



Custom Responsive Theme Workshop

WordCamp NEO 2016

#wcneo | #wcthemedev

About Me

- Web Developer for 7+ years. Including 2+ years of WordPress experience
- **Owner of Infinite Reality LLC, a web and graphic design firm, since since 2007 and have been working in higher education for the last three years.**
- **Recently began teaching a Web Design 2 class at Lorain County Community College with emphasis on WordPress**
- Built 6+ WordPress themes with theme specific plugins
- Migrated and designed 6 complete sites with Wordpress



Topics

- Why Build a Theme?
- Available Responsive Themes
- Basic Framework For a Responsive Theme
- Theme Building Tips
- Code Demo
- Taking The Responsive Theme One Step Further

Why Build a Theme?

- Unable to find a theme that can either be modified or have a child theme created for it.
- Work with a number of sites, tend to use the same type of theme code over and over.
- Rapid build out of new sites.
- Just to see if you can!

Theme Building Tips

- Wireframe
- Plan features
- Configure a Dev environment
- CSS or SASS?
- codex.wordpress.org
- www.wphierarchy.com – Cool Resource
- Before using the theme or submitting to the repo, make sure the header of the CSS stylesheet is filled out.

Theme Building Tips

- Theme Check – Plugin from Repository
- Theme Unit Test – codex.wordpress.org/Theme_Unit_Test
- Functions - Theme vs Plugin
 - If the function is for the content, put it in a plugin
 - If it doesn't matter where the function is placed, put it in a plugin.
 - Very rarely is there a good reason to add a function to the theme.
- Keep your dev environment updated, core and plugins!

Available Responsive Themes

- Free Themes from WordPress.org Respository
 - Select Responsive Layout from feature filter – 1,601 themes.
- Premium Themes with responsive layouts
 - StudioPress – Genesis
 - Woo Themes – Canvas & Storefront
 - Wpmudev –UpFront

Basic Frameworks For a Responsive Theme

- CSS Frameworks
 - Bootstrap
 - Foundation
- WordPress Starter Theme
 - Underscores

How To Build Your Responsive Theme

- Download a fresh copy of
 - Underscores - <http://www.underscores.me>
 - Foundation 5 - <http://foundation.zurb.com>
 - **Optional:**
Zurb Foundation Icons: <http://zurb.com/playground/foundation-icon-fonts-3>
- Unzip underscores into a directory for your theme.
 - Create a foundation directory inside of your theme directory, and unzip the foundation download here.
- Unzip and Copy Foundations folders into underscore theme folder
- Optional - Configure Gulp (but highly recommended)

How To Build Your Responsive Theme

Foundation Initialization File

- Create a New File in your favorite code editor
- Paste the following code:

```
jQuery (document) .ready (function ($) {  
  $(document) .foundation ();  
});
```

- Save the file as “ **foundation.js** ” in your **theme folder**

How To Build Your Responsive Theme

- Open **Functions.php** , located in your Underscores theme, in your favorite code editor
- Look for a Function Titled: **function themename_scripts**
example: **dabrattoli_theme_2016_scripts**
- Click right above it and hit return a few times
- Then Add This Code
function wcneo_foundation_scripts() {

}
- Then **Save** the file

How To Build Your Responsive Theme

Theme Functions.php

- Find the stylesheet enqueues.
 - Search for “Enqueue scripts and styles”
- Adding Foundation CSS : Add the following two lines to your function.

```
wp_enqueue_style( 'foundation-app',  
get_stylesheet_directory_uri() . '/foundation/css/app.css' );
```

```
wp_enqueue_style( 'foundation-css',  
get_stylesheet_directory_uri() . '/foundation/css/  
foundation.css' );
```

```
wp_enqueue_style( 'foundation-icons',  
get_template_directory_uri() . '/foundation/foundation_icons/  
foundation-icons.css' );
```

How To Build Your Responsive Theme

Theme Functions.php

- Add a call for your custom styles to your function.

```
wp_enqueue_style( 'demo-custom-style',  
get_stylesheet_directory_uri() . '/  
custom.css', array(), '1' );
```

- Add the calls for the foundation js files to your function.

```
wp_enqueue_script( 'foundation-js',  
get_template_directory_uri() . '/foundation/js/  
foundation.min.js', array( 'jquery' ), '1',  
true );
```

```
wp_enqueue_script( 'foundation-modernizr-js',  
get_template_directory_uri() . '/foundation/js/  
vendor/modernizr.js', array( 'jquery' ), '1',  
true );
```

How To Build Your Responsive Theme

Theme Functions.php

- Add a call for the foundation init, to your function.

```
wp_enqueue_script( 'foundation-init-js',  
get_template_directory_uri() . '/foundation.js',  
array( 'jquery' ), '1', true );
```

Adding the Foundation Script and Stylesheets to WordPress

Theme Functions.php

```
function wcneo_foundation_scripts() {  
    wp_enqueue_style( 'foundation-app', get_stylesheet_directory_uri() . '/  
foundation/css/app.css' );  
  
    wp_enqueue_style( 'foundation', get_stylesheet_directory_uri() . '/  
foundation/css/foundation.css' );  
  
    wp_enqueue_style( 'demo-custom-style',  
get_stylesheet_directory_uri() . '/custom.css', array(), '1' );  
  
    wp_enqueue_script( 'foundation-js', get_template_directory_uri() . '/  
foundation/js/foundation.min.js', array( 'jquery' ), '1', true );  
  
    wp_enqueue_script( 'foundation-modernizr- js',  
get_template_directory_uri() . '/foundation/js/vendor/modernizr.js',  
array( 'jquery' ), '1', true );  
  
    wp_enqueue_script( 'foundation-init-js', get_template_directory_uri() .  
'/foundation.js', array( 'jquery' ), '1', true );  
}  
  
add_action( 'wp_enqueue_scripts', 'wcneo_foundation_scripts' );
```

How To Build Your Responsive Theme

- You now have Foundation plugged into Underscores!
- Build your design out!



Code Demo



Leveraging Responsive Theme As A Framework

Why use a framework?

- Ease of development – Framework can drastically reduce the development time and make things easier when working on new projects in the future.
- Built-in functionality – Gives you built in functionality of built-in widgets, and additional functionality including responsive capabilities. Allows you to re-use functionality code that you use on regular basis in one central location.
- Code Quality – It allows you to create a coding standards for your own work, along when working with others.
- Upgrades – The ability to upgrade several sites without losing any styling functionality of child themes.
- Reusability
- Build a Child Theme on top of the framework

Building a Child Theme

- There are two popular ways of creating a Child Theme

1) The first being take a copy of your parent theme, and then just remove what you don't need.

Pros:

- You get only the code you want

Cons

- Renaming all the Parent Functions to represent the Child Theme

2) The Second Being create and download a new theme from underscores.me

Pros:

- You get a clean slate with everything formatted for the new child theme

Cons

- You can get extra code and files you don't need or want

Building a Child Theme

- Start Building your Child Theme
- Files can be opened from the Parent Framework, saved to the child theme folder and then modified.
- Only the files you need to change, need to be copied over.

```
wp_enqueue_style( 'demo-style',  
get_stylesheet_directory_uri().'/style.css' );  
}  
  
add_action( 'wp_enqueue_scripts', 'demo_scripts' );
```

Building a Child Theme

- **Comment Out Functionality Functions From Parent's Theme Functions.php**

```
/** * Implement the Custom Header feature. */  
require get_stylesheet_directory() . '/inc/custom-header.php';  
  
/** * Custom template tags for this theme. */  
require get_stylesheet_directory() . '/inc/template-tags.php';  
  
/** * Custom functions that act independently of the theme  
templates. */  
require get_stylesheet_directory() . '/inc/extras.php';  
  
/** * Customizer additions. */  
require get_stylesheet_directory() . '/inc/customizer.php';  
  
** * Load Jetpack compatibility file. */  
require get_stylesheet_directory() . '/inc/jetpack.php';
```

Building a Child Theme

Adding Parent Theme Styles Foundations Support to Child's Theme Functions.php

```
/* Add Parent Theme Styles */

wp_enqueue_style( 'demo-framework-style',
get_template_directory_uri() . '/style.css' );

/* ----- Add Foundation Support From Parent Theme ----- */

/* Add Foundation CSS */

wp_enqueue_style( 'foundation-app',
get_template_directory_uri() . '/foundation/css/app.css' );

wp_enqueue_style( 'foundation', get_template_directory_uri() .
'/foundation/css/foundation.css' );

wp_enqueue_style( 'foundation-icons',
get_template_directory_uri() . '/foundation/foundation_icons/
foundation-icons.css' );
```

Building a Child Theme

Adding Parent Theme Scripts to Child's Theme Functions.php. Including the responsive theme framework (Foundation) init script

```
/* Add Foundation JS */

wp_enqueue_script( 'foundation-js', get_template_directory_uri() .
'/foundation/js/foundation.min.js', array( 'jquery' ), '1',
true );

wp_enqueue_script( 'foundation-modernizr-js',
get_template_directory_uri() . '/foundation/js/vendor/
modernizr.js', array( 'jquery' ), '1', true );

/* Foundation Init JS */

wp_enqueue_script( 'foundation-init-js',
get_template_directory_uri() . '/foundation.js',
array( 'jquery' ), '1', true );
```


Building a Child Theme

- Final Result of Child Theme code

```
function irllc_foundation_scripts() {  
    /* Add Foundation CSS */  
    wp_enqueue_style( 'foundation-app', get_template_directory_uri() . '/  
    foundation/css/app.css' );  
    wp_enqueue_style( 'foundation-css', get_template_directory_uri() . '/  
    foundation/css/foundation.min.css' );  
    wp_enqueue_style( 'foundation-icons', get_template_directory_uri() . '/  
    foundation/foundation_icons/foundation-icons.css' );  
    /* Add Foundation JS */  
    wp_enqueue_script( 'foundation-js', get_template_directory_uri() . '/  
    foundation/js/vendor/foundation.js', array( 'jquery' ), '1', true );  
    wp_enqueue_script(  
    'foundation-modernizr-    js', get_template_directory_uri() .    '/  
    foundation/js/vendor/what-input.js', array( 'jquery' ), '1', true );  
    wp_enqueue_script( 'foundation-init-js', get_template_directory_uri() .  
    '/foundation-init.js', array( 'jquery' ), '1', true );  
    }  
    add_action( 'wp_enqueue_scripts', 'irllc_foundation_scripts' );
```

Building a Child Theme

- Last Step is to modify the **style.css** file in your child theme.
- Open **style.css** from the child theme folder in your editor
- Find the **style header**: For example:

```
/*Theme Name: WCNEO  
Theme URI: http://underscores.me/  
Author: David Brattoli  
Author URI: http://www.dabrattoli.com  
Description: Responsive Theme for WCNEO  
Version: 1.0.0  
License: GNU General Public License v2 or  
later  
License URI: http://www.gnu.org/licenses/  
gpl-2.0.html  
Text Domain: wcneo */
```

Building a Child Theme

- Add a blank line between **Author and Version**
- Then **add**
Template: folder name of the framework
- Then **Save**



Code Demo

Questions?

David Brattoli

Web:

www.Infiniterealityllc.com

www.Dabrattoli.com

Twitter: [@DavidMBrattoli](https://twitter.com/DavidMBrattoli)

Email: davidbrattoli@yahoo.com

david.brattoli@infiniterealityllc.com

Github:

<https://github.com/dabrattoli>

Resources Mentioned

- underscores.me – Automattic Starter Theme
- foundation.zurb.com – Foundation5 CSS Responsive Framework
- foundation.zurb.com/docs – Foundation Documentation
- foundation.zurb.com/docs/components/kitchen_sink.html – Foundation Examples
- wphierarchy.com – Clickable Template Hierarchy
- wordpress.org/plugins/what-the-file – Plugin that display which template file is being used to display the content.

Resources Mentioned

- pxtoem.com – Pixels to EM convertor
- brackets.io – Multi-Platform Code Editor Environment
- gruntjs.com – Grunt (Command Line Automation Tool)

Code Examples

Sample files from this presentation can be found at:
<http://bit.ly/wceno-crtw-samples>

Slides Can be found at
<http://www.slideshare.net/DavidBrattoli/responsive-themeworkshop-wcneo2016>