Pragmatic Responsive Web Design

From John Tsevdos / @tsevdos

Hi I'm John and I'm social...

- tsevdos.com
- phrappe.com
- @tsevdos
- github.com/tsevdos
- linkedin.com/in/tsevdosjohn

Responsive Web Design

- why?
- how?
- tips and techniques

Why responsive?

- and not native
- and not mobile site

Because with a responsive web site we provide support

- for all devices, including tablets, phones and tvs
- for all screen sizes
- for all browsers
- and because we love the "one site to rule them all" attitude

Responsive web design principles

- flexible layout (grid)
- media queries
- flexible media (images, videos, etc.)

Being pragmatic

- think different
- support as many devices/browsers your budget allows
- solve problems with simple solutions

First things first

- think out of the box, think responsivly
- forget pixel-perfection
- layout (wireframe it)
- forget colors, images, logos etc.
- prototype it (browser-ready protype)

Mobile first

- mobile first content?
- mobile first styles

Question : why mobile first?

Set your major and minor break points

- the first break point is the no "break point"
- primary (mobile first) styles
- set break points where your content needs adjusting
    - phones (320/480)
    - tablets (768/1024)
    - (small) desktop (1024)
    - (large) desktop (1200 and over)

Set your major and minor break points

```
@media only screen and (min-width:480px) {
}
@media only screen and (min-width:768px) {
}
@media only screen and (min-width:1024px) {
}
@media only screen and (min-width:1200px) {
}
```

Question : does it make sense to create a break point (and make adjustments) for every known device?

CSS : Flexible layout

The % way

- do the right math (target ÷ context = result)

```
#container { width:100%; max-width:1200px; }
#main { float:left; width:75%; } /* 900 / 1200 */
#sidebar { float:right; width:25%; } /* 300 / 1200*/
```

- flexible margin / paddings

```
#main { float:left; width:71.666666%; /* 860 / 1200 */ padding:1.666666
%; /* 20 / 1200 */ }
```

- tip: use % for fonts as well

CSS : Flexible layout

The em way

- same as above, but with ems
- how much is 1em?

```
1em = 100% from 16px = 16px
1.5em = 150% from 16px = 24px
0.5em = 50% from 16px = 8px
```

- tip: use ems for fonts as well

CSS : Flexible media

```
img { max-width:100%; }
```

```
img, object, embed, video { max-width:100%; }
```

```
img { max-width:100%; height:auto !important; /* just in case, to force
 correct aspect ratio */ }
```

```
img { -ms-interpolation-mode: bicubic; } /* IE scaling fix */
```

- fluid images from unstoppable robot ninja post
- fitvids.js

CSS : The rest

- normalize
- set a good base (typography, colors, helpers, etc.)
- use CSS preprocessors (SASS, LESS, etc.)

Markup

- use HTML5
- take control

```
<meta name="viewport" content="width=device-width, initial-scale=1" />
```

- be nice to IE6-8

```
<meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1">
```

  - use html5shiv or modernizr
  - respond.js

Markup

- Easy mobile-friendly enchantments
  - Tel:

    ```
    <a href="tel:+306948123456">+306948123456</a>
    ```

  - SMS:

    ```
    <a href="sms:+306948123456">+306948123456</a>
    ```

  - iPhone/iPad icon:

    ```
    <link rel="apple-touch-icon" sizes="57x57" href="touch-icon-iphone-114.png" />
    ```

Markup

Easy mobile-friendly inputs

```
<!-- Default Keyboard -->
<input type="text" />

<!-- Numeric Keyboard -->
<input type="number" />

<!-- Pattern Keyboard -->
<input pattern="[0-9]*" type="text" />

<!-- Number Keyboard -->
<input type="tel" />

<!-- URL Keyboard -->
<input type="url" />

<!-- e-mail Keyboard -->
<input type="email" />
```

Responsive web design is great, but you'll need more

- unobtrusive javascript
- feature detection
  - modernizr
- enrich touch devices (tablets/mobiles)
  - navigation
  - touch events (on image galleries, etc.)
  - geolocation

Hand-code or frameworks

that's a tough one, it depends

- budget
- time
- project scale

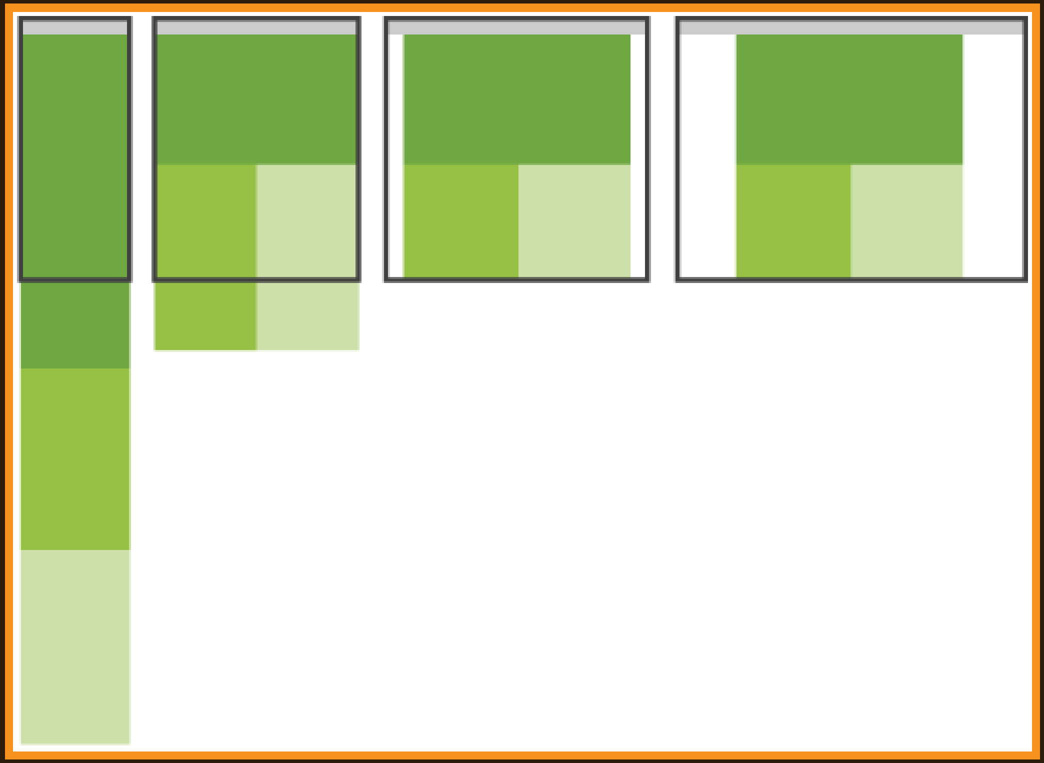
Responsive web design frameworks

- Bootstrap
- Foundation

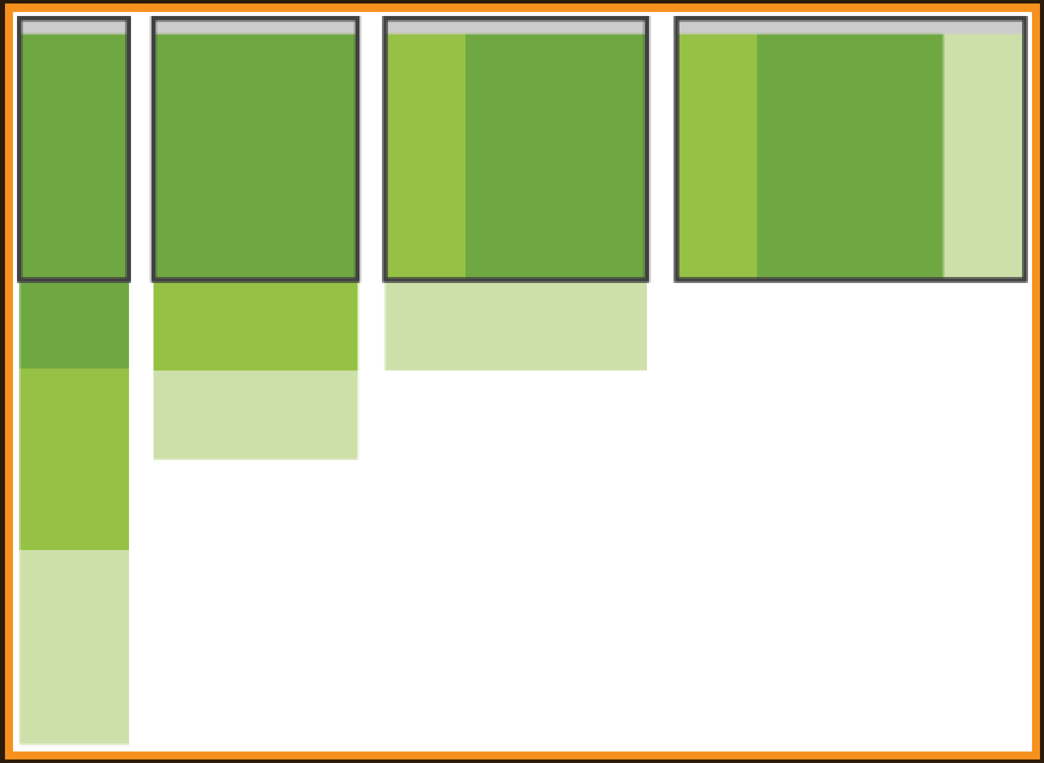
Responsive web design patterns

- Multi-Device Layout Patterns (all images are from the post)

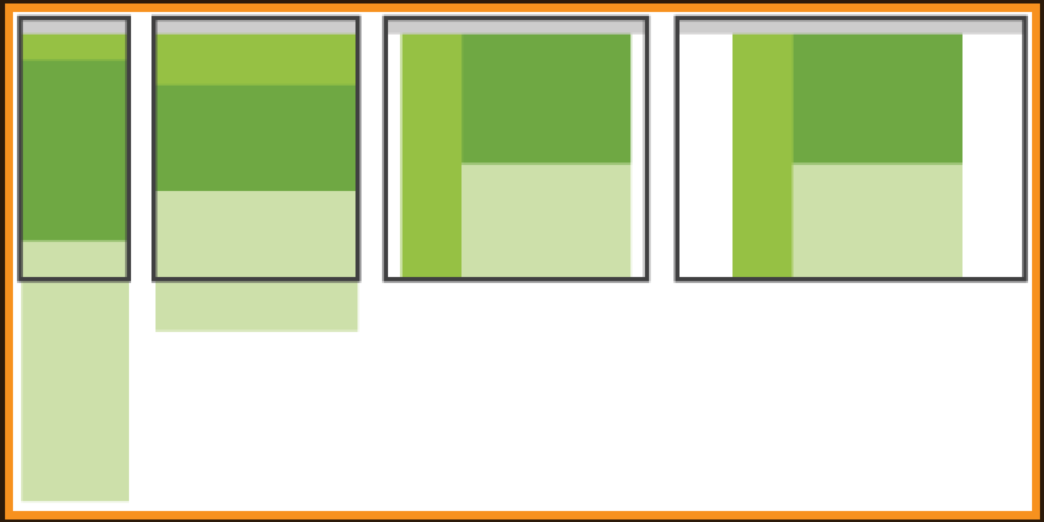
Mostly fluid



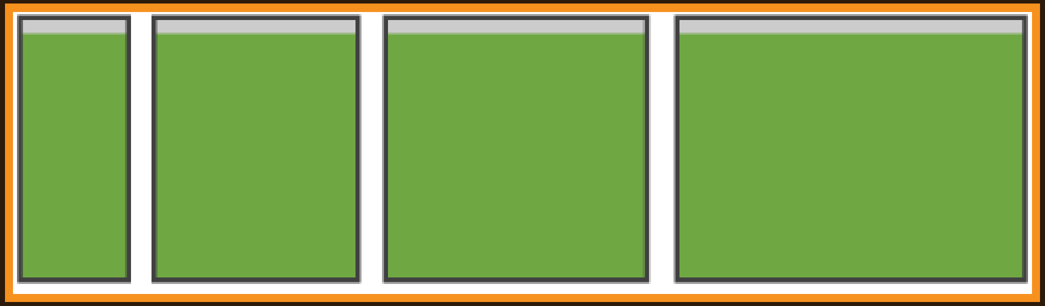
Column drop



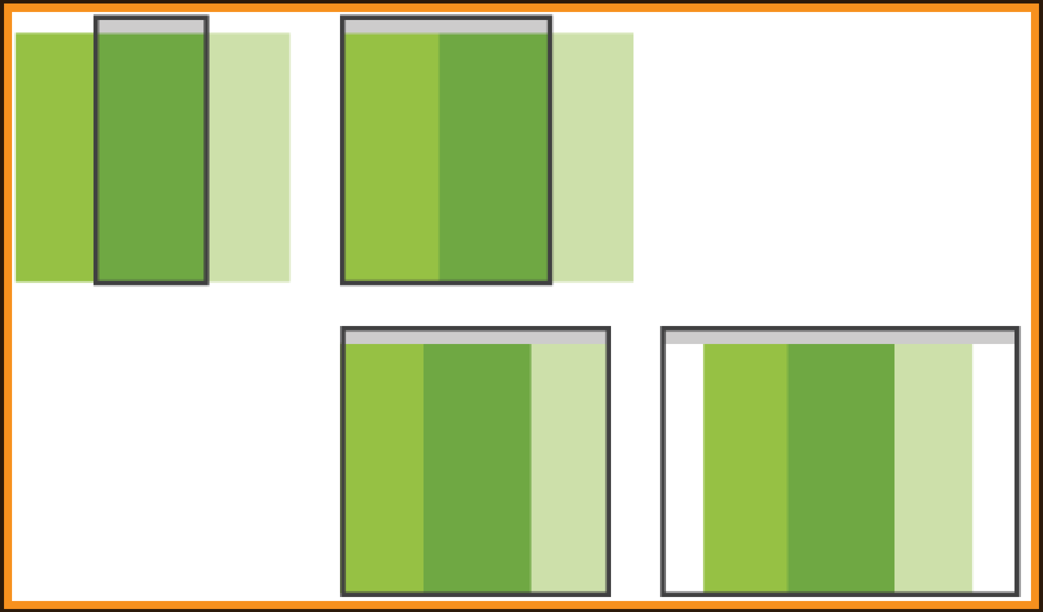
Layout shifter



Tiny tweaks



Off canvas



Responsive web design patterns

- Responsive Patterns

Toolbox

- text editor / IDE
- Chrome / Firefox
- iPad
- iPhone
- Adobe Edge Inspect

Testing

- don't rely on emulators and other desktop tools
- try to test to as many devices / browsers / OS you can
- you're not magician, it may not work on your mum's mobile...

Thank you

Questions ?