

УДК 621.391

## РАЗРАБОТКА ВЕБ-ПРИЛОЖЕНИЯ ДЛЯ ОБРАБОТКИ ЦИФРОВЫХ ИЗОБРАЖЕНИЙ

Н.И. ЛИСТОПАД, О.Н. ВИНИЧУК

*Белорусский государственный университет информатики и радиоэлектроники, Республика Беларусь*

*Поступила в редакцию 01 ноября 2019*

**Аннотация.** Данная работа посвящена анализу существующих веб-приложений, выявлению их достоинств и недостатков, а также реализации веб-приложения для обработки цифровых изображений

*Ключевые слова:* Веб-приложения, векторное и растровое представления, многостраничные и одностраничные приложения.

### Введение

На сегодняшний день достаточно сложно подобрать графический редактор для обработки цифровых изображений ввиду огромного количества как десктоп версий, так и веб-приложений, каждый из которых обладает рядом функциональных возможностей и, так или иначе, дополняет друг друга.

При создании html-страниц цифровые изображения подвергаются коррекции. Зачастую, материал для верстки html-страниц представлен не в psd-макете (исходный материал для верстки сайта), а в качестве скомпонованных по разделам файлов, которые и требуют последующей обработки.

Все компьютерные изображения, которые размещаются в качестве графического представления на html-странице, как известно, можно разделить на два формата – векторные и растровые, – различающиеся, прежде всего, уровнем абстракции, примененной к изображению [1]. При этом если векторная графика пытается имитировать восприятие изображений человеком, то растровый формат хранит графику в том виде, в каком она легче всего обрабатывается компьютером. Соответственно, векторная графика в большинстве своем создается пользователем с нуля используя векторные редакторы. Основным поставщик растровых изображений – цифровое изображение, то есть в существенной своей части автоматический процесс с легко оцифровываемыми результатами.

### Анализ методов и моделей обработки изображений

В последнее время список сервисов и ресурсов в Интернете достаточно расширился. Интернет превратился из однообразных статических html-страниц в мощный инструмент интерактивности и общения с конечными пользователями. В связи с этим веб-приложения в настоящее время приобрели небывалую популярность, ввиду предоставления различных преимуществ, которые отсутствуют в обычных десктоп-приложениях. Исходя из этого, была сформулирована гипотеза: веб-приложения можно выделять как отдельное звено из всей цепочки приложений. К сожалению, десктоп-приложения развиваются не так быстро и стремительно, как веб-технологии. Большое множество компаний переходят с обычных приложений на веб-приложения, именно потому, что видят в них будущее. На основании

вышеописанной характеристики десктоп-приложений и веб-приложений были выделены основные преимущества работы с веб-приложениями [2]:

1. Установка веб-приложений дешевле и намного проще. Для работы с веб-приложениями, как правило, не требуется установка дополнительного ПО, пользователю для полноценной работы достаточно стандартного браузера, который идет в пакете установленных программ любой операционной системы (далее ОС), а вот при работе с десктоп-приложениями обязательно требуется установка данного приложения и, возможно, для бесперебойной работы, дополнительного программного обеспечения (далее ПО, как правило, это дополнительные драйверы или расширения).

2. Обновление веб-приложений дешевле и намного проще. Для того чтобы совершить обновление веб-приложения, его необходимо обновить только на сервере и для того, чтобы работать с новой версией пользователю достаточно перезагрузить страницу. Обновление ПО десктоп-версии очень похоже на его установку, поэтому преимущества, которые были упомянуты выше имеют, место и в данной ситуации.

3. Веб-приложения более универсальны и практичны для конечного пользователя. Ввиду того, что веб-приложение хранится на сервере, доступ к нему можно получить с любого устройства, где подключен Интернет и установлен браузер, то есть можно сделать вывод, что доступ к веб-приложению возможен из любой ОС и с любого устройства. К сожалению, для реализации таких возможностей десктоп-приложений существует только один выход: разработка отдельных приложений под ОС и отдельных приложений для различных устройств, однако, в случае, если приложения будут иметь общий доступ к БД, которая должна храниться на сервере, необходимо будет наличие сети Интернет;

4. Веб-приложения облегчают организацию хранения данных. Если есть необходимость обращаться к одним и тем же данным из разных мест, то намного проще организовать их хранение в одном месте, как вариант, облачное хранилище данных (модель онлайн-хранилища, в котором данные хранятся на многочисленных распределенных в сети серверах, предоставляемых в пользование клиентам).

Чтобы обеспечить стабильную работу сложных веб-приложений, желательно использовать технологии, которые дадут наилучшую производительность и скорость [2]. На сегодняшний день существует два основных способа разработки веб-приложений:

- SPA (Single Page Applications) – одностраничные приложения;
- MPA (Multi Page Applications) – многостраничные приложения.

MPA – это традиционное многостраничное веб-приложение. При таком подходе каждый раз, когда приложение запрашивает данные либо отправляет их на сервер, оно вынуждено получать новую страницу в полном объеме, а затем отображать ее в браузере. При использовании такого способа разработки веб-приложений тратится время и снижается производительность на отображение одних и тех же элементов, что влияет на скорость и производительность в целом. Многие веб-разработчики для того, чтобы хоть как-то повысить скорость и уменьшить нагрузку на сервер используют библиотеку jQuery.

Для относительно простых веб-приложений такой способ вполне подходит, но ввиду того, что веб-приложение для обработки изображений большого объема обладает рядом функциональных возможностей, которые будут нагружать веб-приложение, то речь может идти о том, как ускорить сам процесс обработки данных. Все это ведет к тому, что размер трафика у MPA заметно увеличивается, и при многостраничном подходе неминуемо появляются проблемы с производительностью.

Для решения проблемы с производительностью был выбран способ SPA для создания веб-приложений. Архитектура у SPA устроена таким образом, что при переходе на новую страницу, обновляется только часть контента. Таким образом, нет необходимости повторно загружать одни и те же элементы. Это очень удобно не только для пользователей, но и для разработчиков.

Для разработки SPA используется один из самых популярных языков программирования – JavaScript, а также библиотека jQuery.

При создании html-страниц с использованием способа SPA можно выделить ряд достоинств по отношению к MPA:

- более быстрая загрузка веб-страниц;
- улучшенное восприятие пользовательского интерфейса, поскольку загрузка данных с сервера происходит в фоновом режиме;
- нет необходимости разрабатывать код на сервере для визуализации страницы;
- разделение на Front-end (клиентская сторона пользовательского интерфейса к программно-аппаратной части сервиса.) и Back-end (программно-аппаратная часть сервиса) – разработку;
- упрощенная разработка под мобильные приложения, а именно, возможность использования одного и того же серверного кода как для веб-приложений, так и для мобильных приложений.

Проведенный анализ показал, что SPA – это своеобразная эволюция способа MPA и технологии обмена данными в фоновом режиме AJAX. При таком подходе только структура веб-страницы строится на сервере, все остальное генерируется средствами JavaScript. SPA запрашивает html-разметку и данные отдельно и визуализирует результаты непосредственно в браузере.

Описанный выше подход представлен на рис. 1.

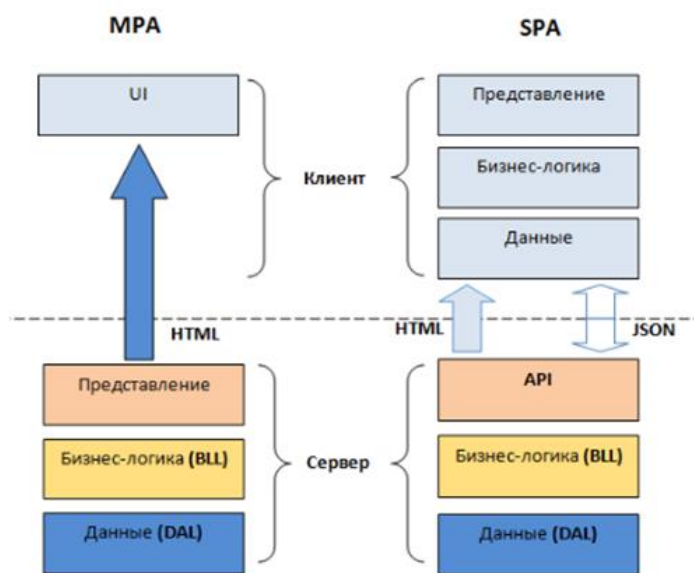


Рис. 1. MPA и SPA: способы загрузки и отображения данных

### Заключение

Данная работа посвящена анализу существующих веб-приложений, выявлению их достоинств и недостатков, а также реализации веб-приложения для обработки цифровых изображений. Использование вышеописанных моделей и алгоритмов позволяет сократить объем памяти, необходимой для хранения изображений, времени обработки изображений, а также обеспечить качественное отображение изображений на веб-страницах.

Таким образом, предлагаемая одностраничная основа SPA может быть эффективно применена при создании веб-приложения для обработки изображений большого объема.

## **DEVELOPMENT OF A WEB APPLICATION FOR PROCESSING DIGITAL IMAGES**

N.I. LISTOPAD, O.N. VINICHUK

**Abstract.** This work was devoted to the analysis of existing web applications, identifying their advantages and disadvantages, as well as the implementation of a web application for processing digital images.

*Keywords:* Web applications, vector and raster representations, multi-page (MPA) and single-page (SPA) applications.

### **Список литературы**

1. Флэнаган Д. JavaScript. Подробное руководство. СПб: Символ-Плюс, 2008.
2. Прохоренок Н.А. HTML, JavaScript, PHP и MySQL. Джентльменский набор Web-мастера. СПб.: БХВ-Петербург, 2010.