**Cms**

**Конструктор CMS WORDPRESS**

**Wordpress** - это свободно распространяемая, соответствующая стандартам, быстрая, легкая и бесплатная платформа для персонального блоггинга с практичными настройками, с гибким и настраиваемым ядром.

**Архитектура**: ядро + тема + дополнение

Преимущества:

* Наиболее популярная
* Бесплатная
* Широкий набор плагинов, тем, виджетов
* WYSIWYG редактор
* Технический опыт не обязателен

Недостатки:

* Как использовать все возможности правильно
* Возможны проблемы при установке

**Управление и Администрирование**

* Управление пользователями
* Профили пользователей
* Динамическая генерация страниц
* Интернационализация и локализация

**Темы**

**Плагины**: /wp-content/plagins backup

* Обеспечение безопасности сайта
* Борьба со спамом
* Техническая оптимизация ресурса в соответствии с требованиями поисковых систем (создание и обновление карты сайта, канонизация URL, оптимизация title страниц, автоматическая генерация метаданных страниц, блокирование индексации дублированного контента, создание микроразметки сайта).
* Повышение функциональности и юзабилити

**Меры безопасности**

1. Двухфакторная аутентификация
2. Логины и пароли
3. Зашифровать все
4. Обновления CMS

Основы кэширования в Wordpress

**Кэш** - это промежуточный буфер, который позволяет хранить наиболее часто используемые данные в памяти или на жестоком диске.

**Три основных вида кэширования:**

* Кэширование страниц (WP Super Cache, W3 Total Cache). Позволяет сохранять страницу целиком в кэш и передавать оттуда при последующем запросе. По умолчанию в ядре wp не реализовано.
* Кэширование объектов(функции wp\_cache\_add, wp\_cache\_set, wp\_cache\_get). Позволяет кэшировать данные произвольного типа. В ядре по умолчанию.
* Транзитное кэширование. (функции get\_transient, set\_transient, delete\_transient)

Для адаптации под мобильные устройства:

1. WPTouch
2. Jetpack
3. WPSmart Mobile
4. Responsible

**Какие проекты не стоит делать на вордпресс:**

WordPress хоть и гибкий, но область применения не безгранична.Есть два типа которые не рекомендуем делать на WP:

1. Одностраничные сайты и лендинги. Данные ресурсы представляют из себя сайты состоящие из одной — десяти страниц, в основном для продажи товаров и привлечения людей в разного типа тренинги и курсы. Такие сайты пишутся на простом HTML. База данных и сложная, большая CMS, как WordPress, тут не нужна. Существуют конструкторы и программы, лучше использовать их.
2. Сложные сервисы. Не пытайтесь сделать на вордпрессе какой-либо сервис обрабатывающий большое количество процессов.

CMS Wordpress (работа с контентом)

Таблицы бд по умолчанию имеют префикс wp\_.

Основная таблица wp\_posts.

Две таблицы не ни с одной другой - wp\_options и wp\_links.

Wp\_users и wp\_comments могут быть не связаны между собой.

Групповая установка сайтов подразумевает некоторые дополнительные таблицы.

**4 типа контента:**

* Сообщения (записи)
* Комментарии
* Пользователи
* Ссылки

**Сообщения**

‘post\_type’ => ‘post’

Существует 5 типов записей по умолчанию встроенных в WordPress:

* ﻿﻿запись, сообщение (post);
* ﻿﻿страница (page);
* ﻿﻿вложение (attachment);
* ﻿﻿редакция (revision);
* ﻿﻿элемент меню навигации (navigation menu item)

**Записи**:

single-post-{ярлык\_записи).php

single-post.php

single.php

singular.php

index.php

**Страницы**:

Http://ваш-сайт.ru/ярлык\_страницы

{любое\_название}.php (шаблон страницы)

page-{urldecode\_post\_name}.php

page-{post\_name}.php

page-{ID\_записи}.php

page.php

singular.php

index.php

*Особенности* страницы: могут использовать произвольный файл в качестве шаблона; имеют иерархию; имеется возможность сортировать по порядку.

**Вложения**

Предназначены для управления файлами, загруженными через стандартный загрузчик wp, а также для хранения данных о них.

**Редакции**

Это черновики, которые автоматически сохраняются в процессе редактирования записи. Каждая редакция сохраняется в таблице **wp\_hosts** вместе с записью, с которой она связана как родитель посредством поля **post\_parent**.

**Элементы Меню**

Данный тип хранит информацию об элементах навигации. Это единственный тип, который не используется для работы с контентом. Сохраняются как записи. Выводится с помощью запроса при отображении меню.

**Комментарии**

Комментарии хранятся в отдельной таблице базы данных — wp\_comments.

Комментарий может иметь прикрепленные через таблицу wp\_commentmeta метаданные.

комментарии связаны друг с другом с помощью поля comment\_parent.

Комментарии могут быть связаны таблицей wp\_users.

**Пользователи**

Имеют собственные таблицы - wp\_users. А также таблица с метаданными wp\_usermeta.

**Ссылки**

Таблица wp\_links.

Ссылки работают подобно записям. У них есть содержание, им можно присвоить элементы таксономии. Но им нельзя назначить автора.

**Замечания** по поводу метаданных!

Только три типа контента могут содержать метаданные:

* Записи
* Комментарии
* Пользователи

Виджеты

**Регионы и сайдбары**

**Виджеты** - это независимые блоки содержимого, которые можно размещать в областях предусмотренных темой.

В Wp существует определенно количество регионов, куда можно добавлять виджеты.

Виджеты бывают **типовые** или созданы плагинами и темами(с помощью наследования класса **WP\_Widget** и функции **register\_widget**). или определены прямо в админке с помощью html.

Сайдбары необходимо регистрировать в файле **functions.php** при помощи функции **register\_sidebar.**

**Joomla**

Позиции.

Ссылка «Предварительный просмотр»

(расширения -> шаблоны)

(Preview link in Extensions →

Templates Manager →

Manager).

Группы интерфейсов/уровни доступа к системе:

Гость - Просмотр сайта и общедоступных статей (public);

Зарегистрированный пользователь – Привилегии Гостя + Просмотр статей, доступных для пользователей;

Автор - Привилегии Зарегистрированного пользователя + создание новых статей + редактирование + создание новых + просмотр спецконтента;

Редактор - привилегии Автора + изменение всех статей, в том числе и опубликованных;

Издатель - Привилегии Редактора + Публикация статей;

**Группы админки:**

Менеджер - привилегии издателя + доступ администратора;

Администратор – привилегии менеджера + создание новых пользователей + установка расширений;

Супер юзер – права администратора + изменение шаблона сайта + изменение общих настроек;

Права доступа: Система - Общие настройки - Пользователи

Установка расширений:

Из Jed

Загрузка пакетов

Из директории

Через URL

**Drupal**

Это цмс написанная на php и использует реляционные бд.

Разработана в 2000году Дрис Бёйтарт.

Основные понятия:

**Нода - э**то элемент контента которая основывается на одной и той же базовой структуре.

**Таксономия**

Состоит из двух элементов:

* Словари
* Термины

**Формат ввода - э**то определенный набор прав применяющиеся при отображении страницы после ввода содержимого

**Блоки**

**Права доступа –** имеет три роли: анонимный пользователь, зарегистрированный пользователь, админ.

**Меню**

Admin toolbar

Модули устанавливаем через расширения.

Располагаются на официальном сайте.

Все модули и темы могут быть представлены в нескольких версиях:

* Стабильная
* Альфа/бета
* Разрабатываемая

Модуль содержит:

* Название
* Описание
* Ссылки на скачивание
* Статистика
* Ссылка на документацию, данного модуля

Ноды

Специальный раздел для типов материалов: Структура-Типы материалов

Типы блоков:

* предоставляемые модулями
* Созданные вручную

**Роль плагинов в расширении функциональности CMS**

**Плагин –** дополнительная программа, необходимая для расширения функциональности сайтов (использующих CMS), браузера или программы.

**Причины применения плагинов:**

1. Адаптация под конкретные нужды пользователя;
2. Предоставляют возможность не перегружать систему, за счёт сохранения компактного размера системы;
3. Улучшения функционала CMS;

**Основные преимущества плагинов CMS:**

**-** Лёгкость интеграции;

**-** Широкий выбор решений;

**-** Экономия времени и ресурсов;

**-** Поддержка и обновление;

**-** Управление пользователями и ролями;

**В DRUPAL как таковыми плагинами являются модули:**

- Ядровые;

- Контрибные

- Пользовательские

**Интересные модули в DRUPAL:**

* Geolocation
* Captcha Riddler
* Commerce Augment Reality
* Flag
* Moodboards

Plugin API - позволяет создавать блоки

Entity API - позволяет управлять настройками блока и определять настройки видимости блока.

**Создание модуля**

Modules/custom/[module:name]

Файл имя\_модуля.info.yml

Кастомные модули размещать в папке custom(создать)

Создаем контроллер

Modules/[module:name]/src/Controller

Создаем в корневой папке модуля новую папку src.

Создаем в папке src новую папку:

Controller.

Создаем файл с нашим контроллером

HelloWorldController.php.

Создаем файл маршрутизации

Имя\_модуля.routing.yml

Располагается в корневой папке. Служит для описания маршрута по базовому шаблону.

В качестве параметров маршрута выступают такие значения, как:

* Заголовок страницы
* URL
* Контроллер
* Права доступа
* И т.д.

**Apex**

**Apex** – инструмент ускоренной разработки web-приложений для БД Oracle

Версии системы:

* 1. Скачать дистрибутив, установить Oracle и приложение apex на комп
  2. Облачная версия (не нужна установка). Ограничен объём БД и не разрешено коммерческое использование.

Встроенные в Apex средства

* + 1. Темы пользовательского интерфейса
    2. Управление навигацией
    3. Управление формами
    4. Гибкие отчёты
* Возможности Apex
  1. среда разработки имеет простой и эффективный веб-интерфейс
  2. помощники миграции из настольных БД и электронных таблиц
  3. встроенный мастер генерации отчётов в формате pdf
  4. инструменты для интеграции и веб-сервисами
  5. большое кол-во шаблонов пользовательского интерфейса
  6. интуитивно-понятное управление рабочим пространством
  7. управление объектами по принципу Drag&drop
  8. графический помощник создания sql-запросов
  9. защищённость данных сессии после авторизации пользователя
  10. встроенный редактор PL/SQL
  11. мастер создания диаграмм и отчётов на сайте
  12. поддержка более 20 языков, включая русский

Apex – кросс-платформенная система.

Разработка в Apex может вестись на нескольких языках: ?????????????????????????

Во время работы:

* 1. выполнение аутентификации и авторизации
  2. контроль выполнения приложений на разных уровнях запуска
  3. обработка запросов и проверка состояния «качества»
  4. управляет состоянием сессии
  5. генерирует и обрабатывает страницы

**Архитектура**

1. Oracle Application Express размещается в БД;
2. представляет собой набор таблиц с данными PL/SQL код

Oracle применяет HTTP Server(Apache), в котором осдержится плагин mod\_plsql. Этот плагин перенаправляет все запросы между элементами APEX в БД???????????????????????????

Apex генерирует из данных, находящихся в БД Oracle, динамические страницы, при этом позволяет нам обрабатывать их в режиме реального времени. При создании приложений, написанных на Apex, сгенерированные метаданные, которые хранятся в таблицах БД Oracle, модифицируются. Во время работы программы Apex считывает эти метаданные и подставляет их в программу, отображая в браузере.

**Структура главной страницы**

В состав Apex входит 4 компонента:

* + 1. App Builder – для создания html-интерфейса. Каждое приложение представляет собой коллекцию страниц, связанных друг с другом при помощи вкладок, кнопок или гипертекстовых ссылок.

Номер приложения – целое число, идентифицирующее приложение в системе в целом; номер страницы?????????????????????77

* + 1. SQL Workshop – среда управления объектами БД
    2. Team Development
    3. Gallery

**Использование SQL рабочей области**

Apex позволяет использовать элемент управления деревьями для просмотра свойств объекта и создания новых объектов.

SQL Commands позволяет?????????????????????

SQL Scripts -

Utilities:

* + - 1. Data workshop – загружает и разгружает данные в тексте, XML и форматах электронной таблицы
      2. Object Reports - ??????????????????????????????

Схема – логических контейнер для объектов БД. В списке Schesm отображаются ???????????????