**Important**

Node version: **10.16.2**

PHP version: **5.6.4**

**О фреймворке**

При вёрстке страниц встречают очень много разных и сильно отличающихся друг от друга дизайнов и получается так, что при каждом новом проекте применяется разный подход к работе. Идея заключается в том, чтобы создать некоторую систему, которая будет сохранять стиль вёрстки и будет иметься уже готовая минимальная система, которая упростит старт проекта.

Фреймворк предназначен только для одиночной работы и не может быть использован в командной разработке. Он будет упрощать вёрстку и экономить персональное время на фрилансе.

Основная концепция данной системы состоит в том, чтобы создать ядро, где будет содержаться минимальное количество стилей, которые используются в большинстве случаев вёрстки HTML.

Исходя из этих соображений, можно сказать, что вёрстка страниц будет означать достраивание этого ядра непосредственно в его код, то есть создание переменных уже не будет обязательным, если клиент заранее не попросит об этом.

Один из возможных причин создания новых переменных – это долгосрочный и расширяющийся проект, то есть клиент будет постоянно просить новые дополнения и часто изменять их. В таком случае необходимо создавать и группировать объекты с переменными, создавая возможность лёгкого их редактирования.

**Строение фреймворка**

Для упрощения вёрстки используется PHP. PHP папка состоит из двух папок exmaples и project. Examples - это примеры блоков и компонентов сайта. Project -это папка, где находятся непосредственно php файлы проекта, которые генерируются как html. Стрелками обозначены выходы каждой папки:

src/php/project -> dist

src/php/examples -> dist/examples

**Применение**

Фреймворк может быть использован для вёрскти обычных **HTML** страниц и для прямой натяжки на **Wordpress**:

***HTML Вёрстка***

* Запускаем **npm** install, после чего запускаем **gulp**.
* Если при запуске **gulp** будет выдавать ошибку, то заменить папку **gateway** в папке **node\_modules** на старую версию **gateway**, которая находится на компьютере по адрессу: *D:\ARTYOM\WEB\FRAMEWORKS\sass-framework\fixed node modules for php\*
* Поменять префикс классов можно при помощи поиска и замены (search and replace).

***Wordpress Натяжка***

* Скопировать папку **sass-framework,** в корневую директорию **Wordpress.**
* Установить значение "**html\_mode = false**, **wp\_mode = true**" после чего установить путь **css, js, images, fonts** и т.д. папок. Например **"../wp-content/theme-name/.."**.
* Запускаем **npm** install, после чего запускаем **gulp**.
* Заходим в папку /**src/sass/abstracts/theme-variables.scss** и меняет **$class-prefix** на другое значение. После чего при помощи search replace меняем префиксы во всех **php** темплейтах. Удаляем **php** папку шаблонизатора и **examples.**
* Заменяем слово **text\_domain** во всех **php** файлах на соответсвенное короткое название темы.
* В экспортном файле **acf** находятся минимальные кастомные поля. Их можно ипортировать при необходимости.

**Структура SCSS папки**

Необходимо придерживаться именно той структуре папки, которая уже создана. Все новые части или блоки сайта нужно заполнять в папку **"layout",** но при большой необходимости можно создать дополнительную папку **"blocks".**

**Алгоритм разработки.**

* Делаем все необходимые установки, которые были приведены выше.
* Исследуем тщательно дизайн, делим страницы на блоки, ищем повторяющиеся части, или даже похожие.
* Настраиваем переменные типографии.
* Орпеделяем основные цвета сайта, создавая их соответственные переменные.
* Настраиваем контейнеры, секции, брейкпоинты и всё, что связано с размерами разметки.
* Начинаем вёрстку с шапки сайта, подгоняя его переменные под требуемый дизайн.

Вся вёрстка осуществляется при помощи php шаблонизатора, который позволит создавать циклы для быстрого заполнения и будет обеспечивать лёгкость в изменении html.