

شروع کار با CSS

بخش دهم: مباحث پیشرفته تر در طراحی قالب

انواع طراحی وب، طراحی واکنش گرا

(Responsive Web Design)

سید کاوه احمدی



Ethan Marcotte, Responsive Web Design (2010)
<http://alistapart.com/article/responsive-web-design>

“Responsive web design consists of three components: a fluid grid, fluid media and media queries”

- Ethan Marcotte

چرا به یک وب سایت واکنش گرا نیاز داریم؟

■ گسترش تولید، استفاده و تنوع دستگاه‌های هوشمند

— هر روز ۱.۴ میلیون دستگاه هوشمند در برابر ۳۱۷,۱۲۴ نوزاد متولد می‌شوند.

■ گسترش استفاده از دستگاه‌های هوشمند

■ تنوع در اندازه صفحات نمایش

■ تنوع سیستم عامل و نوع مرورگر

آیا من به یک طراحی واکنش‌گرا نیاز دارم؟

- بازدیدکنندگان در چه دیوایس‌هایی سایت را مشاهده خواهند کرد.
- طراحی واکنش‌گرا مستلزم کار بیشتر و ایجاد سایت‌هایی با حجم بیشتر خواهد بود. بنابراین پیش از هر چیزی باید دید آیا واقعا به یک سایت واکنش‌گرا نیاز داریم یا نه.

مخاطب اهمیت دارد

- امروزه زمانی که سایتی را برای طیف وسیعی از کاربران ایجاد می‌کنید، باید موارد زیر را در انتخاب نوع قالب در نظر بگیرید:
- Screen resolution,
- Browser choice,
- Whether or not the browser is maximized,
- Extra toolbars open in the browser (History, Bookmarks, etc.),
- Even the operating system and hardware.

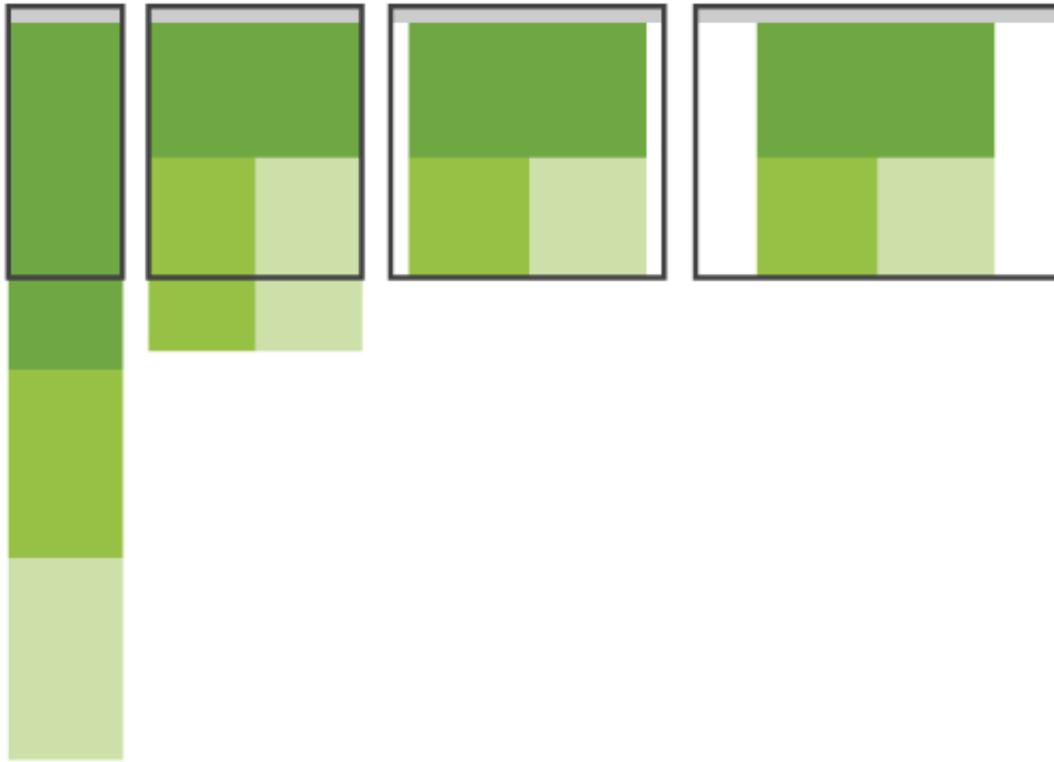
طراحی واکنش گرا چیست؟

- در یک طراحی استاندارد واکنش گرا، با توجه به خصوصیات دستگاه‌ها و صفحات نمایش، ساختار چینش وب سایت و حتی نوع نمایش آنها باید با استفاده از استایل‌های متفاوت برای حالت‌های متفاوت تغییر کند.

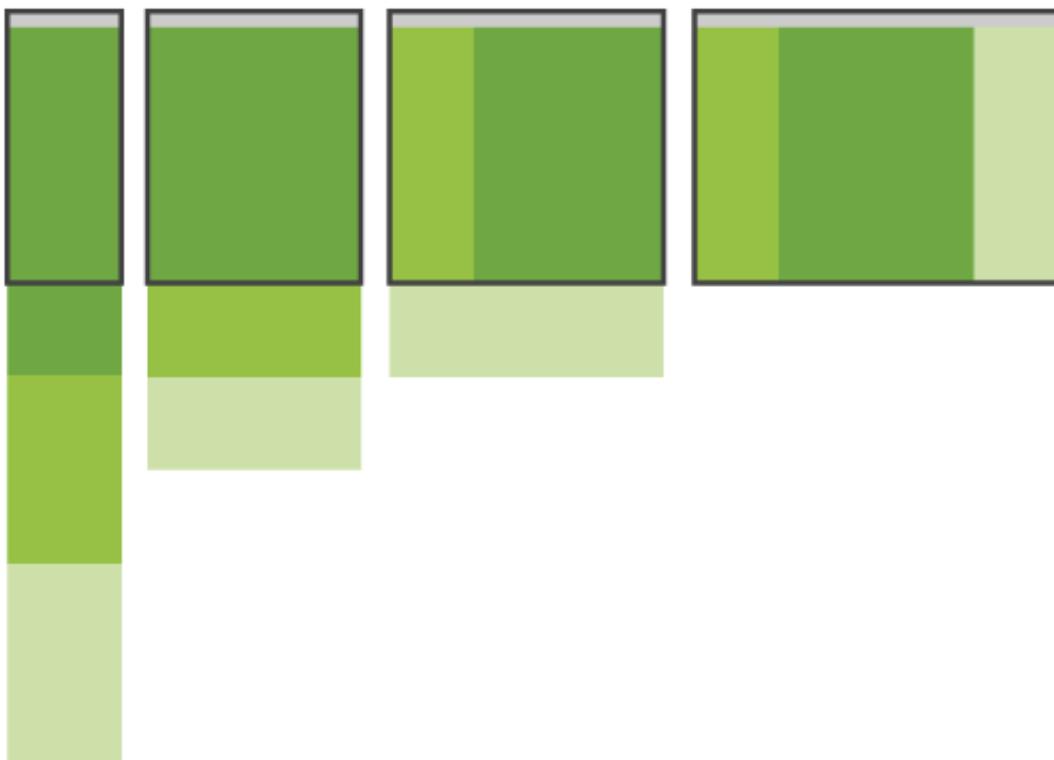
ساختارهای متداول در طراحی واکنش گرا چگونه‌اند؟

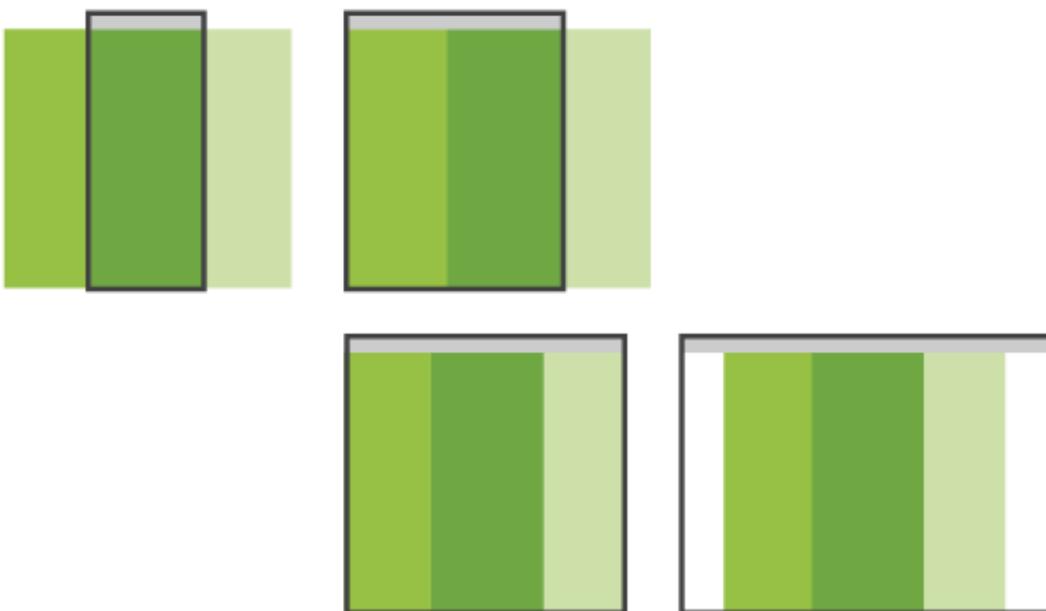
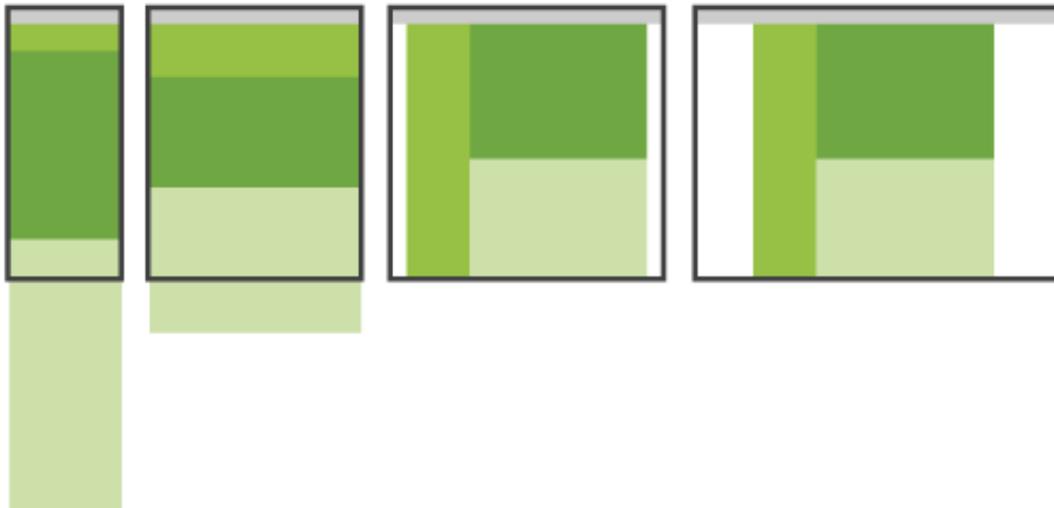
- با توجه به اندازه صفحات نمایش، مهمترین نکته در یک طراحی واکنش گرا در دسترس قرار دادن مطالب و قسمت‌های مهم وب سایت به صورت کاملا ساده، در دستگاه‌ها و اندازه صفحات نمایش متفاوت است.
- مسلما نمی‌توان یک سایه بار با دو ستون از تبلیغات را در یک نسخه موبایل به کاربر نمایش داد!
- وب سایت باید با تغییر عرض صفحات ساده‌تر و ساده‌تر شود و با توجه به درجه اهمیت عناصر در صفحه به ترتیب از نمایش آنها چشم پوشی شود و ساختار به یک ساختار خطی عمودی تبدیل شود.

ساختار عمومی



ساختار ریزشی





طراحی موبایل در برابر طراحی واکنش‌گرا

▪ اگر کاربران موبایل زیادی داریم، بهتر است یک طراحی اختصاصی برای موبایل

داشته باشیم

– حجم و کار کمتر

– ایجاد Native app

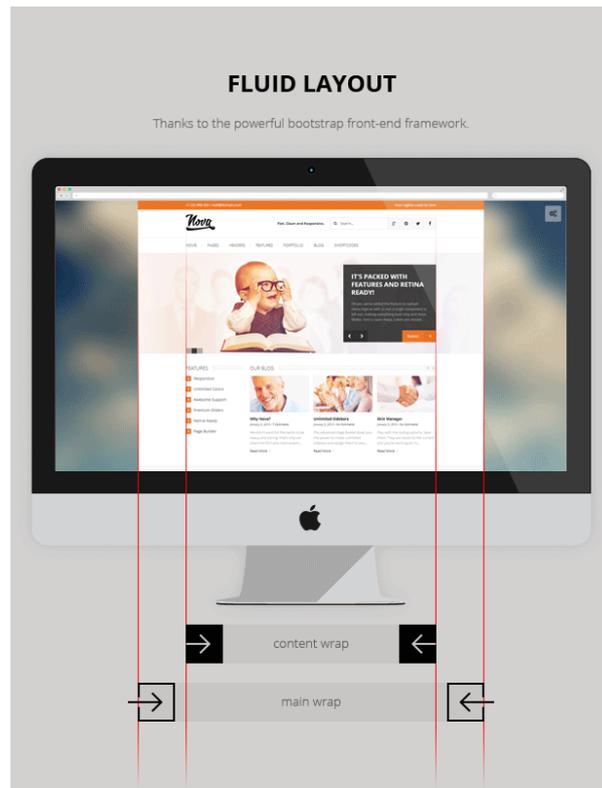
▪ app محلی فیس‌بوک

– قابلیت استفاده از رویدادهای ویژه موبایل

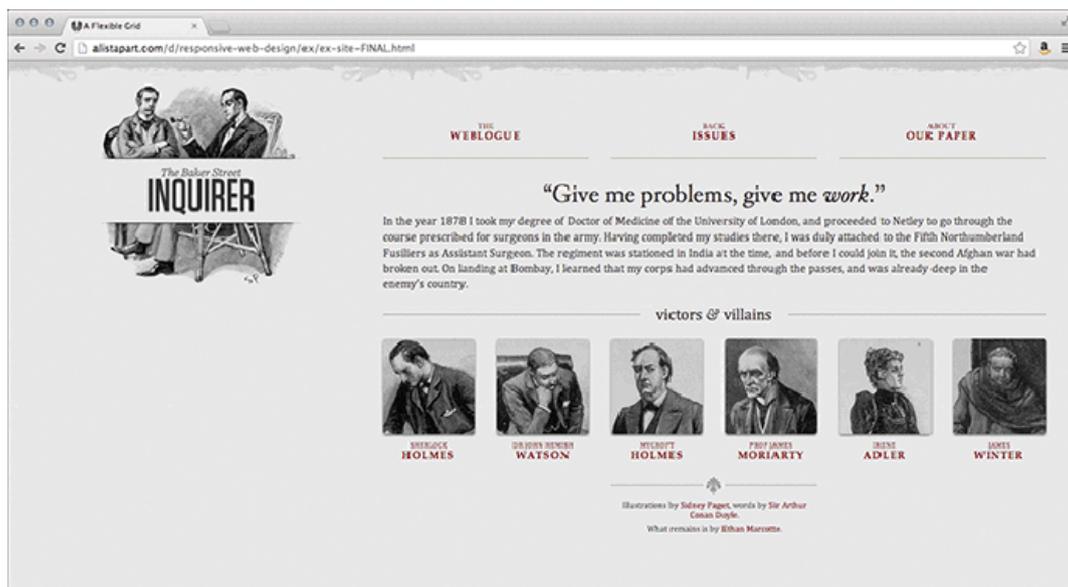
Flexible Grid

Fluid Layouts + Responsive CSS Grids

Fluid Layouts vs RWD



Fluid Layouts vs RWD



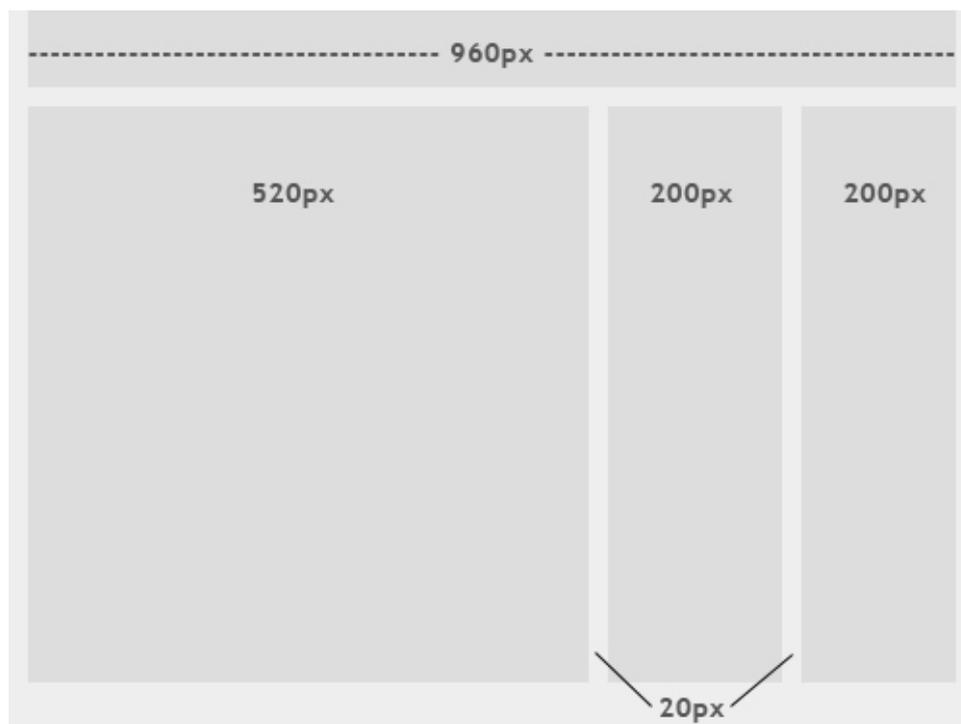
Popular Layout Technics

- Fixed-width layouts
- Fluid (liquid) layouts
- Elastic layouts
- Hybrid layouts

Fixed Website Layouts

- قالب درون المانی قرار گرفته که عرض ثابتی دارد.
- عرض المان‌های اصلی قالب براساس درصدی از عرض مشخص شده یا مقداری ثابت مشخص می‌شود.
- نکته مهم این است که قالب طراحی شده تا تغییری در شکل آن به وجود نیاید.
— بدون در نظر گرفتن این نکته که رزولوشن صفحه کاربر چقدر است.

Fixed Website Layouts



Fixed Web Page Design

- بسیاری از طراحان این سبک طراحی را ترجیح می دهند.
- ساده تر است.
- تضمینی وجود دارد که همانچه طراح می بیند را کاربر نیز خواهد دید.

Fixed Web Page Design

■ مزایا:

- طراحی ساده‌تر
- عرض سایت در همه جا یکسان است بنابراین قرار دادن تصاویر، ویدئوها، فرم‌ها و... کمتر مشکل ایجاد می‌کند.
- نیازی به استفاده از `min-width` یا `max-width` که توسط همه مرورگرها پشتیبانی نمی‌شوند نخواهیم داشت.
- حتی با طراحی سایت برای کمترین رزولوشن (800×600) محتوا محل کافی برای نمایش در تمامی رزولوشن‌ها را خواهد داشت.

Fixed Web Page Design

■ معایب:

- محل خالی (`white space`) زیادی در رزولوشن‌های بالا خواهیم داشت.
- نقض نسبت الهی (`divine proportion`) و قانون سوم توازن و اصول طراحی
- رزولوشن‌های پایین‌تر اسکرول افقی می‌خورند (براساس عرض ثابت استفاده شده)
- از نظر کاربرد پذیری (`usability`) امتیاز کمتری خواهد گرفت.
- عدم استفاده مناسب از عرض صفحه و ایجاد صفحات طولانی در حالتی که می‌توانستیم از صفحه استفاده بهتری کنیم.

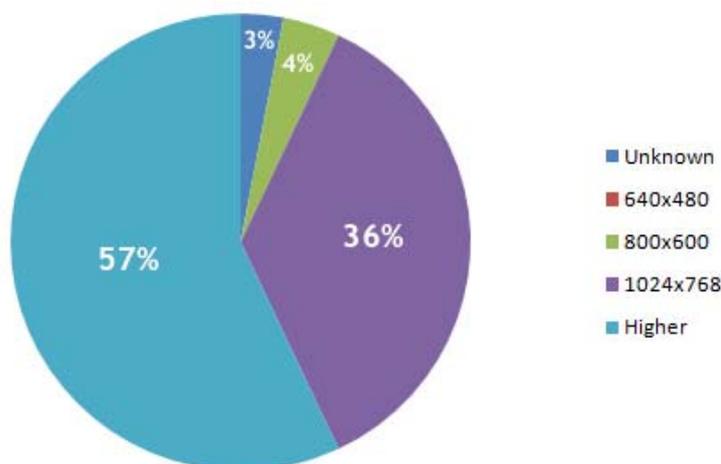


Fixed Web Page Design

■ از چه عرضی برای صفحه استفاده کنیم؟

— توجه کنید که این آمار اساساً چندان دقیق نیست و در مناطق جغرافیایی مختلف می‌تواند

متفاوت باشد.



Fixed Web Page Design

■ این آمار را از کجا بیاورم؟

- <http://www.w3counter.com/globalstats.php>
- http://www.w3schools.com/browsers/browsers_display.asp
- http://www.onestat.com/html/aboutus_pressbox31.html

Fixed Web Page Design

■ بالاخره چه عرضی؟

- بسیاری از طراحان یکی از اندازه‌های 960px یا 760px را استفاده می‌کنند.
- قالب با عرض 960 برای کاربران با رزولوشن 1024x768 و بالاتر مناسب است.
- اگر کاربرانی دارید که از رزولوشن 800x600 استفاده می‌کنند از عرض 760 استفاده کنید.
- در هر حال استفاده از `margin: 0 auto;` فراموشتان نشود.

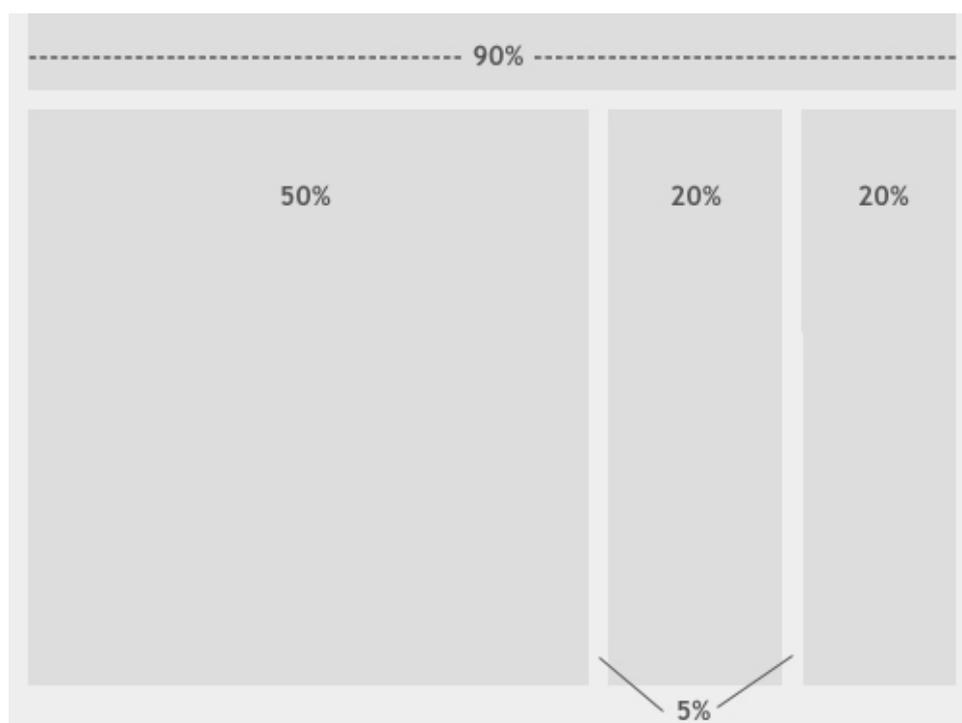
■ چرا ۹۶۰؟

- بر ۱۲، ۱۰، ۸، ۶، ۵، ۳ و ۲ بخش پذیر است.

Fluid (Liquid) Website Layouts

- عرض اجزای اصلی قالب به شکل نسبی (%) مشخص می شود.
- بنابراین با توجه به رزولوشن صفحه کاربران وفق داده می شود (تنظیم می شود).

Fluid (Liquid) Website Layouts



Fluid Web Page Design

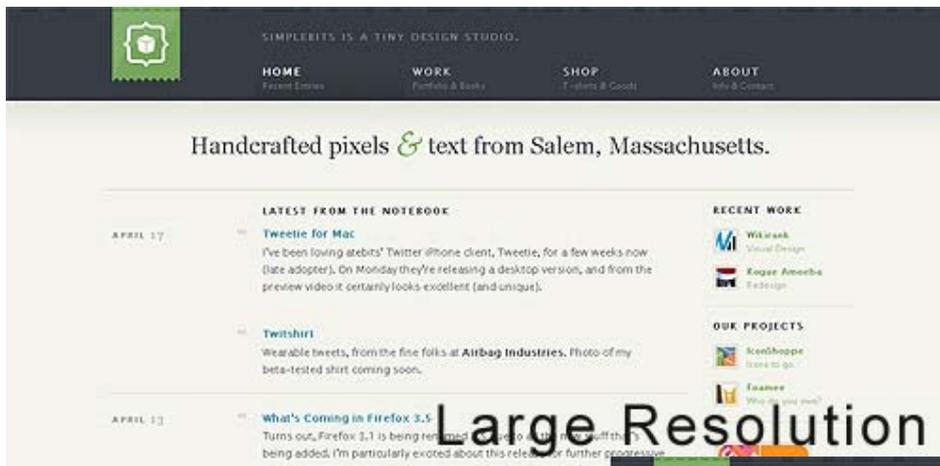
■ مزایا:

- این قالب‌ها کاربرپسندتر (user-friendly) هستند چراکه خود را با تنظیمات کاربر وفق می‌دهند.
- می‌تواند از ایجاد فضای خالی زیاد در رزولوشن‌های بالا و اسکرولر بار افقی در رزولوشن‌های پایین جلوگیری کند.
- در یک کلام از فضای صفحه به شکل بهتری استفاده می‌کند.

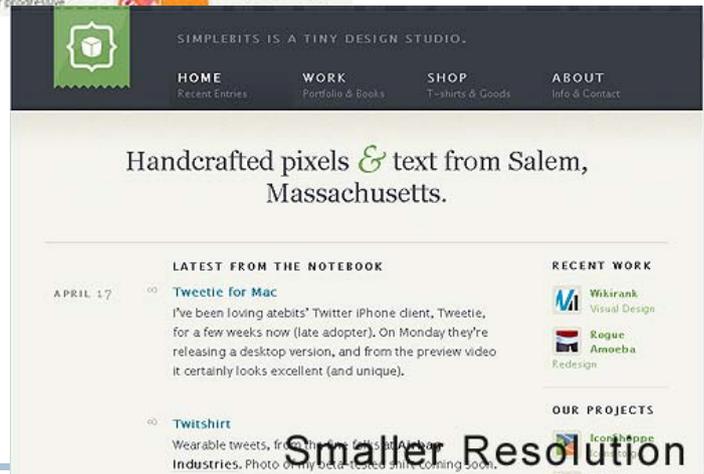
Fluid Web Page Design

■ معایب:

- طراح کنترل کمتری روی آنچه کاربر می‌بیند خواهد داشت و ممکن است مشکلاتی در رزولوشن‌های خاصی پدید آید.
- تصاویر، ویدئوها و... باید برای نمایش در اندازه‌های مختلفی و رزولوشن‌های متفاوتی آماده باشند.
- در رزولوشن‌های خیلی بالا ممکن است سایت در یک ارتفاع کم نمایش داده شود و فضای خالی زیادی در پایین آن ایجاد شود.



Large Resolution



Smaller Resolution

Fluid Web Page Design

■ نکاتی که باید در نظر داشت:

— استفاده از یک طراحی ساده

— مشخص کردن `max-width` و `min-width`

Elastic Design

- برخی مواقع مورد توجه طراحان قرار می‌گیرد چون به نحوی ترکیبی از دو طراحی قبلی است.

- ایده کلی استفاده از **em** (و **rem**) برای اندازه دهی به المان‌ها است

- کنترل قوی تایپوگرافی

“A pixel is an unscalable dot on a computer screen, whereas an em is a square of its font size. Because font sizes vary, the em is a relative unit that responds to users’ text-size preferences.”

Elastic Design

- مزایا:

- اگر درست طراحی شود، قالب کاملا کاربرپسند خواهد بود.

- معایب:

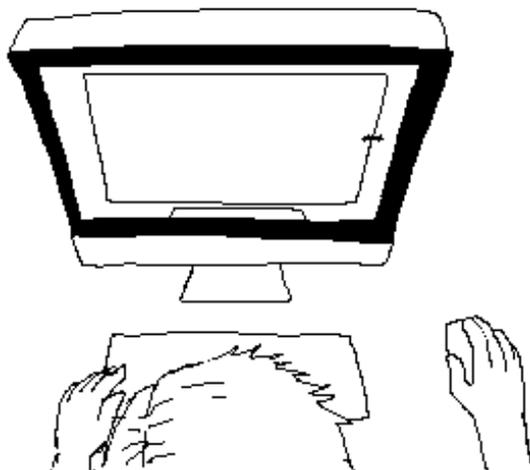
- نیاز به بررسی زیادی برای اطمینان از کاربرد پذیری دارد.

- هزینه زیاد طراحی

- وجود مشکلاتی در نمایش در مرورگرهای متفاوت

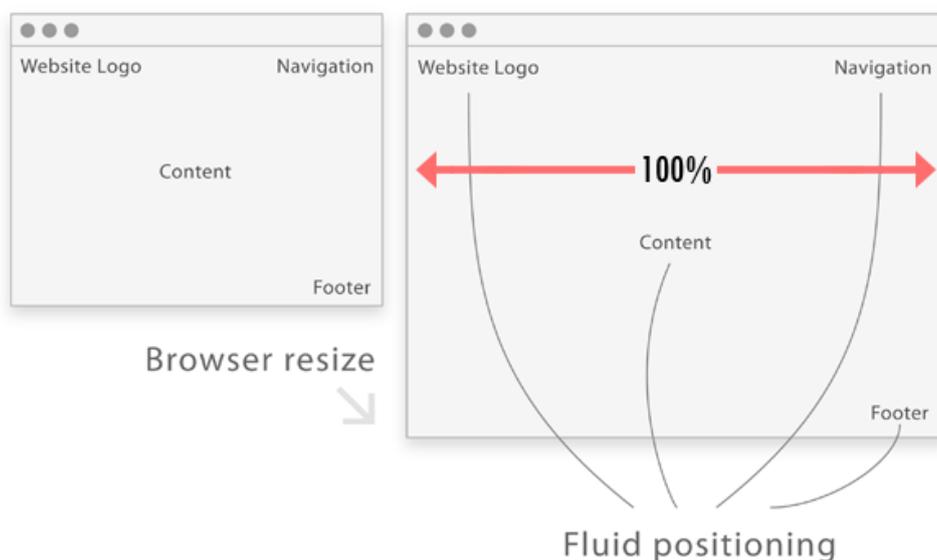
Fluid Layouts, A Journey

- مزیت استفاده از گریدهای شناور این است که می‌توانیم از `max-width` استفاده کنیم و به این ترتیب در صفحات عریض‌تر، صفحه براساس محاسباتی قابل نمایش است.

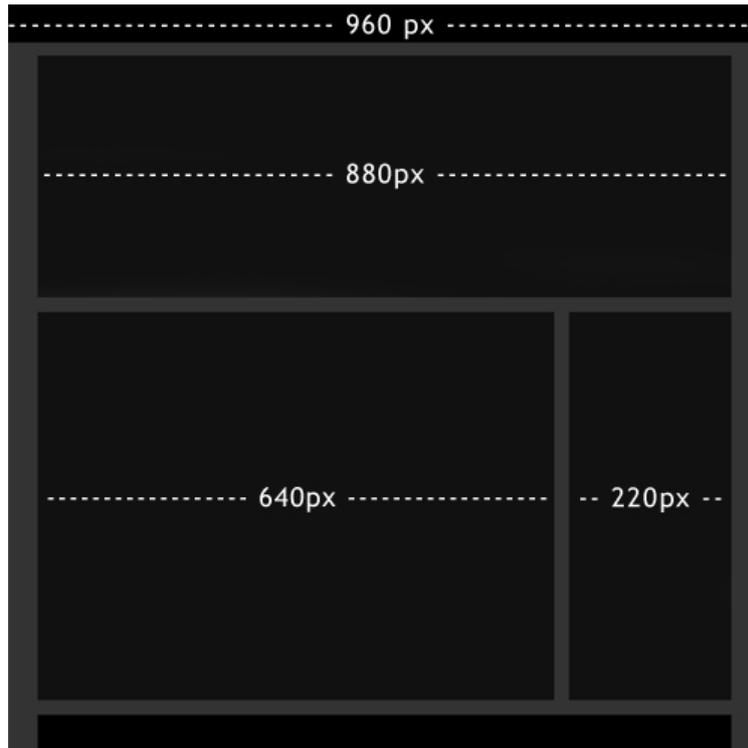


Fluid Layouts, A Journey

- ساختار شناور می‌تواند با تغییر اندازه صفحه نمایش منطبق شود. در این ساختار از اندازه‌های نسبی (، %، `em` و..) برای تعیین اندازه استفاده می‌شود.



ایجاد یک fluid layout



ایجاد یک fluid layout

TARGET / CONTEXT = RESULT

Don't round off the decimals

#container / base font size
960 / 16

THE RATIO REVOLUTION WILL NOT BE TELEVISED

A BLOG ENTRY.

Anyone else tired of Helvetica?

I mean it: I've got your *Neue Haas Grotesk* right here, buddy. Along with some lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

Yeah, you heard me.

Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

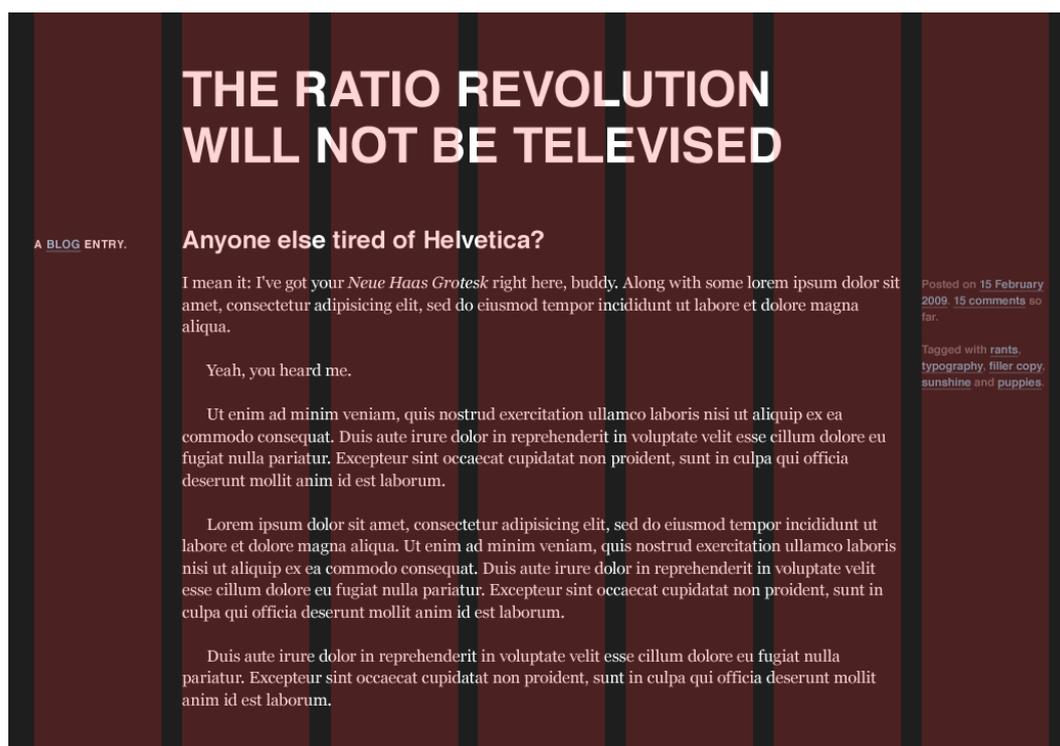
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

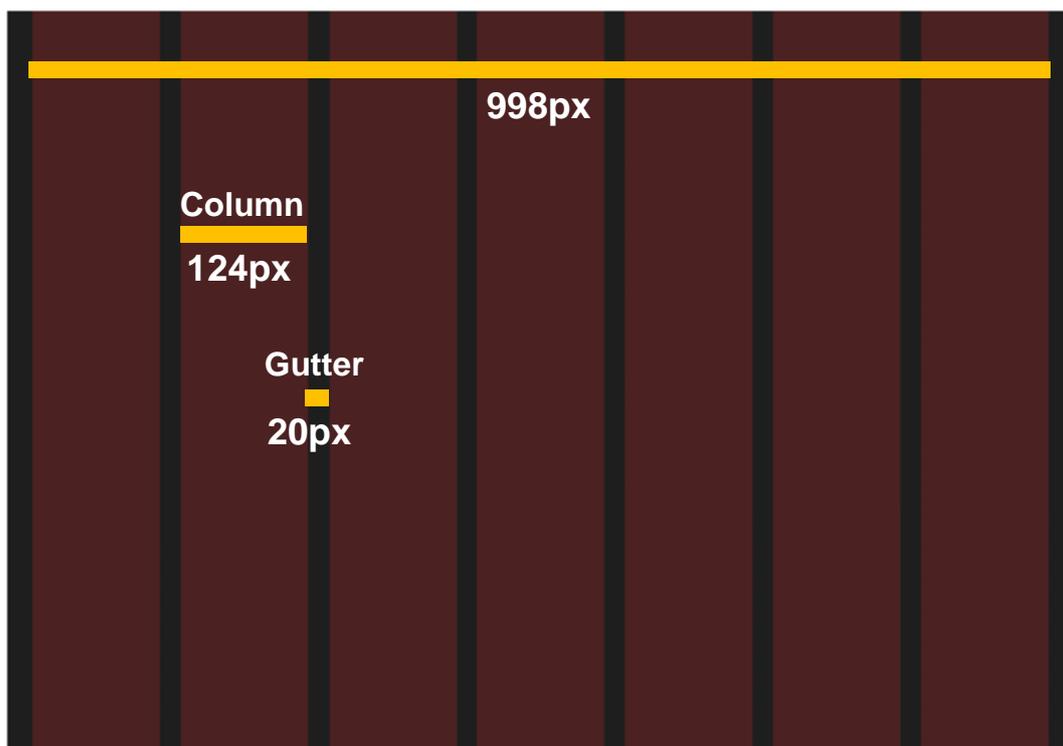
Posted on 15 February 2009. 15 comments so far.

Tagged with [rants](#), [typography](#), [filler copy](#), [sunshine](#) and [puppies](#).

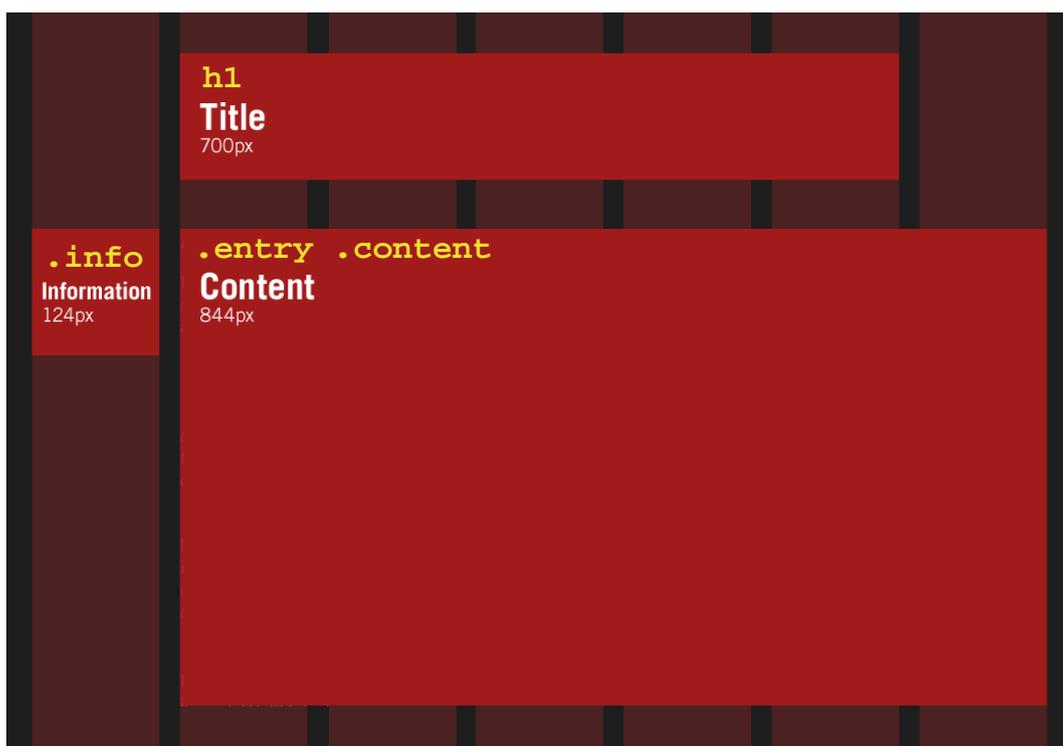
- این صفحه را می‌توان به صورت گرید با ۷ ستون ۱۲۴ پیکسلی در نظر گرفت که مابین ستون‌ها شیارهایی با عرض ۲۰ پیکسل قرار گرفته (عرض کلی ۹۹۸).



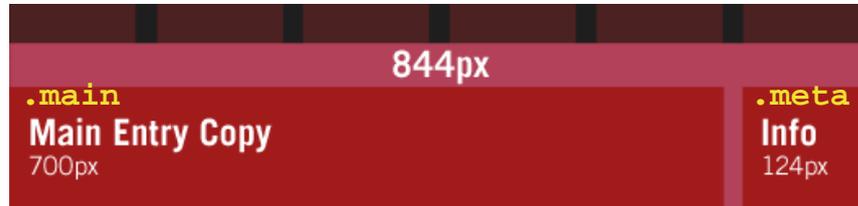
ایجاد یک fluid layout



ایجاد یک fluid layout



ایجاد یک fluid layout



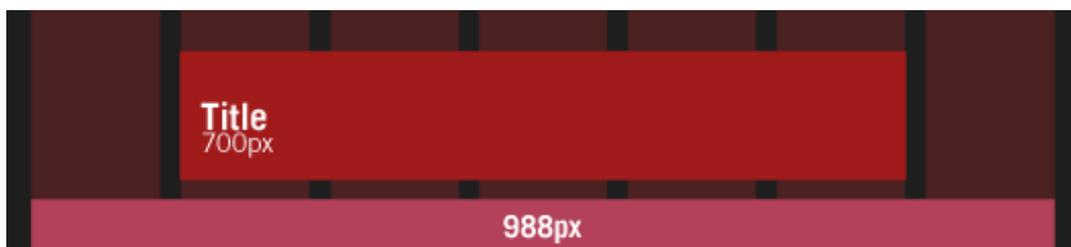
ایجاد یک fluid layout

```
<div id="page">
  <h1>The Ratio Revolution Will Not Be Televised</h1>
  <div class="entry">
    <h2>Anyone else tired of Helvetica?</h2>
    <h3 class="info">A <a href="#">Blog</a> Entry:</h3>
    <div class="content">
      <div class="main">
        <p>Main content goes here. Lorem ipsum etc.</p>
      </div>
      <div class="meta">
        <p>Posted on etc., etc.</p>
      </div>
    </div>
  </div>
</div>
```

ایجاد یک fluid layout

```
#page {  
  margin: 40px auto;  
  padding: 0 1em;  
  max-width: 61.75%;      /* 988px / 16px = 61.75% */  
}
```

ایجاد یک fluid layout



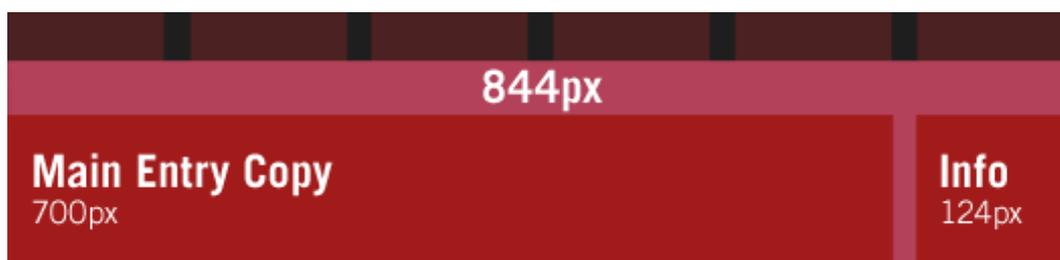
```
h1 {  
  margin-left: 14.575%; /* 144px / 988px = 0.14575 */  
  width: 70.85%;      /* 700px / 988px = 0.7085 */  
}
```

ایجاد یک fluid layout



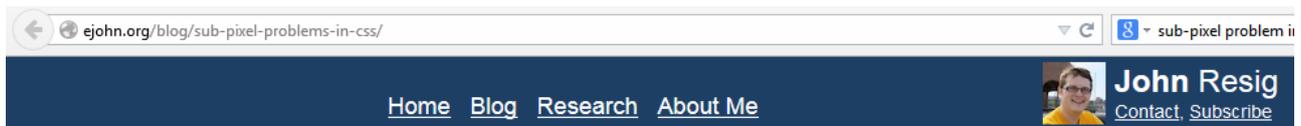
```
.entry h2,  
.entry .content {  
  float: right;  
  width: 85.425%; /* 844px / 988px = 0.85425 */  
}  
  
.entry .info {  
  float: left;  
  width: 12.551%; /* 124px / 988px = 0.12551 */  
}
```

ایجاد یک fluid layout



```
.entry .main {  
  float: left;  
  width: 82.938%; /* 700px / 844px = 0.82938 */  
}  
  
.entry .meta {  
  float: right;  
  width: 14.692%; /* 124px / 844px = 0.14692 */  
}
```

Cross Browser?



Sub-Pixel Problems in CSS

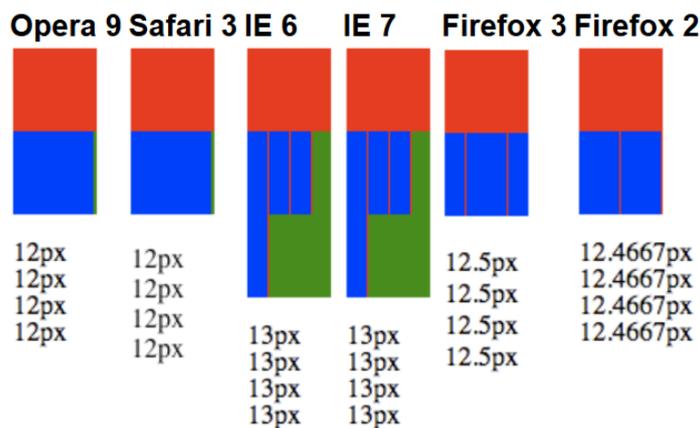
Something that jumped at me, recently, was a rendering dilemma that browsers have to encounter, and gracefully handle, on a day-by-day basis with little, to no, standardization.

Take the following page [for example](#). You have 4 floated divs, each with a width of 25%, contained within a parent div of width 50px. Here's the question: **How wide are each of the divs?**

The problem lies in the fact that each div should be, approximately, 12.5px wide and since technology isn't at a level where we can start rendering at the sub-pixel level we tend to have to round off the number. The problem then becomes: Which way do you round the number? Up, down, or a mixture of the two? I think the results will surprise you, as they did me.

Cross Browser?

- عرض المان‌های آبی ۲۵٪ است و عرض المان پدر ۵۰ پیکسل
 - عرض هریک از المان‌های آبی باید ۱۲.۵ پیکسل شود.
 - مرورگرها نمی‌توانند نیم پیکسل را نمایش دهند.
 - فقط IE عدد را به سمت بالا رند می‌کند:



Cross Browser?

■ یک راه حل ساده:

– عرض ستون را برای IE یک پیکسل کمتر در نظر بگیرید و بعد نسبت را محاسبه کنید!

Grid, Grid, Grid

■ صفحه بندی از زمان نوشتن روی پایپروس

– استفاده از گرید در چاپ و وب

■ چرا؟

– به صفحه ساختار می دهد

– کنترل حرکت چشم کاربر در پیمایش صفحه

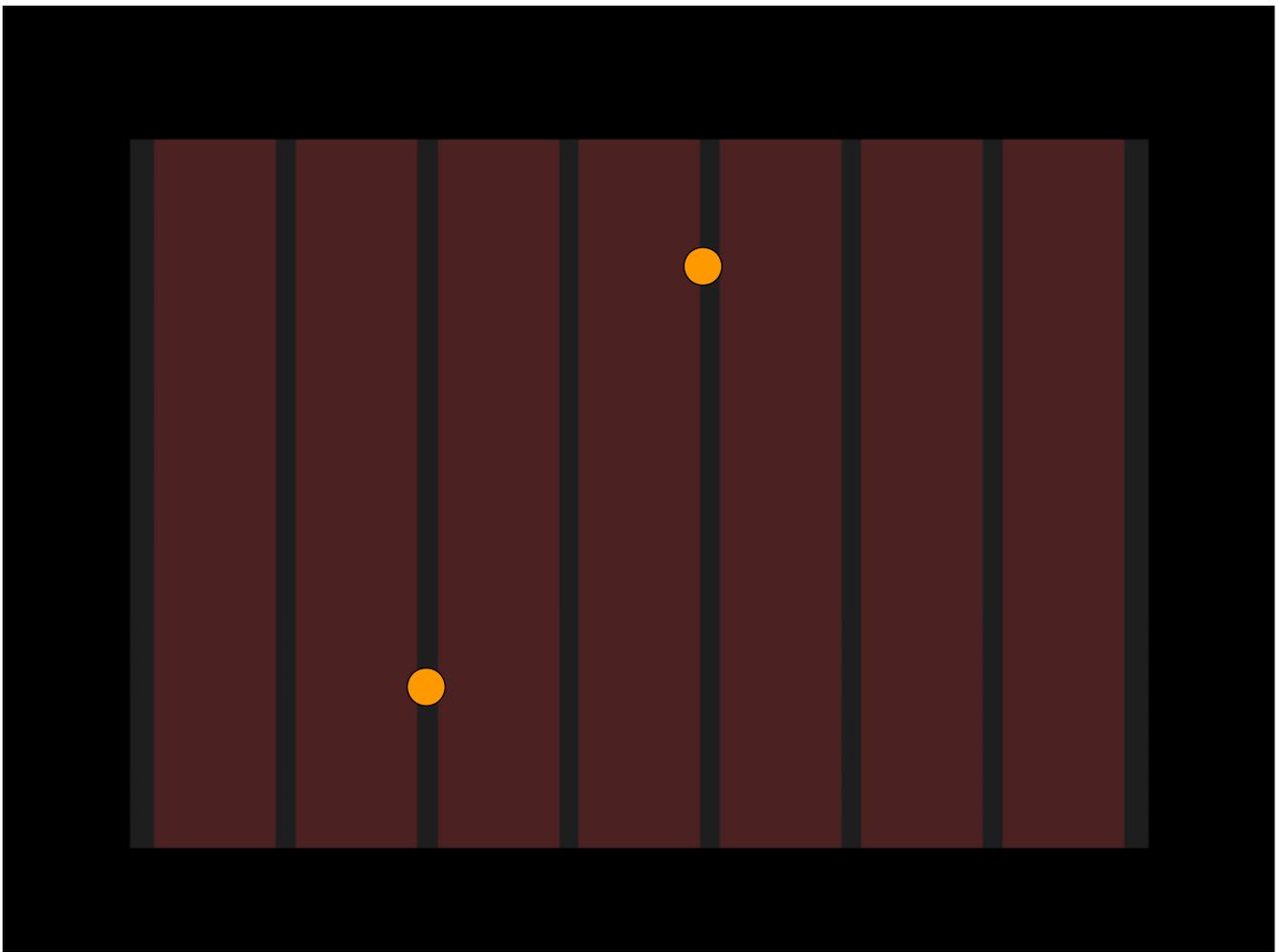
– ایجاد margin و negative space بین المان های درون صفحه

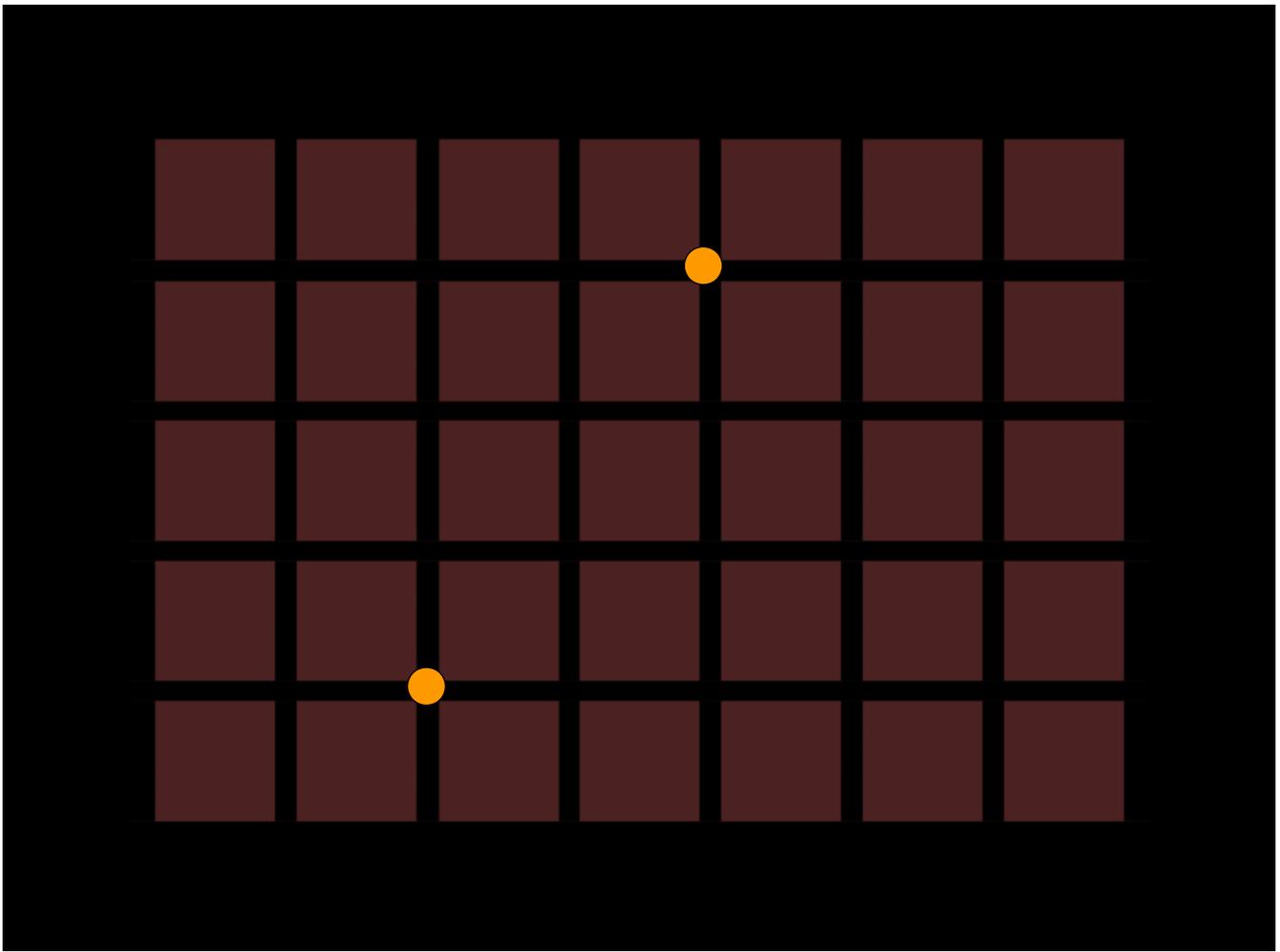
– هدایت کاربر بر اساس جریان (flow) صفحه

– گرید نمایش داده نمی شود و پشت صحنه کار

Dot to Design







Looking Room



حذف یک ستون از سمت چپ و اضافه کردن
یک ستون به سمت راست
ایجاد حس حرکت

Grid, Grid, Grid

- گریدها همانند المان‌های وب که از **Box Model** پیروی می‌کنند، شامل وجه‌های **Box Model** هستند و از این جهت به سادگی قابل ترکیب شده با هم هستند.

طراحی شناور به تنهایی؟

- طراحی شناور نمی‌تواند به تنهایی یک طرح واکنش‌گرا - با خصوصیات استاندارد - را بسازد.
- با استفاده از یک ساختار شناور شما فقط می‌توانید یک وب‌سایت انطباق‌پذیر با صفحات نمایش متفاوت داشته باشید. این کافی نیست.

Media Types

- یکی از پرکاربردترین استفاده‌ها از **media types**، طراحی نسخه پرینت به کمک **CSS** است.

```
<link rel="stylesheet" type="text/css"
      href="core.css" media="screen" />
```

```
<link rel="stylesheet" type="text/css"
      href="print.css" media="print" />
```

Media Types

- با استفاده از **media types** می‌توانید نسخه‌های متفاوتی از یک سند سی اس اس را به صورت اختصاصی، برای دستگاه‌های متفاوت، به صفحه وب اضافه کنید.
 - all, braille, embossed, handheld, print, projection, screen, speech, tty, tv

Media Query

■ مدیا کوئری این توانایی را به طراحان وب می‌دهد تا علاوه بر نوع دستگاه، ویژگی‌های فیزیکی دیگری نیز از دستگاه را مورد هدف قرار داده و شرطی سازی انجام دهد.

■ با کمک **Media Query** می‌توانید استایل‌های خود را برای بسیاری از خصوصیات فیزیکی دستگاه شرطی سازی نمائید.

- width, height, device-width, device-height, orientation, aspect-ratio, device-aspect-ratio, color, color-index, monochrome, resolution, scan, grid

چگونه از Media Query استفاده کنیم؟

■ استفاده به شکل **external** و **internal**

- وارد کردن در یک **style sheet** و وارد کردن به عنوان یک **style sheet** مجزا

نحوه استفاده internal یا داخلی

- اگر شرط برقرار باشد و اندازه دستگاہ بزرگتر از 501px شود، رنگ پس زمینه به خاکستری تغییر پیدا خواهد کرد.

```
body {  
    background: blue;  
}  
  
@media screen and (min-width:501px) {  
    body {  
        background: gray;  
    }  
}
```

نحوه استفاده internal یا داخلی

- اگر شرط برقرار باشد و اندازه دستگاہ کوچکتر از 500px شود، رنگ پس زمینه به آبی تغییر پیدا خواهد کرد.

```
body {  
    background: gray;  
}  
  
@media screen and (max-width:500px) {  
    body {  
        background: blue;  
    }  
}
```

نحوه استفاده internal یا داخلی

■ تعیین یک بازه از عرض

– اگر شرط برقرار باشد و اندازه دستگاه بزرگتر از 500px و کمتر از 699px شود، رنگ پس زمینه به آبی تغییر پیدا خواهد کرد.

```
body {
  background: gray;
}

@media screen and (min-width:500px) and (max-width:699px) {
  body {
    background: blue;
  }
}
```

نحوه استفاده external یا خارجی

■ استفاده از link و @import برای وارد نمودن استایل خارجی

```
<link href="narrow.css" rel="stylesheet"
      media="only screen and (max-width:500px)">
```

OR

```
@import url(narrow.css) only screen and (max-width:500px);
```

Embedded vs External media queries

Embedded خصوصیات منفی

بارگذاری حجم زیادی از فایل در صورت پشتیبانی یا عدم پشتیبانی از این تکنولوژی برای تمام مرورگرها
استفاده از جاوااسکریپت برای هماهنگ سازی با نسخه های ۸ و پایین تر اینترنت اکسپلورر
سخت تر بودن کنترل و سازماندهی در صورت گسترش استایلها

External خصوصیات مثبت

حجم کم فایل استایل اصلی در صورت عدم پشتیبانی از این تکنولوژی
توانایی خصوصی سازی برای مرورگرهای اینترنت اکسپلورر به وسیله کامنت های شرطی
آسان تر بودن کنترل و سازماندهی در صورت گسترش استایلها

Embedded خصوصیات مثبت

HTTP request های کمتر
بیشتر جلوی چشم است، بنابراین سخت تر فراموش می شود

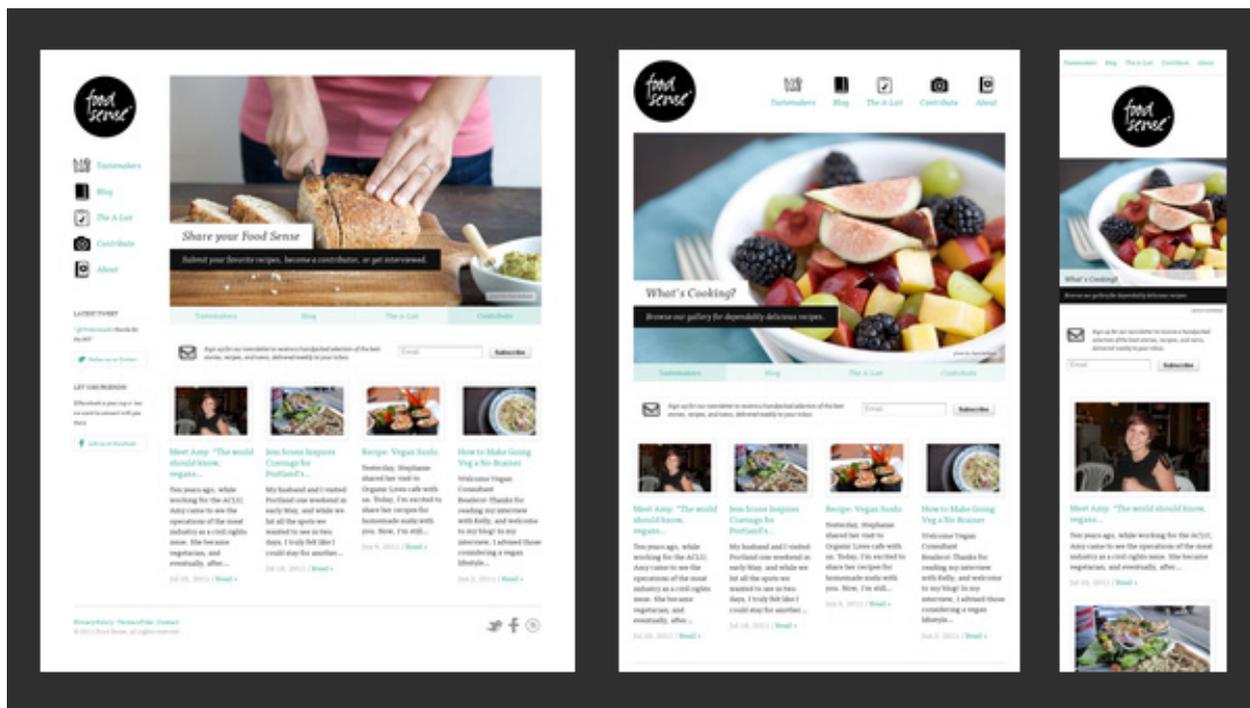
External خصوصیات منفی

HTTP request های بیشتر
کمتر جلوی چشم است، بنابراین بیشتر فراموش می شود

مثالی از طراحی واکنش گرا

- نمایش صفحه وب در شکل هایی مناسب با اندازه صفحه (در اینجا ۳ شکل برای اندازه های ۱.۲۴، ۵۰۰ و کوچکتر)

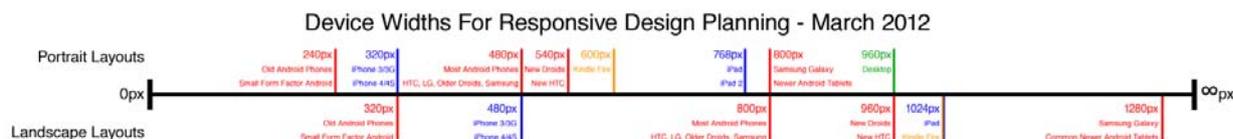




نقاط شکست (breakpoints)

- نقاطی که قصد داریم طراحی و نحوه چینش محتوا در آن و پس از آنرا تغییر دهیم.
- نوع طراحی وب سایت عامل تعیین کننده‌ای در انتخاب نقاط بحرانی است.
- ممکن است نقاط بحرانی با روش آزمون و خطا تعیین شوند.
- بیشترین رزولوشن‌های مورد استفاده در صفحات نمایش:
 - اندازه‌های زیر نشان دهنده عرض حداقل (min-widths) هستند:
 - 320px, 480px, 600px, 768px, 900px, 1200px

نقاط شکست (breakpoints)



http://en.wikipedia.org/wiki/List_of_displays_by_pixel_density

مشکل تفاوت تفکیک پذیری پیکلسی دیوایس‌ها علاوه بر ابعاد

- ممکن است ابعاد دیوایس کوچک باشد اما تعداد پیکسل‌های بیشتری داشته باشیم (تفکیک پذیری بالاتر)
- A Laptop
 - 13.3"
 - 1280 * 800 PX
 - 72 PPI
- A Tablet
 - 5"
 - 1920 * 1080 PX
 - 441 PPI

تصاویر انعطاف پذیر (Flexible Image)

- تصاویر بخش عمده‌ای از محتوای وب سایت را تشکیل می‌دهند.
- در طراحی شناور یکی از مهمترین چالش‌ها ، تغییر اندازه تصویر همراه با تغییر اندازه صفحه نمایش است.
- این چالش در هنگام استفاده از ویدئو و فلش نیز وجود خواهد داشت!
- اگر اندازه صفحه نمایش کوچکتر از اندازه طبیعی یا تعیین شده این عناصر شود، مشکل شروع خواهد شد!
- هیچ کس یک تصویر بسیار بزرگ یا یک ویدئوی بزرگ را در یک صفحه نمایش کوچک نمی‌پسندد!
- مشکل تفکیک‌پذیری‌های متفاوت را نیز در نظر بگیرید.

تشخیص تفکیک‌پذیری با Media Query

```
@media screen and (-webkit-min-device-pixel-ratio: 2), (min-resolution: 192dpi) {  
  /* Retina stuff here */  
}
```

<http://css-tricks.com/snippets/css/retina-display-media-query/>

حل مشکل تفکیک‌پذیری‌های متفاوت

■ استفاده از viewport

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

- با مقدار دهی **device-width** به تگ **viewport** به مرورگر می‌گوییم عرض دستگاه را با عرض صفحه نمایش یکی کن مثلا اگر عرض دستگاه ۳۲۰ پیکسل است، به جای آن که مقدار پیش فرض ۹۸۰ پیکسل را به عرض صفحه اختصاص دهد همان مقدار را به عرض صفحه اختصاص بدهد.

حل مشکل تفکیک‌پذیری‌های متفاوت

■ استفاده از viewport

— باعث کاهش کیفیت تصاویر با ابعاد یکسان در دیوایس‌های با تفکیک‌پذیری بالا می‌شود.

- استفاده از تصاویر بزرگتر
- استفاده از font icon
- استفاده از تصاویر برداری (SVG).

پشتیبانی از Media Query

CSS3 Media Queries - Recommendation

Global user stats*:
 Support: 86.92%
 Partial support: 0.01%
 Total: 86.93%

Method of applying styles based on media information. Includes things like page and device dimensions

Resources: [WebPlatform Docs](#) | [IE demo page with information](#) | [Media Queries tutorial](#) | [Polyfill for IE](#)

	IE	Firefox	Chrome	Safari	Opera	iOS Safari	Opera Mini	Android Browser	BlackBerry Browser	Opera Mobile	Chrome for Android	Firefox for Android	IE Mobile
29 versions back			4.0										
28 versions back		2.0	5.0										
27 versions back		3.0	6.0										
26 versions back		3.5	7.0										
25 versions back		3.6	8.0										
24 versions back		4.0	9.0										
23 versions back		5.0	10.0										
22 versions back		6.0	11.0										
21 versions back		7.0	12.0										
20 versions back		8.0	13.0										
19 versions back		9.0	14.0										
18 versions back		10.0	15.0										
17 versions back		11.0	16.0										
16 versions back		12.0	17.0		9.0								
15 versions back		13.0	18.0		9.5-9.6								
14 versions back		14.0	19.0		10.0-10.1								
13 versions back		15.0	20.0		10.5								
12 versions back		16.0	21.0		10.6								
11 versions back		17.0	22.0		11.0								
10 versions back		18.0	23.0		11.1								
9 versions back		19.0	24.0		11.5								
8 versions back		20.0	25.0		11.6								
7 versions back		21.0	26.0	3.1	12.0			2.1					
6 versions back	5.5	22.0	27.0	3.2	12.1			2.2		10.0			
5 versions back	6.0	23.0	28.0	4.0	15.0	3.2		2.3		11.0			
4 versions back	7.0	24.0	29.0	5.0	16.0	4.0-4.1		3.0		11.1			
3 versions back	8.0	25.0	30.0	5.1	17.0	4.2-4.3		4.0		11.5			
2 versions back	9.0	26.0	31.0	6.0	18.0	5.0-5.1		4.1		12.0			
Previous version	10.0	27.0	32.0	6.1	19.0	6.0-6.1		4.2-4.3 7.0	12.1				
Current	11.0	28.0	33.0	7.0	20.0	7.0	5.0-7.0 4.4	10.0	16.0	33.0	26.0	10.0	
Near future		29.0	34.0		21.0								
Farther future		30.0	35.0		22.0								
3 versions ahead		31.0	36.0										

Note: Incomplete support by older webkit browsers refers to only acknowledging different media rules on page reload

<http://caniuse.com/css-mediaqueries>

پشتیبانی مرورگرهای موبایل

iOS		Android		Chrome		Opera			BlackBerry				Nokia			UC			Puffin	Nin tendo	Dol phin	Tizen	One	IE		Firefox	
6	7	2	4	18	30	Mini	Mob 12	Mob 16	6	7	PE	10	Xpress	Mee Go	Anna	Belle	Mini	8	9					9	10	OS	And
por-trait	lay-out	phys-ical	p+p	yes	layout	yes	both	yes	both	avail	yes	both	buggy	phys-ical	physical			p+p	p+p	phys-ical	yes						

<http://www.quirksmode.org/mobile/tableViewport.html>

طراحی واکنش‌گرا و موتورهای جستجو

- یکسان بودن آدرس اینترنتی به کاربران شما کمک خواهد کرد تا راحت‌تر آدرس وب‌سایت شما را به خاطر بسپارند، راحت‌تر صفحات شما را به اشتراک بگذارند و یا لینک کنند.
- در صورت استفاده از نسخه واکنش‌گرا نیازی به بررسی مجدد نسخه موبایل وب‌سایت توسط ربات‌های جستجوگر نخواهد بود.
- استفاده از نسخه واکنش‌گرا کمک خواهد کرد از درجات کسب شده توسط وب‌سایت، در دنیای دستگاه‌های هوشمند نیز استفاده شود.

چالش‌های طراحی واکنش‌گرا

- قدرت پردازش و حجم حافظه
- پهنای باند
- رویدادهای (events) صفحات لمسی
- غیر فعال شدن تصاویر توسط کاربران
- پشتیبانی از تکنولوژی‌های وب jQuery، CSS3 و...

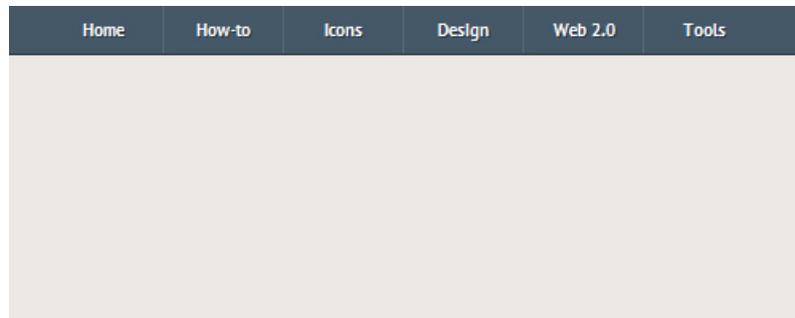
MOBILE FIRST

- با توجه به اهمیت تجارت و بازرایابی توسط موبایل ها و دیوایس های متنوع و متفاوت بحث های بسیار جدی در مورد شروع طراحی برای نسخه های موبایل مطرح می شود
- این نوع از تفکر کاملا به نوع تجارت وابسته است

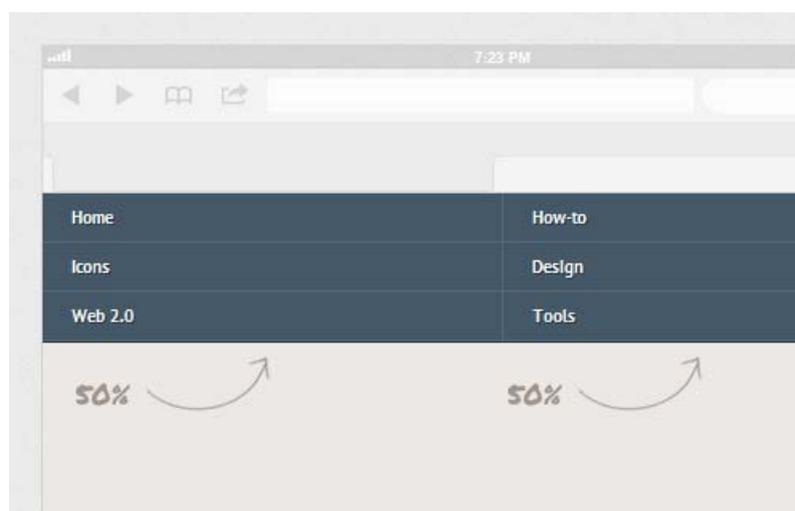
هماهنگ سازی با مرورگرهای قدیمی

- با استفاده از پلاگین های ساخته شده با استفاده از جاوااسکریپت امکان پشتیبانی از مدیا کوئری توسط مرورگرهای قدیمی امکان پذیر است.
 - Respond.js
 - CSS3-Mediaqueries.js
 - Adapt.js

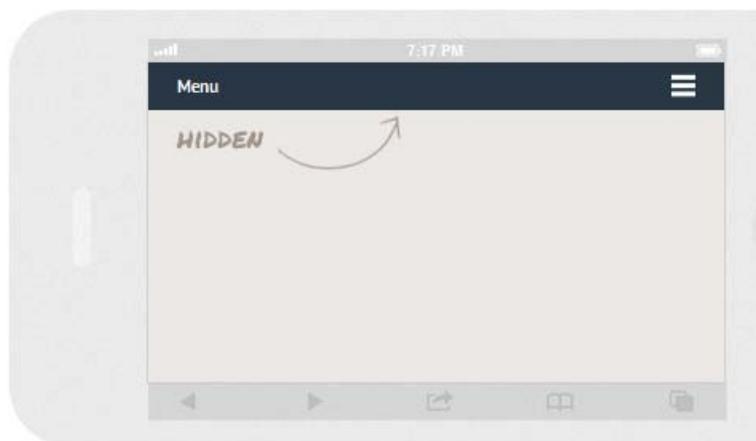
Responsive Navigation



Responsive Navigation



Responsive Navigation



<http://www.hongkiat.com/blog/responsive-web-nav/>

منابع

- Ethan Marcotte: Fluid Grids (2009)
 - <http://alistapart.com/article/fluidgrids>
- Ethan Marcotte: Responsive Web Design (2010)
 - <http://alistapart.com/article/responsive-web-design>
- Ethan Marcotte: Fluid Images (2011)
 - <http://alistapart.com/article/fluid-images>

■ طراحی قالب واکنش‌گرا و استفاده از گرید فریم‌ورک‌های موجود جهت طراحی واکنش‌گرا، شاهو طوفانی، ارائه در اولین همایش تخصصی واکنش‌گرا آذر ماه سال ۱۳۹۲ – تهران

■ طراحی **Responsive** یا واکنش‌گرا، امیر سروری، ارائه در اولین همایش استانداردهای وب و بستر وب باز تیرماه سال ۱۳۹۱ – تهران