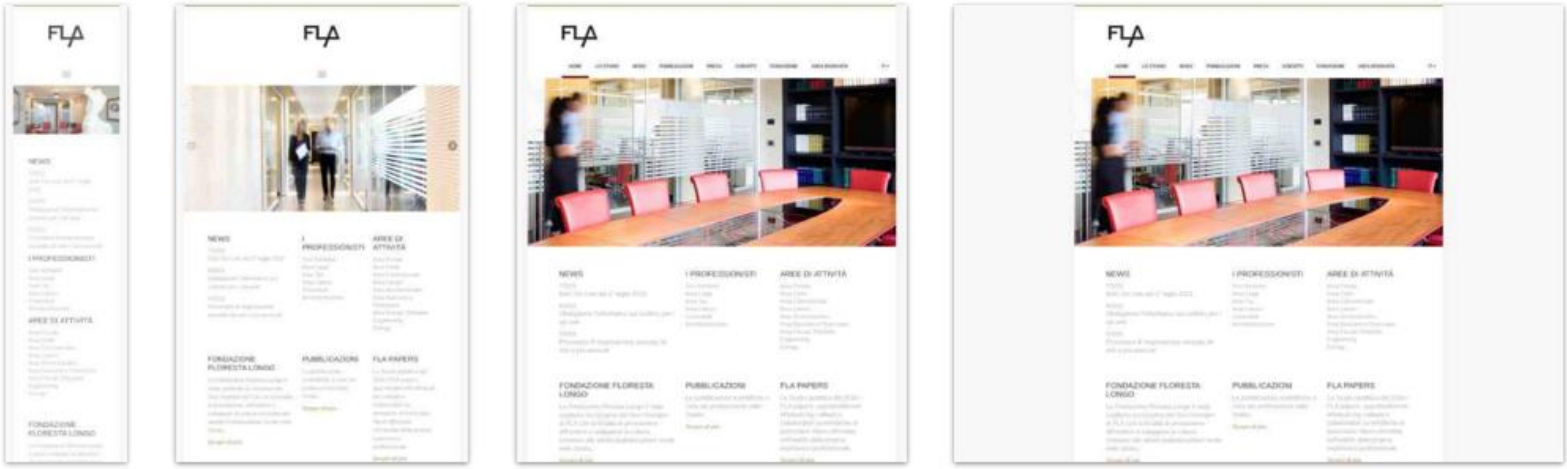
- Go to https://www.google.com/maps and find location
- Share / Embed map as iframe on your web page.
- Zoomable, draggable, personalized places, etc.



Exercise

- #18 Google static maps





Responsive layout

Intro

- Focus first on content in the site structure, not layout.
  - Site structure is hard to create & change.
  - Layout is fast.
- Continual change requires layout flexibility

Flexible / fluid grids

- Convert fixed positioned units to fluid units
  - Target / context = Result
- Manage proportions, not fixed units.
  - Percentages, not pixels.
- Width, hierarchy, interaction and density all influence layout.

UX patterns - responsive

- The Select Menu

UX patterns - responsive

- The Left Nav Flyout

Transforming a layout

- Work from the top down
- The more narrow the screen, the more the layout became vertical, and use more centering to make things work.
- Hide things (via display: none) that were nice at bigger sizes but that weren’t needed at smaller sizes.

Fluid grids

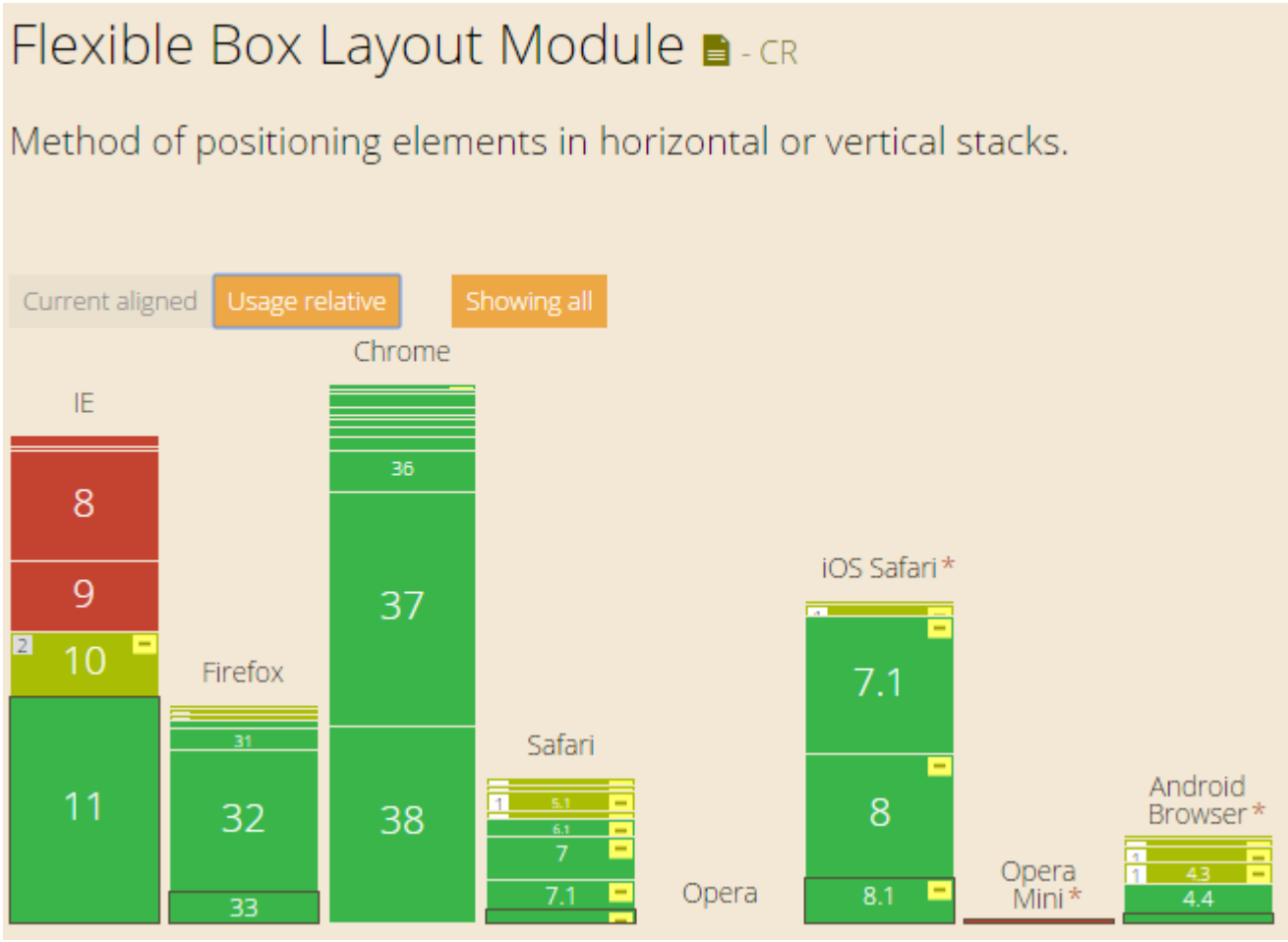
- Ethan Marcotte - Mar 2009
  - http://www.alistapart.com/articles/fluidgrids/
- Based around full grid systems.
  - Use a calculator to work out the numbers
  - http://csswizardry.com/2011/06/fluid-grid-calculator/

Fluid grids

- parent container (context)
  - max-width, padding, etc. based on text size in ems
- child container (target)
  - width, margin, etc. based on percentages
  - select non-fluid layout in px
  - **target** px **/ context** px **= result** in percentage
- IE always rounds up on fractional pixels
  - you may need to decrease by one pixel before calculating

Flexbox

- fluid by design
- semantic not tr, td, th…
- recently updated
- Waiting for IE9 to go



Flexbox

- RWD layout - http://codepen.io/team/css-tricks/pen/jqzNZq
- https://css-tricks.com/snippets/css/a-guide-to-flexbox/
- http://www.flexboxpatterns.com/home
- http://maxsteenbergen.com/fibonacci/ - interactive tool
- http://flexbox.io/#/ - 20 videos

Fluid CSS tips - Brad Frost

- display: inline-block
- box-sizing: border-box
  - fluid width items with fixed amounts of padding
    - .photogallery li { float: left; box-sizing: border-box; width: 50%; padding: 1em; }
  - Also for full-width form fields:
    - input[type=search] { box-sizing: border-box; width: 100%; padding: 1em; }
  - Handy for centering inline li's:
    - .nav { text-align: center; } .nav li { display: inline-block; margin: 0 0.5em; }

Responsive HTML email layouts

- Cerberus
  - http://tedgoas.github.io/Cerberus/
- Zurb
  - http://foundation.zurb.com/emails.html
- Patterns
  - http://responsiveemailpatterns.com/

Email layouts

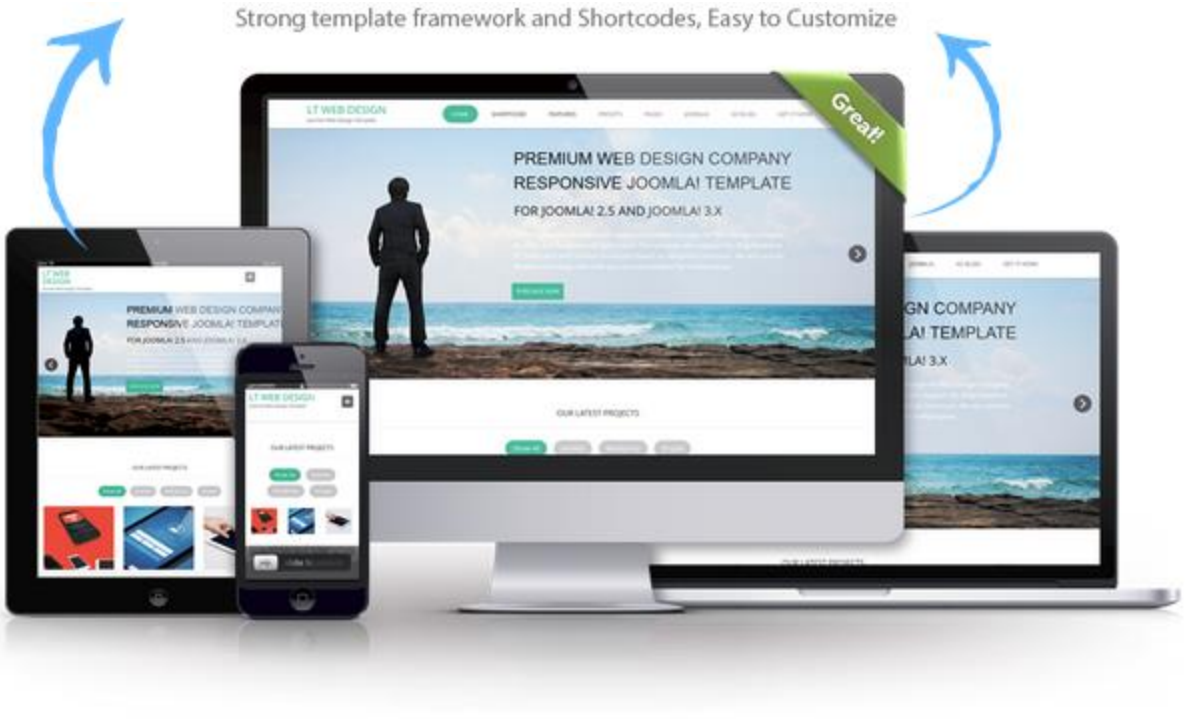
- use a Gmail First strategy
  - http://julie.io/writing/gmail-first-strategy-for-responsive-emails/
- keep your max width to 600px

Fullscreen F11 on mobile

- To imitate the native app look
- Techniques
  - Fake it: auto-hide the address bar
  - Request the browser to go fullscreen in response to a user gesture.
  - Install the app to the home screen
- http://www.html5rocks.com/en/mobile/fullscreen/

Exercises

- #15 Fluid layouts
- #16 CSS Grid

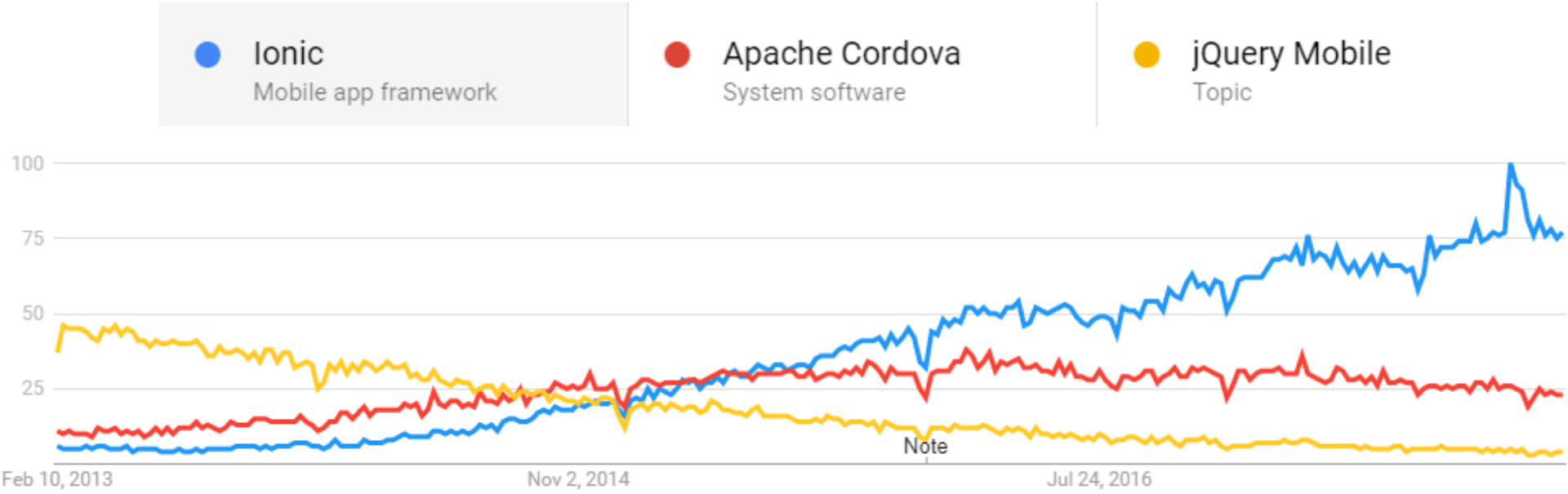

Strong template framework and Shortcodes, Easy to Customize
LT WEB DESIGN
PREMIUM WEB DESIGN COMPANY
RESPONSIVE JOOMLA! TEMPLATE
FOR JOOMLA! 2.5 AND JOOMLA! 3.X
Great!

Frameworks

Frameworks

- UI driven - opinionated
  - jQuery Mobile
    - https://jquerymobile.com/
- Code driven
  - Apache Cordova
  - Ionic
  - Electron

Frameworks - trends



Responsive frameworks

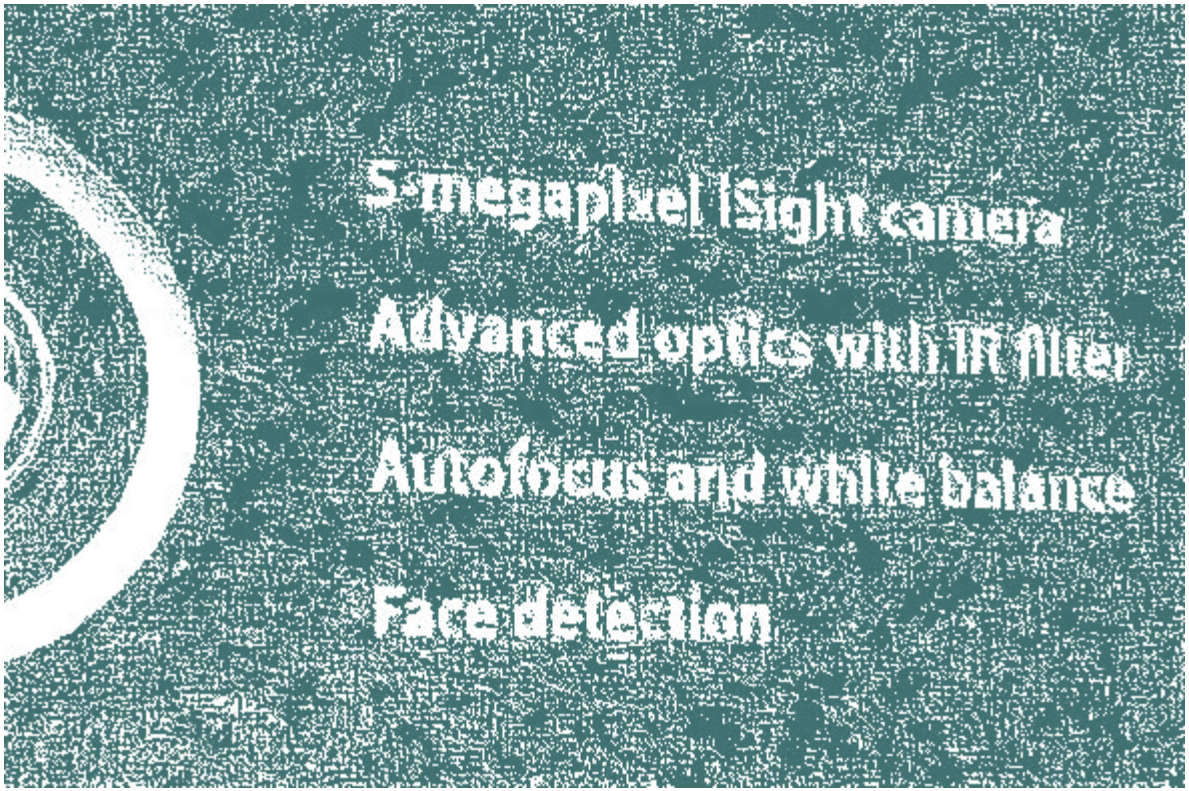
- Foundation 5 by ZURB
  - http://foundation.zurb.com/
  - boilerplate framework and components
  - SASS
- Twitter Bootstrap 4
  - http://getbootstrap.com/
  - boilerplate framework and components
  - look for free templates
  - visual compose - http://www.layoutit.com/

UI libraries

- UI libraries
  - Twitter Bootstrap
  - Zurb Foundation
  - Materialize
- Bootstrap 4 interface builders
  - https://www.bootply.com/
  - https://pingendo.com/

Exercise

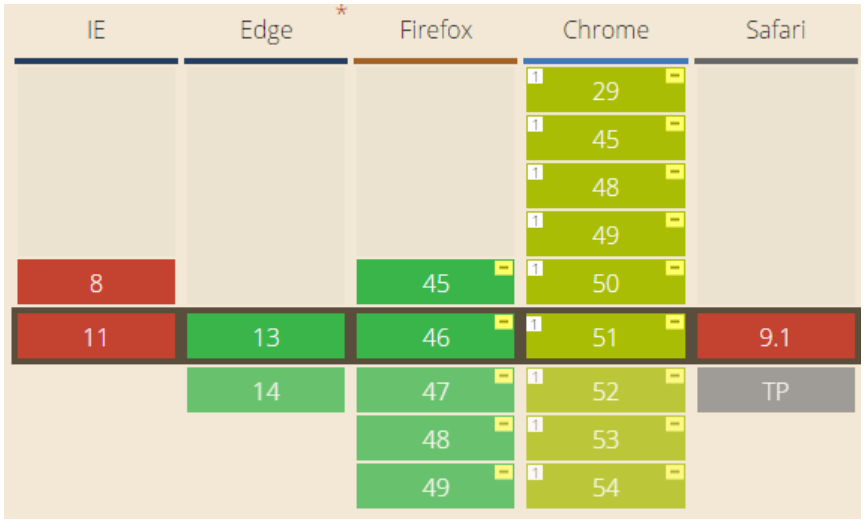
- #19 Twitter Bootstrap
- #20 Zurb Foundation
- #21 Material Design and Materialize

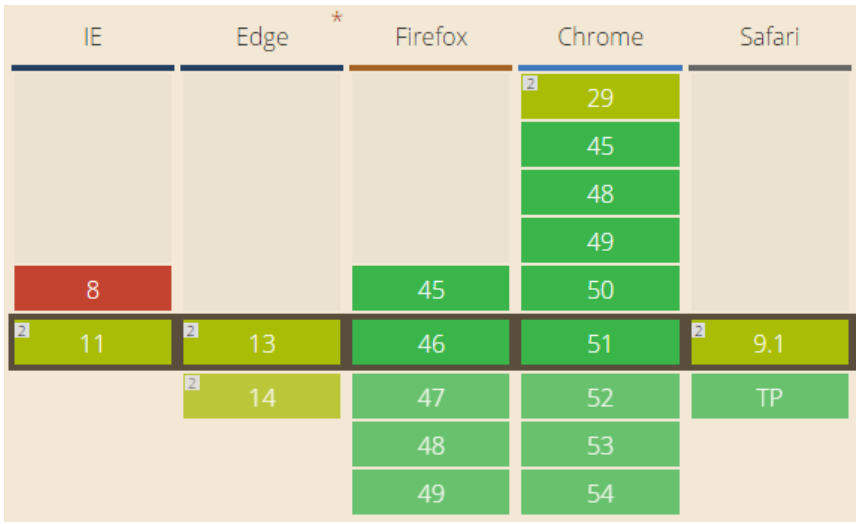


Device access

APIs

- getUserMedia/Stream API
  - WebRTC (peer to peer)
  - https://developer.mozilla.org/en-US/docs/Web/API/Media_Streams_API
- File API
  - https://developer.mozilla.org/en-US/docs/Using_files_from_web_applications





APIs

- Camera API
  - not W3C, Firefox OS
  - https://developer.mozilla.org/en-US/docs/Web/API/Camera_API/Introduction
- Example
  - media/capture.html

Camera iOS

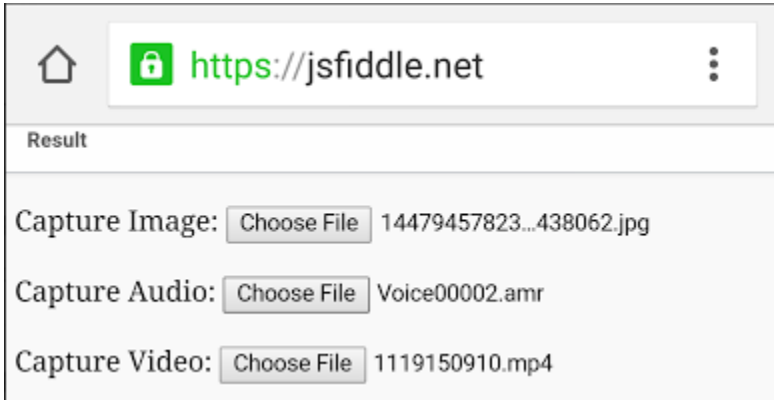
- <input type=‘file’ " /> support iOS6+
  - accept="image/*, audio/*, video/* to limit
- Pic up app
  - Native app to upload files to a web site
  - http://picupapp.com/
  - File-upload form fields don't work in Mobile Safari, a webapp can instruct Picup to choose and upload a photo. Once the upload is complete, Picup returns control to the webapp with information about the upload.

Camera Android

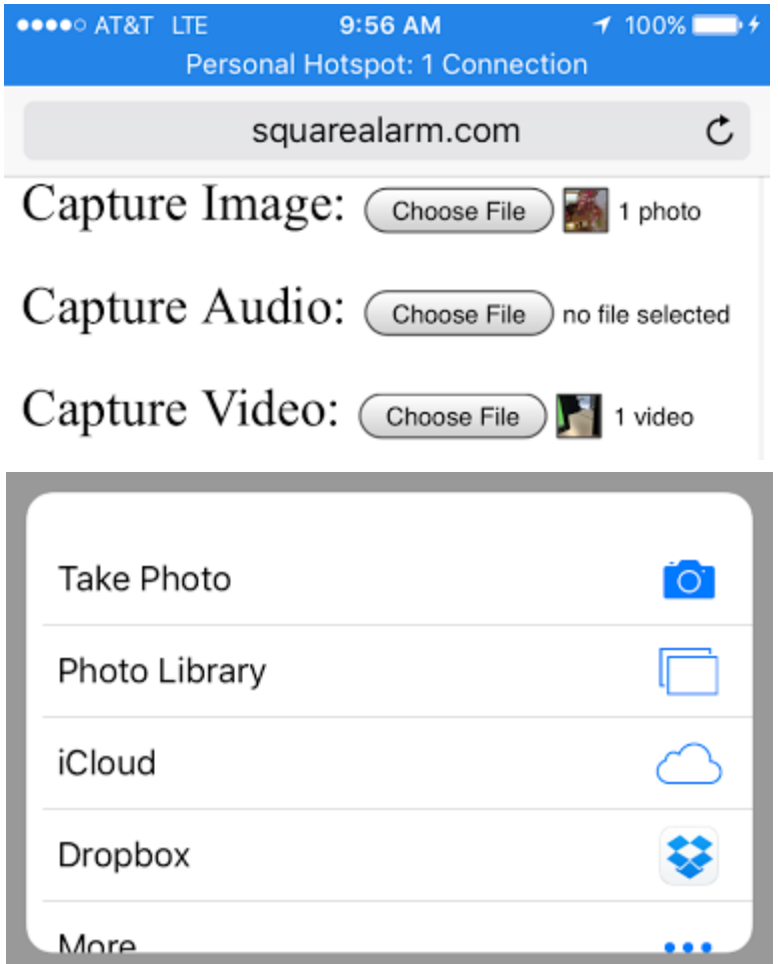
- 3.0 SDK supports image, audio, and video capture. Feb 2011 (capture optional)
  - <form enctype="multipart/form-data" method="post">
  - <h2>Regular file upload</h2>
  - <input type="file"></input>
  - <h2>capture=camera</h2>
  - <input type="file" accept="image/*;capture=camera"></input>
  - <h2>capture=camcorder</h2>
  - <input type="file" accept="video/*;capture=camcorder"></input>
  - <h2>capture=microphone</h2>
  - <input type="file" accept="audio/*;capture=microphone"></input>
  - </form>

Screen shots

- Android



- iOS



Camera resources

- Packages
  - WebcamJS - https://github.com/jhuckaby/webcamjs
  - Script Cam: http://www.scriptcam.com/ (jQuery)
  - jQuery Webcam: http://www.xarg.org/project/jquery-webcam-plugin/

Phone <a href=?

- To call, tel:{phone-number}
- To SMS, sms:{destination}?body={message}.
  - The body might be ignored by some platforms.
  - iOS supports HTML on the body.
- To start mail, mailto:{to}?subject={subject}&body={message}
  - iOS supports HTML on the body.

Phone <a href=?

- Facetime on iOS, facetime:{number or user}
- To Skype call, skype:{user}?call
- To tweet with app, twitter://post?message={message}
  - Tweet

Phone <a href=?

- Maps on Android and iOS < 6, http://maps.google.com?q={query}
- <a href=”http://maps.google.com?q=golden+gate+bridge”> Open Map</ a>
- Navigation on Android and iOS < 6, http://maps.google.com?sa ddr={point1}&daddr={point2}
- Navigate to Pier 39

Phone <a href=?

- Maps on iOS >=6, http://maps.apple.com?q={query}
- Open Map
- Navigation on iOS >= 6, http://maps.apple.com?saddr={point1} &daddr={point2}
- Navigate to Pier 39

Remove automatic linking

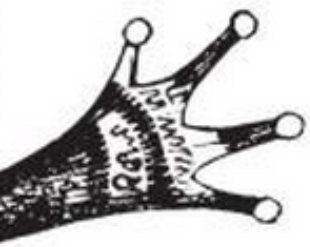
- <meta name=”format-detection” content=”telephone=no”> <meta name=”x-rim-auto-match” content=”none” forua=”true”>

Web sites

- Bruce Lawson
  - http://html5doctor.com/getusermedia/
  - http://introducinghtml5.com/ - his book examples, links

Exercise

- Media capture

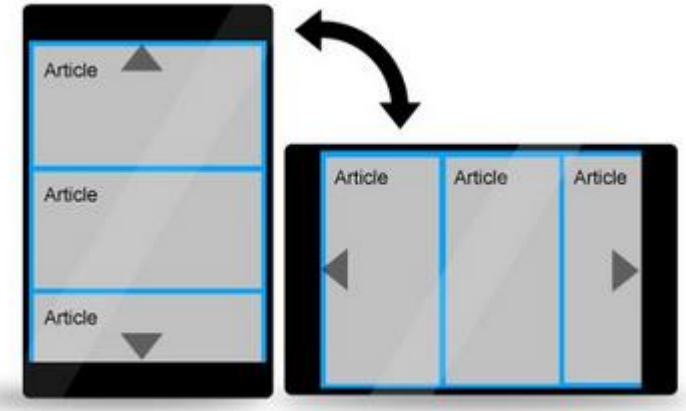


It’s a constraint of not having a mouse but a capability of touch.

Orientation and touch

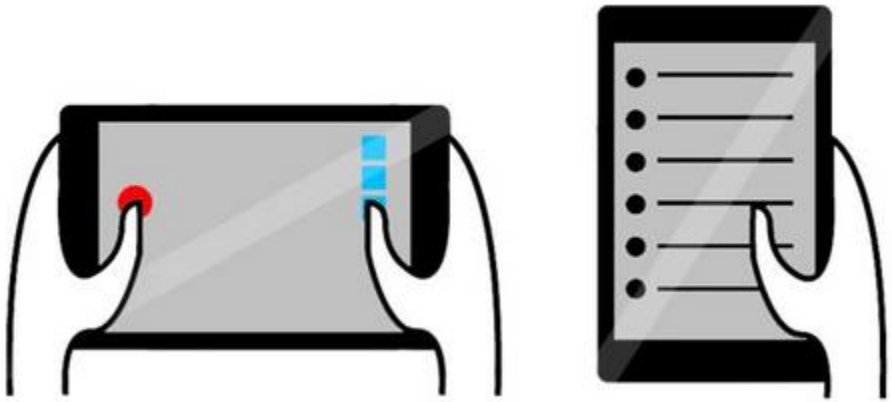
Orientation

- Two angles (landscape and portrait),
- Vertical scrolling is natural to portrait mode.
- Horizontal swiping is more natural to landscape mode.
- Think about different styles.



Orientation

- Re-think your menu layout for orientation.
- Horizontal layouts work for landscape mode.
- Portrait view is better to place the menu items underneath each other. Even 2 columns.

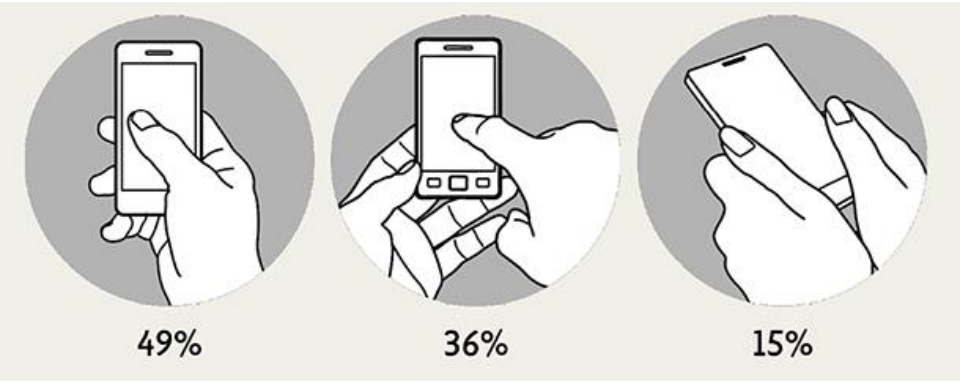


Responsive orientation

- @media screen and (orientation : landscape) {
  - section { }
- }
- @media screen and (orientation : portrait) {
  - section { }
- }

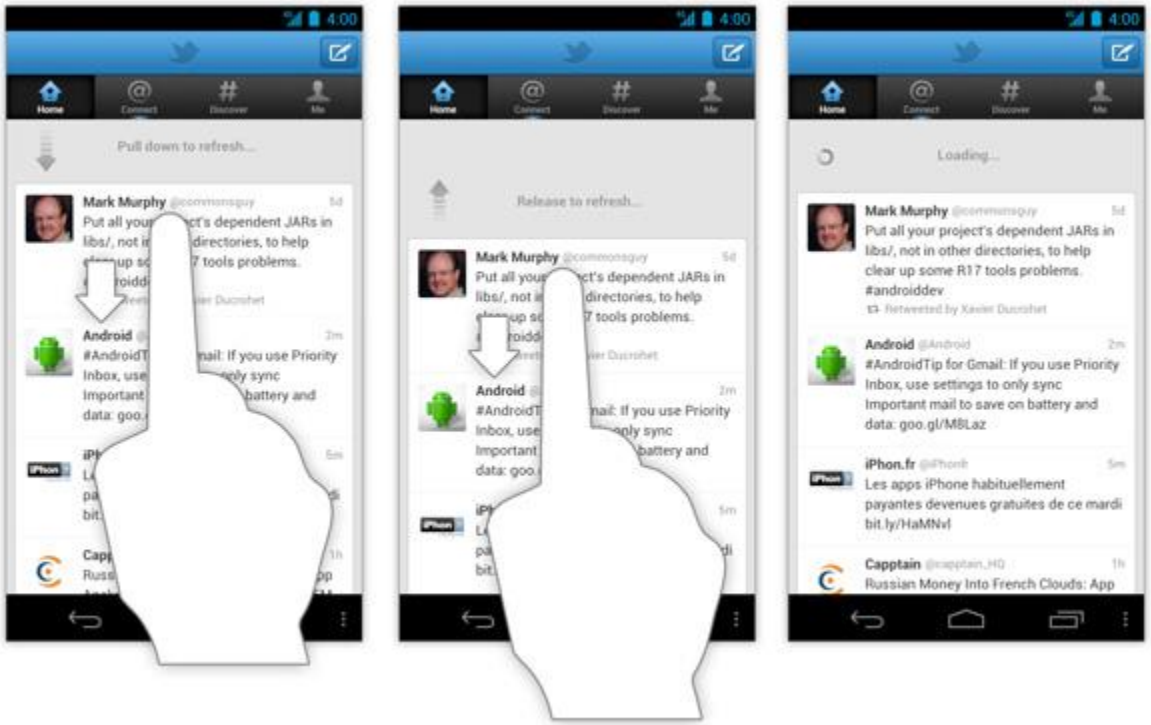
Touch

- Small devices require maximum interactive surface.
- Touch is winning
- Best practice designs
  - right size targets
  - right position
  - use common gestures
    - hover not used



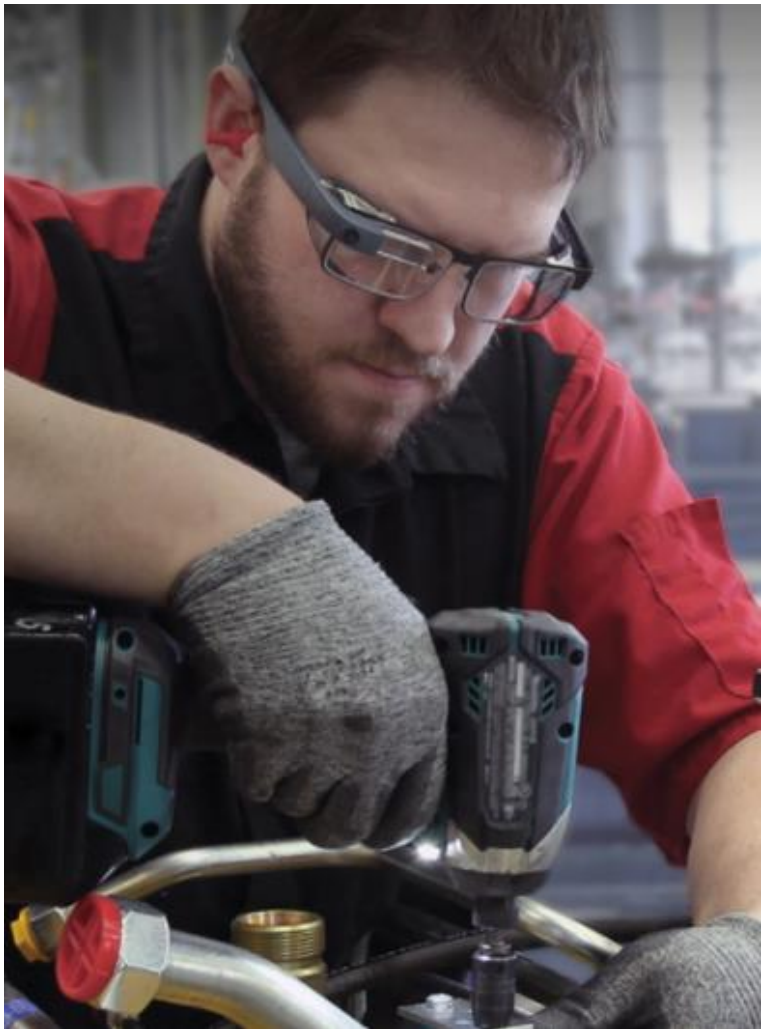
Touch

- A natural interface
- A new set of interactions
  - pull down to refresh
  - swipe for more options
  - draw to select



Touch – VR / AR

- Leap Motion Controller – VR development https://www.leapmotion.com/
- Google Glass
  - https://x.company/glass/
- Microsoft Hololens
  - http://www.microsoft.com/en-us/hololens



Touch - projection

- OmniTouch
  - http://www.youtube.com/watch?v=Pz17lbjOFn8
- Cicret bracelet - http://cicret.com/
  - pico projection

Touch - retrofitting

- Retrofitting PCs - http://air.bar/
  - $75-79 – Windows 13.3 - 15.6-inch screens
  - $99 – MacBook Air 13.3”

Go small by going big

- Shrink content to fit?
- Increase size for convenient UX
  - fingers are bigger than mouse pointers
  - fingers slip
  - iOS says 44x44 points (0.61 in.)
  - MS says 9mm (0.35 in.)
- Visual representation can be up to half of actual target

Go small by going big

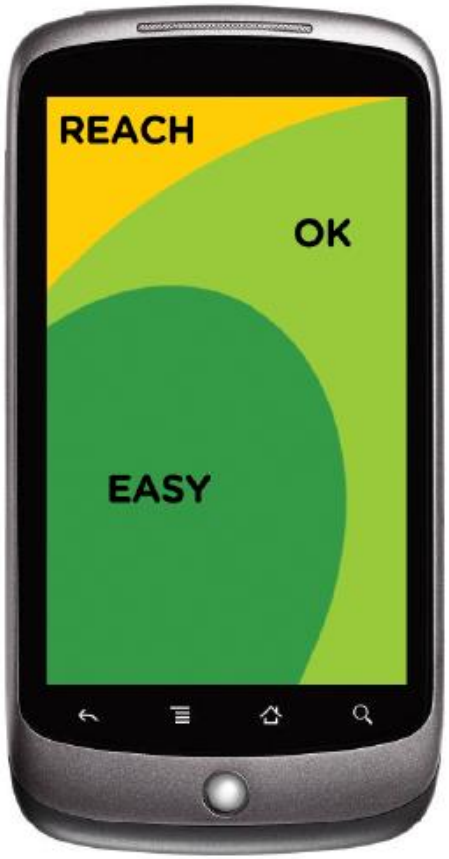
- Make targets bigger when:
  - frequently touched
  - the result of a touch error is severe or really frustrating
    - the UI element is located toward the edge of the screen or difficult to hit
  - when the UI element is part of a sequential task—like using the dial pad

Go small by going big

- Spacing
  - bigger helps
  - separation also helps

Where do we touch?

- Bottom is where we hold the phone
  - we use thumbs often
  - most people are right-handed
- Primary actions in middle or bottom of screen
  - left to right layout
- Upper left corner for uncommon actions



Bark

NUI is natural

- Natural User Interface
  - the content is the interface
  - direct interaction with content not chrome
  - reduce visuals that are not content
- NUI videos
  - http://vimeo.com/channels/nui/46022904

NUI is natural

- Icons, menus & pointers are replaced by gestures, intuition and fingers
- Direct, not indirect, is how we really interact
- Still building a common set
  - beginning guides help
  - the iPad four finger swipe

Hover - a no-touch event

- Tool tips that appear on hover (desktop) don't work, so no help there.
  - yes, no giant menus crowding the screen!
- Options to replace
  - put on screen
  - on tap / swipe
  - put on separate screen
  - get rid of it

Hover - a no-touch event

- Support for trackpads, trackballs, keypads, scrollwheels, keyboards?
- :hover can be used to highlight control without JS
  - :focus is not always explicit state
- Older devices still out there. Use:
  - smaller targets
  - progressive enhancement
    - graceful degradation is a desktop first strategy

Touch

- Low level spec
  - touchstart, touchend
    - area for activate, area for deactivating
  - touchmove
  - touchcancel (Safari)
- Event payload (multi-touch)
  - touches
  - targetTouches
  - changedTouches

Touch targets

- Josh Clark - “The 44-pixel block is, in many ways, the basic unit of measurement for the iPhone interface, establishing the visual rhythm of many iPhone apps.”
  - Tab bars are 49px
  - Nav bars are 74px
  - Navigation bar, toolbars, table cells are 29 pixels tall with tap height of 44 pixels

Touch targets

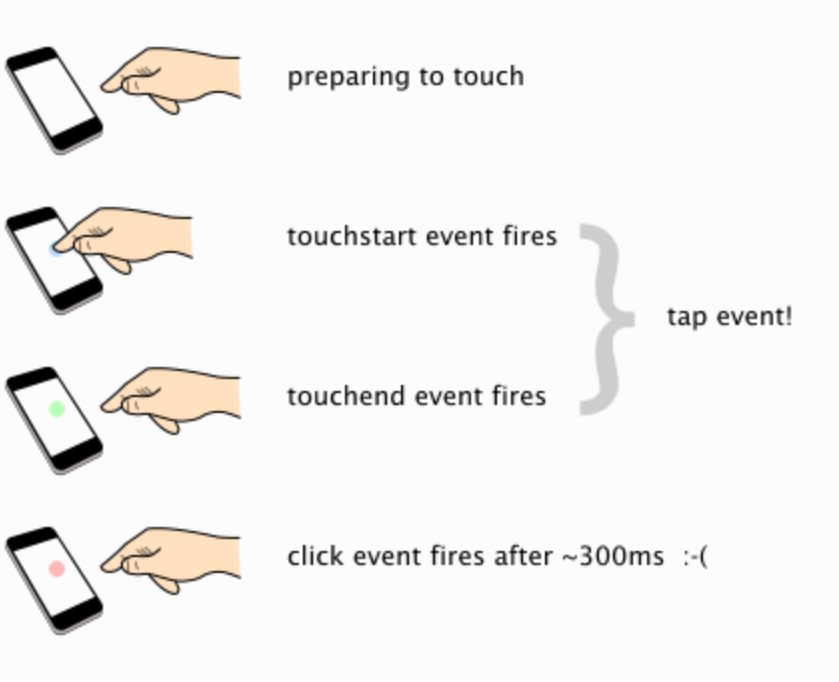
- Put the anchor tag around the block parent element
  - remove the a style
  - apply it to a strong and markup the original text with strong.

Gestures

- No standard gesture API yet
- Apple
  - proprietary API for gesture* events
  - http://developer.apple.com/library/ios/#DOCUMENTATION/AppleApplications/Reference/SafariWebContent/HandlingEvents/HandlingEvents.html
  - http://developer.apple.com/library/safari/#documentation/UserExperience/Reference/GestureEventClassReference/GestureEvent/GestureEvent.html#//apple_ref/doc/uid/TP40009353

Click delays

- Mouse click event delayed by 300ms
- Chrome Android removed it in v32
  - 5/2015 – v.42
- JS library to remove
  - https://github.com/ftlabs/fastclick



Disable zooming and scrolling

- When you need more control
- No zooming (viewport scaling)
  - <meta name="viewport" content="width=device-width,
    initial-scale=1.0, user-scalable=no">
- No scrolling
  - document.addEventListener('touchmove',
    function(e) {
      e.preventDefault( ); });

Exercise

- Touch detection with Modernizr and event names
  - test in Chrome emulator, mobile devices

Touch libraries

- *Hammer.js
  - pan, pinch, press, rotate, swipe, tap
  - http://eightmedia.github.com/hammer.js/
  - Used by Microsoft templates, Zurb Foundation
- *jQuery UI Touch Punch – maps click to touch
  - http://touchpunch.furf.com/
    - will ignore text inputs when form is large on JellyBean
    - MC Hammer Can't Touch This

Touch library - jQM

- When using jQuery UI or any other package that needs touch support use
  - jQuery Mobile Download Builder (Custom Download)
  - http://jquerymobile.com/download-builder/
  - Select Events / Touch and download custom build

Fires a resize event with a slight delay to prevent excessive callback invocation

- ☑ Touch

  Touch events including: touchstart, touchmove, touchend, tap, taphold, swipe, swipeleft, swiperight, scrollstart, scrollstop

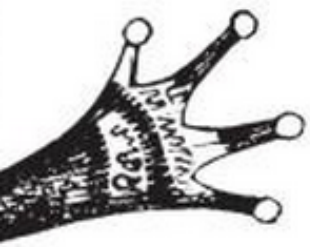
- ☐ Events

Touch enabled plugins

- *Swiper – image slider
  - http://www.idangero.us/sliders/swiper/
  - onTouchStart, onTouchMove, onTouchEnd, onSlideReset, onSlideChangeStart, onSlideChangeEnd, onSlideNext, onSlidePrev, onSlideClick, onSlideTouch

Exercises

- Finger Painter



Videos

- Stephen Woods of Flickr – Touch interactions in JavaScript
  - YouTube - http://youtube.com/watch?v=IcD9CF0bxyk
  - Slides - http://www.slideshare.net/ysaw/creating-responsive-html5-touch-interfaces

Pressure

- Apple
- Force Touch
  - Apple Watch
  - less sensitivity
- 3D Touch
  - peek (light = preview)
  - pop (heavy = tap)

- http://pressurejs.com/

Google advice

- https://developers.google.com/web/fundamentals/design-and-ui/input/touch/

Popular packages

- OwlCarousel2 - responsive carousel slider
  - https://owlcarousel2.github.io/OwlCarousel2/
- Slideout - touch slideout navigation menu
  - https://slideout.js.org/
- Swiper - touch slider (templates also)
  - http://idangero.us/swiper/#.WaAGcSiGOCg
- Sortable - reorderable drag-and-drop lists
  - http://rubaxa.github.io/Sortable/

Exercise

- Hammer test
- Drag and drop with jQueryUI and TouchPunch
  - swap out TouchPunch with Hammer

End matter

Web sites - Email

- Litmus - https://litmus.com - $$
  - Preview/update across 30+ email clients and devices
- Campaign Monitor -
  - http://www.campaignmonitor.com/testing/
  - 20+ clients, price scales by emails sent

Tools

- Mobile devopment software stacks
  - https://stackshare.io/stacks/mobile
- Mobile design tools
  - http://www.uxpin.com/
  - https://webflow.com/
  - https://www.flinto.com/ - Mac
  - http://www.irise.com/

Web sites

- Mobile Design patterns
  - http://mobile-patterns.com/
- HTML5 Rocks
  - http://www.html5rocks.com/en/mobile
    http://www.html5rocks.com/en/tutorials/
    http://www.html5rocks.com/en/mobile/mobifying.html
- Scripp's National Spelling Bee case study
  - http://www.smashingmagazine.com/2015/09/rapid-app-development-buzzworthy-spelling-bee-app/

Web sites - links

- mobiForge
  - https://mobiforge.com/
- Programmable Web
  - https://www.programmableweb.com/

Conferences / videos

- *Breaking Development - Apr
  - http://bdconf.com
  - http://vimeo.com/bdconf
- Fluent (O'Reilly)
  - http://fluentconf.com/
  - http://www.youtube.com/playlist?list=PL75AC4484E6866741
- An Event Apart (A List Apart)
  - http://aneventapart.com/
  - http://aneventapart.com/news/tag/video

Conferences / meetings

- International - Lanyrd
  - http://lanyrd.com/topics/mobile-web/
  - http://lanyrd.com/topics/html5/
  - http://lanyrd.com/topics/javascript/
  - http://lanyrd.com/topics/web-development/
- Local - Meetup
  - http://www.meetup.com/Mobile-Media-Club/
  - http://www.meetup.com/KCWebCore/
  - http://www.meetup.com/Kansas-City-Mobile-App-Developers-Group/

Final items

- Certificates
- Evaluation
  - http://www.metricsthatmatter.com/centriqfoss1