Требования к верстке.

**Оглавление**

1. Соответствие макету
2. Кроссбраузерность, кодировка и DOCTYPE
3. Корректная работа при вбивании реального текста, надёжность вёрстки
4. Отсутствие глупостей в html и css, единообразие, аккуратность

## #1. Соответствие макету

1. Для проверки соответствия вашей верстки макету, необходимо использовать [Pixel Perfect](https://chrome.google.com/webstore/detail/perfectpixel-by-welldonec/dkaagdgjmgdmbnecmcefdhjekcoceebi?hl=ru). Суть работы плагина: дизайн макет страницы накладывается на вашу верстку в целях проверки соответствия вашей верстки макету.
2. Старайтесь добиться идеального сходства.
3. При явных проблемах в дизайн макете (ошибках дизайнера) вы вправе придерживаться здравого смысла.
4. Допустимо, чтобы по ширине текст расходился с макетом (связано с межбуквенным интервалом. Не нужно пытаться менять межбуквенный интервал при помощи CSS)
5. При изменениях контента, размеры блоков могут меняться (по высоте, например).

## #2. Кроссбраузерность, кодировка и DOCTYPE

1. **Кодировка: UTF-8**

Зачем нужно: UTF-8 это универсальность и совместимость. Это современный стандарт, за ним даже не будущее, а настоящее.

Эта кодировка должна использоваться для всех файлов: html, css, js (если файлы в разных кодировках почти наверняка будут проблемы)

**2. DOCTYPE: HTML5**

Зачем нужно: наличие корректного doctype необходимо чтоб страницы отображались в соответствии со стандартами. HTML5 doctype позволяет нам смело использовать современные тэги (canvas, header, article,...) и старые проверенные решения, ранее бывшие в опале (например embed). HTML5 — это современный стандарт, в нём можно писать и в строгом XHTML-синтаксисе.

**3. Кроссбраузерность**

* Chrome (последний)
* Firefox (последний)
* Safari (последний)
* Opera (последний)
* IE10+

## #3. Надёжность вёрстки

Вёрстка должна тянутся, не разваливаться и не терять дизайнерский вид при изменении контента на странице. Его может быть больше или меньше чем на макете.

1. **Проверка ввода и удаления данных.**

Проверяется: на странице с контентом, пробуем добавлять и удалять содержимое – «что будет когда текста много?», «а когда мало?».

Обязательно пробовать менять расположение элементов, чтоб после того как вы поменяли блоки местами не развалилось оформление (из-за каскада).

1. **Проверка корректности работы стилей.**

Каждый логический блок представляет из себя независимый модуль. И его перемещение в другую структурную часть сайта, не должно влиять на его стили.

Это нужно чтоб на живом сайте потом не появились проблемы при заполнении реальными данными.

Использовать html5-тэги для разметки: header, footer, aside, nav, section, article и т.д. Кроме того что это семантично, также повышается надёжность, «пуленепробиваемость» вёрстки. Лишний открытый или закрытый div легко может поломать вёрстку. Но когда каркас сайта — атомарные и редко повторяющиеся html5-тэги, то «поломка» локализуется в пределах html5-тэга.

## #5. Отсутствие глупостей в html и css, единообразие, аккуратность

**Плохо:**

* Отступы между блоками на сайте должны быть за счёт margin у блоков, а не выпирающих наружу margin у содержимого блоков.
* Нежелательно когда вёрстка содержит пустые блоки для презентационных целей, для этого существуют псевдоэлементы
* Плохо когда нет базовых стилей у стандартных элементов. Т.е. просто h1,h2,ul,table,etc без классов должны смотреться красиво и органично. Проще говоря — используйте Normalize, a не Reset CSS.
* Плохо (недопустимо!) вешать стили на селекторы вложенных стандартных тэгов, без классов. Т.е. писать что-то типаh2 a span {}.

**Хорошо:**

* По возмжоности использовать[БЭМ](https://ru.bem.info/method/key-concepts/)! Важно понимать что это методология, а не инструменты. Нужно обязательно уходить от каскада, а БЭМ — это один из отличных вариантов решения.
* Хорошо — структурировать код в блоки описывающие логику документа. Т.е. создавать div даже там, где он для презентационных целей не нужен. И наоборот — стараться не ставить лишний div там, где структура этого не требует, а это нужно лишь для визуальных эффектов.

# Важно!

* Лого на внутренних страницах должно вести на главную.
* Все ссылки должны как-то реагировать на :hover, :active и :focus — показыванием/убиранием подчёркивания, сменой цвета, чем угодно.
* В вёрстке не должны оставаться закомментированные «на всякий случай» куски кода, лишние неиспользуемые файлы, старые версии файлов и т.п.
* Ссылки на сторонние ресурсы должны быть с target="\_blank".
* Разумеется картинки должны быть в отдельной папке, css — в отдельной и js — в отдельной. Графика, не являющаяся частью дизайна (всякие иллюстрации, фото в новостях и т.д.) — нужно положить в отдельную папку, например userfiles.