

О концепциях выбора технических средств для разработки веб-проекта Атрощенко Н. А.

*Атрощенко Натэлла Александровна / Atroshchenko Natella Alexandrovna – старший преподаватель,
кафедра экономической информатики, инженерно-экономический факультет,
Белорусский государственный университет информатики и радиоэлектроники, г. Минск*

Аннотация: в статье анализируются вопросы выбора технических средств при разработке веб-ресурса с учётом его технической оптимизации.

Ключевые слова: веб-проект, техническая оптимизация веб-проекта, стандартизация.

Вопрос о технической оптимизации веб-ресурса чрезвычайно важен уже на этапе принятия стратегии решения: на стадии реализации кардинально изменить его будет весьма затруднительной задачей. Веб-проект - это, в общем случае, совокупность языков программирования, технологий и фреймворков, применяемых архитектурных решений и паттернов проектирования, систем безопасности, библиотек и хранилищ данных, коммуникационных сетевых служб, включая сокетные соединения, и т. д. Выбор эффективного программного продукта для разработки проекта нужен не только с точки зрения хороших навыков владения им разработчиков. При принятии решения об использовании программных средств и инструментов и внедрением решений в проект важно отметить ряд необходимых критериев:

- сочетание с используемыми библиотеками, технологиями, фреймворками;
- актуальность и современность выбранных версий;
- степень того, насколько успешно этот продукт успел зарекомендовать себя на рынке;
- устойчивость в авральные условия работы, низкая степень отказов в нестандартных ситуациях;
- гибкость организации и единых концепций на стадии разработки;
- возможность альтернативных решений при возникновении аварийных обстоятельств;
- чётко изложенная техническая документация, поддержка и консалтинг;
- наличие всех хорошо отработанных аспектов безопасности.

Создание интернет-проектов без системного подхода к планированию и разработке интернет-проекта при отсутствии самого понятия о качестве создаваемого продукта, незнание государственных стандартов, требований технической оптимизации, юридических основ по размещаемой в сети информации приводит к засилью в сети интернет некачественных сайтов, замусоривающих просторы Интернет. Для того, чтобы создать эффективно работающий веб-сайт, изначально нужно опираться на основополагающий фундамент тех стандартов и требований, которые заложены и существуют уже достаточно продолжительное время на просторах СНГ и международных сообществ, например: ГОСТ 9126. Информационная технология. Оценка программного продукта, ANSI/NCITS 354-2001: Common Industry Format for Usability Test Reports (Промышленный стандарт описания результатов юзабилити-тестирования), ГОСТ 34.603-92. Виды Испытаний Автоматизированных Систем, Международные стандарты серии ISO 9000 и др. [1] Ориентирование на соответствие стандартам и требованиям качества готового приложения при выборе инструментов и библиотек при разработке веб-проекта, системный подход к планированию и возможности дальнейшей поддержки и развития веб-проекта – фундамент для эффективного веб-проекта. На этапе разработки следует держать под контролем по возможности все аспекты оптимизации проекта на клиентской стороне. Исходя из требований к качеству веб-продукта непосредственно потребителями: высокой скорости загрузки, интуитивно понятной навигации, usability, удобства восприятия информации, удовлетворённости от использования и степени успешности решаемых пользователем задач [2], разработчики должны определить приоритетные требования, как к создаваемому продукту, так и к инструментам для его создания. Простые и допотопные средства разработки, без сомнения, отрицательно скажутся на качестве веб-продукта. Самый яркий пример негативного влияния на дальнейшую техническую оптимизацию сайта – автоматическая генерация HTML кода и вёрстки веб-страницы с помощью весьма распространённых бесплатных инструментов. Излишний, многократно повторяющийся код, утечки памяти, «битые» указатели, неграмотно используемые структуры данных и пространств имён, отсутствие высокоуровневого прототипирования в масштабах всего приложения – свидетельство достаточно низкой планки профессионализма создателей сайта, либо игнорирование актуальных программных продуктов. Сегодня существует множество средств модульного программирования, библиотек, средств отладки, шаблонизаторов, возможности асинхронной обработки данных, инструментов для анализа и сокращения кода, его быстрой редакции, читабельности и минимального обращения к серверу и хранилищу данных, что, безусловно, будет влиять на время загрузки содержимого веб-страницы. Поэтому каждую абстрактную единицу кода на клиентской стороне, будь то: объект, файл, блок, модуль, статическая или динамическая библиотека и т. д., необходимо рассматривать с точки зрения двух принципов технической оптимизации: меньше данных, меньше соединений. В итоге оптимальное построение приложения нужно вести исходя из: размера кода,

времени загрузки, времени взаимодействия с пользователем. Однако никто не застрахован от ситуации, когда найденные на просторах интернета эффективные плагины, прекрасно работающие поодиночке, отказываются совместно работать с одним и тем же набором данных из-за различного внутреннего представления этих данных или других причин несовместимости. В настоящее время существует немало технических средств, достаточно серьёзно уменьшающих нагрузку на сервер. Есть готовые решения, ориентированные на облачные технологии, а также сторонние решения, например, Sophos UTM & Next-Gen Firewall, регулирования пропускной способности (средствами Киберсейф Межсетевой экран), используются специальные плагины (например, WPLANG Lite), документо-ориентированные хранилища данных, а также частичное сохранение данных на клиентской стороне. Защищая сервер от большого количества запросов и DOS-атак, пишутся специальные приложения для контроля фильтра запросов к серверу. Для технической оптимизации на стороне сервера можно предпринять следующие шаги: настройка Gzip-сжатия, правильного кэширования, защита от повышенной нагрузки от ботов, использование системы доставки трафика (Content Delivery Network или Content Distribution Network, CDN). Уход от многопоточного обращения к серверу, принцип асинхронного неблокирующего ввода-вывода на платформах, подобных node.js – одно из приоритетных направлений ориентации создания веб-приложений на сегодняшний день.

Литература

1. Консорциум Всемирной паутины [англ. World Wide Web Consortium, W3C]: Официальный сайт. [Электронный ресурс]. Режим доступа: <http://WWW.W3.ORG> (дата обращения: 27.10.2016).
2. International Organization for Standardization, ISO = Международная организация по стандартизации: Официальный сайт. [Электронный ресурс]. Режим доступа: <http://WWW.ISO.ORG> (дата обращения: 27.10.2016).