

УДК: 004.415.25

СПОСОБЫ РЕАЛИЗАЦИИ ИНФОРМАЦИОННЫХ СИСТЕМ ПОД МОБИЛЬНЫЕ УСТРОЙСТВА

Шарафиев Дмитрий Евгеньевич

Студент 3-го курса

Казанский государственный энергетический университет

Аннотация: в статье рассмотрена реализация информационных систем, используемые при этом архитектурные решения, а также детально рассмотрена адаптация систем под мобильные устройства.

Ключевые слова: веб система, адаптивный сайт, мобильное приложение, REST, WebSocket, Android, iOS.

METHODS FOR IMPLEMENTATION OF INFORMATION SYSTEMS FOR MOBILE DEVICES

Sharafiev D.E.

Abstract: the article discusses the implementation of information system, the architecture solutions used in this and also considered in detail the adaptation of system for mobile devices.

Keywords: web system, responsive website, mobile application, REST, WebSocket, Android, iOS.

В настоящее время идет тенденция на реализацию информационных систем с адаптацией под мобильные устройства. Веб системы без отображения на мобильные устройства становятся все менее актуальными, за исключением внутренних корпоративных систем компаний, но и для таких систем все чаще руководящий состав заказывает разработку мобильных приложений или интеграцию системы с уже существующим мобильным приложением. Совершенствование технологий и средств реализации, позволяет усовершенствовать системы, усложняя их, но в то же время делая их более удобными для пользователей. Существует два способа адаптации веб систем под мобильные гаджеты: адаптивная верстка сайта и разработка мобильного приложения, которые могут быть реализованы с помощью кроссплатформенных или нативных технологий.

Веб системы реализуют клиент-серверную архитектуру, которая представляет из себя клиентскую часть и серверную часть.

Клиентская часть – это веб страница, которую пользователь открывает в браузере для работы с этим веб ресурсом.

Серверная часть представляет из себя бизнес логику работы данного ресурса, а также базу данных, в которой хранятся все данные этого веб ресурса.

Взаимодействие серверной и клиентской части называется application programming interface (API) [1]. Схема взаимодействия клиента и сервера представлена на рисунке 1.

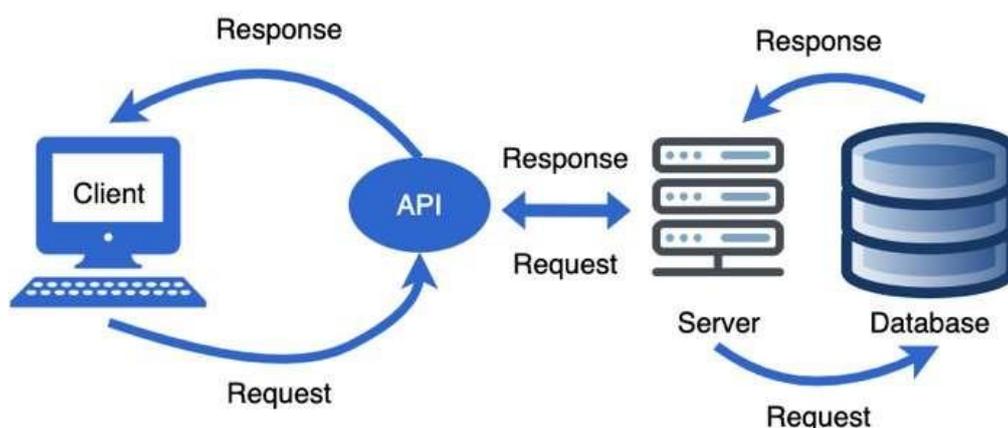


Рис. 1 Схема взаимодействия клиента и сервера.

Существует две известных реализации API: Representational State Transfer (REST) и Web Sockets.

REST — это архитектура организации взаимодействия приложения/сайта с сервером посредством протокола HTTP (HyperText Transfer Protocol). Особенность REST [2] в том, что сервер не запоминает состояние пользователя между запросами - в каждом запросе передаётся информация, идентифицирующая пользователя и все параметры, необходимые для выполнения операции. Всё взаимодействие с сервером сводится к четырём операциям, называемым CRUD (READ, CREATE, DELETE, UPDATE) [3]. Каждая операция должна выполнять определённые правила:

- READ - получение данных с сервера (обычно в формате JSON, или XML);
- CREATE - добавление новых данных на сервер;
- UPDATE - модификация существующих данных на сервере;
- DELETE - удаление данных на сервере;
- каждый URL является ресурсом.

Операция получения данных не может приводить к изменению состояния сервера. Для каждого типа операции используется свой метод HTTP-запроса:

- получение – GET;
- добавление – POST;
- модификация – PUT;
- удаление – DELETE.

Протокол WebSocket (стандарт RFC 6455) предназначен для решения любых задач и снятия ограничений обмена данными между браузером и сервером. Протокол WebSocket работает над TCP. Благодаря WebSocket, можно реализовать привычный пользователю чат, в котором новые сообщения приходят на устройство без какого-либо действия пользователя.

Он позволяет пересылать любые данные, на любой домен, безопасно и почти без лишнего сетевого трафика [4].

Адаптация таких систем под мобильные устройства может быть реализована тремя способами: адаптация сайта под мобильные устройства, разработка мобильной версии сайта и разработка мобильного приложения.

Адаптивная верстка сайта под мобильные устройства, представлена на рисунках 2 и 3.

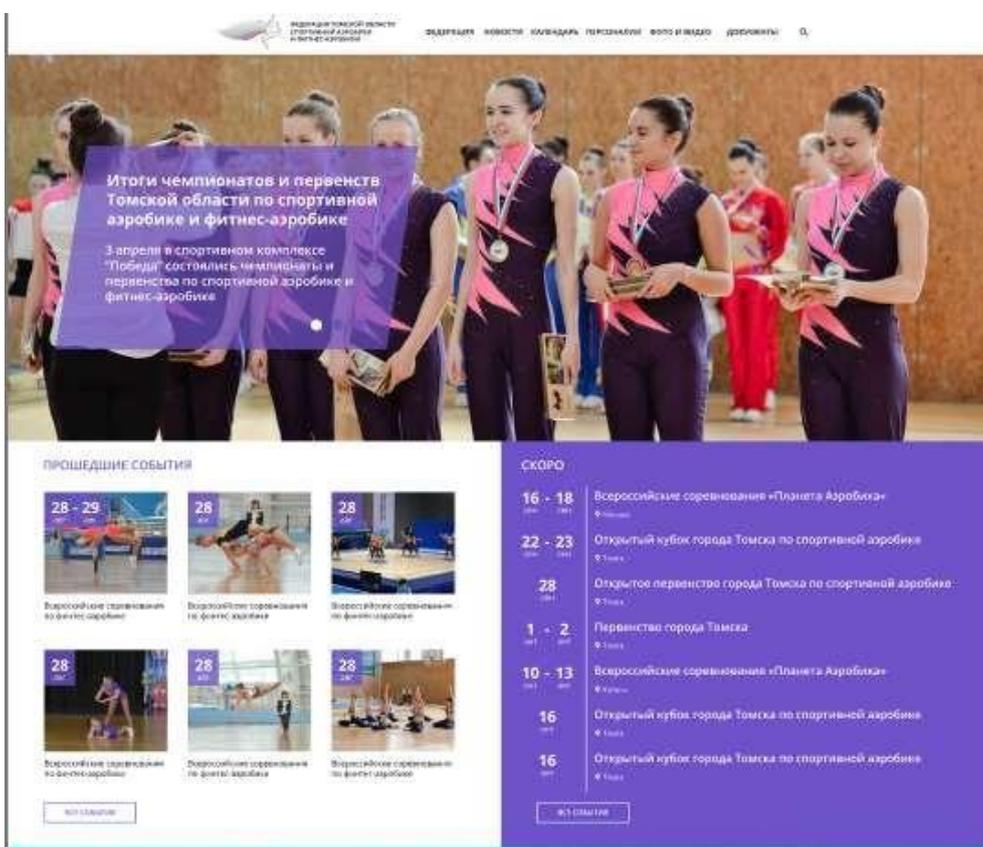


Рис. 2 Скриншот адаптивного сайта с компьютерного монитора

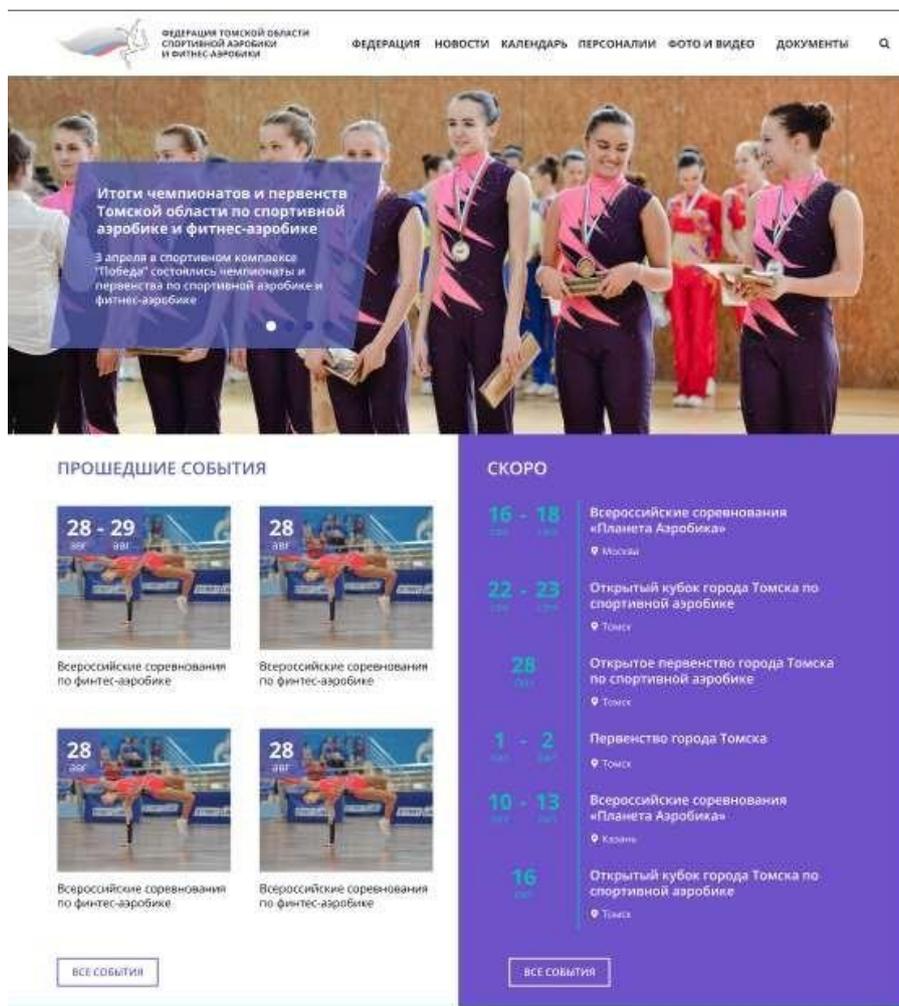


Рис. 3 Скриншот адаптивного сайта с мобильного устройства

Адаптивная верстка сайта подразумевает адаптацию элементов пользовательского интерфейса под разные экраны от маленьких экранов мобильных телефонов до больших экранов планшетов. Преимуществами такого подхода являются:

- адаптация сразу под все мобильные устройства, вне зависимости от операционной системы, будь то iOS, Android или Windows Phone;
- изменения на сайте отображаются и на мобильных устройствах;
- не требует дополнительной разработки, за исключением веб системы.

Недостатками данного подхода являются:

- весь контент, который есть на сайте также будет виден в мобильном приложении. Некоторые части сайта могут быть абсолютно бесполезными для мобильного приложения и их лучше не показывать;
- необходимо постоянное подключение к интернету;
- медленный в плане производительности и скорости работы на мобильном устройстве.

Мобильная версия сайта представляет из себя отдельное веб приложение, вынесенное в другой поддомен веб системы.

Преимущества данного подхода, следующие:

- в мобильную версию сайта включен только тот контент, который нужен пользователю мобильного приложения;
- знакомая и удобная пользователю навигация, например, в виде нижней панели или «бургер» навигации.
- не нужно адаптировать элементы пользовательского интерфейса под компьютерные мониторы.

Недостатками данного подхода, следующие:

- необходима разработка отдельного веб приложения для мобильной версии сайта;
- медленный по скорости в мобильном устройстве;
- необходимо постоянное подключение к интернету;
- при внесении изменений на основной сайт, нужно вносить изменения в мобильную версию сайта.

Мобильное приложение представляет из себя отдельное приложение, оно содержит нужную логику, которая предоставляет пользователю весь необходимый персонал. В нем, как правило, есть удобная навигация, которая позволяет быстро переключаться между разделами мобильного приложения. Также, в мобильном приложении, для удобства пользователя, может быть реализовано локальное хранение данных для доступа к ним в offline режиме.

Преимущества мобильного приложения, следующие:

- главное преимущество – скорость работы и время отклика;
- привычный для пользователя интерфейс;
- доступ к функциям телефона, таким как: камера, галерея, список контактов, календарь, push уведомления, биометрические способы подтверждения личности владельца телефона и т.д;
- есть возможность реализовать offline режим для работы в приложении без интернета.

Недостатки мобильного приложения, следующие:

- необходимо создание двух мобильных приложений, в случае нативного способа разработки, или одного мобильного приложения, в случае кроссплатформенного подхода, что все равно трудозатратней, чем написать адаптивный сайт или мобильной версии сайта;
- при внесении изменения на сайт, необходимо вносить изменение в одно приложение, если оно кроссплатформенное, или в два разных приложения, если оно реализовано нативно.

Заключение

В данном материале рассмотрены варианты адаптации информационных систем под мобильные устройства с использованием мобильного приложения и адаптивной верстки.

Список литературы

1. Что такое API? Простое объяснение для начинающих [Электронный ресурс]. Режим доступа: <https://dev.by/news/что-такое-api-простым-языком>, свободный.

2. REST: простым языком [Электронный ресурс]. Режим доступа: <https://medium.com/@andr.ivas12/rest-%D0%BF%D1%80%D0%BE%D1%81%D1%82%D1%8B%D0%BC-%D1%8F%D0%B7%D1%8B%D0%BA%D0%BE%D0%BC-90a0bca0bc78>, свободный.

3. CRUD [Электронный ресурс]. Режим доступа: <https://dic.academic.ru/dic.nsf/ruwiki/1266403>, свободный.

4. WebSocket [Электронный ресурс]. Режим доступа: <https://learn.javascript.ru/websockets>, свободный .