Проект по информатике на тему:

«Разработка web-сайтов с использованием языка разметки гипертекста HTML»

Выполнил:

Ученик 9а класса

Ильинов Павел

Оглавление

1. [Цели и задачи.](#Цели_и_задачи)
2. [Что использовал при работе над проектом?](#Что_использовал_при_работе_над_проект)
3. [Описание работы.](#Описание_работы)
   1. Создание и настройка проекта.
   2. Блок первый. Header.
   3. Блок второй. Navbar.
   4. Блок третий. About.
   5. Блок четвертый. Services.
   6. Блок пятый. Get in touch.
   7. Блок шестой. Signup.
   8. Блок седьмой. Footer.
4. Заключение.

Цели и задачи:

Занимаясь данным проектом, я поставил себе такие цели:

* Закрепить знание HTML
* Закрепить базовые знание селекторов в CSS
* Научиться писать простые скрипты на JS

Что использовал при работе над проектом?

Во время работы с проектом я использовал такие программы:

* Visual Studio Code – текстовый редактор, для редактирования кода
* Figma – графический редактор, для просмотра макета
* Google Chrome – браузер, для просмотра сайта

И такие инструменты:

* HTML, CSS, JS
* SASS – препроцессор CSS, для ускорения написания стилей
* Gulp.js – для сборки проекта
* Node.js – нужен для работы gulp

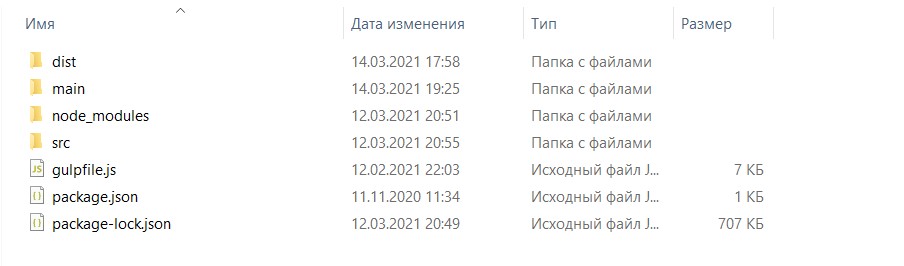
Описание работы.

Часть 1.

Создание и настройка проекта.

Самое главное во время разработки сайта, это правильно созданные папки. Когда все файла лежат в одном месте - это неудобно, и можно легко запутаться, особенно когда много файлов.

Сначала создаем папку с названием “Project\_for\_school” где будут храниться все наши файла (рис. 1).



Рисунок

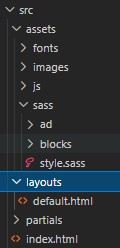
В нем я создал папки dist, main, src и файлы *gulpfile.js* и *package.json*.

Папка *node\_modules* и файл *packege-lock.json* были созданы gulp-ом.

В *main* папка со скриншотами, этот файл и файл с ссылками.

В *dist* уже собранный проект.

В *src* проект разделенный по блокам, в котором мы и будем писать.

В файлу *gulpfile.js* сценарий, который будет выполнять gulp а в *package.json* прописаны модули которые будет устанавливать node.js при работе.

В папке *src* лежат такие папки и файлы (рис. 2)

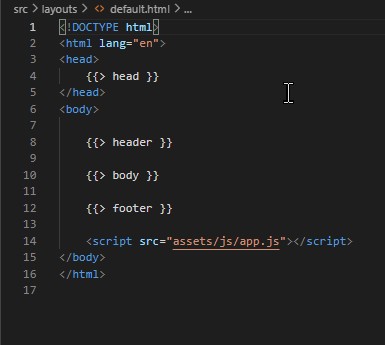
В *assets* папки с шрифтами, картинки, js-скрипты и папка с sass файлами.

Рисунок

В папке *ad* хранится файл *reset.sass* со сбросом стилей.

В папке *blocks* стили для отдельных блоков.

В файле *style.sass* все это стили импортируются.

** В файле *default.html* такой код (рис. 3).

Рисунок

Там я импортирую JS скрипты, head, header, body и footer.

Теперь рассмотрим код, который находится в файле *style.sass* (рис. 4). В первых строчках мы импортируем файлы из папки *ad* в последних стили блоков из папки *blocks*. Также задаем стиль блоку с классом *container*.

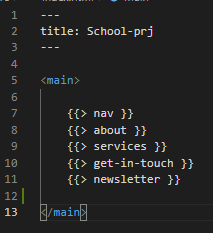


Рисунок 4

Рисунок 5

Так выгллядит файл *index.html* (рис. 5).

На 2 строчке мы называем проект, а в теге *main* импортируем все блоки.

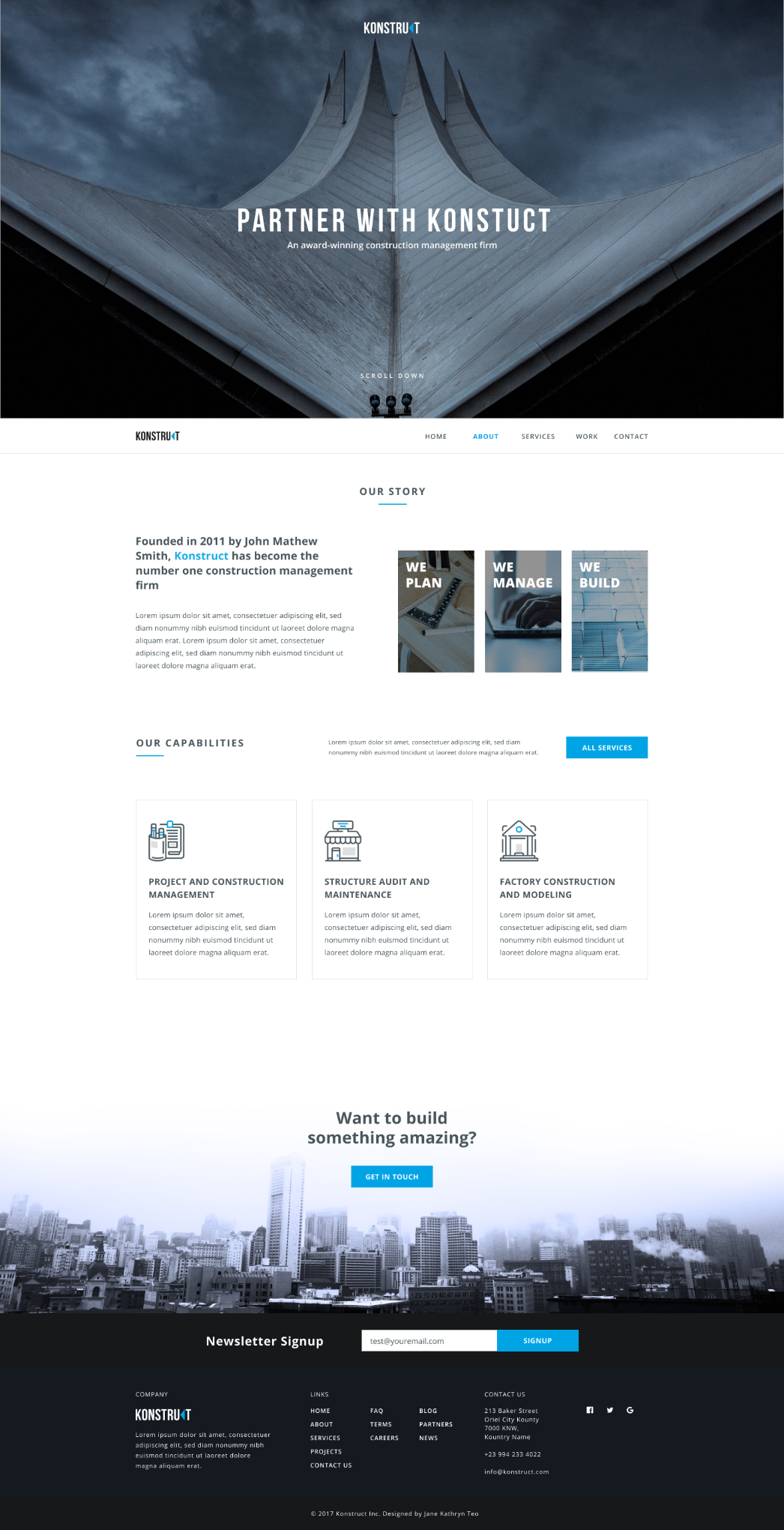
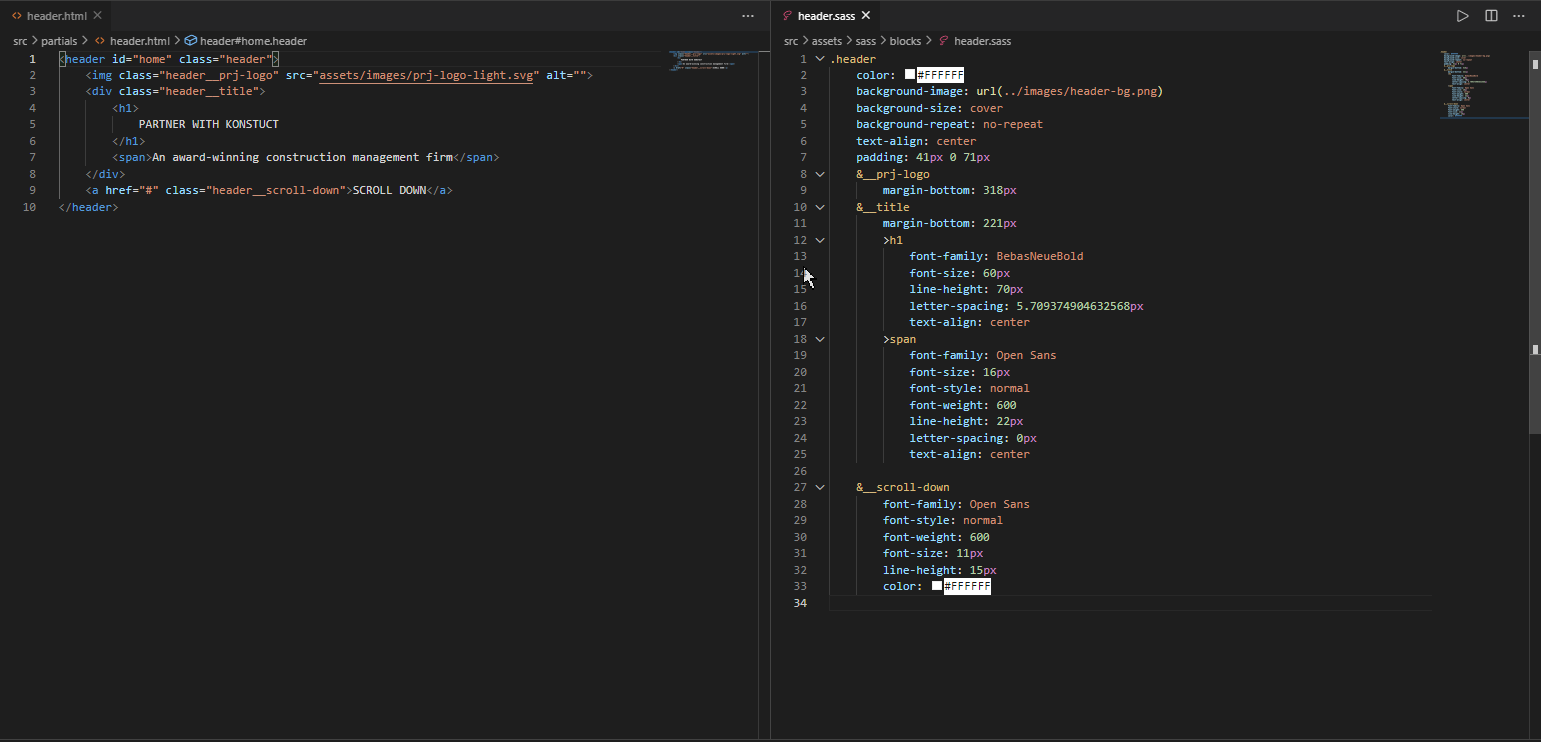


Рисунок 6

Таким образом выглядит сайт, который я собираюсь сверстать (рис. 5).

Часть 3. Блок первый. Header.

Начнем с верхней части. Так будет выглядеть наш код (рис. 7).



Рисунок

Справа html файл, а слева sass файл.

Тегом *header* я разграничиваю блок. Так он должен выглядеть (рис. 8).

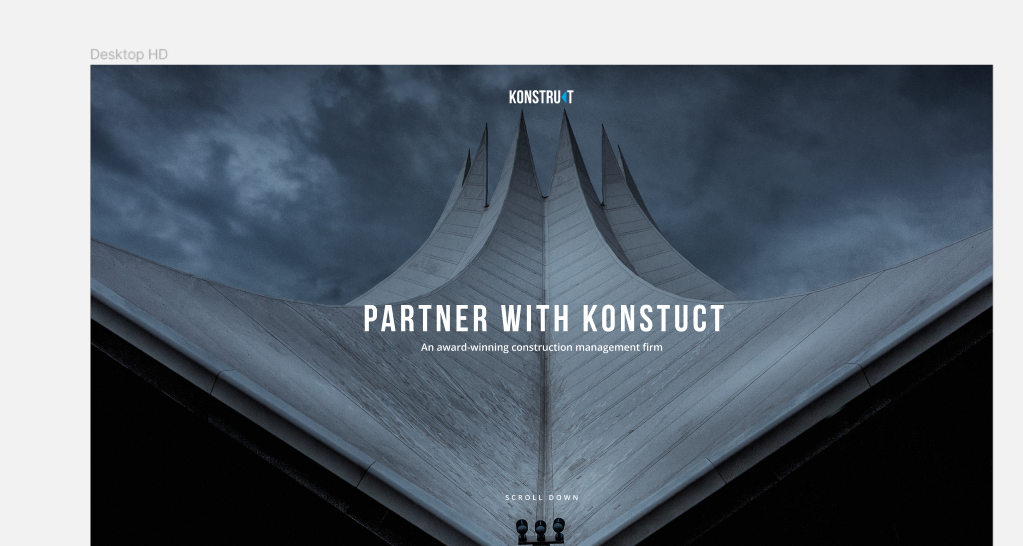


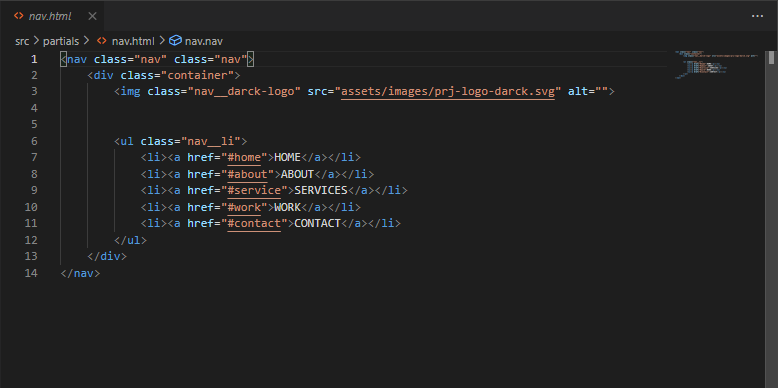
Рисунок 8

В тэг *img* я вставляю логотип. В файле *header.sass* выравниваем его по центру и делаем отступ от верхнего края.

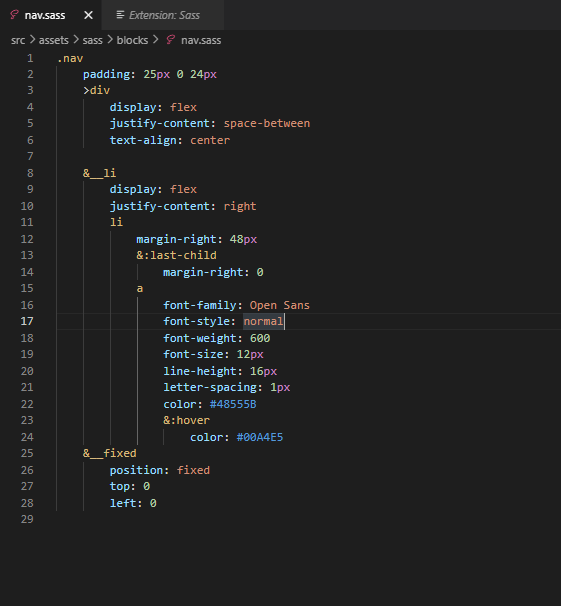
Потом я создаю отдельный контейнер *header\_\_title* и в тэге *h1* вписал заголовок, а в тэге *span* вписал текст.

Тэгом *a* я добавил кнопку “scroll down”.

Часть 4. Блок второй. Navbar.



Рисунок

Следующий блок это Navbar. Это самый легкий блок.

Рисунок

Сначала я объединю все в тэг, под названием *nav*.

В нем будет тэг *div* с классом *container*. Этот *div* я буду использовать, если мне нужно будет выровнять текст. Мы пропишем стили для него только один раз, и будем его вызывать с помощью этого класса.

Тэгом *img* вставляем логотип.

Дальше создаем кнопки навигации. Их мы будем располагать с помощью таблицы (тэга *ul*). Каждая кнопка — это отдельная строчка в таблице (тэг *li*).

И чтобы они работали добавим тэг *a*.

Чтобы они были в строчку, а не в столбик, я использую *display:* *flex*.