**ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ МОЛДОВЫ**

**ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ**

**ДЕПАРТАМЕНТ ИНФОРМАТИКИ**

**Мораренко Оксана**

**BOOTSTRAP и ОТЗЫВЧИВЫЙ ВЕБ-ДИЗАЙН**

**0613.5 Прикладная информатика**

**Курсовая работа**

**Директор Департамента: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Капчеля Титу, доцент, доктор наук**

**Научный руководитель: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Плешка Наталья, лектор**

**Автор: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Кишинев, 2023**

**Содержание**

**Введение**...............................................................................................................................

**I. Фронтенд фреймворки для создания веб отзывчивого интерфейса** ......................

1.1Веб интерфейсы для пользователей …………………………………................................................

1.2Viewport в HTML. CSS и mediaQueries.............................................................

1.3Фреймворки для создания и стилизации веб интерфейса………………………………………………....

1.4Отзывчивый веб дизайн

1.5 Bootstrap5 и responsive веб-design

**II.** **Создание адаптивного темплейта веб-сайта**

* 1. Описание задачи..................................................................................................
  2. Описание предназначения сайта...................................................................................................
  3. Описание архитектуры…………………………………………………………………
  4. Описание разработки сайта при помощи Bootstrap
  5. Анализ представления сайта в разных браузерах

**Выводы**...................................................................................................

**Библиография**.........................................................................................................................................

**Anexe**...............................................................................................................................................**.**

ВВЕДЕНИЕ

В современных условиях уже довольно сложно представить себе развитие общества без использования такого блага цивилизации, как Интернет. Сейчас практически каждая организация, клуб или какая-либо личность имеют свой собственный сайт. В основном все сайты основываются на трех основных функциях:

* Информационная: Под информационной функцией понимается, что сайт будет основываться на предоставление, какой-либо информации пользователям, относящейся к тематике сайта.
* Рекламная: Под рекламной функцией понимается, что сайт будет основываться на привлечение пользователей, с помощью рекламы какой-либо деятельности или о предоставляемых услугах.
* Удаленное приложение: Под удаленным приложением понимается, что сайт будет основываться на предоставление функций пользователям, чтобы упростить выполнение каких-либо действий. Но на данное время сайты, стали выполнять не только какую-либо из представленных функций, а совмещать их.

        Сайт туристического агентства «MOXI». Основой сайта, является предоставление информации о деятельности агентства, о том какие туры оно предоставляет и контактах. Что касается интерфейса сайта, то можно сказать, что приятный, не нагруженный сайт, удобный в использовании, правильно подобранная цветовая гамма, которая хорошо сочетается с тематикой сайта.

**I. Фронтенд фреймворки для создания веб отзывчивого интерфейса** ......................

1.1Веб интерфейсы для пользователей …………………………………................................................

**Веб-интерфейс** — это совокупность средств, при помощи которых пользователь взаимодействует с веб-сайтом или любым другим приложением через браузер. Веб-интерфейсы получили широкое распространение в связи с ростом популярности всемирной паутины и соответственно — повсеместного распространения веб-браузеров.

Распространение веб-интерфейса связано с ростом популярности сети интернет и повсеместного использования веб-браузеров. Чтобы страница могла соответствовать требованиям веб-интерфейса, она должна иметь одинаковый внешний вид с одинаковым функциональном при работе в различных браузерах.

Классический метод [создания веб-интерфейсов](https://gusarov-group.by/dev/sozdanie-dizajna-interfejsa/) — использование кода HTML с CSS и JavaScript’a. Но различная реализация HTML, CSS, DOM и пр. в браузерах может вызывать проблемы в разработке веб-приложений. Работе интерфейса также могут мешать возможность пользователя настраивать такие параметры браузера, как шрифт, цвет, поддержка сценариев и др.

Есть и другой, менее популярный и универсальный способ создания веб-интерфейса —  использование Adobe Flash, Silverlight или Java-апплетов. Большинство современных браузеров поддерживает эти технологии, не без помощи плагинов, работать с Flash- или Java-приложениями можно легко.

**Для чего применяют веб-интерфейсы?**

Web interface применяют для работы с различными онлайн-сервисами. Это может быть обычная электронная почта или специальные системы веб-аналитики. Иногда на сайтах веб-интерфейс именуют «Личным кабинетом».

Само понятие говорит за себя. Так, приставка «веб» означает удаленную работу, вдали от компьютера пользователя, на интернет-сервере. Взаимодействие с самим сервисом происходит через «интерфейс» (вторая часть слова) — специальная графическая оболочка, содержащая кнопки, окна, поля для заполнения и прочих элементов.

К примеру, веб-интерфейс электронной почты.

Еще 15-20 лет назад электронная почта работала через программу-клиент. Пользователь просто устанавливал ее на свой компьютер и мог отправлять и получать письма. Но постепенно функции приложения перенесли в веб, что упростило работу самой почты.

Веб-интерфейс электронной почты работает также, как программа-клиент. Он принимает, отправляет и обрабатывает письма, перенаправляет и сортирует их. Однако веб-интерфейс почты содержит еще и папки для разделения почты, то есть письма сортируются в исходящие, входящие, удаленные, спам и пр.

А получить доступ к почтовому интерфейсу можно только после введения пароля, указанного при регистрации ящика.

Обзор основных элементов пользовательского интерфейса (UI) и пользовательского опыта (UX) поможет лучше понять, как эти составляющие влияют на взаимодействие пользователей с веб-сайтами и приложениями. Вот краткий обзор каждого из этих элементов:

1.Элементы пользовательского интерфейса (UI):

* -Текстовые элементы: Заголовки, подзаголовки, параграфы, кнопки, ссылки и другие текстовые элементы используются для передачи информации и взаимодействия с пользователем.
* -Графические элементы: Иконки, изображения, логотипы и другие графические элементы добавляют визуальный интерес и помогают пользователю лучше понять контекст и функциональность.
* -Формы и поля ввода: Поля для ввода текста, флажки, кнопки выбора, выпадающие списки и другие формы позволяют пользователям взаимодействовать с сайтом и передавать информацию.
* -Навигационные элементы: Меню, панели навигации, ссылки и кнопки навигации обеспечивают легкую навигацию по веб-сайту и помогают пользователям быстро находить нужную информацию.
* -Раскладка и сетка: Организация содержимого на странице с помощью разметки и сетки позволяет создать понятную и удобную структуру, упрощая восприятие пользователем контента.

2.Пользовательский опыт (UX):

* -Исследование пользователей: UX-дизайнеры проводят исследования и анализ поведения пользователей, чтобы лучше понять их потребности, ожидания и проблемы.
* -Пользовательские пути и взаимодействие: UX-дизайнеры создают логику и структуру взаимодействия пользователя с веб-сайтом, оптимизируя пользовательские пути, потоки и задачи.
* -Информационная архитектура: UX-дизайнеры организуют информацию и контент на сайте таким образом, чтобы пользователи легко могли найти нужную информацию и выполнить задачи.
* -Визуальный дизайн: UX-дизайнеры разрабатывают эстетически приятные и функциональные интерфейсы с помощью цветовой схемы, шрифтов, композиции и других визуальных элементов.
* -Проверка удобства использования: UX-дизайнеры проводят тестирование интерфейса с реальными пользователями, чтобы выявить слабые места и внести улучшения в дизайн.

В целом, элементы пользовательского интерфейса и пользовательский опыт взаимосвязаны и влияют друг на друга. Хорошо спроектированный пользовательский интерфейс и удобный пользовательский опыт помогают пользователям взаимодействовать с веб-сайтом более эффективно и приятно.

1.2 Viewport в HTML. CSS и mediaQueries:

Viewport — это область, которую видит пользователь на экране, когда заходит на страницу сайта с любого устройства.

Раньше все было просто: разрешение экранов более или менее одинаковое, делай себе одну ширину макета, и никто не будет жаловаться. Сейчас стало труднее: пользователи заходят на страницы с разных устройств, разрешение экрана сильно различается.

Чтобы пользоваться сайтами было удобно, нужно правильно масштабировать страницы. Для этого используется метатег viewport. Он не делает верстку адаптивной. Его предназначение — контроль масштаба отображения страницы.

**Роль CSS и mediaQueries в создании отзывчивого дизайна**.

CSS и mediaQueries играют важную роль в создании отзывчивого дизайна веб-интерфейсов. Вот как они содействуют достижению отзывчивости:

1.CSS (Cascading Style Sheets):

* Разделение структуры и представления: CSS позволяет разделить структуру HTML-кода от его внешнего вида. Это позволяет разработчикам изменять визуальное оформление элементов без изменения самой структуры документа.
* Стилизация элементов: CSS предоставляет широкий набор свойств и селекторов, которые позволяют стилизовать элементы веб-страницы. Это включает изменение цвета, размера, шрифта, отступов, границ и других визуальных атрибутов элементов.
* Медиа-запросы: CSS позволяет использовать медиа-запросы для определения различных условий отображения страницы в зависимости от характеристик устройства или размера экрана, на котором происходит просмотр. Медиа-запросы позволяют применять разные стили или изменять расположение элементов для достижения отзывчивого дизайна.

2.MediaQueries:

* Определение точек разрыва (breakpoints): MediaQueries позволяют разработчикам задать точки разрыва для разных размеров экранов или устройств. Такие точки могут быть определены на основе ширины экрана, плотности пикселей или других характеристик устройства.
* Адаптивное применение стилей: С помощью MediaQueries можно определить различные стили для разных точек разрыва. Это позволяет адаптировать внешний вид и расположение элементов на странице в зависимости от размера экрана или устройства.
* Мобильное первое (Mobile-first) и Десктопное первое (Desktop-first) подходы: MediaQueries поддерживают оба подхода к разработке отзывчивого дизайна. В мобильном первом подходе стили для мобильных устройств определяются сначала, а затем добавляются стили для более крупных экранов. В десктопном первом подходе стили для больших экранов определяются сначала, а затем добавляются стили для меньших экранов.

Сочетание CSS и mediaQueries позволяет разработчикам создавать отзывчивый дизайн, который адаптируется к различным размерам экранов и устройствам. Это позволяет пользователям получать оптимальный пользовательский опыт, независимо от того, на каком устройстве или экране они просматривают веб-сайт.

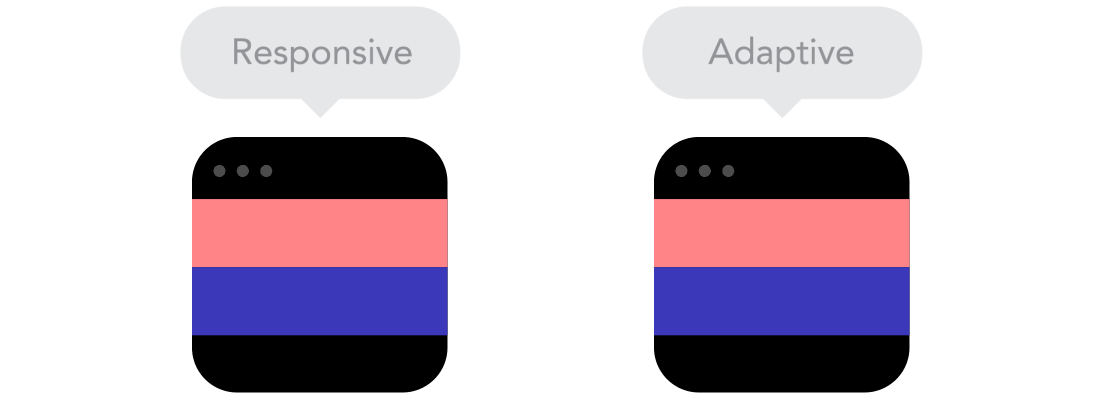
1.3 Фреймворки для создания и стилизации веб-интерфейса:

1.4 Отзывчивый (responsive) веб-дизайн:

Отзывчивый дизайн — отличное решение проблемы корректного отображения сайта на разных экранах. Однако новичкам зачастую трудно понять основы, обучаясь только по книгам/статьям. С каждым днём появляется всё больше различных устройств, имеющих разные размеры экрана, поэтому создание дизайна в пикселях и только для настольных компьютеров/смартфонов остаётся в прошлом. Именно поэтому сейчас стоит изучить принципы отзывчивого дизайна — дизайна, совмещающего в себе адаптивность и резиновость.

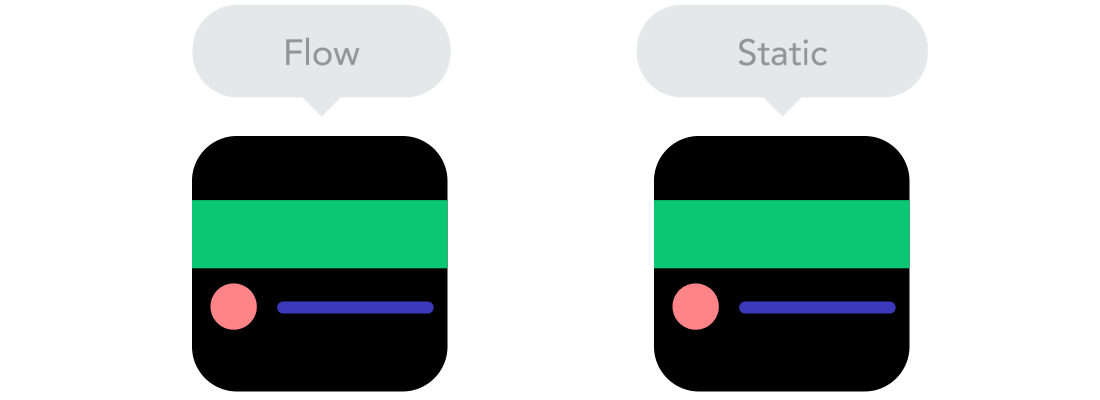
## Отзывчивый vs Адаптивный веб-дизайн

Может показаться, что это одно и то же. Однако оба этих вида дизайна дополняют друг друга и подходят каждый для своей задачи.



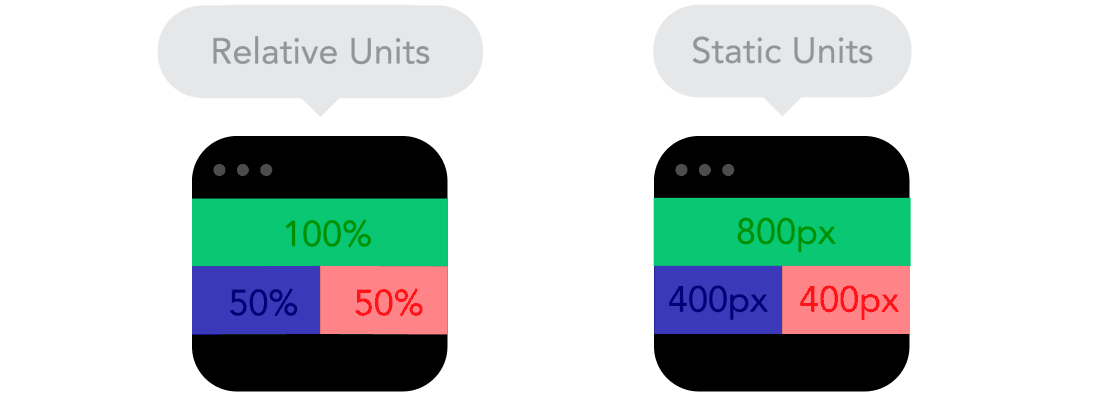
## Поток

Когда размер экрана уменьшается, содержимое страницы начинает занимать больше высоты, и элементы начинают смещаться вниз. Такое поведение называется потоком. Это может быть труднодоступным для понимания, если раньше вы использовали фиксированные размеры элементов. Однако, когда вы начнёте использовать поток, вы поймёте принцип его работы.



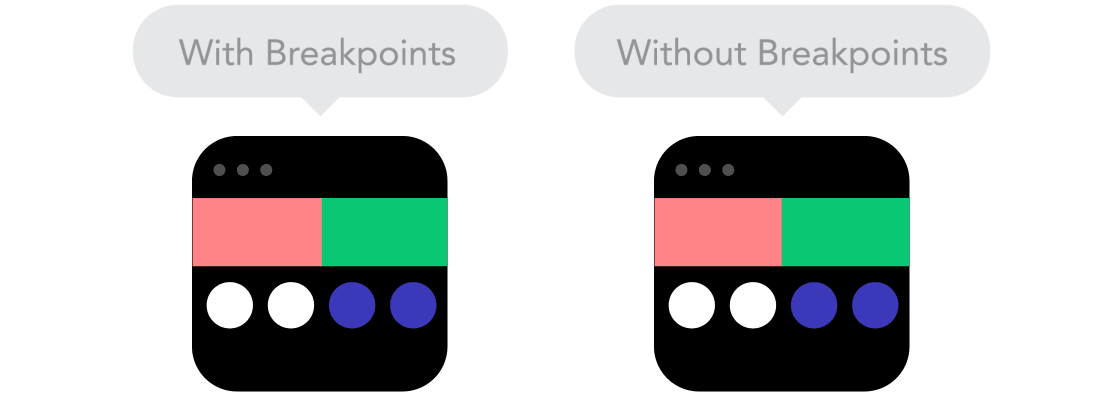
## Относительные единицы измерения

Область просмотра страницы может быть монитором, экраном мобильного или чем угодно ещё. Плотность пикселей на разных экранах также разная, поэтому нам нужны гибкие единицы измерения, работающие везде. Отзывчивый дизайн — именно тот случай, когда относительные единицы измерения вроде процентов становятся действительно полезными. С помощью процентов мы можем задать блоку ширину в 50%, и на любом устройстве он будет занимать лишь половину экрана.



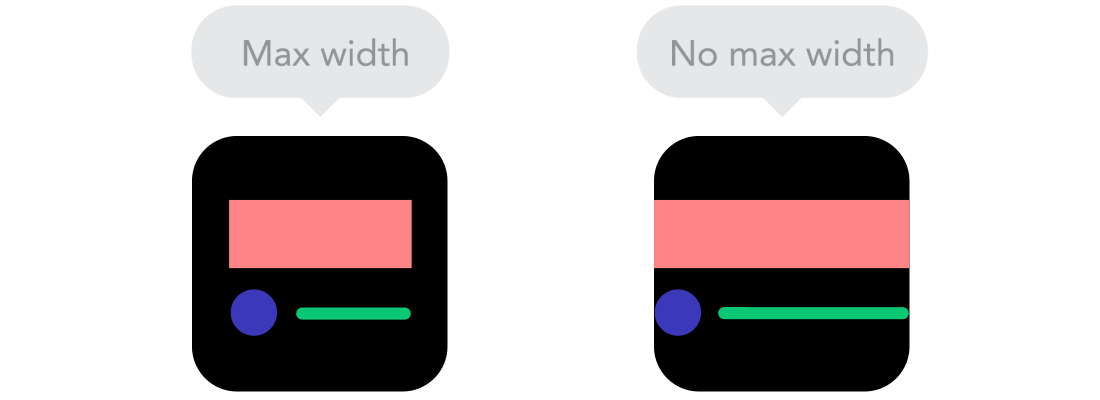
## Контрольные точки (Breakpoints)

Контрольные точки позволяют менять расположение блоков на странице только в случае использования экрана с определёнными размерами. Например, на настольных компьютерах у сайта будет три колонки, а на мобильных — только одна. Как вы расставите контрольные точки, зависит от поведения контента. Если содержимое страницы «ползёт», то определённо следует добавить контрольную точку. Однако, контрольные точки следует использовать с осторожностью — вы можете в них запутаться.



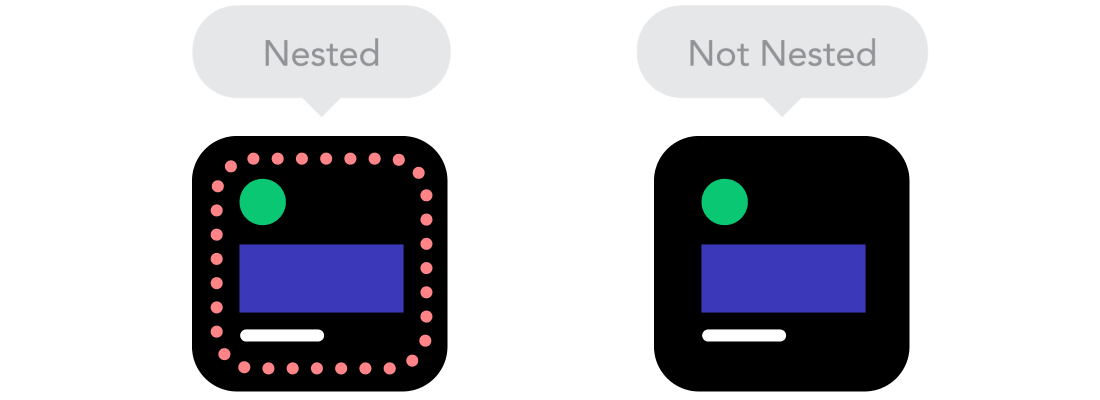
## Max- и min-значения

Контент, занимающий всю ширину экрана — это здорово, если он отображается на мобильном. А если вы откроете страницу через ваш телевизор? Вряд ли увиденная картина обрадует вас. Поэтому здравым решением будет использование минимальных и максимальных значений. Например, если задать блоку свойства `width: 100%` и `max-width: 1000px`, то он будет отображаться на весь экран, если ширина экрана меньше 1000 пикселей; в противном случае, блок будет занимать 1000 пикселей.



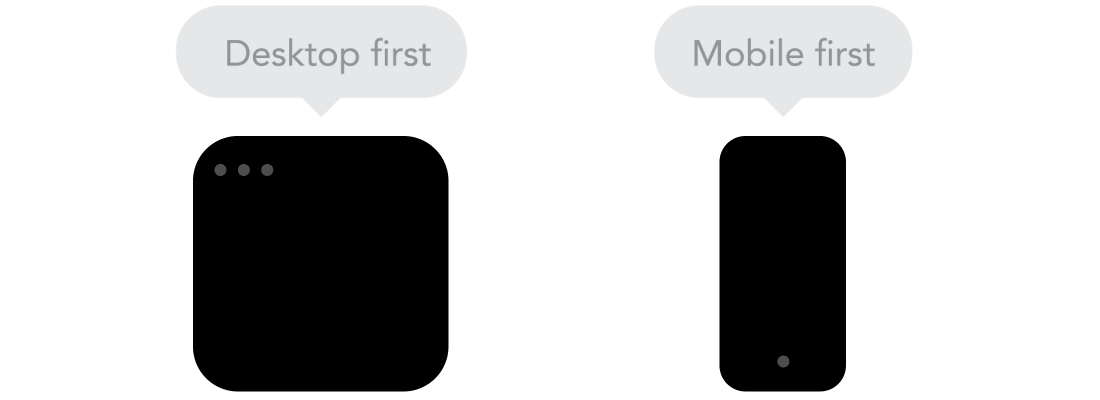
## Вложенные объекты

Помните position: relative? Если у вас будет много элементов, зависящих от расположения других элементов, то их будет тяжело контролировать. Намного проще и правильнее обернуть эти элементы в один контейнер. Кстати, это тот случай, когда статичные единицы измерения вроде пикселей помогут вам. Они полезны для содержимого, которое вы не хотите адаптировать к размеру экрана — например, это может быть логотип или кнопка.



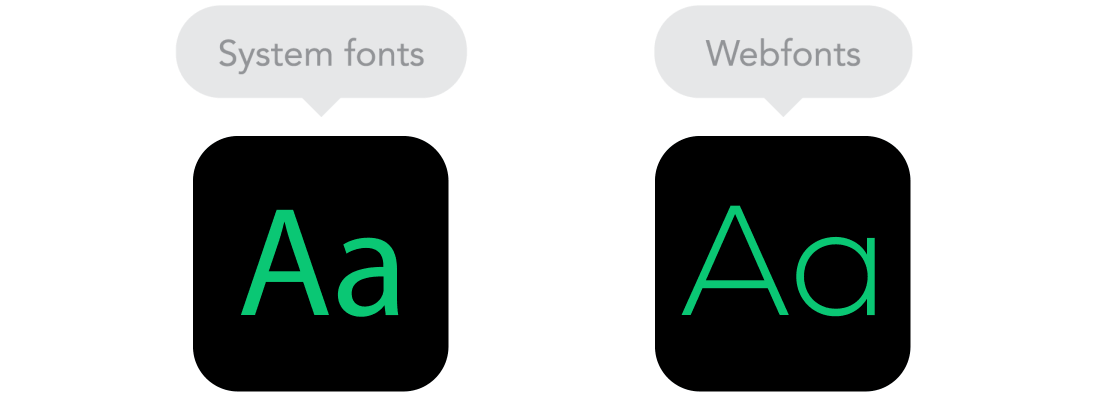
## Desktop или mobile first

С технической стороны нет никаких отличий: вы можете писать базовую разметку для мобильных, и расставлять ключевые точки для десктопов (mobile first) и наоборот. Часто люди не знают, какой подход выбрать. Подумайте как следует и выберите подходящий вам подход.



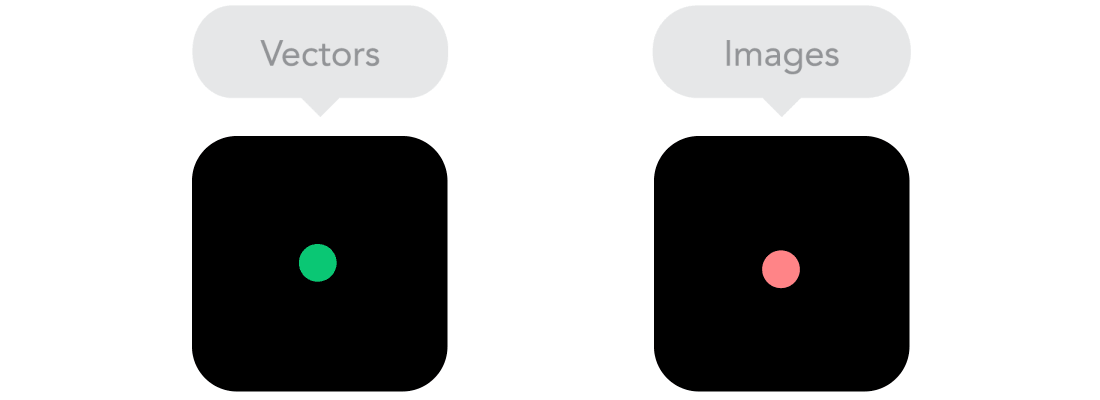
## Веб-шрифты vs системные шрифты

Хотите использовать на своём сайте круто смотрящуюся гарнитуру Futura или Didot? Используйте веб-шрифты! Хоть они и выглядят красиво, не стоит забывать, что каждый подключённый шрифт будет загружен. Соответственно, чем больше шрифтов, тем медленнее загружается страница. С другой стороны, системные шрифты загружаются моментально за исключением случаев, когда пользователь не имеет локально установленного шрифта, используемого на странице. В таких случаях браузер будет использовать шрифт по умолчанию.



## Растровые vs векторные изображения

Имеет ли ваше изображение множество мелких деталей и впечатляющих эффектов? Если да, то используйте растровый формат. В противном случае используйте векторный формат. Для растровых изображений используйте форматы jpg, png или gif, для векторных лучшим выбором будут SVG и иконочные шрифты. Каждый из форматов имеет свои преимущества и недостатки. В любом случае, помните о размере изображений — ни одна картинка не должна попасть в онлайн, не будучи оптимизированной (сжатой). Векторные изображения зачастую избавлены от лишнего размера, однако они не поддерживаются старыми браузерами. Также стоит помнить, что если векторное изображение содержит много деталей, то оно может весить больше растрового.



**II.** **Создание адаптивного темплейта веб-сайта**

* 1. Описание задачи..................................................................................................
  2. Описание предназначения сайта

Веб-сайт нашего туристического агентства сфокусирован на то, чтобы посодействовать нашим клиентам в планировании, а также бронировании поездок. Он считается основным онлайн-ресурсом, на котором люди имеют все шансы отыскать требуемые данные касательно туров, гостиницах, перелетах, а также иные предложения, предлагаемые нашим агентством.

Основные функции сайта:

* + 1. Обеспечение данных касательно туров, а также путешествиях: На сайте можно найти обширные данные касательно разных известных туристических направленностях, достопримечательностях, зонах проживания, а также туровых пакетах. Наши клиенты имеют все шансы исследовать, пронаблюдать и рассмотреть самостоятельно разнообразные виды, а также подобрать более оптимальный вариант с целью собственных нужд и предпочтений.
    2. Онлайн-бронирование: Наш сайт дает возможность нашим покупателям резервировать турпоездки, гостиницы, авиабилеты, а также прочие услуги, конечно же, онлайн. Мы рекомендуем комфортную концепцию поиска и бронирования, что дает возможность покупателям подбирать даты, номера в отелях , а также прочие вспомогательные услуги.
    3. Поддержка и консультации: Наш сайт предоставляет клиентам возможность обратиться к нам через сайт для получения консультации и ответов на вопросы. Мы предоставляем контактные данные и формы обратной связи, чтобы клиенты могли связаться с нами легко и удобно.
    4. Представление специализированных услуг и промоакций: Рекомендуем наш веб-сайт с целью представления супер предложений, бонусов, а также промоакций, которые находятся в открытом и легком доступе для наших клиентов. Это позволяет нам привлечь внимание и предложить особые выгодные условия для путешествий.

Целью нашего сайта является создание удобной и информативной платформы, где клиенты могут получить все необходимые сведения о турах и бронировать путешествия без лишних сложностей. Мы стремимся предоставить высокий уровень обслуживания и помочь нашим клиентам сделать их путешествия незабываемыми и беззаботными.

* 1. Описание архитектуры…………………………………………………………………
  2. Описание разработки сайта при помощи Bootstrap
  3. Анализ представления сайта в разных браузерах