

# One Page Everywhere

*Fluid, Responsive Design with Semantic.gs*

# The Semantic Grid System

- Grid System
- Fluid or Fixed
  - Responsive
    - Semantic
- Sass or LESS

# Grid Systems



[Download](#) - CSS, sketch paper, and templates for: Acom, Fireworks, Flash, InDesign, GIMP, Inkscape, Illustrator, OmniGraffle, Photoshop, QuarkXPress, Visio, Exp Design. Repository at [GitHub](#).

## 960 GRID SYSTEM

ADS BY FUSION

Big ol' **DOWNLOAD** button :)

 **INTERVIEW ABOUT 960.gs**

**VIEW SLIDES ABOUT THE 960 GRID SYSTEM**

**ADAPT.JS - ADAPTIVE CSS**

**CUSTOM CSS GENERATOR**

**GRID OVERLAY BOOKMARK**

### Essence

The 960 Grid System is an effort to streamline web development workflow by providing commonly used dimensions, based on a width of 960 pixels.

### Dimensions

The 12-column grid is divided into portions that are 60 pixels wide. The 16-column grid consists of 40 pixel increments. Each column has 10 pixels of

### Purpose

The premise of the system is ideally suited to rapid prototyping, but it would work equally well when integrated into a production environment.

# Grid System Fixed Size

The screenshot displays the 960 Grid System website. At the top left is the Twitter logo. The main header features the '960 GRID SYSTEM' logo in a large, stylized font. Below the logo, there is a 'Download' link and a list of supported applications: Acom, Fireworks, Flash, InDesign, GIMP, Inkscape, Illustrator, OmniGraffle, Photoshop, QuarkXPress, Visio, and Exp Design. A prominent 'Big ol' DOWNLOAD button :)' is centered on the page. Below this are several navigation links: 'the changelog INTERVIEW ABOUT 960.gs', 'VIEW SLIDES ABOUT THE 960 GRID SYSTEM', 'ADAPT.JS - ADAPTIVE CSS', 'CUSTOM CSS GENERATOR', and 'GRID OVERLAY BOOKMARK'. The main content area is divided into three columns: 'Essence', 'Dimensions', and 'Purpose'. The 'Essence' column explains the goal of streamlining web development. The 'Dimensions' column details the 12-column and 16-column grid structures. The 'Purpose' column describes the system's suitability for prototyping and production. Below these columns are sections for 'More Columns' and 'Source Order'. At the bottom, two example wireframes are shown: 'Keynote Wireframe Toolkit' and 'Brand Rich Media', each with a 'SHOW GRID' button.

Twitter

960  
GRID SYSTEM

Download - CSS, sketch paper, and templates for: Acom, Fireworks, Flash, InDesign, GIMP, Inkscape, Illustrator, OmniGraffle, Photoshop, QuarkXPress, Visio, Exp Design. Repository at [GitHub](#).

Big ol' DOWNLOAD button :)

the changelog INTERVIEW ABOUT 960.gs

VIEW SLIDES ABOUT THE 960 GRID SYSTEM

ADAPT.JS - ADAPTIVE CSS

CUSTOM CSS GENERATOR

GRID OVERLAY BOOKMARK

**Essence**

The 960 Grid System is an effort to streamline web development workflow by providing commonly used dimensions, based on a width of 960 pixels. There are two variants: 12 and 16 columns, which can be used separately or in tandem. [Read more.](#)

**Dimensions**

The 12-column grid is divided into portions that are 60 pixels wide. The 16-column grid consists of 40 pixel increments. Each column has 10 pixels of margin on the left and right, which create 20 pixel wide gutters between columns. [View demo.](#)

**Purpose**

The premise of the system is ideally suited to rapid prototyping, but it would work equally well when integrated into a production environment. There are printable sketch sheets, design layouts, and a CSS file that have identical measurements.

**More Columns**

For those more comfortable designing on a 24-column grid, an alternative version is also included. It consists of columns 30 pixels wide, with 10 pixel gutters, and a 5 pixel buffer on each side of the container. This keeps text from touching browser chrome — helpful for devices like the iPhone, where a lower-case "l" or "1" might be easily missed. [View demo.](#)

**Source Order**

By utilizing the *push\_XX* and *pull\_XX* classes, elements can be rearranged, independent of the order in which they appear in the markup. This allows you to keep more pertinent info higher in the HTML, without sacrificing precision in your page layout. For instance, view the source code of this page to see how the *H1* tag has been re-positioned.

Keynote Wireframe Toolkit — 12 col [SHOW GRID](#)

Brand Rich Media — 12 col [SHOW GRID](#)

Wireframe Toolkit  
For Keynote  
version 1.1

Examples Buzz Media FAQ Tips \$12 BUY NOW

BR BRAND RICH MEDIA  
DESIGN, TOOLS & TEMPLATES

PROJECTS CONTACT

# Grid System Fixed Size



# Semantic.gs: Fixed or Fluid

```
// Specify the number of columns and set column and gutter widths
```

```
$columns: 12;
```

```
$column-width: 60;
```

```
$gutter-width: 20;
```

```
// Remove the definition below for a pixel-based layout
```

```
$total-width: 100%;
```

# Grid System Clutter

```
<body>
  <div class="container_12">

    <h1 class="grid_4 push_4">
      960 Grid System
    </h1>
    <!-- end .grid_4.push_4 -->

    <p id="description" class="grid_4 pull_4">
      ...
    </p>
    <!-- end #description.grid_4.pull_4 -->
```

# Semantic.gs: Layout in Stylesheets

```
<body>
  <div id="main-content">
    <header id="banner">
      <h1> ... </h1>
    </header>
    <section id="history">
      <h2>History</h2>
      <p> ... </p>
    </section>
```

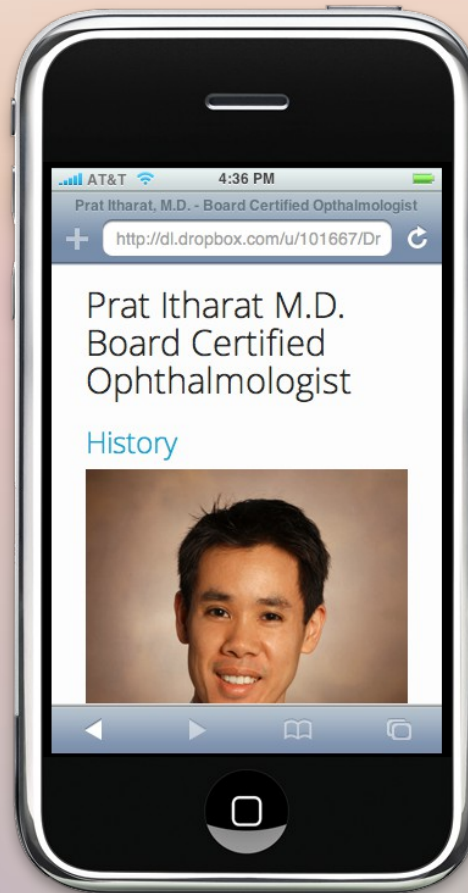
```
#banner {
  @include column(12);
  padding-top: 3em; }

#history {
  @include column(6); }

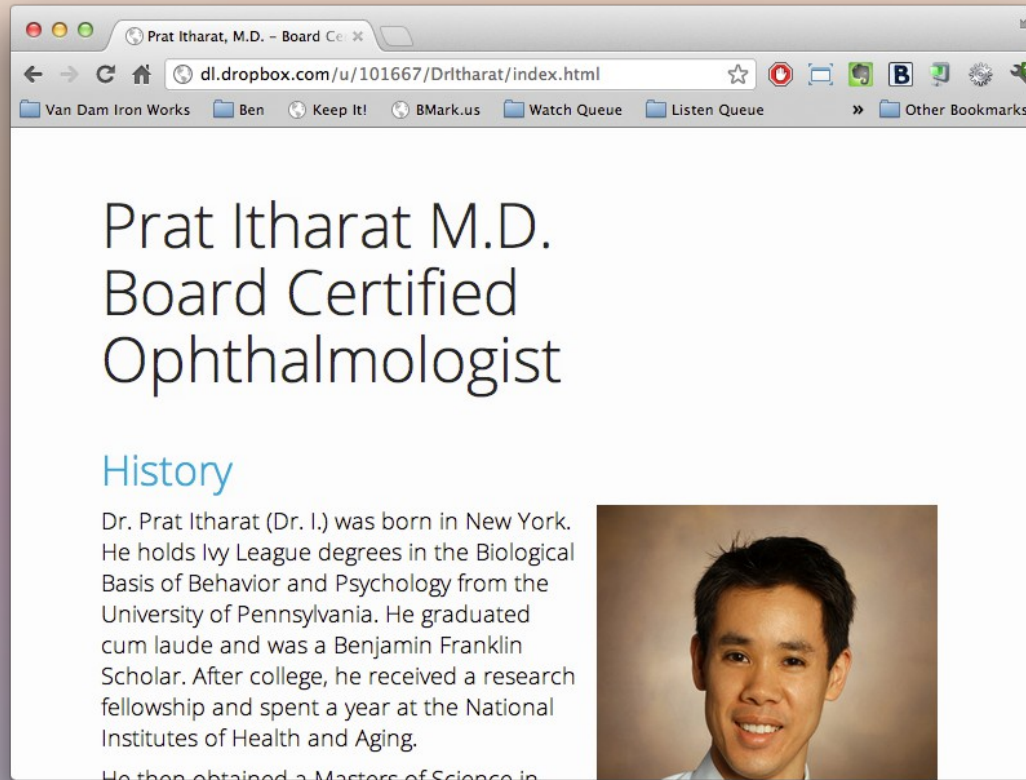
#contact {
  @include column(6); }
```



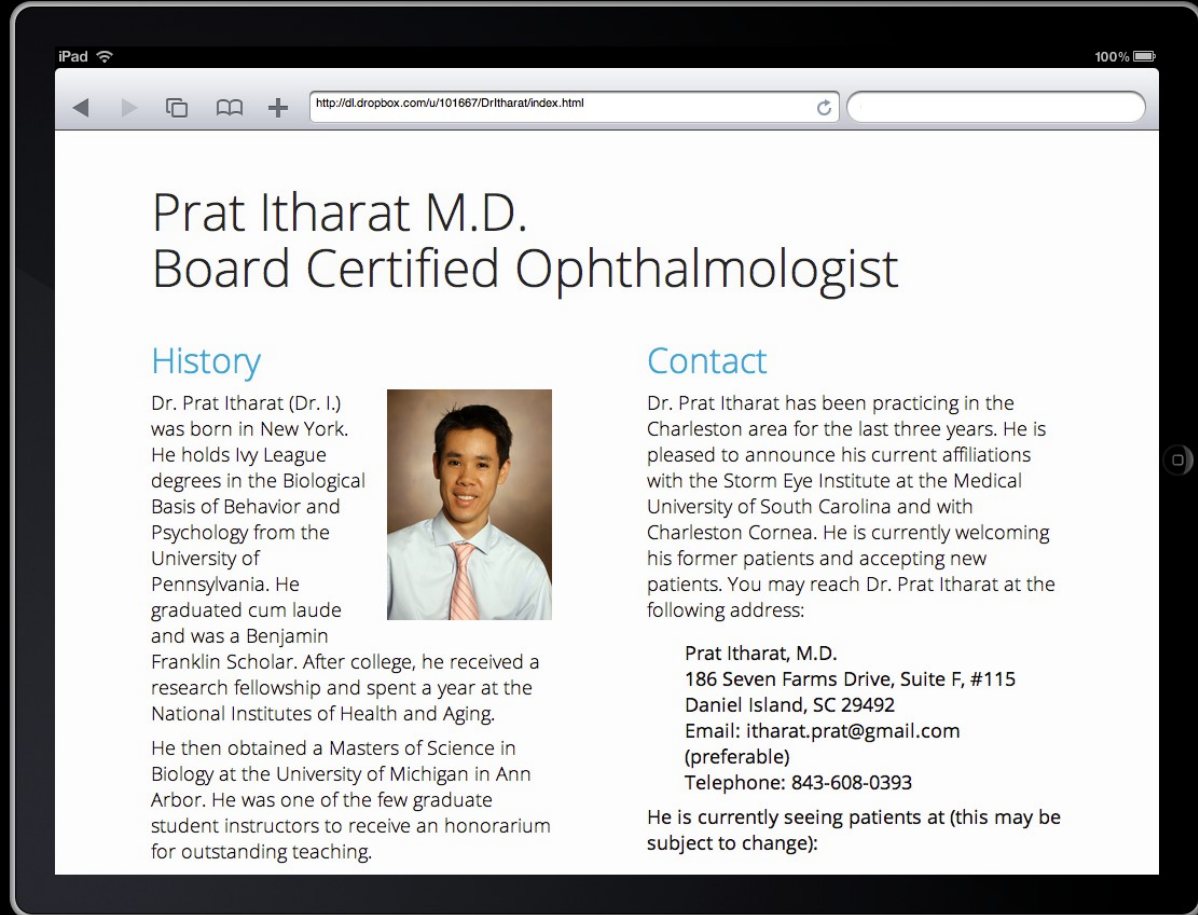
# Semantic.gs: Responsive (iPhone)



# Semantic.gs: Responsive (800x600)



# Semantic.gs: Responsive (iPad)




iPad 100%

<http://dl.dropbox.com/u/101667/DrItharat/index.html>

## Prat Itharat M.D. Board Certified Ophthalmologist

### History

Dr. Prat Itharat (Dr. I.) was born in New York. He holds Ivy League degrees in the Biological Basis of Behavior and Psychology from the University of Pennsylvania. He graduated cum laude and was a Benjamin Franklin Scholar. After college, he received a research fellowship and spent a year at the National Institutes of Health and Aging.



He then obtained a Masters of Science in Biology at the University of Michigan in Ann Arbor. He was one of the few graduate student instructors to receive an honorarium for outstanding teaching.

### Contact

Dr. Prat Itharat has been practicing in the Charleston area for the last three years. He is pleased to announce his current affiliations with the Storm Eye Institute at the Medical University of South Carolina and with Charleston Cornea. He is currently welcoming his former patients and accepting new patients. You may reach Dr. Prat Itharat at the following address:

**Prat Itharat, M.D.**  
186 Seven Farms Drive, Suite F, #115  
Daniel Island, SC 29492  
Email: [itharat.prat@gmail.com](mailto:itharat.prat@gmail.com)  
(preferable)  
Telephone: 843-608-0393

He is currently seeing patients at (this may be subject to change):

# Semantic.gs: Responsive (1920x1080)

Prat Itharat, M.D. - Board C

dl.dropbox.com/u/101667/DrItharat/index.html

## Prat Itharat M.D. Board Certified Ophthalmologist

### History

Dr. Prat Itharat (Dr. I) was born in New York. He holds Ivy League degrees in the Biological Basis of Behavior and Psychology from the University of Pennsylvania. He graduated cum laude and was a Benjamin Franklin Scholar. After college, he received a research fellowship and spent a year at the National Institutes of Health and Aging.


He then obtained a Masters of Science in Biology at the University of Michigan in Ann Arbor. He was one of the few graduate student instructors to receive an honorarium for outstanding teaching.

Dr. Itharat then returned to New York to complete his medical school training at the State University of New York - Upstate (Syracuse). He graduated in the top 10% of his medical school class and was elected to the Alpha Omega Alpha honors society. After internship, he finished a three year ophthalmology residency at the prestigious Vanderbilt University in Nashville TN, training with some of the most renowned physicians in ophthalmology. At Vanderbilt, Dr. Itharat was elected as Chief Resident by his peers and faculty. He taught medical students and residents and served on the residency admission committee.

Dr. Itharat is board certified by the American Board of Ophthalmology (the nation's oldest medical specialty certifying board), having scored in the top 10% of ophthalmologists.

He is trained in the latest, small incision, sutureless cataract surgery. He practices comprehensive ophthalmology, including but not limited to cataracts, dry eyes, glaucoma, diabetes, macular degeneration, ocular trauma and injuries.

A great believer in education, Dr. Itharat currently serves as a clinical instructor at the Storm Eye Institute/Medical University of South Carolina. He teaches the next generation of ophthalmologists medical and surgical knowledge and techniques. His dedication to teaching



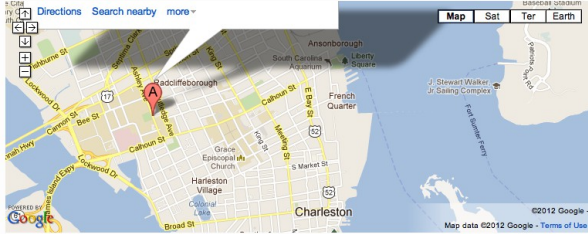
### Contact

Dr. Prat Itharat has been practicing in the Charleston area for the last three years. He is pleased to announce his current affiliations with the Storm Eye Institute at the Medical University of South Carolina and with Charleston Cornea. He is currently welcoming his former patients and accepting new patients. You may reach Dr. Prat Itharat at the following address:

Prat Itharat, M.D.  
186 Seven Farms Drive, Suite F, #115  
Daniel Island, SC 29492  
Email: itharat.prat@gmail.com (preferable)  
Telephone: 843-608-0393

He is currently seeing patients at (this may be subject to change):

Storm Eye Institute  
167 Ashley Avenue  
Charleston, SC 29425  
843-792-8100  
(note: wait times at this clinic may be lengthy)



# Project Walk-through


- File locations
- semantic\_gs files

# HTML Walk-through

- Single stylesheet
- IE fixes
- Semantic elements
- Few layout divs

# SCSS Walkthrough

- Imports
- Variables
- Columns and Gutters
- *@include* Columns
- Few sizes in pixels, all % and em



**Crabby old man kitteh**

**tellz u boring storeez**



# Fixes for Old Internet Explorers

## **Html5shiv**

<https://github.com/aFarkas/html5shiv>

## **Respond.js**

<https://github.com/scottjehl/Respond>

# Things to Watch Out For

- Right and left padding and margins on grid elements can be finicky. Best to just wrap them with a div.
- Image sizing: small screens get large images and have to scale the images down.
- Can't really re-order sections.

# Choosing Media Query Breakpoints

1. Start big and go to small
2. Shrink window until something looks funky
3. Create a breakpoint, fix what looks funky
4. Repeat from #2
5. No need to worry about specific devices!

# Credits

- The Semantic Grid System <http://semantic.gs/>
- iPhone screenshots simulated using iPhony
  - <http://www.marketcircle.com/iphoney/>
- iPad screenshots simulated using iPad Peek
  - <http://ipadpeek.com/>
- Some slides from <http://icanhascheezburger.com/>

# Some Useful Utilities

- Chrome Window Resizer Extension
  - <https://chrome.google.com/webstore/detail/kkelicaakdanhinjdeammilcgefongh>
- HTML5 Shiv
  - <https://github.com/aFarkas/html5shiv>
- Respond.js
  - <https://github.com/scottjehl/Respond>

# Contact Me

- Meatspace: Ben Rousch
- Email: [brousch@gmail.com](mailto:brousch@gmail.com)
- Twitter: [@brousch](https://twitter.com/brousch)
- Google+: <https://plus.google.com/102663141609195877664/>

# One Page Everywhere

*Fluid, Responsive Design with Semantic.gs*

Greeting

## The Semantic Grid System

- Grid System
- Fluid or Fixed
  - Responsive
  - Semantic
- Sass or LESS

We're going to transform this one page, static website into a fluid, responsive website using The Semantic Grid System.

Semantic.gs is:

12 column, 960 pixel grid system

Which can be fluid or fixed

It is responsive

It is semantic

It uses Sass or LESS

We'll talk about a few of these terms as we go on.



## Grid Systems



CSS grid systems have been around for a while now. I've used Blueprint in the past, but there are many others available.

How many of you have used a grid system at some point?

Because keeping things lined up in CSS is hard, the idea is that the grid system gives you a framework of columns and gutters, the space in between columns, that let you line things up nicely on your page.

## Grid System Fixed Size

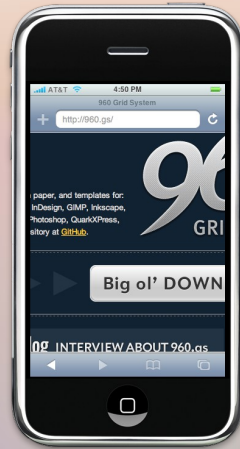


But there are a couple of problems with grid systems:

1. They are usually a fixed width. Often 960 pixels. This does not lend itself well to responsive design, which needs to adapt to many different screen sizes.

Here is the 960 grid on a 1080p monitor. The empty areas on the side are pretty excessive.

## Grid System Fixed Size



Here is the same site on an iPhone.

I don't think I need to say anything.

## Semantic.gs: Fixed or Fluid

```
// Specify the number of columns and set column and gutter widths
$columns: 12;
$column-width: 60;
$gutter-width: 20;

// Remove the definition below for a pixel-based layout
$total-width: 100%;
```

The Semantic Grid System lets you use a fixed size if you want to, but it can also provide a fluid grid for you.

Fluid means that the sizes of elements change for different screen sizes. You use percentages and ems to lay things out instead of pixels.

\* Compare fixed Drl to fluid \*

We'll be using the fluid grid today. And I think once you go fluid you won't want to go back to fixed.

## Grid System Clutter

```
<body>
  <div class="container_12">

    <h1 class="grid_4 push_4">
      960 Grid System
    </h1>
    <!-- end .grid_4.push_4 -->

    <p id="description" class="grid_4 pull_4">
      ...
    </p>
    <!-- end #description.grid_4.pull_4 -->
```

Another problem with typical grid systems is they clutter up your HTML with extra divs and span or grid properties.

Here's some of the HTML from that 960Grid page we just saw.

This is layout stuff. It really doesn't belong in your HTML. It belongs in your stylesheets. And that's where The Semantic Grid System puts it.

## Semantic.gs: Layout in Stylesheets

```
<body>
  <div id="main-content">
    <header id="banner">
      <h1> ... </h1>
    </header>
    <section id="history">
      <h2>History</h2>
      <p> ... </p>
    </section>
  </div>
</body>
```

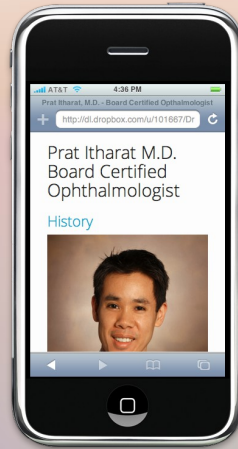
```
#banner {
  @include column(12);
  padding-top: 3em; }

#history {
  @include column(6); }

#contact {
  @include column(6); }
```

This is a snippet from the final HTML and stylesheet we'll create today. You can see the column information is in the CSS. There's no sign of it in the HTML.

## Semantic.gs: Responsive (iPhone)



Responsive refers to a website's ability to adapt to different resolutions, screen sizes, and devices.

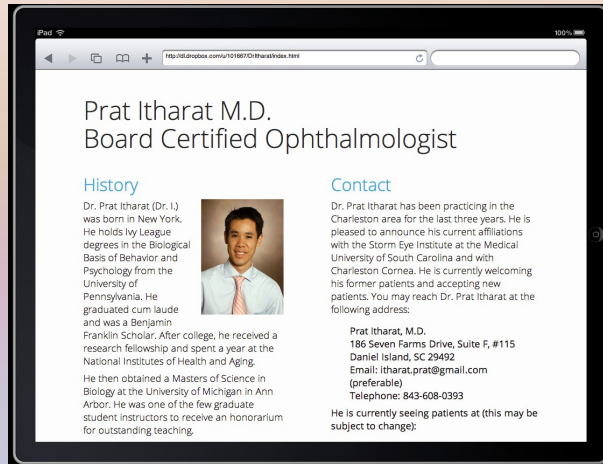
## Semantic.gs: Responsive (800x600)



A typical grid system works well for a typical sized desktop or laptop screen, but it doesn't work well for very large screens or smartphones.

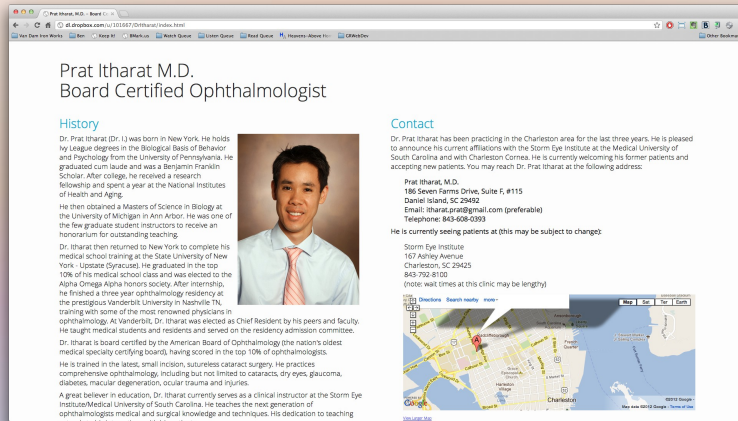


## Semantic.gs: Responsive (iPad)



This is partly because the grid is defined in the HTML, which means you can't change it on the fly without resorting to Javascript or different versions of the page for different sized screens..

## Semantic.gs: Responsive (1920x1080)



These are screenshots of our finished website on an iPhone, at 800x600 resolution, on an iPad, and at 1920x1080 resolution.

Let's see them again.

## Project Walk-through

- File locations
- semantic\_gs files

## HTML Walk-through

- Single stylesheet
- IE fixes
- Semantic elements
- Few layout divs

## SCSS Walkthrough

- Imports
- Variables
- Columns and Gutters
- @include Columns
- Few sizes in pixels, all % and em



Media queries don't work on these crusty old browsers.  
So do we just say forget those guys? Of course not.  
Much of Dr I's audience is likely to be old people still  
running Windows XP.

## Fixes for Old Internet Explorers

### **Html5shiv**

<https://github.com/aFarkas/html5shiv>

### **Respond.js**

<https://github.com/scottjehl/Respond>

Luckily I found an easy fix.

## Things to Watch Out For

- Right and left padding and margins on grid elements can be finicky. Best to just wrap them with a div.
- Image sizing: small screens get large images and have to scale the images down.
- Can't really re-order sections.

There a few things you need to watch out for as you design your site using The Semantic Grid System.



## Choosing Media Query Breakpoints

1. Start big and go to small
2. Shrink window until something looks funky
3. Create a breakpoint, fix what looks funky
4. Repeat from #2
5. No need to worry about specific devices!

With a pre-existing site, like to start with the largest version of the website and shrink it. Other people like to go mobile first and make media query breakpoints as the site gets bigger. Try them both, see what you like.

## Credits

- The Semantic Grid System <http://semantic.gs/>
- iPhone screenshots simulated using iPhony
  - <http://www.marketcircle.com/iphoney/>
- iPad screenshots simulated using iPad Peek
  - <http://ipadpeek.com/>
- Some slides from <http://icanhascheezburger.com/>

## Some Useful Utilities

- **Chrome Window Resizer Extension**
  - <https://chrome.google.com/webstore/detail/kkelicaakdanhinjdeammilcgefonfh>
- **HTML5 Shiv**
  - <https://github.com/aFarkas/html5shiv>
- **Respond.js**
  - <https://github.com/scottjehl/Respond>

## Contact Me

- Meatspace: Ben Rousch
- Email: [brousch@gmail.com](mailto:brousch@gmail.com)
- Twitter: [@brousch](https://twitter.com/brousch)
- Google+: <https://plus.google.com/102663141609195877664/>