Semantic UI for WordPress: Developer Edition

The Semantic UI WordPress starter/developer theme.

`build` `passing`　`Licenses` `CC-BY-SA 4.0 and MIT`

This project incorperates Semantic UI into a developer theme for WordPress. This project also includes some of my favorite techniques for creating fast, responsive, and easy-to-maintain themes for WordPress. *Please keep in mind this theme is meant to be developed for your specific application; and is not meant to be used "as-is."*

Download:

`release` `1.0.0-beta2`　`branch` `develop`

Features

Semantic UI
Responsive Design | **Google Web Fonts**
Image Optimization | **Theme Options Page** | **Woocommerce Support**
Font Awesome | **Webicons** | **LESS/SASS Support** | **Normalize CSS**
jQuery | **Gulp** | **Google Microdata** | **Unit Testing**
Highlight.js | **Keyboard Shortcuts**
Well Commented Code

Installation

Requirements

- PHP 5.4 or later
- WordPress 3.9.0 or later

Install Guide

1. Download the release version of the theme
2. Unzip to your `wp-content/themes` directory
3. Set the theme to "active" in your WordPress dashboard

Building From Source

To build from source you need to have **Node.js** installed and in your `$path` (**win/mac/unix**). You should also have **PHPUnit** and **Composer** installed and in your `$path` as well.

**Please note that when building from source, the `/dist` directory should be renamed and then put in your `/wp-content/themes` directory

Mac & Unix:

1. Download or clone the master branch
2. Open your command line and navigate to where you deployed the code
3. Run `npm install && sudo npm install -g gulp` enter your password and then wait for it to finish.
4. Run `gulp` to see a list of available tasks. Running `gulp build` will regenerate `/dist` from scratch.

Windows:

1. Download or clone the master branch
2. Open your command line and navigate to where you deployed the code
3. Run `npm install && npm install -g gulp` and then wait for it to finish.
4. Run `gulp` to see a list of available tasks. Running `gulp build` will regenerate `/dist` from scratch.

Usage

This is a developer theme designed to be developed for your specific application. The default state of this theme is meant to be plain & organized; while not being bias to any particular design. (thus everything defaults to black and white)

Designing Pages

In this developer theme there are 4 important parts to generate a page:

- Templates
- Includes
- Contents
- Assets

A template file generates the general layout of a page. They generally call the header and footer functions, get content files, and should call any includes the page will need.

A include file adds functionality to a page and should have no output unless a function or method is called in a template or content file. Include files are usually used for API classes and libararies, but can be used for any kind of functionality.

A content file generates a group of elements and often has functionality calls for specific elements, such as dynamic text and images. Content files typically output the most HTML.

Assets are typically static files that are commonly used. These files are usually images, stylesheets, fonts, and javascript files, and might also include other files that need to be precompiled before they can be used. (such as LESS and SASS/SCSS files)

Contributing

Contributing to Semantic UI

Visit **this page** to learn how to contribute to Semantic UI.

Contributing to This WordPress Theme

Contributing *via* Suggestions:
The best way to submit a suggestion is to open an issue on Github and prefix the title with `[Suggestion]`. Alternatively, you can email your suggestions to projectcleverweb (at) gmail (dot) com.

Contributing *via* Reporting Problems:
All problems must be reported via Github's **issue tracker**.

Contributing *via* Code:

1. Fork the repo on Github: **github.com/ProjectCleverWeb/Semantic-UI-WordPress**
2. Make your changes.
3. Send a pull request to have your changes reviewed.

License