

Код документа	0204
Ревизия	1.1.247
Дата	26.11.2018

Верстка HTML шаблона

Техническое задание

Таблица обновлений документа

Дата	Ревизия	Автор	Примечание	Проверил

Статистика

Файл 0204 - project - ТЗ_ВерсткаHTMLШаблонов.docx
Размер: 73 Кбайт
Дата создания: 30.10.2016
Дата редактирования: 26.11.2018
26.11.2018

Дата печати:

Содержание

Введение	5
Сокращения.....	5
Ссылки	5
1. Введение	6
2. Организационные моменты:	7
3. Требования	8
3.1 Общие требования	8
3.1.1 Структура HTML-документа.....	8
3.2 Адаптивная верстка типовых страниц в формате языка гипертекстовой разметки (HTML), включая каскадные таблицы стилей (CSS).....	9
3.3 Брекпойнты	9
3.3.1 1920 x 1200 и выше	9
3.3.2 1920 x 1080	9
3.3.3 1600 x 1200	9
3.3.4 1600 x 900	9
3.3.5 1440 x 1080	9
3.3.6 1440 x 900	9
3.3.7 1366 x 768	9
3.3.8 1280 x 1024	9
3.3.9 1024 x 768	9
3.4 Верстка должна быть выполнена без применения каких-либо фреймворков/библиотек типа Bootstrap. Чистый HTML/CSS.....	10
3.5 Верстка должна быть выполнена без применения js-скриптов.....	10
3.6 Верстка не должна конфликтовать с Bootstrap4.x/Bootstrap3, элементы которых могут быть использованы в верстаемых типовых страницах.....	10
3.7 Осмысленное наименование тегов.....	10
3.8 Верстка должна проходить валидацию	10
3.8.1 HTML	10
3.8.2 CSS.....	10
3.9 CSS.....	10
3.9.1 Для создания CSS должны использоваться пре/пост процессоры	10
3.9.2 Недопустимо использование миксинов (@include) и экстендов(@extend).....	11
3.9.3 Нельзя использовать вендорные префиксы в препроцессорах, которые могут быть добавлены автопрефиксером.....	11
3.9.4 Использование Flex и Grid должно быть обосновано.....	11
3.10 Осмысленное наименование CSS-классов	11
3.10.1 Нельзя использовать !important.	11
3.10.2 Сброс стилей – normalize.css или reset.css	11
3.10.3 Преимущественное использование единиц em и rem, а не px	11
3.10.4 Верстка должна быть чистой.....	11

3.10.5	Верстка должна проходить тесты на переполнения и незаполнения.....	11
3.11	Верстка должна корректно отображаться в следующих браузерах	11
3.11.1	Internet Explorer.....	11
3.11.2	Firefox.....	11
3.11.3	Chrome	12
3.11.4	Edge.....	12
3.12	Верстка должна корректно отображаться при различных коэффициентах масштабирования шрифта (устанавливается на десктопах)	12
3.13	Верстка должна корректно отображаться при различных размерах (масштабе) фонта, устанавливаемого в браузерах.	12
4.	Шаблон Page.....	13
4.1	Header	13
4.2	SubHeader	14
4.3	LeftSide	14
4.4	Body	14
4.5	RightSide.....	14
4.6	Footer.....	15
5.	Блок Body	16
5.1	BodyHeader	16
5.2	Body	16
5.3	ScrollBar.....	16
6.	Карточка Исполнителя	17

Введение

Данный документ представляет собой техническое задание на верстку структуры (шаблона) HTML-документа.

Сокращения

Аббревиатура	Значение

Ссылки

- 1.
- 2.

1. Введение

HTML-шаблон будет использоваться в проекте на ASP.NET Core.

Типы разрабатываемых страницы приведены в документе Layout.xlsx.

Пример семантической структуры HTML-шаблона в файле TemplateHTML.html.

2. Организационные моменты:

1. Исполнитель предоставляет график выполнения работ:
 - Разбиение работы на этапы (при наличии)
 - Срок готовности всей работы/каждого этапа
 - Стоимость всей работы/каждого этапа
2. При условии соблюдения конечного срока выполнения всей работы, допускаются перерывы в работе до 2х дней. Работа в выходные дни и государственные праздники остается на усмотрение исполнителя. Об изменениях Исполнитель должен уведомлять заблаговременно, а не по факту/постфактум.
3. Ежедневная отчетность и информирование о ходе работ.
4. Работа ограничена 3-мя итерациями – основная и 2 правки.

В первой исполнитель делает как считает нужным, но допускает незначительное вмешательство заказчика в рамках корректировки глобальных моментов на которые нужно обратить внимание, чтобы потом не переделывать двойную работу. Далее собирается пакет правок и отправляется на исправление. После этого может быть еще одна такая итерация правок.
5. Перед началом работы Исполнитель выбирает и согласовывает инструменты для:
 - валидации HTML-файлов
 - валидации CSS-файлов
 - используемые пре/постпроцессоры
 - проверки при различных разрешениях
 - проверки в различных браузерах

3. Требования

3.1 Общие требования

Основной рабочий режим сайта разрабатывается под разрешение монитора 1600 x 900. Далее это разрешение называется **базовым**.

3.1.1 Структура HTML-документа

```
<!DOCTYPE html>
<html lang="ru">
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>Title</title>
  <link rel="stylesheet" href="CssFile" />
</head>

<body>
  <header></header> <!-- Header -->
  <aside>
    <div class="LeftSide">
    </div>
  </aside> <!-- Left Side-->
  <main>
  </main> <!-- Main-->
  <aside>
    <div class="RightSide">
    </div>
  </aside> <!-- Right Side-->
  <footer></footer> <!-- Footer -->

</body>
</html>
```


3.2 Адаптивная верстка типовых страниц в формате языка гипертекстовой разметки (HTML), включая каскадные таблицы стилей (CSS)

Под адаптивной версткой подразумевается отзывчивая (англ. - “responsive”) верстка, при которой содержимое страницы подстраивается под фактические размеры окна браузера, без рывков по ширине глобальных контейнеров на ключевых точках (“брейкпойнтах”). Другими словами, глобальные контейнеры всегда должны быть 100% ширины.

3.3 Брейкпойнты

Основной рабочий режим сайта разрабатывается под разрешение монитора 1600 x 900. Далее это разрешение называется **базовым**.

3.3.1 1920 x 1200 и выше

Используется верстка для 1600 x 900, выровненная по центру экран с пустым пространством по краям.

3.3.2 1920 x 1080

Используется верстка для 1600 x 900, выровненная по центру экран с пустым пространством по краям.

3.3.3 1600 x 1200

Используется верстка для 1600 x 900

3.3.4 1600 x 900

Базовый брейкпойнт.

3.3.5 1440 x 1080

3.3.6 1440 x 900

3.3.7 1366 x 768

3.3.8 1280 x 1024

3.3.9 1024 x 768

Используется верстка 1280 x 1024 с горизонтальной линией прокрутки.

3.4 Верстка должна быть выполнена без применения каких-либо фреймворков/библиотек типа Bootstrap. Чистый HTML/CSS

3.5 Верстка должна быть выполнена без применения js-скриптов.

Если какая-либо часть задания не может быть выполнена без скриптов, это должно быть оговорено перед началом работы.

3.6 Верстка не должна конфликтовать с Bootstrap4.x/Bootstrap3, элементы которых могут быть использованы в верстаемых типовых страницах.

3.7 Осмысленное наименование тегов

3.8 Верстка должна проходить валидацию

3.8.1 HTML

Например, W3C - validator.w3.org

Если верстка не проходит валидацию, должно быть грамотное объяснение причины. Объяснения типа - «Я всегда так делаю. Все отображается нормально. Никто не жаловался.» - не прокатят.

3.8.2 CSS

Например, cslint.net.

Если верстка не проходит валидацию, должно быть грамотное объяснение причины. Объяснения типа - «Я всегда так делаю. Все отображается нормально. Никто не жаловался.» - не прокатят.

3.9 CSS

3.9.1 Для создания CSS должны использоваться пре/пост процессоры

В порядке приоритетности, исполнитель в праве выбрать: sass, less.

3.9.2 Недопустимо использование миксинов (@include) и экстендов(@extend)

3.9.3 Нельзя использовать вендорные префиксы в препроцессорах, которые могут быть добавлены автопрефиксером.

3.9.4 Использование Flex и Grid должно быть обосновано

Поскольку создается не полноценный дизайн, а базовая структура документа, не требующая полноценной адаптивности, необходимость применения Flex и Grid должна быть обоснована.

3.10 Осмысленное наименование CSS-классов

3.10.1 Нельзя использовать !important.

3.10.2 Сброс стилей – normalize.css или reset.css

3.10.3 Преимущественное использование единиц em и rem, а не px

Использование px должно быть обосновано.

3.10.4 Верстка должна быть чистой

Структура документа должна верстаться чистой разметкой, пренебрегая правилами БЭМ.

3.10.5 Верстка должна проходить тесты на переполнения и незаполнения.

То есть также быть отзывчивой по высоте элементов. Кроме специально оговоренных случаев, недопустимы фиксированные значения height.

3.11 Верстка должна корректно отображаться в следующих браузерах

3.11.1 Internet Explorer

не ниже 10.0. Платформа — операционные системы семейства Windows.

3.11.2 Firefox

не ниже 61.

3.11.3 Chrome

не ниже 67.

3.11.4 Edge

не ниже 11.

3.12 Верстка должна корректно отображаться при различных коэффициентах масштабирования шрифта (устанавливается на десктопах)

При разных размерах фонта, установленного в системе – Мелкий (по умолчанию)/Средний(125%)/Крупный(150%).

3.13 Верстка должна корректно отображаться при различных размерах (масштабе) фонта, устанавливаемого в браузерах.

При разных размерах фонта, установленного в браузере – 80% - 100% - 120 % - 150%.

4. Шаблон Page

Состоит из следующих блоков:

- Header
- SubHeader
- LeftSide
- Body
- RightSide
- Footer

Размеры задаются у всех блоков, кроме Body. Body занимает все оставшееся после других блоков пространство по ширине.

Возможные начальные значения, например, минимальные и максимальные высоты блоков, задаются статически, при разработке.

Размеры блоков задается для каждого разрешения (брекпойнта) отдельно.

Начальное состояние при отсутствии внутри каждого из блоков каких-либо элементов и не заданной минимальной высоте блоков:

- Header не виден
- SubHeader не виден
- LeftSide не виден
- RightSide не виден
- Footer не виден
- Body занимает все пространство

Каждый из блоков имеет два состояния – обычное и фиксированное. В фиксированном состоянии при прокрутке контента в Body блок остается на своем месте. Переход в фиксированное состояние происходит добавлением класса/атрибута/модификатора к стандартному классу:

Было: `<header class="standart"></header>`

Стало: `<header class="standart fixed " ></header>`

Вертикальный скроллбар постоянно присутствует, даже если контента в Body мало.

4.1 Header

Высота этого блока зависит от высоты находящихся в нем элементов.

Если никаких элементов нет – высота = 0, т.е. блок скрыт.

Может быть задана минимальная высота.

Может быть задана максимальная высота. При ее превышении, находящиеся в нём элементы обрезаются.

Может быть задана фиксация блока. При отсутствии фиксации при прокрутке блок уходит вверх. При фиксации блок остается на месте, прижат к верхней части экрана. Содержимое Body прокручивается за ним.

4.2 SubHeader

Высота этого блока зависит от высоты находящихся в нем элементов.

Если никаких элементов нет – высота = 0, т.е. блок скрыт.

Может быть задана минимальная высота.

Может быть задана максимальная высота. При ее превышении, находящиеся в нём элементы обрезаются.

Может быть задана фиксация блока. Фиксация может быть задана независимо от фиксации блока Header. Если Header зафиксирован, SubHeader фиксируется под ним. Если Header не зафиксирован, при прокрутке, после “ухода” Header-а, SubHeader фиксируется к верхней части экрана. Содержимое блока Body прокручивается за ним.

При отсутствии фиксации при прокрутке блок уходит вверх.

4.3 LeftSide

Может быть задана минимальная ширина.

Ширина блока – по ширине содержимого.

Может быть задана максимальная ширина блока, при ее превышении содержимое обрезается.

4.4 Body

Может быть задана фиксация ширины по ширине экрана.

Если задана фиксация ширины, то содержимое, выходящее за ширину экрана обрезается.

Может быть задана максимальная высота блока. В этом случае, при большом контенте, активизируется скроллбар.

4.5 RightSide

Может быть задана минимальная ширина.

Ширина блока – по ширине содержимого.

Может быть задана максимальная ширина блока, при ее превышении содержимое обрезается.

4.6 Footer

Высота этого блока зависит от высоты находящихся в нем элементов.

Если никаких элементов нет – высота = 0, т.е. блок скрыт.

Может быть задана минимальная высота

Может быть задана максимальная высота. При ее превышении, находящиеся в нём элементы обрезаются.

Может быть задана фиксация блока.

При фиксации блок всегда прижат к нижней части экрана. При прокрутке содержимое Body проходит за ним.

При отсутствии фиксации блок размещается за блоком Body, даже если высота Body превышает высоту экрана, т.е. смещается вниз.

При отсутствии фиксации при прокрутке вверх блок не поднимается выше нижней части экрана.

5. Блок Body

Состоит из:

- BodyHeader
- ScrollBar
- Body

5.1 BodyHeader

Высота меню – по высоте содержимого или задается.

Если высота меню задана, то при её превышении содержимое обрезается.

Меню может быть зафиксирован. В этом случае при прокрутке меню фиксируется у верхней границы блока, а содержимое прокручивается под ним.

5.2 Body

Блок растягивается на всю ширину контейнера, в котором он содержится.

Может быть задана максимальная высота блока.

Максимальная высота не может быть меньше, чем высота окна браузера минус высоты Header-а и Footer-а.

Справа оставляется место под scrollbar.

Если содержимое вмещается по высоте в блок – скроллбар не показывается, но место под него остается зарезервировано и не занимает контентом блока.

При превышении контентом высоты блока:

- Если блок зафиксирован - активизируется скроллбар блока Body.
- Если блок не зафиксирован - работает скроллбар браузера.

5.3 ScrollBar

Ширина скроллбара должна настраиваться.

6. Карточка Исполнителя

Наименование	
Фамилия, имя, отчество	
Контакты	
E-mail	
Телефон	
Сайт/Ссылки	
Другое	
Информация	Если есть, что сказать, говорите здесь.
Проект	
Срок выполнения	
Дата начала	
Общая стоимость	
Оплата	Яндекс.Деньги
Разбиение по этапам	
Этап 1	
Описание	
Сроки	
Цена	
Инструменты	
HTML-checker	
CSS-checker	
JS-checker	
Проверка при разных разрешениях	
Проверка в различных браузерах	
Препроцессор CSS	
Постпроцессор CSS	

Дополнительные сведения	