**Разработков веб-сайтов**

**[слайд 0]**

Билл Гейтс сказал: “Если вашего бизнеса нет в интернете, значит вас нет в бизнесе”

**[слайд 1]**

Каждый день, мы сталкиваемся с результатом области разработки веб-сайтов

**[слайд 2]**

и начиная рассматривать тему разработки веб-сайтов, мы должны поднять несколько важных тем:

**[слайд 3]**

области применения, технологии и как вообще начать?

**[слайд 4]**

Вся разработка сайтов в той или иной мере вращается вокруг трех вещей

**[слайд 5]**

Эти три вещи - HTML (Hypertext markup language), CSS (Cascading Style Sheets), JS (JavaScript)

**[слайд 6]**

Hypertext markup language или HTML

**[слайд 7]**

Это язык, который определяет структуру контента веб-страницы

**[слайд 8]**

Язык использует для структурирования теги, так например мы имеем открывающий и закрывающий теги, которые образуют элемент для структурирования и организации текста

**[слайд 9]**

Помимо парных тегов, существует и одинарные

**[слайд 10]**

Имея все это на руках и обычный текст, мы можем его структурировать и организовать

**[слайд 11]**

Так например h1 (header1) является тегом заголовка первого уровня

**[слайд 12]**

p (paragraph) является тегом параграфа

**[слайд 13]**

И самое важное - тег a (anchor), который является тегом ссылки

**[слайд 14]**

Современный html предоставляет теги, которые способны работать с таким контентом как видео, аудио и даже что-либо рисовать произвольно

**[слайд 15]**

Типичный html элемент имеет открывающий и закрывающий тег, а в центре контент, это может быть текст

**[слайд 16]**

Или другие html элементы, которые называют детьми

**[слайд 17]**

Также открывающий тег может иметь один или более атрибутов. Атрибуты представляет из себя пару ключ-значение. Это позволяет кастомизировать поведение элемента

**[слайд 18]**

Cascading Style Sheets или CSS - это язык таблиц стилей, которые делает страницу красивой

**[слайд 19]**

По умолчанию сайт, написанный на html представляет из себя лишь структуру или скелет

**[слайд 20]**

Но css делает сайт презентабельным

**[слайд 21]**

С помощью описания свойств элемента. Так например вы можете описать

**[слайд 22]**

Размер

**[слайд 23]**

Цвет заднего фона

**[слайд 24]**

Цвет и размер текста

**[слайд 25]**

Выравнивание

**[слайд 26]**

И например закругление краев

**[слайд 27]**

JavaScript или js это

**[слайд 28]**

Высокоуровневый

**[слайд 29]**

Однопоточный

**[слайд 30]**

Автоматизировано собирающий мусор

**[слайд 31]**

Интерпретируемый или JIT (just in time) компилируемый

**[слайд 32]**

Прототипный

**[слайд 33]**

Мультипарадигменный

**[слайд 34]**

Динамический язык

**[слайд 35]**

Использующий неблокирующий цикл событий

**[слайд 36]**

С крайне простой целью - быть простым для изучения скриптовым языком

**[слайд 37]**

Язык является неким мозгом веб-сайта

**[слайд 38]**

А свое огромное влияние он получил относительно недавно

**[слайд 39]**

Чаще всего мы видим его участие в разработке интерфейсных приложений

**[слайд 40]**

Ведь именно JS

**[слайд 41]**

Помимо WebAssembly

**[слайд 42]**

Имеет нативную поддержку всеми браузерами

**[слайд 43]**

При этом мы можем иметь одну языковую базу, для написания серверных-приложений

**[слайд 44]**

Мобильных приложений

**[слайд 45]**

А также компьютерных приложений

Именно это и делает язык таким удобным

**[слайд 46]**

Но - все это лишь поверхностное описание широких возможностей сферы веб-разработки и является некоторой частью основы. Помимо базы стоит учитывать знание безопасности, умение работать с инструментами и различными фреймворками, которые образуют современные тенденции веб-разработки

**[слайд 47]**

Возвращаюсь к нашей теме, можно выделить обобщенную схему создания сайта.

Начиная от идеи и заканчивая сборка проекта, далее мы пробежимся по каждому из этих элементов. При этом проецируя на реальный проект.

**[слайд 48]**

Идея = концепция = суть сайта

Если говорить про один из моих проектов, то идея следующая…

**[слайд 49]**

После идеи следует документировать все вышесказанное, дабы оно не потерялось как в памяти, так и направляло ход разработки.

Для задания вектора развития проекта существует документ дорожной карты, который поэтапно описывает весь процесс, а для описания всех технический деталей можно сделать технический документ.

**[слайд 50]**

Следующим на очереди идет дизайн и дизайн на самом деле разделяется на три направления, которые мы не затронем, но о которых я упомяну.

UX - user experience

UI - user interface

TD - tech design

**[слайд 51]**

Одну позицию делят такие понятия как frontend и backend, в свою очередь клиент и сервер соответственно, которые тесно связаны друг с другом и лишь в такой связи образуют полноценный и защищенный функционал сайта.

Но, каким же образом они связаны?

**[слайд 52]**

Дело в том, что для передачи информации между клиентами и сервером используется сетевой протокол HTTP или Hypertext transport protocol, который очень удобен для передачи различной информации. Суть его проста - клиент отправляет запрос, а сервер отвечает на него

Следуя всему вышесказанному, текущую идеи сайта, можно представить следующим образом…

**[слайд 53]**

Серверу необходимо регулярно запрашивать актуальные данные рынка драгоценных металлов и кэшировать их в базе данных.

**[слайд 54]**

В свою очередь сервер, при запросе стоимости радиодетали от клиента, на основе имеющихся у него данных, сможет посчитать данную стоимость и ответить этой информацией клиенту.

**[слайд 55]**

Тестирование является немаловажным аспектом процесса разработки, так как предоставляет возможность выявить существующие проблемы перед конечным формированием продукта

**[слайд 56]**

Сборка является заключительным этапом и подразумевает создает всех необходимых условий для дальнейшего удобного обслуживания продукта, это могут быть системы управления версиями, обновлениями, а также какие-либо виртуализированные среды запуска.

**[слайд 57]**

Заключение