

АНАЛИЗ ИНСТРУМЕНТОВ ДЛЯ СОЗДАНИЯ САЙТОВ

А.В. Шейн, Л.Н. Самуилик, 3 курс

Научный руководитель – И.А. Янковский, к.э.н., доцент

Полесский государственный университет

Динамика роста количества веб-ресурсов (сайтов, приложений, сервисов) говорит о том, что технология создания сайта становится более доступной [1] и является актуальной темой. В данной работе проанализируем инструменты, с помощью которых можно создать свой собственный сайт.

Прежде чем перейти к анализу инструментов для создания сайтов, разберёмся с определением сайта. «Сайт - в компьютерной сети объединённая под одним адресом (доменным именем или IP-адресом) совокупность документов частного лица или организации. По умолчанию подразумевается, что сайт располагается в сети Интернет». [2]

Основные инструменты создания сайтов

Существуют различные способы создания сайта, однако наиболее распространёнными являются: написание кода с нуля, разработка на конструкторе или создание на готовой CMS [3]. Также хочется отметить новый подход к созданию сайтов: создание сайта с помощью нейросети. Основные различия данных способов — ядро или же «движок», определяющие базовые функции веб-сайта.

Создание сайта написанием программного кода с нуля

Разработка на чистом коде с помощью языков программирования считается наиболее правильным вариантом, позволяющим реализовать любую задачу. Для разработки простого сайта или лендинга потребуются знания стыка технологий, например, HTML, CSS и JS [11].

Из основных **преимуществ** разработки данным способом стоит выделить:

- Чистый валидный код — разрабатывая сайт с нуля, заказчик имеет возможность сразу писать код, соответствующий стандартам качества, не имеющий некорректных решений, ошибок или ненужных строк.
- Безопасность и конфиденциальность — создание на конструкторе или CMS не всегда может гарантировать защиту данных компании, особенно если используются шаблоны. Написание кода с нуля позволяет предотвратить в ядре любые уязвимости.
- Отсутствие ограничений — при написании сайта с помощью кода можно реализовать любое решение. Через код можно добавить нужные функции, провести интеграцию других IT-продуктов, настроить структуру и навигацию [10].

К плюсам также можно отнести: меньший объем кода, экономичный расход ресурсов сервера, не требуется обновление «движка» или отдельных модулей, упрощенная система создания бэкапа, стабильность (если ошибка допущена на одной странице, она не может затронуть работу всего сайта).

Теперь обсудим **недостатки** данного способа. Основной недостаток разработки с помощью кода возникают из преимуществ данного способа. Ввиду сложности разработки реализовать проект с чистым кодом затратно как по финансам, так и по времени. Для создания высоконагруженных сервисов или проектов со сложной структурой потребуется серьезный бюджет и привлечение большого штата специалистов [6]. К недостаткам также можно отнести:

- Сложность внесения изменений, которые повторяются на всех страницах (или на большинстве). К таким изменениям можно отнести пункты меню, телефоны, адреса, акции или любую другую справочную информацию. Сложность заключается в том, что эти данные необходимо вручную изменять на каждой странице. И если сайт состоит из нескольких сотен страниц, то этот процесс может показаться весьма затруднительным.
- Отсутствие админ-панели, в которой более простым и понятным способом выводится информация о сайте и упрощен способ наполнения сайта контентом.
- Для поддержки и наполнения сайта, необходимо обладать базовыми знаниями HTML [12].

В дополнение можно сказать, что для создания сайта, используя чистый код, необходимо, непосредственно, знание самого кода, а, следовательно, потребуется затратить немалое количество времени для его изучения.

Создание с помощью конструктора сайтов

Данный способ считается наименее затратным. Большинство конструкторов распространяются на бесплатной основе и часто предоставляют услуги хостинга, онлайн-кассы и т. д. [9]

Основные **преимущества** использования конструкторов:

- Удобный интерфейс.
- Простота настройки.
- Масса готовых шаблонов и бесплатный хостинг позволяют быстро запустить проект в сеть.
- Скорость разработки. По факту, сайт уже создан. Пара часов на заполнение – и сайт готов к использованию.

- Минимальные знания в веб-программировании. Человек без особых знаний в этой сфере при желании может создать сайт, и сделать это будет не сложнее, чем заполнить профиль в социальных сетях.

Более того, за небольшую доплату большинство конструкторов предлагают дополнительные опции и функции для роста и расширения проекта.

Из **недостатков** конструкторов следует выделить:

- Ограниченность настроек в бесплатных версиях. Большинство бесплатных решений предлагают весьма типовые опции, не позволяют перенести разработанный сайт на личный домен.
- Отсутствие доступа к коду.
- Ограничение на полноценное продвижение. Даже в платных версиях конструкторов есть много ограничений. Из-за этого не получится настроить более тонкие опции SEO-продвижения.
- Шаблонность. Созданный вами сайт не будет отличаться уникальностью.
- Медленная работа сайта. Связано это с тем, что обычно сайты-конструкторы располагаются на самых дешёвых серверах и не способны выдержать постоянной и большой нагрузки на сайт, а также страницы сайта весят по несколько мегабайт, что тоже приводит к медленной загрузке и работе сайта [9].

Создание сайта на CMS-системе

Система управления — универсальное решение, позволяющее запустить проект быстро, недорого и с требуемой функциональностью. CMS представляет собой программное обеспечение, включающее движок и базовые функции, позволяющее работать со структурой и контентом. Благодаря CMS можно быстро создать и настроить свой сайт, не привлекая для этого сторонних специалистов.

Основная идея CMS - это простота создания сайта. Любой желающий может разобраться в принципах работы понравившейся ему CMS. Интерфейс любой из систем интуитивно понятен, и человек без каких-либо знаний PHP, CSS, HTML и других инструментов веб-разработки, сможет наполнять свой сайт без помощи сторонних программистов [7].

Для опытных программистов CMS также полезна, ведь система облегчает множество повседневных задач и упрощает систематизацию информации.

Оптимальными системами управления для коммерческих проектов считаются 1С-Битрикс, WordPress, Drupal.

Преимущества использования CMS:

- Функциональность и обилие настроек — с помощью базовых функций CMS можно реализовать любую структуру веб-сайта. Кроме того, ядро CMS предполагает простую интеграцию с большинством сторонних IT-продуктов.
- Оптимизированность — ресурс на базе CMS, как правило, уже имеет техническую и поисковую оптимизацию, хорошо ранжируется в поисковых системах.
- Доступность - для старта достаточно скопировать систему на сервер и приступить к работе, что в большинстве случаев бесплатно. Информация для изучения также находится в открытом доступе, что не может не радовать. Единственные затраты на изучение и стар создания сайта - это оплата интернет-трафика и время освоения CMS.
- Актуальность на рынке труда - разработка сайтов - это хороший способ заработка, а разработка сайтов на CMS - это ещё и доступный способ заработка. Не каждый будет вникать в работу и функционирование CMS, поэтому человек, освоивший эту систему, может стать ценным кадром в сфере сайтостроения. А благодаря четкому механизму расширения функционала сайтов, создан-

ных на CMS, у них очень низкий порог входа, что позволяет разрабатывать сайты людям, разбирающимся в программировании поверхностно [7].

Недостатки сайта на CMS:

- Дорогостоящая лицензия.
- Слабо проработанный движок, который может быть уязвимым.
- Также, для «рядового» пользователя создание сайта на CMS может вызвать некоторые трудности, связанные с обилием настроек и функций, а также потребовать немало времени для освоения данного способа.
- Некоторые CMS очень требовательны к хостингу, и поэтому, пытаясь, сэкономить на хостинге, можно получить медленный и регулярно «падающий» сайт.
- Популярные плагины могут сильно нагрузить сайт, если бездумно использовать их все подряд.
- Снова хочется вспомнить о **шаблонности**. Если человек остановится лишь на освоении работы системы, то его работы приобретут шаблонность, что, несомненно, является минусом на рынке труда и в сфере создания контента в целом. Какая бы гибкая ни была платформа, все равно нельзя создать полностью уникальный сайт, на нем наверняка останутся некоторые признаки шаблонности. Поэтому для создания уникального сайта всё же придётся освоить основы веб-разработки.
- Также к минусам многих CMS можно отнести то, что каждая из них обладает **собственными особенностями**. Например, для решения какой-нибудь задачи путём написания сайта вручную хватило бы пару строчек кода, когда на CMS пришлось бы долго искать пути реализации данного решения.
- Для написания простых сайтов использовать CMS неоправданно, поскольку его функциональность окажется невостребованной, но при этом он будет загружать сервер [8].

Создание сайта с помощью нейросети

Было бы удобно, если бы пользователи могли быстро создавать сайты и веб-приложения для своих целей: не шаблонные, а именно такие, какими они хотят видеть продукт. И такой инструмент есть. Весной 2017 года IT-сообщество всколыхнула новость о том, что датский стартап Uizard Technologies разработал нейросеть pix2code для автоматизации фронтенд-разработки. Спустя восемь месяцев разработчики FloydHub на её основе создали собственную нейросеть для верстки, которая может сгенерировать разметку и стили, ориентируясь на изображение макета [6].

Принцип работы такого алгоритма следующий:

- Пользователь загружает макет страницы в формате JPEG,
- Нейросеть трансформирует картинку в HTML и CSS-код,
- Пользователь получает сверстанную страницу [5].

Алгоритм умеет верстать простые блоки, определять теги и расставлять их в нужном порядке. Его главное преимущество — постоянная обучаемость. По мнению разработчиков, с помощью такого обучения в течение трех лет нейросеть могла бы изменить разработку: полностью автоматизировать верстку и снизить порог вхождения в профессию. Правда стоит уточнить, что такое заявление было сделано 3,5 года назад, а алгоритм до сих пор не нашёл активного применения в веб-разработке [4].

Заключение. Чтобы понять, какая технология для создания сайта лучше всего подходит для вашего проекта, необходимо понимать, какой сайт в результате вы желаете получить, какие задачи были поставлены. Например, язык серверных скриптов выбирают исходя из решаемых задач. При грамотном подходе к выбору используемых технологий и инструментов, на выходе вы получите качественный продукт [2]. Однако какая бы технология не была выбрана, для создания действительно качественного и уникального продукта пригодятся знания программного кода.

Список использованных источников

1. Технологии для создания сайтов [Электронный ресурс]. Дата доступа: 25.03.2022. Режим доступа: <https://studiobit.ru/blog/sozдание-web-saytov/tekhnologii-sozdaniya-saytov/>
2. Способы создания сайта [Электронный ресурс]. Дата доступа: 25.03.2022. Режим доступа: <https://studiobit.ru/blog/sozдание-web-saytov/sposoby-sozdaniya-sayta/#:~:text=%D0%9E%D1%81%D0%BD%D0%BE%D0%B2%D0%BD%D1%8B%D0%B5%20%D0%BC%D>

0%B5%D1%82%D0%BE%D0%B4%D1%8B%20%D1%81%D0%BE%D0%B7%D0%B4%D0%B0%D0%BD%D0%B8%D1%8F%20%D1%81%D0%B0%D0%B9%D1%82%D0%BE%D0%B2.%D0%BE%D0%BF%D1%80%D0%B5%D0%B4%D0%B5%D0%BB%D1%8F%D1%8E%D1%89%D0%B8%D0%B5%20%D0%B1%D0%B0%D0%B7%D0%BE%D0%B2%D1%8B%D0%B5%20%D1%84%D1%83%D0%BD%D0%BA%D1%86%D0%B8%D0%B8%20%D0%B2%D0%B5%D0%B1%2D%D1%81%D0%B0%D0%B9%D1%82%D0%B0.

3. 15 инструментов для создания сайта «с нуля» [Электронный ресурс]. Дата доступа: 03.04.2022. Режим доступа: <https://www.imena.ua/blog/website-tools/>

4. Искусственный интеллект для создания сайтов (веб-дизайна) [Электронный ресурс]. Дата доступа: 27.03.2022. Режим доступа: https://www.liveintellect.ru/tags/ii_dlja_sozdaniija_sajtov/

5. Заменят ли верстальщиков нейросети и конструкторы для создания сайтов? [Электронный ресурс]. Дата доступа: 06.04.2022. Режим доступа: <https://habr.com/ru/company/htmlacademy/blog/566132/>

6. Нейронная сеть, которая верстаёт сайты по картинке [Электронный ресурс]. Дата доступа: 27.03.2022. Режим доступа: <https://vc.ru/future/31732-neyronnaya-set-kotoraya-verstaet-sayty-po-kartinke>

7. Плюсы и минусы создания сайтов на популярных CMS [Электронный ресурс]. Дата доступа: 06.04.2022. Режим доступа: <https://jino.ru/journal/articles/plyusy-minusy-sayty-cms/>

8. Плюсы и минусы популярных бесплатных CMS для сайтов [Электронный ресурс]. Дата доступа: 07.04.2022. Режим доступа: <https://xn---7sbbaghlkm9ah9aiq.net/news-new/25.html>

9. Сайт на конструкторе: плюсы, минусы и когда стоит их использовать [Электронный ресурс]. Дата доступа: 08.04.2022. Режим доступа: <https://olprime.ru/blog/konstruktor-saytov-vse-za-i-protiv/>

10. Разница между HTML-версией и CMS-версией сайта [Электронный ресурс]. Дата доступа: 08.04.2022. Режим доступа: <https://www.rserdyuk.ru/html-vs-cms.html>

11. HTML или CMS – что лучше для коммерческого сайта [Электронный ресурс]. Дата доступа: 08.04.2022. Режим доступа: <https://vzh.ru/article/html-ili-cms-cto-luchshe-dlya-kommercheskogo-sajta/>

12. Что лучше использовать – чистый HTML, самописный движок или CMS WordPress? [Электронный ресурс]. Дата доступа: 06.04.2022. Режим доступа: <https://denis-creative.com/cto-luchshe-ispolzovat-chistyj-html-samopisnyj-dvizhok-ili-cms-wordpress/#:~:text=%D0%9C%D0%BE%D0%BC%D0%B5%D0%BD%D1%82%D0%B0%D0%BB%D1%8C%D0%BD%D0%B0%D1%8F%20%D1%81%D0%BA%D0%BE%D1%80%D0%BE%D1%81%D1%82%D1%8C%20%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D1%8B%20%D1%81%D0%B0%D0%B9%D1%82%D0%B0%2C%20%D1%81%D0%B0%D0%B9%D1%82,%D0%B4%D0%B5%D1%88%D0%B5%D0%B2%D1%8B%D0%B9%20%D1%85%D0%BE%D1%81%D1%82%D0%B8%D0%BD%D0%B3%20%D1%81%20%D0%BC%D0%B8%D0%BD%D0%B8%D0%BC%D0%B0%D0%BB%D1%8C%D0%BD%D1%8B%D0%BC%D0%B8%20%D0%BF%D0%B0%D1%80%D0%B0%D0%BC%D0%B5%D1%82%D1%80%D0%B0%D0%BC%D0%B8>