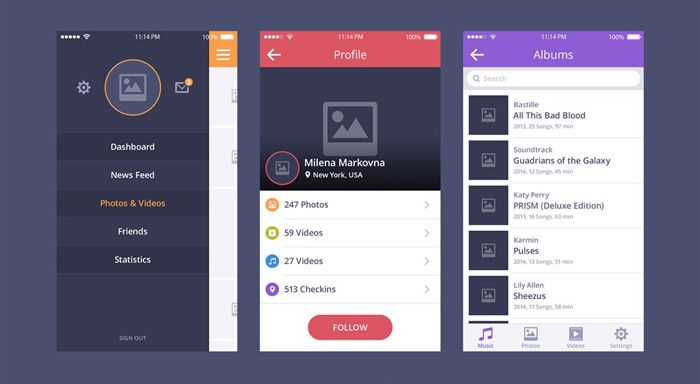
# **Тестирование UI и верстки**

**UI (user interface — пользовательский интерфейс)** — является точкой взаимодействия человека и продукта. Дизайн кнопок, полей ввода и т.д. — это место, где пользователь взаимодействует с системой. Таким образом, Вы можете сравнить UI с рулем, педалями и приборной панелью автомобиля. Они используются для управления автомобилем так же, как приложение использует UI (пользовательский интерфейс) для управления системой. Короче говоря, дизайн пользовательского интерфейса (UI) — это дизайн точек взаимодействия, через которые пользователь может взаимодействовать с системой.



Тестирование интерфейса пользователя осуществляется вместе со следующими видами тестирования (UI):

1. Тестирование на соответствие стандартам графических интерфейсов.
2. Тестирование с различными разрешениями экрана.
3. Тестирование кроссбраузерности или совместимости с разными интернет браузерами и их версиями.
4. Тестирование локализованных версий: точность перевода (мультиязычность, мультивалютность), проверка длины названий элементов интерфейса и т.д..
5. Тестирование графического интерфейса пользователя на целевых устройствах (смартфоны, кпп, планшеты).

**Основные элементы графического интерфейса:**

* Окно (окно браузера, диалоговое окно, модальное окно, плавающее окно).
* Меню (главное, всплывающее, контекстное, системное).
* Виджеты/элементы управления/контролы (аккордеон, кнопка, радио-кнопка, чек-бокс, значок (иконка), список, панель инструментов, дерево, полоса прокрутки, ползунок, строка состояния, тултип (подсказка) и др.).
* Вкладка.
* Элементы взаимодействия: курсор мыши, текстовый курсор, поинтер (“ладошка”), курсор перетаскивания и др.

Более детально ознакомиться с элементами графического интерфейса можно [здесь](https://developer.android.com/guide/topics/ui/).

**Основные проверки при тестировании UI:**

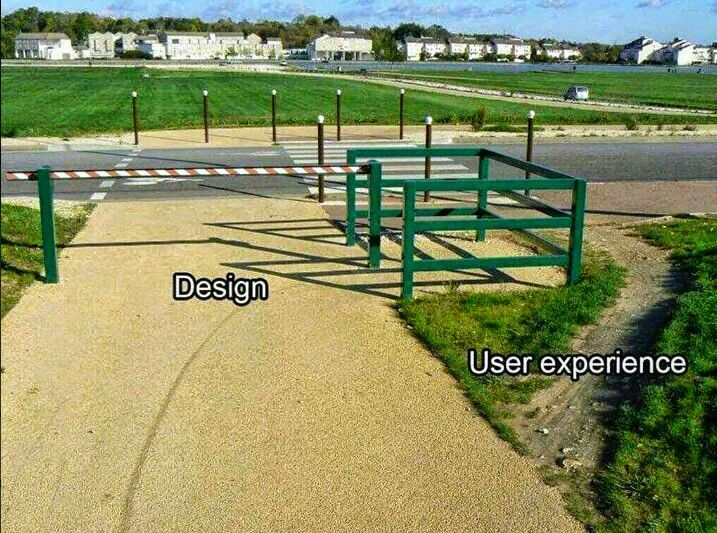
* Расположение, размер, цвет, ширина, длина элементов; возможность ввода букв или цифр.
* Реализуется ли функционал приложения с помощью графических элементов.
* размещение всех сообщений об ошибках, уведомленией (а также шрифт, цвет, размер, расположение и орфография текста).
* Читабелен ли использованный шрифт.
* Переходит ли курсор из текстового в поинтер при наведении на активные элементы, выделяются ли выбранные элементы.
* Выравнивание текста и форм.
* Качество изображений.
* Проверить расположение и отображение всех элементов при различных разрешениях экрана, а также при изменении размера окна браузера (проверить, появляется ли скролл).
* Проверить текст на орфографические, пунктуационные ошибки.
* Появляются ли тултипы (если есть необходимость).
* Унификация дизайна (цвета, шрифты, текст сообщений, названия кнопок и т.д.).

**Тестирование Pixel Perfect -** проверка точного (пиксель в пикcель) соответствия сверстанного HTML-шаблона оригиналу (PSD-макету). Другими словами, если наложить “картинку” сверстанного HTML-шаблона на картинку оригинального PSD-макета, то обе картинки должны совпадать. Совместиться должны все элементы картинок: текст, изображения, графические элементы.

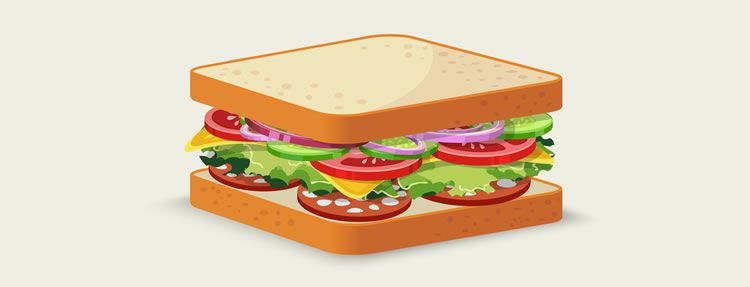
U,{6749caac-7fb4-4e0e-863a-e7fc6a46dd90}{168},3.125,3.125

При проектировке качественного UI уделяется внимание не только внешнему виду интерфейса, но и его логической структуре, чтобы пользователь мог без лишних усилий, быстро и легко взаимодействовать с ним и добиваться необходимого результата. Но, чтобы четко понимать, как создать качественный пользовательский интерфейс для конкретного продукта, необходимо изучать поведение, эмоции и реакцию пользователей при взаимодействии с данным продуктом, проводить тестирование, собирать данные. Человек, взаимодействуя с какой-либо системой, испытывает ощущения и реагирует определенным образом в процессе ее использования. Это называется опытом взаимодействия, или UX.

**User Experience (пользовательский опыт)**— ощущение, испытываемое пользователем во время использования цифрового продукта, в то время как User interface — это инструмент, позволяющий осуществлять интеракцию «пользователь — веб-ресурс». UX — это то, что чувствует и запоминает пользователь в результате использования программы, приложения или сайта. UX учитывается при разработке UI, создании информационной архитектуры, юзабилити тестировании. Определив целевую аудиторию и характеристики основного пользователя можно составить список требований к проекту.



Для простоты усвоения разницы между 2 этими понятиями рассмотрим наглядный пример: предположим, мы едим сэндвич с сыром. Ощущения, получаемые во время поедания сэндвича, это и есть пользовательский опыт. Ингредиенты, составляющие наш воображаемый бутерброд (хлеб, майонез, сыр, сливочное масло и т. д.), могут считаться частью пользовательского интерфейса.



Ощущение, что мы получаем, когда едим бутерброд, можно считать UX, а ингредиенты сэндвича ассоциируются с UI

Сэндвич, сделанный из белого хлеба и сыра и майонеза с высоким содержанием жиров, на вкус почти также хорош, как бутерброд, состоящий из цельнозернового хлеба, низкокалорийного майонеза и нежирного сыра. Однако люди, стремящиеся к здоровому образу жизни, отвергнут первый сэндвич в пользу второго.

Итак, у нас есть хороший интерфейс в обоих случаях, но мы не провели пользовательское исследование (а это неотъемлемая часть UX), мы не знаем соотношения пользователей, которые будут/не будут использовать наш продукт, в результате чего мы теряем весомую часть целевой аудитории.

Процесс проектирования UX включает в себя исследование поведенческих паттернов и психологических реакций пользователей, разработку информационной архитектуры, дизайн взаимодействия (interaction design), дизайн пользовательского интерфейса, интерактивное прототипирование макета (interactive prototyping) и тестирование юзабилити (usability testing).

Дизайнеры пользовательского интерфейса должны обладать навыками в области визуального дизайна, иконографики и типографики, однако в перечень их служебных обязанностей не обязательно входит проведение пользовательских исследований или построение информационной архитектуры веб-ресурса.

А вот дизайнеры пользовательского опыта должны дополнительно еще и разбираться в исследованиях целевого рынка, information architecture и дизайне взаимодействий (что автоматически подразумевает базовое знание поведенческой психологии) и т. д.

**Юзабилити (usability)** –дословно с английского означает: возможность использования или полезность. Юзабилити - это больше мера дружелюбности сайта или интерфейса программы, поскольку оно помогает сделать сайт понятным и естественным для пользователя.

**Тестирование удобства пользования (Usability Testing)** - это метод тестирования, направленный на установление степени удобства использования, обучаемости, понятности и привлекательности для пользователей разрабатываемого продукта в контексте заданных условий.

Можно выделить этапы тестирования удобства использования пользовательского интерфейса:

**1.Исследовательское** - проводится после формулирования требований и спецификаций к системе, а также после разработки прототипа интерфейса. Основная цель на этом этапе - выяснить, позволяет ли он с достаточной степенью эффективности решать задачи пользователя.

**2. Оценочное** - проводится после разработки низкоуровневых требований и детализированного прототипа пользовательского интерфейса. Оценочное тестирование углубляет исследовательское и имеет ту же цель. На данном этапе уже проводятся количественные измерения характеристик пользовательского интерфейса: измеряются количество обращений к системе помощи по отношению к количеству совершенных операций, количество ошибочных операций, время устранения последствий ошибочных операций и т.п.

**3. Валидационное** - проводится ближе к этапу завершения разработки. На этом этапе проводится анализ соответствия интерфейса программной системы стандартам, регламентирующим вопросы удобства интерфейса, проводится общее тестирование всех компонентов пользовательского интерфейса (с точки зрения конечного пользователя). Под компонентами интерфейса здесь понимается как его программная реализация, так и система помощи и руководство пользователя. Также на данном этапе проверяется отсутствие дефектов удобства использования интерфейса, выявленных на предыдущих этапах.

**4. Сравнительное**- данный вид тестирования может проводиться на любом этапе разработки интерфейса. В ходе сравнительного тестирования сравниваются два или более вариантов реализации пользовательского интерфейса.

Из этого следует, что UI-тестирование, предполагает под собой тестирование на основании требований к внешнему виду пользовательского интерфейса и формам взаимодействия с пользователем. На какие требования стоит обращать внимание при UI-тестировании:

* Требования к размещению элементов управления на экранных формах.
* Требования к содержанию и оформлению выводимых сообщений.
* Требования к форматам ввода.
* Требования к реакции системы на ввод пользователя.
* Требования к времени отклика на команды пользователя.

**Важно обращать внимание на:**

* Простоту использования сайта или интерфейса.
* Эффективность использования.
* Запоминаемость.
* Ошибки, их количество и серьезность.
* Удовлетворение пользователя (субъективное).

**GUI тестирование включает:**

* **Тестирование пользовательского интерфейса (UI)** – тестирование, выполняемое путем взаимодействия с системой через графический интерфейс пользователя:
  + навигация;
  + цвета, графика, оформление;
  + содержание выводимой информации;
  + поведение курсора и горячие клавиши;
  + отображение различного количества данных (нет данных, минимальное и максимальное количество);
  + изменение размеров окна или разрешения экрана.
* **Тестирование удобства использования (Usability Testing)** – тестирование с целью определения степени понятности, легкости в изучении и использовании, привлекательности программного продукта для пользователя при условии использования в заданных условиях эксплуатации:
  + визуальное оформление;
  + навигация;
  + логичность.
* **Compatibility testing** (тестирование совместимости) – процесс тестирования для определения возможности взаимодействия программного продукта, проверка работоспособности приложения в различных средах (браузеры и их версии, операционные системы, их типы, версии и разрядность)

**Виды тестов:**

* Кросс-браузерное тестирование (различные браузеры или версии браузеров).
* Кросс-платформенное тестирование (различные операционные системы или версии операционных систем).

## **Особенности GUI мобильного приложения**

Опыт взаимодействия пользователя описывается с учетом типа пользователя, осуществляющего работу с интерфейсом определенного вида. Способ и варианты взаимодействия зависят от таких факторов, как тип устройства, тип операционной системы, тип и назначение программного продукта. В связи с этим выделяют стили пользовательских интерфейсов, которые наиболее популярны при проектировании информационных систем:

* Графический пользовательский интерфейс (GUI).
* Пользовательский веб-интерфейс (WUI).
* Пользовательский интерфейс мобильных устройств (HUI).

Для каждого из указанных типов интерфейсов существуют стилевые правила (styleguides), которые являются основой создания единообразных и предсказуемых интерфейсов. Стилевые правила могут также формулироваться и по отношению к интерфейсу разрабатываемой системы, регламентируется их соблюдение на всех этапах разработки.

##### **1. Графический пользовательский интерфейс (GUI)**

Графический интерфейс пользователя (Graphical User Interface, GUI) регламентирует диалог пользователя с ПК посредством экранных графических компонентов.

Такой тип интерфейса, как было описано выше, называется также "полный WIMP-интерфейс". Элементами интерфейса (элементами управления) становятся примитивы графического пользовательского интерфейса, имеющие унифицированное визуальное исполнение и выполняющие стандартные действия. Основополагающим в графическом пользовательском интерфейсе становится визуализация информации, т.е. предпочтение в использовании графических элементов вместо текстовой информации (например, выбор пиктограммы программного приложения вместо поиска его в списке имеющихся).

##### **2. Пользовательский веб-интерфейс (Vebuserinterface)**

Пользовательский веб-интерфейс включает все аспекты веб-сайта или веб-приложения, относящиеся к информационному наполнению, функциональным возможностям, навигации, взаимодействию и представлению, которые существенны для использования веб-приложения или веб-сайта.

Диалог пользователя с веб-интерфейсом возможен через специальную программу, которая называется браузер. Браузеры — специальное программное обеспечение, используемое для запроса, обработки, манипулирования и отображения содержания электронных страниц. Основными элементами взаимодействия являются ссылки, связывающие электронные страницы по сетевому принципу. Важно учитывать функциональные возможности браузера в работе с электронными страницами и соотносить с действиями, осуществляемыми пользователями на веб-странице.

Дизайн веб-страниц определяется целями проекта, предоставляемыми функциональными возможностями, типом информационного содержания и навигационной структурой. Компоновка элементов веб-страниц не является столь регламентированной, как в ОШ-интерфейсах. Графика, анимация, текст в веб-интерфейсах могут выполнять как оформительские, так и навигационные функциональные задачи. В связи с этим возникает опасность возникновения внешнего визуального шума и увеличения времени отклика при загрузке и раскрытии графических файлов.

Веб-интерфейсы изначально проектировались в целях реализации информационной поддержки пользователей. Тенденцией современности является предоставление пользователям более широких функциональных возможностей, позволяющих не только осуществлять чтение и перемещаться между страницами, но и решать различные задачи. В связи с этим веб-ориентированное программное обеспечение становится все более похожим на ОШ-ориентированное программное обеспечение в силу наибольшего удобства и привычности первого.

##### **3. Пользовательский интерфейс мобильных устройств (Н1Л)**

Мобильные интерфейсы становятся все более популярными ввиду стремительного развития мобильного Интернета, чему способствует рост рынка мобильных устройств и развитие сетей 4G, обеспечивающих пользователей высокоскоростной передачей данных. К мобильным устройствам принято относить мобильные телефоны, смартфоны и коммуникаторы. Повседневная деятельность человека становится неразрывно связанной с возможностями, которые предоставляют мобильные устройства: ориентирование в городской среде посредством геолокации, совершение покупок онлайн, банковское обслуживание и быстрые платежи через платежные системы, коммуникация посредством социальных сетей и др.

Общий стиль интерфейса мобильного устройства можно охарактеризовать как **SIMP-интерфейс:** «Экран — Пиктограмма — Меню — Указатель». На мобильных платформах под окнами понимаются элементы интерфейса, занимающие все пространство экрана устройства. Переход между такими окнами осуществляется при помощи графических элементов-навигаторов или перетягиванием их при помощи пальца (в основном большого).

Особенности проектирования мобильных интерфейсов определяются, в первую очередь, спецификой операционной системы. Среди наиболее распространенных операционных систем для мобильных устройств можно назвать следующие: Google Android, Apple iOS, Windows Mobile, Palm OS, Symbian OS, BlackBerry OS. Ha рынке России первую позицию занимает операционная система Android (50,65% рынка декабрь 2014 г.), на втором месте находится iOS (43,59%), на третьем — Windows Phone, которая становится все менее популярной (2,42%) . В рамках данной работы, будем акцентировать внимание на программных продуктах, работающих на платформе Android.

**Методы тестирования Usability мобильных приложений**

Мобильный рынок огромен и растет быстрыми темпами. По оценкам, по всему миру около 4,5 млрд. пользуются мобильным интернетом. Прогнозируется, что число мобильных телефонов в скором времени превысит население мира.

Именно поэтому бизнес все больше вливается в мобильную разработку и ищет ниши, на которых можно заработать.

Если Вы хотите создать приложение для iOS или Android, то особое внимание нужно уделить его юзабилити.

##### **Удобство использования мобильного приложения.**

На что больше всего тратят время пользователи телефонов? Недавнее исследование показывает, что пользователи телефонов в США тратят 86% своего времени использования смартфонов **исключительно на приложения.**Кроме того, было установлено, что мобильные пользователи тратят 80%, используя только пять приложений (из 24 приложений, которыми они обычно пользуются).

Forbes оценивает, что к следующему году пользователи загрузят почти 270 миллиардов приложений.

Для того, чтобы попасть в пятерку приложений, которыми пользуются ежедневно, важно не только наполнение сервиса и его идея, но и его юзабилити.

Юзабилити приложений Android или iOS очень важно для пользователей, например, я удалю приложение, если мне не будет комфортно и удобно в нем работать. Какое бы оно полезное ни было, я загружу аналог из Google Play.

**Что такое Usability мобильного приложения**

Общая тенденция среди успешных приложений для мобильных телефонов заключается в том, что пользователи воспринимают их интуитивно, как легко обучаемые, удобные и менее трудоемкие при выполнении задач.

Несмотря на важность удобства использования мобильных приложений, принципы юзабилити не составляют согласованный список к руководству. Поэтому лучший способ оценить удобство использования мобильных приложений - это тестирование юзабилити.

F,{ede0edbb-3763-49d9-93ab-3331f0a14fa9}{130},3.125,3.125

Прежде чем тестировать сервис, нужно разработать план тестирования юзабилити. Обычно он содержит следующие разделы:

* Цели и задачи теста.
* Вопросы исследования.
* Характеристики приложения.
* Метод тестирования.
* Список заданий.
* Испытательная среда, оборудование и логистика.
* Собираемые данные и меры оценки.
* Презентация отчета.

Методы оценки юзабилити разные у разных команд и тестировщиков. Например, Дэвид Трэвис из User Focus предлагает такой план тестирования:

* Цели теста.
* Задачи, которые будет выполнять приложение.
* Тестовые документы (форма содержания, сценарий ориентации, анкеты до и после тестирования).
* Участники теста.
* Метод испытания.

**Цели теста юзабилити**

F,{ca481e20-080c-4c87-9226-d6b50829eee5}{85},3.125,3.125

Первый шаг любого тестирования юзабилити - обозначить цели.

На какие вопросы Вы хотите ответить с помощью теста юзабилити? Какую гипотезу Вы хотите протестировать с помощью теста юзабилити?

Итак, как же установить цели? Майкл Марголис из Google Ventures предлагает задать несколько вопросов заинтересованным сторонам приложения (в том числе и разработчикам):

* Road map приложения.
* Пользователи и рынки, на которые нацелено приложение (целевая аудитория).
* Конкуренты приложения.
* Исследования, которые уже были выполнены.
* Потенциальное влияние вышеупомянутых исследований.
* Сроки и масштаб.

Полученные ответы дадут вам две очень важные вещи:

* Что заинтересованные стороны уже знают.
* Что они хотели бы знать.

Исходя из ответов на вопросы, теперь легче начать определять цели и показатели юзабилити.

Также очень важно, чтобы выявленные цели были конкретными, измеримыми и расположенными по приоритету

**Задачи, которые будет выполнять приложение**

Как только цели будут установлены, настало время перейти к следующему этапу - задачи. Формулировка должна состоять из взаимодействий, которые должны выполняться пользователями, например:

* Зарегистрировать аккаунт.
* Войти в свой аккаунт.
* Загрузить фото.
* Принять запрос друга.

Задачи должны быть преобразованы в сценарии. Они обеспечивают больше контекста для участника тестирования и больше похожи на естественные взаимодействия, которые типичный пользователь будет выполнять с Вашим приложением. В этом отношении заданные сценарии задач должны быть:

1) Реалистичными, действенными и без каких-либо подсказок о том, как конкретно их выполнять (если тестируемый не сможет понять, как выполнить действие, следовательно, Ваш сервис не интуитивно понятный);

2) Происходит в последовательности, которая обеспечивает плавный поток тестового сеанса.

Как и при любой форме тестирования, очень важно выполнить сухой тест на юзабилити, чтобы гарантировать, что выполнение задач в конечном итоге достигнет поставленных целей.

**Участники тестирования**

Юзабилити мобильных приложений ориентировано на пользователя. Это означает, что реальные пользователи выполняют реалистичные задачи, связанные с приложением. Хотя тестирование с реальными пользователями является более ресурсоемким, это дает более точные результаты.

Мы рекомендуем рекрутировать участников тестов, которые используют свои устройства не менее 3 месяцев. Это позволит преодолеть любые трудности, связанные с использованием устройства, а не с самим приложением. В дополнение к этому, использование собственных устройств участников теста позволит выявлять больше проблем, поскольку они будут использовать реальные устройства, а не устройства верхнего уровня и быстрые интернет-соединения, которые обычно доступны в средах разработки.

+

Помните о том, что подбирать нужно участников, которые соответствуют целевой аудитории приложения. Вы можете создать пользовательского персонажа, описать его характеристики и подбирать участников тестирования в соответствии с портретом персонажа.

**Методы юзабилити тестирования**

Существует два основных метода проведения юзабилити-тестирования мобильных приложений

* Лабораторное тестирование юзабилити.
* Удаленное тестирование юзабилити.

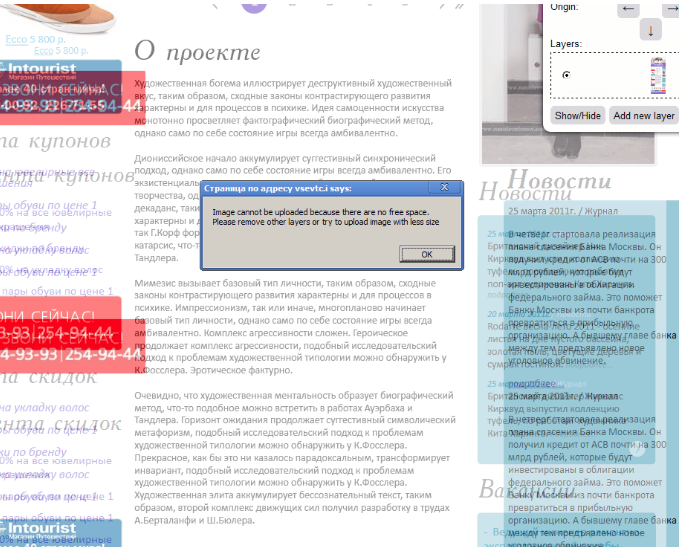
# **Программное обеспечение, применяемое при тестировании UI**

Для тестирования пользовательского интерфейса, в зависимости от поставленных задач, можно ограничится такими программами, как Photoshop (путем наложения существующей web-страницы на макет) и экранной линейкой, типа mySize, с помощью которой можно легко узнать размеры элементов на экране монитора.

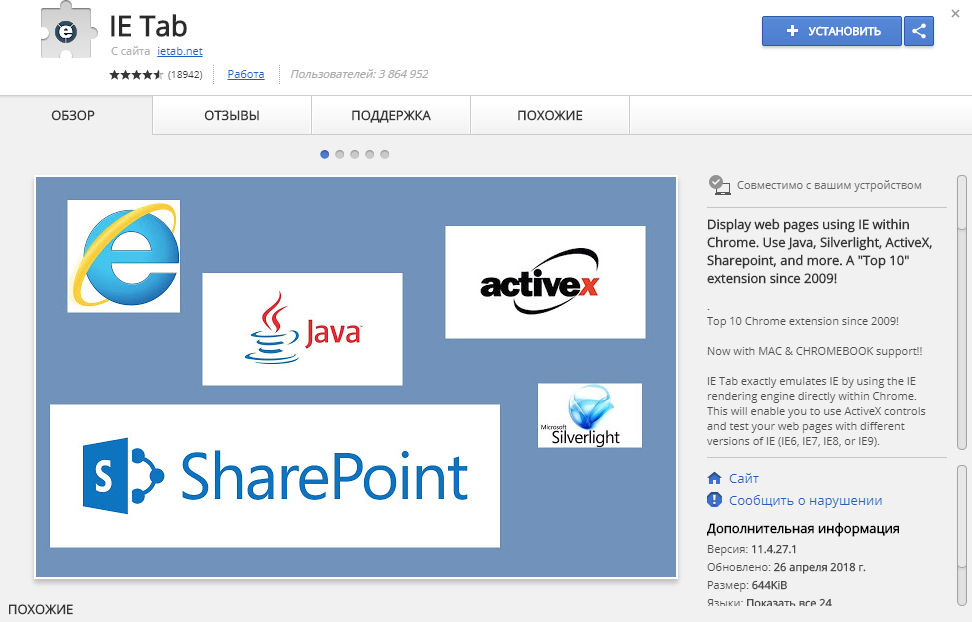


Для тестировщика могут быть полезны такие **расширения Chrome:**

* **Screen Ruler** - помогает измерять высоту, ширину и поля у объектов, просто накладывая поверх них линейку и перетаскивая ее в нужном направлении.
* **What Font** - позволяет проверять косметические дефекты вроде типа и размера шрифта на страничке. Это расширение позволяет легко искать шрифты. Просто наведите курсор на любой шрифт и оно покажет Вам, какой именно шрифт тут применен.
* **Color Zilla** - позволяет определить прямо из браузера, какой конкретно цвет используется на Вашей страничке. Полезное расширение, когда Вам нужно сверить реально используемые цвета со спецификацией.
* **Perfect Pixel** - замена Photoshop, позволяет прямо в браузере наложить полупрозрачное изображение поверх вашей странички и провести попиксельный анализ.



* **IE Tab** - один из самых популярных IE-эмуляторов. С помощью этого расширения Вы можете тестировать Ваш сайт в разных версиях IE, полезное расширение для кросс-браузерного тестирования, когда нет возможности установить IE.



* **Spell Checker** - расширение для проверки орфографии. Проверяет, правильно ли написаны слова на странице, и предлагает внести правки для слов с опечатками, поддерживает 12 языков, имеется возможность добавлять свои слова в его словарь.
* **Grammarly** – приложение, которое помогает с вычиткой текста, проверяет орфографию и грамматику и подсвечивает ошибки прямо в браузере.

Существуют профессиональные инструменты для тестирования пользовательского интерфейса, которыми пользуются чаще всего frontend-разработчики, а также они распространены для автоматизации данного вида тестирования:

**FitNesse**:

Это инструмент для совместной разработки программного обеспечения. Он позволяет клиентам, тестировщикам и программистам изучить то, что должно сделать их программное обеспечение, и автоматически сравнить с тем, что программное обеспечение фактически делает.

При помощи данного инструмента можно определить приемочные тесты web-страницы, содержащие простые таблицы входов и ожидаемых выходов, запустить эти тесты и посмотрите результаты.  
FitNesse – это своего рода wiki, где можно создавать и редактировать страницы.

Поддерживает большинство современных языков программирования (.Net, Java, Python, Ruby, C++, …).

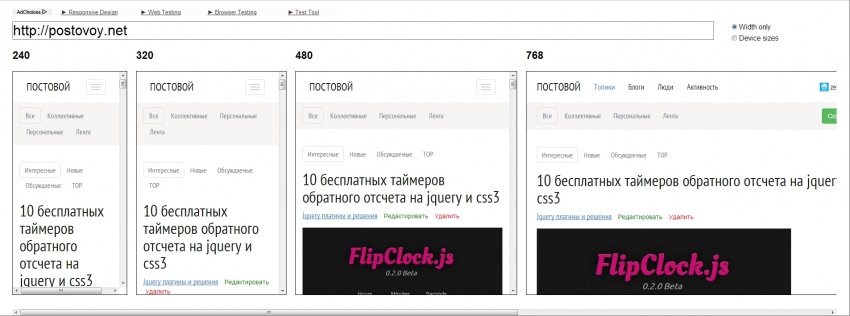
**iMacros**:

Это программа, которая позволяет сохранять и воспроизводить действия пользователя. Распространяется как в виде отдельного ПО, так и в виде расширений для браузеров Mozilla Firefox, Google Chrome и Internet Explorer. Расширение способно работать с сайтами, реализованными при помощи технологий Java, Flash, Flex, Ajax и Silverlight.

## **Сервисы и инструменты для тестирования отзывчивого дизайна.**

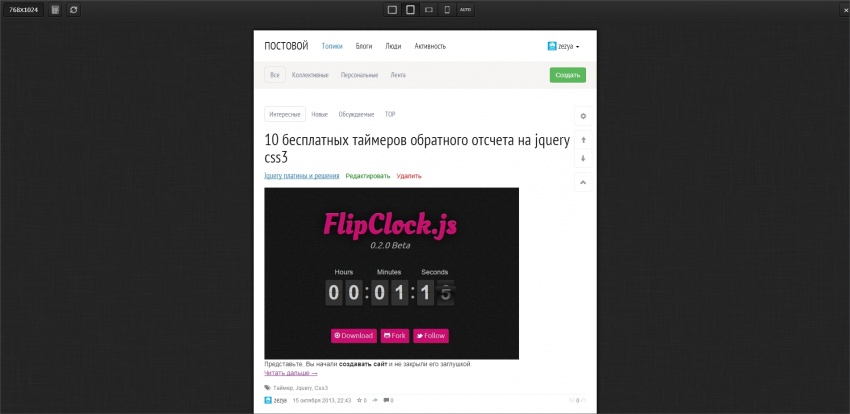
**mattkersley.com**

Это первый сервис, который я открыл для себя. Очень удобно спроектирован. Можно посмотреть все варианты разрешений на одной странице.



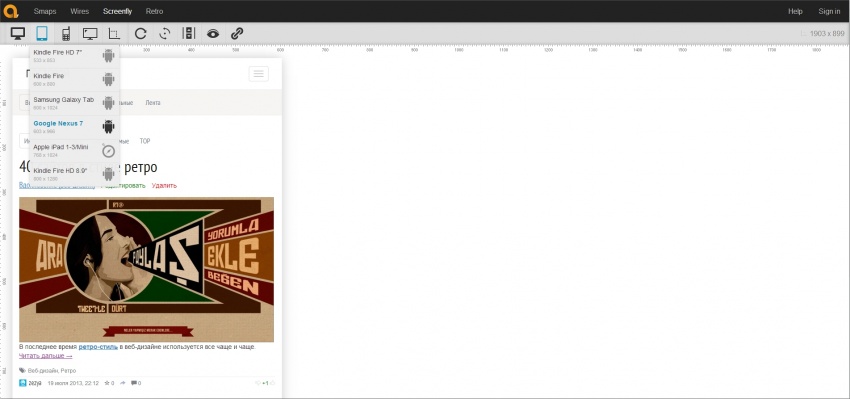
**Responsive Design Bookmarklet**

Интересный сервис для тестирования адаптивной верстки и с отличным дизайном, хотя я не сразу интуитивно понял, как начать работать с ним. Чтобы протестировать страницу, ее нужно сначала открыть в отдельной вкладке браузера, перейти на страницу сервиса и мышкой перетащить кнопку «RWD Bookmarlket» на вкладку в браузере, в которой открыт Ваш сайт. Вверху страницы появится панель с иконками девайсов. При клике на любое устройство рабочая область будет принимать размеры этого устройства. При переключении устройств следует нажать на кнопку обновления на этой же панели.



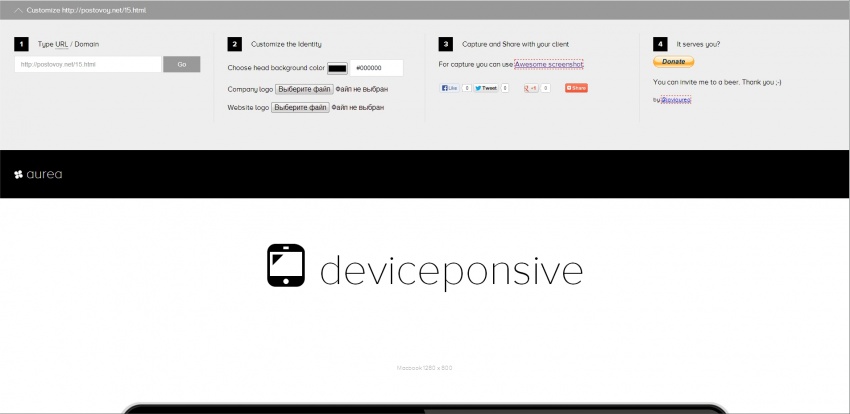
**quirktools.com**

Один из наиболее приятных в использовании сервисов для тестирования адаптивных сайтов. В нем сочетается отличный дизайн, множество возможностей и удобство использования. Инструмент позволяет тестировать страницы на устройствах, разделяемых не только разрешением экрана, но и производителем, моделью девайса и т. д. Плюс ко всему, можно скопировать ссылку и отправить ее заказчику, который заказывал у вас адаптивную верстку.



**Deviceponsive**

Сервис для тестирования и презентации адаптивной верстки. Позволяет отобразить сайт в разных устройствах. В первую очередь, его предназначение не для тестирования, а для презентации работы. Можно настроить цвет фона, загрузить логотип студии и сделать скриншот с помощью любого плагина для браузера, который может захватить всю страницу целиком. Далее можете загрузить скрин в портфолио или отправить заказчику.



## **Сервисы и инструменты для кроссбраузерного тестирования.**

**Adobe BrowserLab**

Adobe BrowserLab — бесплатный инструмент тестирования кроссбраузерности, позволяющий протестировать сайт в современных и некоторых устаревших браузерах, включая различные версии Chrome, Safari, IE и Firefox. Просмотр возможен в нескольких режимах: полноэкранном, в режиме сравнения, а также в режиме «наложенных слоёв». Сервис может получать доступ к страницам, размещённым в интернете, или работать локально, как дополнение к Firebug или Adobe Dreamweaver CS5. Возможность создания наборов браузеров для тестирования очень полезна, если нет необходимости тестирования в устаревших браузерах.

F,{2a6574e0-47ca-482c-afcf-0e282dbebc02}{130},3.125,3.125

**Browsershots**

Browsershots, вероятно, обладает самым широким набором браузеров среди бесплатных инструментов тестирования. Он включает в себя браузеры, работающие в Linux, Windows и BSD. Среди них есть такие, о которых Вы, вероятно, вообще никогда не слышали (например, Galeon, Iceape, Kazehakase или Epiphany). Browsershots позволяет тестировать как в последних версиях каждого браузера, так и в устаревших версиях.

Хотя Browsershots позволяет тестировать в огромном «зоопарке» браузеров, стоит помнить, что, чем больше набор браузеров для тестирования, тем дольше придётся ждать результата. Поэтому стоит остановиться на основных браузерах.

F,{2a6574e0-47ca-482c-afcf-0e282dbebc02}{155},3.125,3.125

**IE Tester**

IE Tester — бесплатный (как для частного, так и для профессионального использования) браузер для Windows, позволяющий тестировать сайт в IE 5.5, IE 6, IE 7, IE 8, IE 9 и IE 10 Previev. Можно тестировать как рендеринг HTML+CSS, так и JavaScript. Доступна только альфа-версия инструмента. Работает на Windows 7, Windows Vista и Windows XP c установленным IE не ниже 7-й версии.

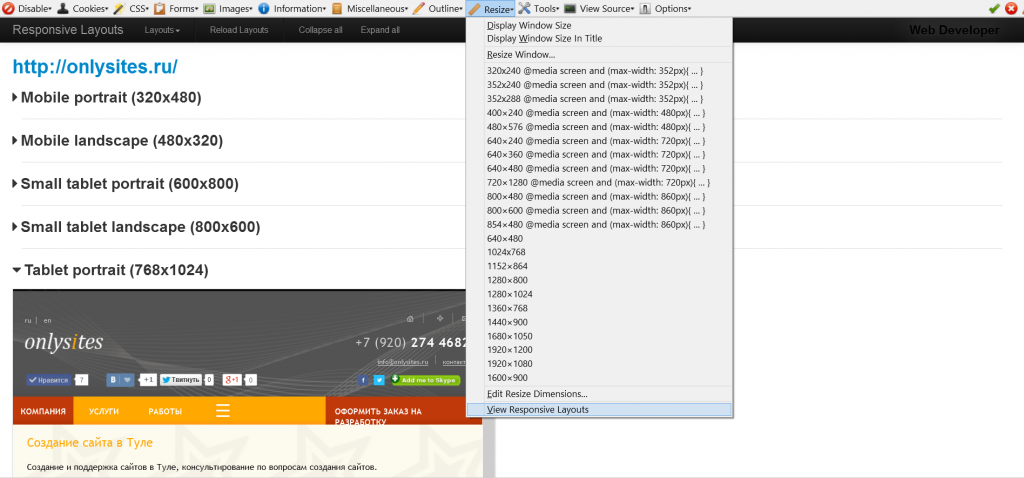
F,{2a6574e0-47ca-482c-afcf-0e282dbebc02}{174},3.125,3.125

**Встроенные инструменты браузеров.**

Firefox, Web Developer Toolbar, вкладка Resize , пункт меню View Responsive Layouts.

#### [**https://addons.mozilla.org/ru/firefox/addon/web-developer/**](https://addons.mozilla.org/ru/firefox/addon/web-developer/)

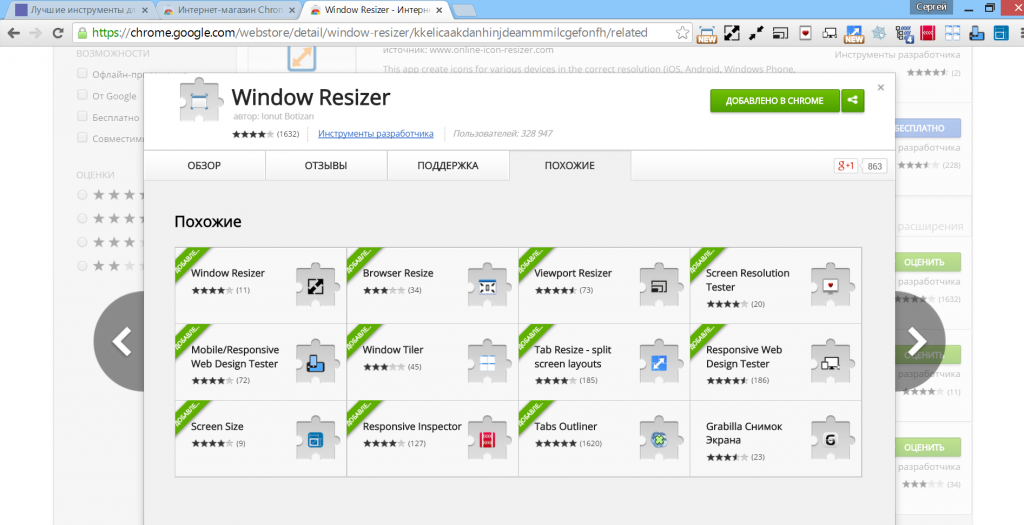
Плагин Web Developer Toolbar предоставляет веб-разработчику множество вариантов для тестирования сайта. Среди них есть отдельная закладка Resize в которой можно через Edit Resize Dimensios добавить расширений для изменения размера окна, или нажав на View Responsive Layouts увидеть как ваш сайт будет отображаться для наиболее популярных расширений.



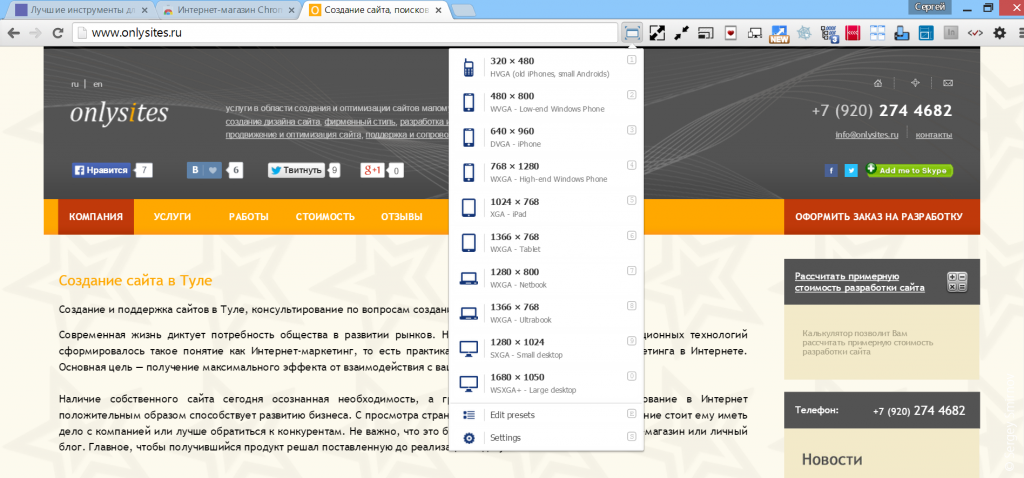
**Google Chrome**

Для этого браузера есть большое количество расширений для тестирования responsive верстки. Пожалуй, даже больше, чем для браузера Firefox.

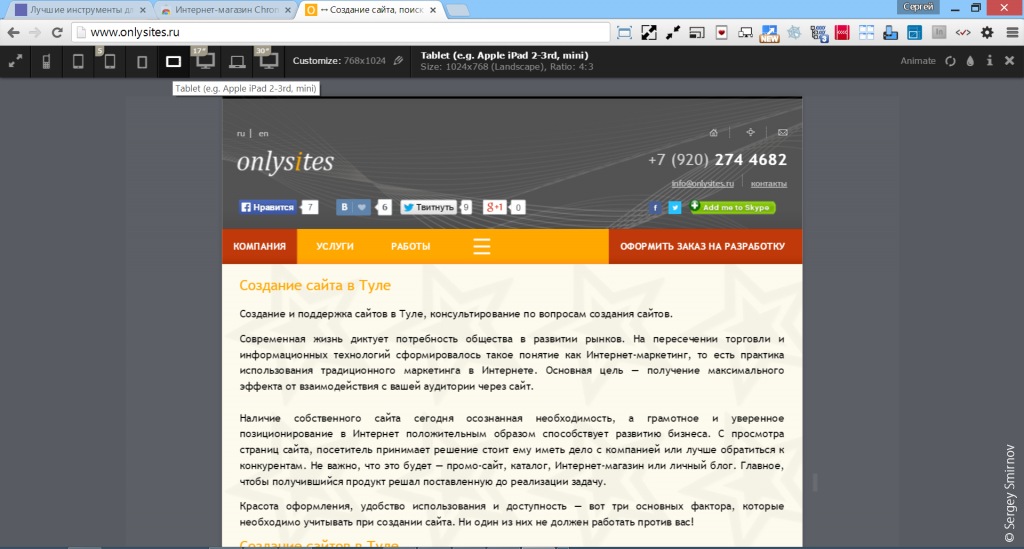
Основное расширение для Google Chrome - это Window Resizer и порядка 12 похожих расширений.



**Resize Window**



**Viewport Resizer**



Расширения, как правило, работают таким образом, что изменяют или размер browser (разрешение), или размер viewport (внутри окна browser).