

УДК 004.4

АНАЛИЗ WEB-ТЕХНОЛОГИЙ ДЛЯ СОЗДАНИЯ ОНЛАЙН-ПЛОЩАДКИ**В. В. Мальцева, А. С. Фролова**

Донской государственной технической университет (Ростов-на-Дону, Российская Федерация)

Рассмотрены технологии для создания онлайн-площадки с помощью языков программирования (HTML, CSS, PHP), CMS (WordPress, Modx, Joomla, Drupal) и конструкторов сайтов. Цель исследования — выявление оптимального способа для создания web-ресурса. Результаты работы позволили сформулировать рекомендации по выбору системы.

Ключевые слова: веб-ресурс, сайт, конструктор, CMS, WordPress, Modx, Joomla, Drupal, языки программирования, HTML, CSS, PHP, система управления контентом.

UDC 004.4

ANALYSIS OF WEB-TECHNOLOGIES FOR CREATING ONLINE PLATFORM**V. V. Maltseva, A. S. Frolova**

Don State Technical University, (Rostov-on-Don, Russian Federation)

This article describes the technologies for creating an online platform using programming languages (HTML, CSS, PHP), CMS (WordPress, Modx, Joomla, Drupal) and various constructors. The purpose of this article is to identify the optimal way to create a web-resource. The results of the article are recommendations for the choice of the system.

Keywords: web resource, website, constructor, CMS, WordPress, Modx, Joomla, Drupal, programming languages, HTML, CSS, PHP, content management system.

Введение. Многие предприятия и организации размещают в интернете свой контент и создают собственные сайты. В рамках данной работы предполагается выявить оптимальные способы создания онлайн-площадки. Для этого необходимо изучить и проанализировать технологии создания web-ресурсов.

Основная часть. Web-сайт необходим для продвижения товаров, услуг и пр. В большинстве случаев сайты (онлайн-площадки) создаются с помощью:

- языков программирования,
- конструкторов,
- системы управления контентом (content management system, CMS).

Итак, сайт можно сделать «вручную», с помощью языков программирования. Для создания структуры используется HTML (от англ. hyper text markup language — язык разметки гипертекста) [1], а для визуализации — CSS (от англ. cascading style sheets — каскадные таблицы стилей) [2]. HTML появился почти одновременно с интернетом. За эти годы он зарекомендовал себя как один из самых простых, надежных и понятных инструментов. В новой версии HTML5.3 расширены возможности создания сайтов. Обновился функционал, с помощью которого можно хранить данные локально в браузерах пользователей, более безопасно передавать контент, легче достигать кроссбраузерности. Что касается CSS — это язык, который отвечает за визуальное представление документов пользователю.

Еще один язык программирования — серверный PHP (от англ. hypertext preprocessor — препроцессор гипертекста). Главная его задача — связать сайт с сервером и его базой данных. Все команды и скрипты, написанные на PHP, выполняются именно на сервере. После этого сервер пе-

редает результат браузеру, который, в свою очередь, придает данным вид, удобный для пользователя. То есть процесс обработки проходит в два этапа:

- PHP-интерпретатор (обработчик) работает с PHP-кодом, и пользователь получает HTML-код;
- браузер обрабатывает полученный результат [3].

Если задействован только HTML, ограничивается функциональность web-страницы, так как нельзя править определенные страницы. Для «оживления» онлайн-ресурса используется технология JavaScript. С помощью такого кода программист определяет, как страница отреагирует на действия пользователя. В браузере JavaScript может выполнять многие операции, в частности перечисленные ниже:

- менять стили элементов,
- реагировать на действия пользователя, клики мыши или перемещения курсора,
- создавать, удалять и редактировать HTML-теги,
- работать с технологиями AJAX (от англ. asynchronous JavaScript and XML, технология взаимодействия с сервером без перезагрузки страницы).

Иными словами, при использовании JavaScript и (или) PHP веб-ресурс действует по заданным алгоритмам в зависимости от выполнения или невыполнения определенных условий.

Таким образом, создание сайта «вручную» позволяет обеспечить как внешнюю, так и внутреннюю работу приложения. Вышеперечисленные технологии дают возможность контролировать работу сайта и выстраивать внутреннюю иерархию. Данный способ подойдет для простых сайтов, которые не требуют большой работы с базами данных (например, сайт-визитка или узкоспециализированные интернет-магазины с небольшим ассортиментом).

Следующий способ создания сайта предполагает использование CMS. В этом случае можно добавлять и редактировать контент сайта, не меняя внутренний механизм организации и вывода страниц. Условно данная технология предполагает наличие двух хранилищ информации. Первое — это базы данных с контентом страниц, во втором содержатся элементы интерфейса (графика, картинки, текст, шаблоны и т. д.) [4]. Для удобства информационного наполнения сайта большинство движков задействуют визуальный редактор WYSIWYG (от англ. what you see is what you get — что видишь, то и получишь). Эта программа создает HTML-код из разметки, облегчающей пользовательское форматирование текста. Поэтому в процессе работы с текстом сразу виден конечный результат.

Основные преимущества CMS [5]:

- доступность разработки,
- простота разработки,
- экономия времени,
- обновление контента,
- мобильность,
- возможность удаленного доступа.

По составу CMS бывают разными. Одни состоят из множества блоков, другие обладают неделимой архитектурой, есть платные и бесплатные решения. Самая популярная CMS — WordPress. Согласно данным Web Technology Surveys на ноябрь 2018 года, на этом движке работает 32,3 % от общего числа существующих сайтов. Еще пару лет назад эту систему использовали 24 % ресурсов на CMS [6]. Анализ различных CMS представлен в табл. 1.

Анализ характеристик CMS

CMS	Битрикс	WordPress	Joomla!	Drupal	Modx
Сложность интерфейса	Для продвинутых	Для новичков			
Шаблоны, расширения, дополнения	Большой выбор, в том числе бесплатных				Узкий выбор
Уровень защиты	Высокий	Низкий		Высокий	Низкий
Загрузка сайта	Быстрая		Медленная	Быстрая	
Освоение	Наличие русскоязычных книг	Большое количество материалов по изучению			Ограниченное количество
Условия доступа	Платно	Бесплатно			
Продвижение	Есть				
Официальная техподдержка	Есть	Нет		Есть	Нет
Система с разделением прав	Есть				
Обновления	По лицензии	Есть			

Серьезные веб-проекты лучше создавать на CMS или самописных движках. Однако иногда такой способ оказывается долгим и трудозатратным. И тогда лучшим решением являются визуальные конструкторы. Суть их реализации заключается в перемещении блоков и настраивании внешнего интерфейса. При этом не нужно исправлять исходный код и разбираться в тегах. Однако из-за шаблонных решений (исходный код закрыт) конструктор не позволяет создать уникальный дизайн. Еще один недостаток — нельзя самостоятельно добавлять новые плагины, скрипты и коды. Конструкторы сайтов предлагают бесплатно минимум услуг, за важные функции необходимо заплатить. Таким образом, конструктор представляет собой простой, но недешевый способ создания сайта, поэтому его использует для следующих задач:

- хорошая визуальная интерпретация идей,
- быстрая разработка и запуск небольших проектов,
- тестирование идеи.

Таким образом, при выборе способа создания веб-ресурса необходимо исходить из направленности сайта, требуемого функционала и имеющихся ресурсов. Основные преимущества и недостатки способов создания сайта представлены в табл. 2.

Достоинства и недостатки способов создания сайта

Способы программирования	Достоинства	Недостатки
Языки программирования	<ol style="list-style-type: none"> 1. Позволяет полностью контролировать работу сайта 2. Нет ограничений функционала 	<ol style="list-style-type: none"> 1. Трудозатратный 2. Необходимо углубленное знание языков программирования
Конструктор	<ol style="list-style-type: none"> 1. Не используется программирование 2. Готовая структура сайта 3. Экономия времени (высокая скорость) 	<ol style="list-style-type: none"> 1. Шаблонный дизайн 2. Долгая загрузка из-за большого объема 3. Невозможно редактировать код вручную 4. Ограничены структура и наполняемость сайта 5. Привязка к определенной площадке 6. Нет возможности расширять сайт (ограниченный функционал) 7. Ограничены или отсутствуют возможности продвижения веб-ресурса
CMS	<ol style="list-style-type: none"> 1. Экономия времени (высокая скорость) 2. Простота разработки сайта 3. Широкий функционал 4. Быстрая загрузка сайта 5. Редактирование контента отдельно от функциональных элементов и дизайна 6. SEO-оптимизация 7. Мобильность 8. Удаленный доступ 9. Регулярное обновление 	<ol style="list-style-type: none"> 1. Низкая безопасность сайта 2. Однотипность сайтов 3. Ограниченный функционал расширения модулей

Заключение. Проведенный анализ позволил установить, что способ создания web-ресурса зависит от направленности, задач и требуемого функционала, материальных ресурсов, а также квалификации программиста. Языки программирования целесообразно использовать для разработки небольших и простых сайтов (визитка или узкоспециализированный интернет-магазин с небольшим ассортиментом). Если необходимо оперативно создать web-ресурс и исключить трудоемкую работу с исходным кодом, можно воспользоваться сайтом-конструктором. Если же в итоге должен получиться подчеркнuto индивидуальный сайт, следует задействовать инструментарий CMS-движков (Битрикс, WordPress, Joomla, Drupal, Modx).

Библиографический список

1. Языки программирования web-сайтов / HineX // hineX.ru : [сайт]. — URL : <http://hineX.ru/yazyki-programirovaniya-web-saitov.php> (дата обращения 07.12.2019).
2. Что такое CSS? / Mozilla and individual contributors // MDN web docs : [сайт]. — URL : https://developer.mozilla.org/ru/docs/Web/Guide/CSS/Getting_started/What_is_CSS (дата обращения: 11.11.2019).
3. Науменко, Д. О. HTML, CSS, PHP, JavaScript, SQL — что и зачем? / Д. О. Науменко // Codeharmony.ru. В гармонии с кодом : [сайт]. — URL : <http://codeharmony.ru/materials/125> (дата обращения 18.11.2018).
4. Что такое CMS и как ее использовать / Rookee // Rookee powered by ingate : [сайт]. — URL : <https://wiki.rookee.ru/cms/> (дата обращения 22.11.2019).
5. Преимущества и недостатки сайта CMS / Denisov // Denisov : [сайт]. — URL : <https://sdvv.ru/articles/testovyy-razdel/preimushchestva-i-nedostatki-sayta-na-cms/> (дата обращения: 22.11.2019).
6. Как создать сайт на WordPress: полное руководство для новичков / TexTerra // TexTerra : [сайт]. — URL : <https://texterra.ru/blog/kak-sozdat-sayt-na-wordpress-polnoe-rukovodstvo-dlyanovichkov.html> (дата обращения 10.11.2019).

Об авторах:

Мальцева Виктория Васильевна, студент Донского государственного технического университета (344000, РФ, г. Ростов-на-Дону, пл. Гагарина, 1), Vikamalt643@gmail.com

Фролова Анна Сергеевна, доцент кафедры «Медиаменеджмент и медиапроизводство» Донского государственного технического университета (344000, РФ, г. Ростов-на-Дону, пл. Гагарина, 1), кандидат философских наук, frannser@gmail.com

Authors:

Maltseva Victoria Vasilevna, student of Don State Technical University (344000, Russian Federation, Rostov-on-Don, Gagarina sq. 1), Vikamalt643@gmail.com

Frolova Anna Sergeevna, associate professor, the department of «Media management and media production» 344000, Russian Federation, Rostov-on-Don, Gagarina sq. 1), PhD in Philosophy, frannser@gmail.com