

# Циклы разработки Web-приложений

Лекция №1

# Web-приложение. Что это?

- **Веб-приложение** — клиент-серверное приложение, в котором клиентом выступает браузер, а сервером — web-сервер. Логика веб-приложения распределена между сервером и клиентом, хранение данных осуществляется, преимущественно, на сервере, обмен информацией происходит по сети. Одним из преимуществ такого подхода является тот факт, что клиенты не зависят от конкретной операционной системы пользователя, поэтому web-приложения являются межплатформенными сервисами.

# Web-приложение. Клиентская часть

- Клиентская часть веб приложения - это графический интерфейс. Это то, что вы видите на странице. Графический интерфейс отображается в браузере. Пользователь взаимодействует с веб-приложением именно через браузер, кликая по ссылкам и кнопкам.
- frontend

# Web-приложение. Серверная часть

- Серверная часть веб-приложения - это программа или скрипт на сервере, обрабатывающая запросы пользователя (точнее, запросы браузера).
- Чаще всего серверная часть веб-приложения программируется на PHP. При каждом переходе пользователя по ссылке браузер отправляет запрос к серверу.
- Сервер обрабатывает этот запрос, вызывая некоторый PHP-скрипт, который формирует веб-страничку, описанную языком HTML, и отправляет клиенту по сети. Браузер тут же отображает полученный результат в виде очередной веб-страницы.
- backend

# Web-приложение. База данных

- База данных (БД, или система управления базами данных, СУБД) - программное обеспечение на сервере, занимающееся хранением данных и их выдачей в нужный момент. В случае форума или блога, хранимые в БД данные - это посты, комментарии, новости, и так далее. База данных располагается на сервере. Серверная часть веб-приложения (то есть, PHP скрипт) обращается к базе данных, извлекая данные, которые необходимы для формирования страницы, запрошенной пользователем.

# Web-приложение. Средства разработки

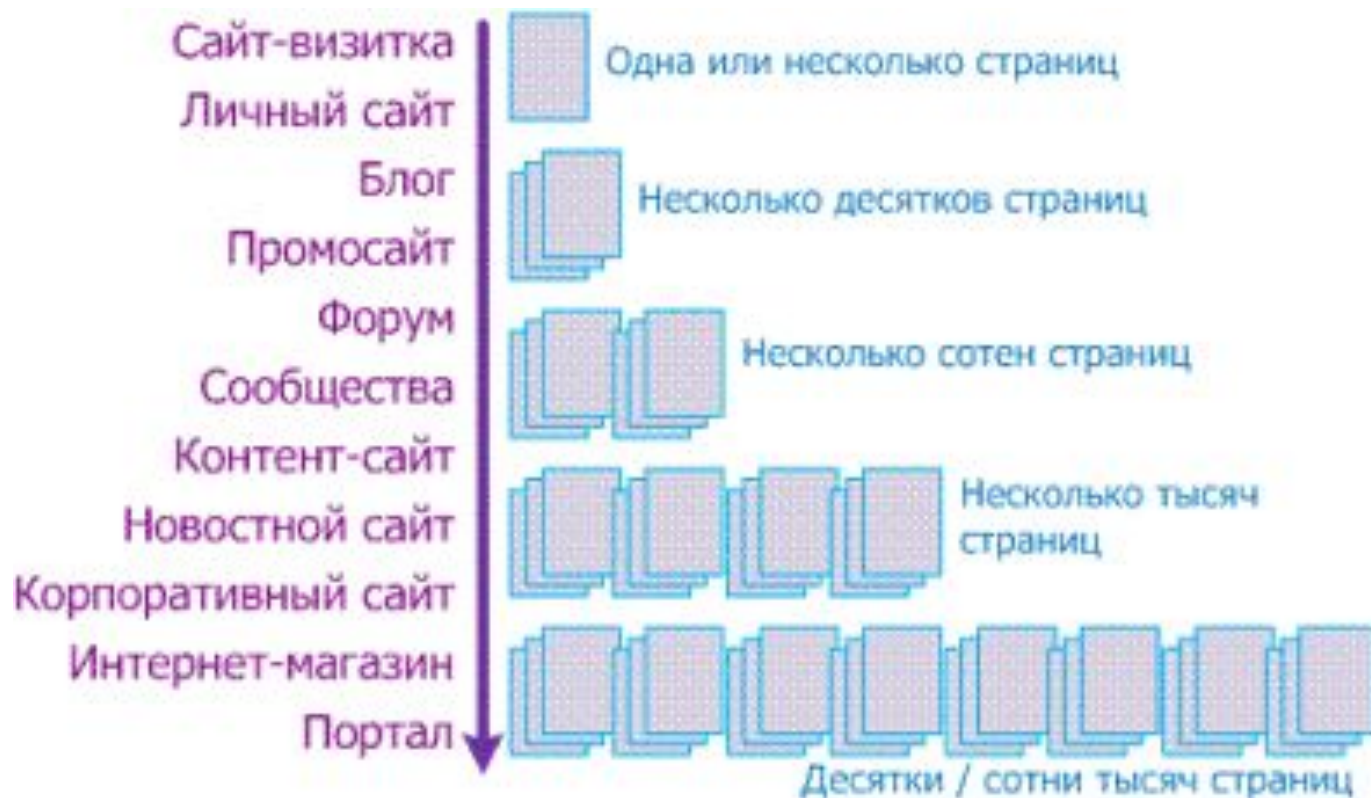
## Front-end

- ✓ HTML
- ✓ CSS
- ✓ JS




## Back-end

- Php, Perl
- Ruby
- Java
- Платформа .Net

# Классификация web-сайтов



# Разработка Web-приложения

- Задача  Решение
- Сложная задача  Сложное решение
- Сложное решение  Процесс

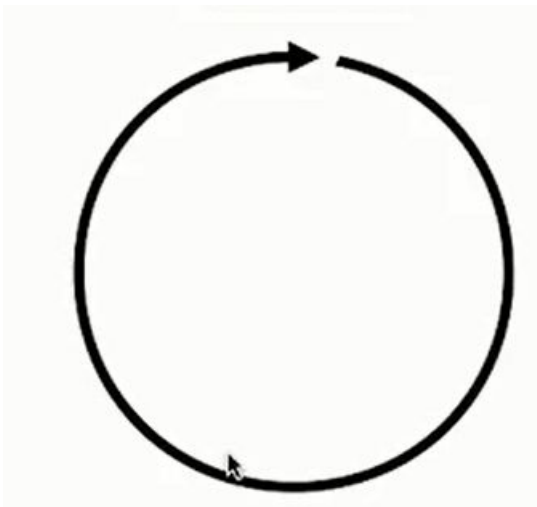


# Процесс разработки

Сайт-визитка:



Промо-сайт:



Сервис:



# Цикл разработки:

- постановка задачи
- дизайн
- разработка
- тестирование
- деплой
- *маркетинг*
- *поддержка*

# Постановка задачи

- **Кто ставит задачу:**

Руководство компании

Руководитель сервиса

Менеджер

- **Что на выходе:**

Продуктовые требования

Фич-листы

Тз

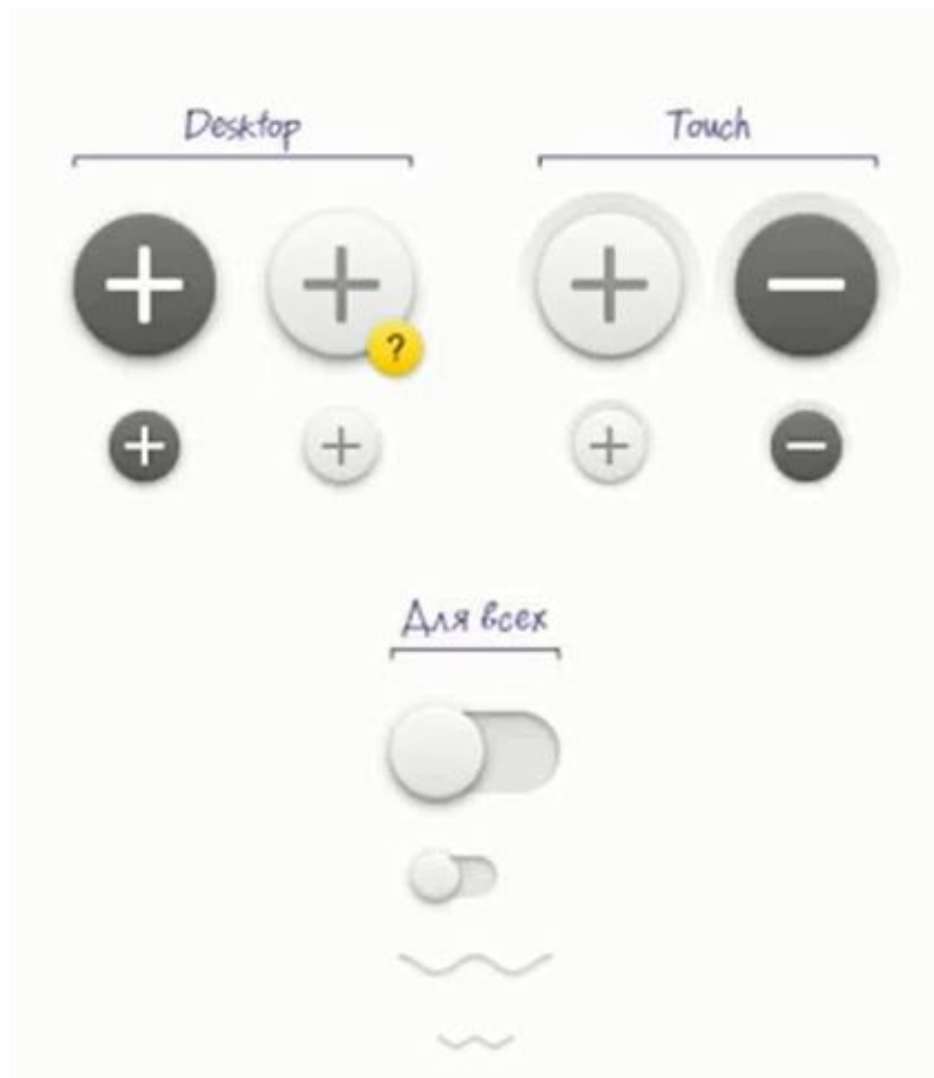
- **Разработчик:**

В курсе

Подает идеи

# Дизайн

- **Кто :**  
дизайнеры  
менеджеры  
все подряд
- **Что на выходе:**  
макеты  
гайды
- **Разработчик:**  
Подает идеи  
Объясняет



# Разработка

- **Кто :**

Верстальщики

Программисты

- **Что на выходе:**

Верстка, шаблоны, код, шаблоны, софт,  
логи...

- **Разработчик:**

Разрабатывает

Первично тестирует

# Progressive Enhancement

- Прогрессивное улучшение предполагает, что веб-интерфейсы должны создаваться поэтапно, циклически, от простого к сложному. На каждом из этапов должен получаться законченный веб-интерфейс, который будет лучше, красивее и удобнее предыдущего. Всего этапов 4:
  - HTML этап
  - CSS этап
  - CSS3 этап
  - JavaScript этап

# Graceful degradation

- возможность работы при отключенном JavaScript,
- аккуратное отображение интерфейса в браузере, не поддерживающем новые свойства CSS3,
- адекватное отображению сайта при отключенных изображениях.

В каждом из этих случаев работа пользователя с интерфейсом будет в принципе возможна, хотя и не так удобна.

# Тестирование

- **Кто :**

тестировщики

Программисты

администраторы

- **Что :**

Автоматизированное

Нагрузочное

Функциональное

- **Разработчик:**

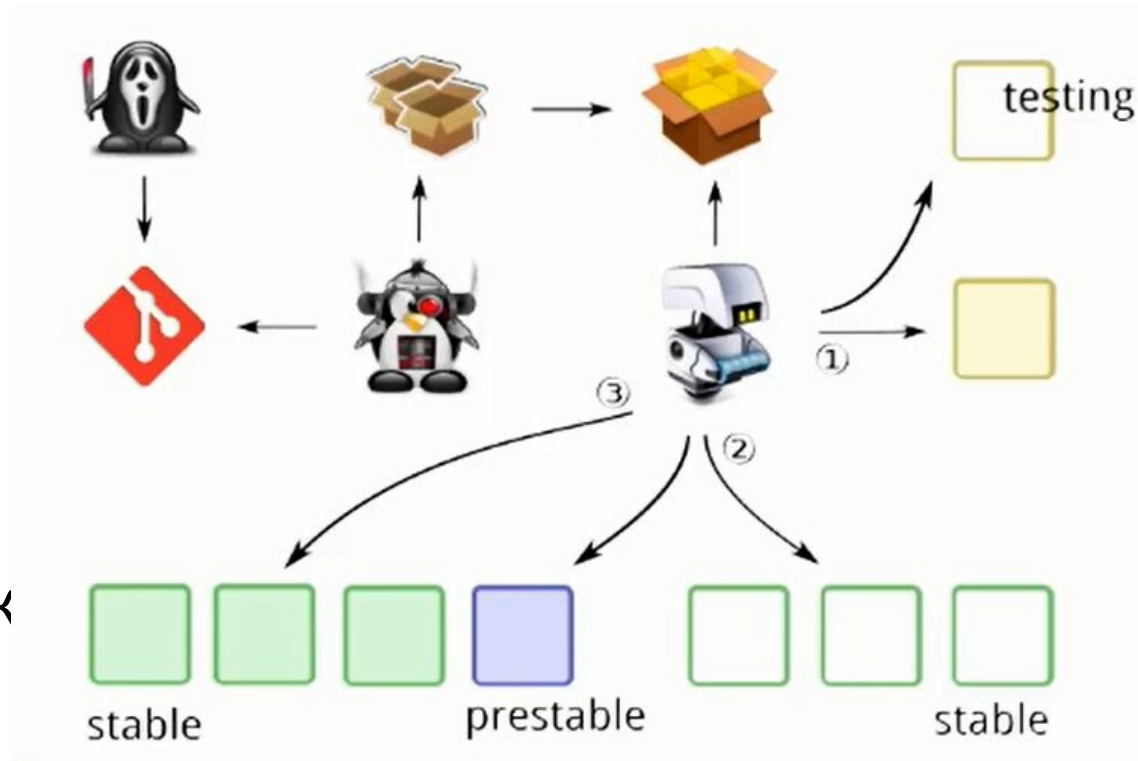
исправляет ошибки

помогает найти другие ошибки



# Деплой

- **Кто :**  
Разработчики  
администраторы
- **Что :**  
Источник данных  
Обработка данных  
Шаблоны/статистик
- **Разработчик:**  
исправляет ошибки  
помогает



# Маркетинг

- **Кто :**

Копирайтеры

Маркетологи

- **Что :**

Группы в социальных сетях

Посты в блогах

Твиттер и тд...

- **Разработчик:**

Следит за корректностью текста

# Поддержка

- **Кто :**

Call-центр  
специалисты

- **Что :**

Консультации  
Баг-репорты

- **Разработчик:**

исправляет  
помогает

# Что потом

## Новый цикл:

- Доработка
- Ошибки
- Исправление
- Эксперименты
- Новые функции

# Примеры циклов разработки

## Эксперименты

размытое ТЗ, несколько реализаций, фидбек

## Прототипы

мусорный код, юзабилити-тестирование

## Бета-версии

быстрый релиз, короткий фич-лист, баги

# Способы разработки web-приложения

- 1) «руками» (конечно, с использованием библиотек)
- 2) wix и другие конструкторы сайтов
- 3) разработка с использованием CMS

# Что такое CMS

- CMS «Content Management Software» («программное обеспечение для управления содержимым»). В нашей стране принято последнюю букву «S» расшифровывать как «System», а по-русски это обычно звучит как «Система управления контентом». Иногда употребляется более простое название - "движок сайта"

# Какие есть CMS?

- Wordpress
- Joomla
- Drupal
- Bitrix
- .....