Отзывчивый и адаптивный дизайн сайта

16.02.2016[22 комментария](https://html5book.ru/otzyvchivyj-dizayn-saita/#comments)



Интернет-пользователи просматривают сайты на разных устройствах с экранами различных размеров. Размеры экранов постоянно меняются, поэтому важно чтобы сайт адаптировался к любому из них. Существует два основных подхода для создания сайтов, легко адаптирующихся для разных типов устройств:

**Responsive Design (RWD)** — отзывчивый дизайн — проектирование сайта с определенными значениями свойств, например, гибкая сетка макета, которые позволяют одному макету работать на разных устройствах;

**Adaptive Design (AWD)** — адаптивный дизайн, или динамический показ — проектирование сайта с условиями, которые изменяются в зависимости от устройства, базируясь на нескольких макетах фиксированной ширины.

* **Содержание:**
* [1. Приемы отзывчивого дизайна](https://html5book.ru/otzyvchivyj-dizayn-saita/#part1)
* [2. Настройка области просмотра с помощью мета-тега viewport](https://html5book.ru/otzyvchivyj-dizayn-saita/#part2)
* [3. Универсальные шаблоны](https://html5book.ru/otzyvchivyj-dizayn-saita/#part3)
* [4. Адаптивный дизайн](https://html5book.ru/otzyvchivyj-dizayn-saita/#part4)
* [5. В чём разница между отзывчивым и адаптивным веб-дизайном](https://html5book.ru/otzyvchivyj-dizayn-saita/#part5)
* [6. Полезные сервисы и инструменты](https://html5book.ru/otzyvchivyj-dizayn-saita/#part6)

1. Приемы отзывчивого дизайна

Философия отзывчивого веб-дизайна заключается в том, что сайт был удобным для просмотра с любого устройства, независимо от размера экрана. Фраза **отзывчивый дизайн**была придумана Ethan Marcotte в 2011 году. Главная особенность отзывчивого веб-дизайна — за счёт подвижной (fluid) сетки макет автоматически реагирует на изменение размеров экрана, раздуваясь или сужаясь, как воздушный шар.

**Отзывчивый дизайн** (англ. **Responsive Web Design**) объединяет в себе три методики — **гибкий макет** на основе сетки, **гибкие изображения** и **медиазапросы**.

**Гибкость макета** базируется на использовании относительных единиц измерения вместо фиксированных пиксельных значений, что позволяет регулировать ширину в соответствии с доступным пространством.

**Гибкость текстового содержимого** достигается путем вычисления размеров шрифта относительно размера шрифта в браузерах по умолчанию 16px, например для фиксированного размера font-size: 42px относительный размер равен 42px / 16px = 2.625em.

Проблема **гибких изображений** решается с помощью правила img {width: 100%; max-width: 100%;} для всех картинок на сайте. Это правило гарантирует, что изображения никогда не будут шире, чем их контейнеры и никогда не превысят своих истинных размеров на больших экранах.

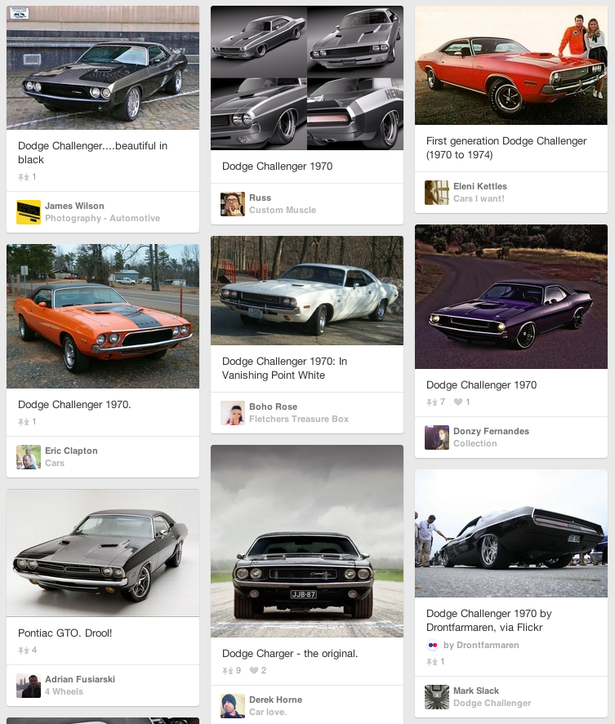
**Медиазапросы** изменяют стили на основании характеристик устройства, связанных с отображением контента, включая тип, ширину, высоту, ориентацию и разрешение экрана. С помощью медиазапросов создается отзывчивый дизайн, в котором к каждому размеру экрана применяются подходящие стили.

РИС. 1. ОТЗЫВЧИВЫЙ ДИЗАЙН

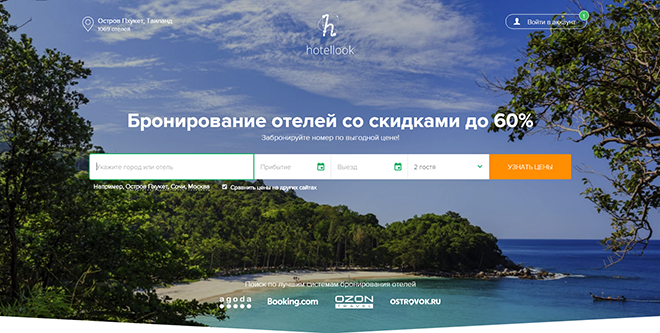
Дополнительные приёмы отзывчивого дизайна

**Масштабируемая векторная графика** — используйте svg-изображения, которые масштабируются до любого размера без потери качества и хорошо выглядят на дисплеях Retina.

**Карточные интерфейсы** — используйте так называемые карточные интерфейсы — прямоугольные фигуры с закруглёнными углами, которые представляют собой контейнеры для контента. Такие блоки являются самодостаточными единицами интерфейса и их легко перемещать по макету.

РИС. 2. PINTEREST, МАКЕТ НА ОСНОВЕ КАРТ

**Оставьте только самое необходимое** — хороший приём, особенно для отзывчивого дизайна. Создавайте отзывчивые и дружелюбные минималистические интерфейсы, которые приобретают всё большую популярность в настоящее время.

РИС. 3. HOTELLOOK, МИНИМАЛИЗМ В ВЕБ-ДИЗАЙНЕ

**Расставьте приоритеты и правильно скройте контент** — используйте скрытые элементы управления, особенно для устройств с небольшими экранами. Всплывающие окна, табы, off-canvas меню и другие подобные приёмы помогут уменьшить количество элементов на странице. Освободив пространство от лишних элементов, вы сделаете интерфейс удобным и дружелюбным для пользователя.

**Создайте большое кликабельное пространство для кнопок** — чем больше активная область кнопки, тем легче пользователю взаимодействовать с ней.

**Добавьте интерактивности в свои интерфейсы** — в ответ на действие пользователя создайте ответное действие — анимацию, которая будет работать при наведении на элемент на десктопных устройствах и при прикосновении к элементу на мобильных устройствах.

2. Настройка области просмотра с помощью мета-тега viewport

Чтобы мобильные браузеры для операционных систем Android и iOS не изменяли автоматически размер страниц сайта, используется специальный тег <meta> с атрибутом name="viewport". В данном теге допускается устанавливать определенное значение для параметров width и initial-scale:

<meta name="viewport" content="width=device-width, initial-scale=1">

HTML

— initial-scale=1 говорит о том, что размер страницы в браузере будет равен 100% величины области просмотра. То есть, соотношение между физическим пикселем и css пикселем будет 1:1;

— width=device-width говорит о том, что ширина страницы будет равна 100% ширины окна любого браузера. То есть, ширина страницы сайта соответствует ширине устройства, поэтому её не надо масштабировать.

<meta name="viewport" content="initial-scale=2.0, width=device-width">

HTML

В данном примере содержимое в окне браузера будет в 2 раза больше по сравнению с физическим размером.

Также тег <meta> можно использовать для контроля над тем, насколько пользователи могут увеличивать и уменьшать ширину страницы:

<meta name="viewport" content="width=device-width, maximum-scale=3, minimum-scale=0.5">

HTML

Данный код позволит увеличивать ширину страницы до величины, равной 3-х кратной ширине экрана устройства и уменьшать её до половины ширины экрана устройства.

Лишить пользователей возможности масштабирования можно с помощью атрибута user-scalable:

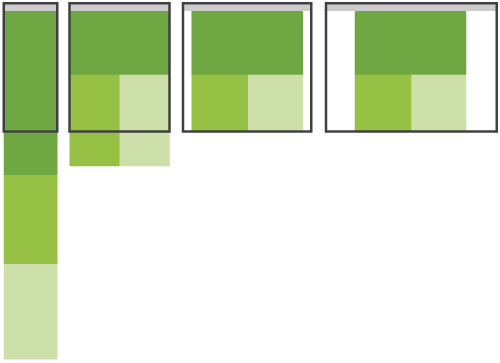
<meta name="viewport" content="initial-scale=1.0, user-scalable=no">

HTML

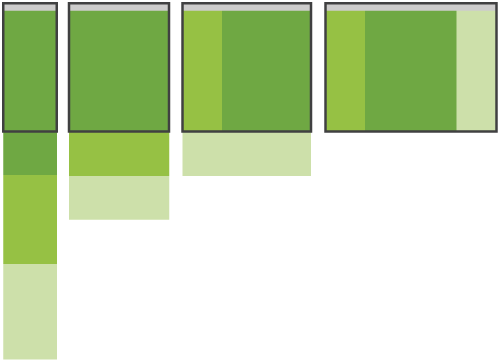
3. Универсальные шаблоны

Большая часть макетов, используемых для создания отзывчивого веб-дизайна, относятся к одной из пяти категорий шаблонов, определенных Люком Вроблевски:  
**Mostly Fluid** (Наиболее резиновый),  
**Column Drop** (Столбцы друг под другом),  
**Layout Shifter** (Двигающийся макет),  
**Tiny Tweaks** (Крошечные изменения),  
**Off Canvas** (Вне экрана).  
В некоторых случаях на странице может использоваться сочетание шаблонов, например Column Drop и Off Canvas.

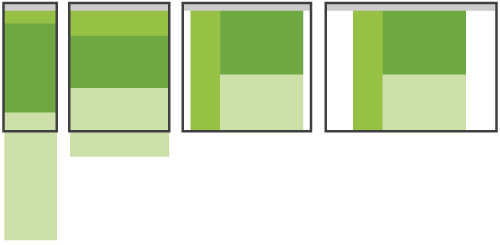
3.1. Mostly Fluid

Популярный макет,  состоит главным образом из резиновой сетки. На экранах большой или средней ширины ее размер обычно остается неизменным, на больших экранах регулируются только поля. На небольших экранах резиновая сетка вызывает перерасчет макета для основного контента, а столбцы размещаются друг под другом. Достоинством шаблона является то, что в нем необходима только одна контрольная точка между небольшими экранами и экранами большого размера.  


3.2. Column Drop

Столбцы размещаются один за одним по вертикали, если ширина окна не может отобразить весь контент. В результате столбцы будут располагаться вертикально друг под другом. Выбор контрольных точек для этого шаблона макета зависит от контента и определяется для каждого варианта дизайна отдельно.  


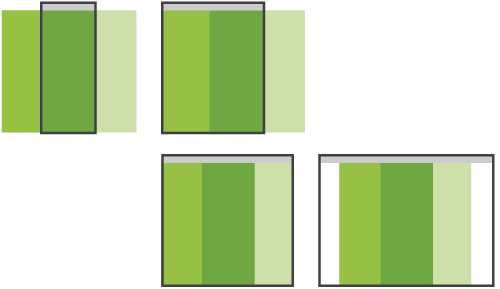
3.3. Layout Shifter

Наиболее отзывчивый шаблон, так как в нем предусмотрено наличие нескольких контрольных точек для экранов различной ширины. Основным отличием этого макета является то, что вместо перерасчета дерева отрисовки и размещения столбцов друг под другом перемещается контент. Из-за значительных различий между основными контрольными точками, поддержка этого макета является более сложной задачей, кроме того, вероятно, придется менять не только общий макет контента, но и его элементы.  


3.4. Tiny Tweaks

Шаблон вносит небольшие изменения в макет, например, регулирует размер шрифта, меняет размер изображений или перемещает контент. Он хорошо работает на макетах, состоящих из одного столбца, например, одностраничные сайты и статьи с большим количеством текста.  


3.5. Off Canvas

Контент, который используется редко, например, элементы навигации или меню приложения, размещаются за пределами экрана, и показываются только тогда, когда это позволяет сделать размер экрана. На небольших экранах контент открывается одним щелчком.  


4. Адаптивный дизайн

В отличие от отзывчивого дизайна, **адаптивный дизайн** (Adaptive Web Design) ориентируется на размеры устройств. Он использует несколько статичных макетов для разных типов устройств (мобильные устройства, планшеты, настольные компьютеры), базируясь на контрольных (переломных) точках. То есть макеты загружаются при определенных размерах окна браузера устройства, а переходы между макетами происходят скачкообразно, а не плавно.

Обычно адаптивные макеты имеют шесть вариантов макетов в зависимости от ширины экрана:  
320  
480  
760  
960  
1200  
1600.

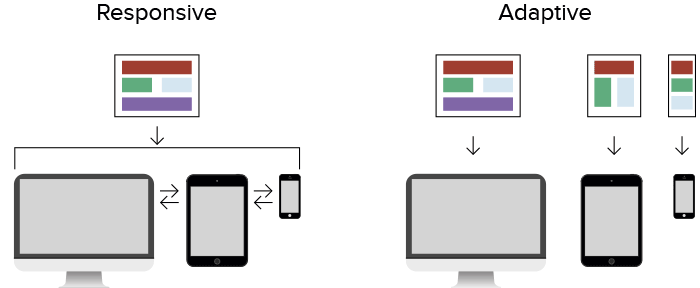
В адаптивных макетах главную роль играет функциональность, то есть при создании дизайна учитываются особенности устройств, например, сенсорное управление для мобильных устройств или большие пространства для настольных мониторов.

Основные приёмы адаптивного дизайна

**Придерживайтесь последовательности** — любой сайт должен создать доверительные отношения с пользователем, чтобы он чувствовал себя комфортно при навигации и взаимодействии с ним. Последовательный дизайн подразумевает, что при переходе на другую страницу сайта у пользователя не возникло чувства, что он попал на другой сайт. Уделяйте внимание мелким деталям, выстраивайте визуальную иерархию, важные элементы выделяйте жирным начертанием. Используйте последовательную цветовую схему по всему сайту, повторно используйте одни и те же элементы для различных ситуаций, например, один и тот же дизайн всплывающих уведомлений.

**Используйте сетку** — структура из 12-ти колонок является более предпочтительной для управления шириной колонок и отступами между ними.

5. В чём разница между отзывчивым и адаптивным веб-дизайном

РИС. 4. ОТЗЫВЧИВЫЙ И АДАПТИВНЫЙ ДИЗАЙН НА РАЗНЫХ УСТРОЙСТВАХ

Для создания отзывчивых макетов используются медиазапросы и относительные размеры элементов сетки, заданные с помощью %. В адаптивном дизайне серверные скрипты сначала определяют тип устройства, с помощью которого пользователь пытается получить доступ к сайту (настольный ПК, телефон или планшет), затем загружает именно ту версию страницы, которая наиболее оптимизирована для него. Для элементов сетки задаются фиксированные pxразмеры.

Поэтому основное отличие между этими приёмами — отзывчивый дизайн — один макет для всех устройств, адаптивный дизайн — один макет для каждого вида устройства.

6. Полезные сервисы и инструменты

[Android SDK](https://html5book.ru/goto/http:/developer.android.com/sdk/) Android-эмулятор для Windows, Linux и Mac OS X. iOS-симулятор доступен только для пользователей Mac OS X и является частью пакета Xcode (его можно бесплатно скачать из Mac App Store).

[Adaptive Images](https://html5book.ru/goto/http:/adaptive-images.com/) PHP-скрипт, работающий на любом веб-сайте, определяет размер экрана и подгоняет под него размер изображения, что в итоге дает малый размер изображения на малых экранах.

[Mydevice.io](https://html5book.ru/goto/http:/mydevice.io/devices/) Таблицы соответствия физических размеров устройств css-значениям высоты и ширины, а также значение pixel-ratio для мобильных устройств.

[Mediaqueri.es](https://html5book.ru/goto/http:/mediaqueri.es/) Коллекция веб-сайтов, использующих медиазапросы и отзывчивый веб-дизайн.

[Skeleton](https://html5book.ru/goto/http:/getskeleton.com/) CSS-фреймворк на основе 12-ти колоночной разметки, максимум 960px. Поддерживается Chrome, Safari, Firefox, IE 7 и выше, мобильными версиями браузеров.

[Bootstrap](https://html5book.ru/goto/http:/getbootstrap.com/2.3.2/) Набор инструментов для разработки веб-приложений. Язык LESS, 12-ти колоночная адаптивная разметка, поддержка мобильных устройств, планшетов и мониторов, множество компонентов, кнопок, выпадающие меню, собственный стиль полей ввода, списков, заголовков, меток, иконок, алерты, табы, прогресс-бары, всплывающие подсказки, «аккордеон», «карусель», и так далее, различные Javascript-плагины, поддержка Scaffolding, в том числе применение Bootstrap-стиля к уже созданным HTML.