

поддержка вовсе не гарантируется, также оставляет желать лучшего работа flex в стандартных браузерах старых устройствах на Android и iOS. Поэтому большинству разработчиков приходится использовать Bootstrap версии 3, либо же другие сетки, фреймворки и библиотеки, которые поддерживают и корректно работают в старых браузерах.

Стоит упомянуть, что существует еще более приемлемый метод решения проблемы отображения сайтов на мобильных устройствах, а именно отзывчивая верстка. Это объединение резиновой и адаптивной верстки. В реализации самая сложная, но позволяет достигнуть лучших результатов.

УДК 621

Капуста Е. В.

СОВРЕМЕННЫЕ ПУТИ МОБИЛЬНОЙ РАЗРАБОТКИ

БНТУ, Минск

Научный руководитель Дробыш А. А.

В настоящее время известны «классические» способы разработки мобильных приложений для платформ iOS и Android (Swift (Objective C) и Java соответственно). Помимо них, существует множество альтернативных способов создания мобильных приложений. Рассмотрим их вкратце.

QT.Ю предоставляет набор инструментов разработчика для написания программ. Программы могут быть написаны один раз (язык программирования – C++), затем с помощью незначительных изменений могут быть скомпилированы для платформ Android, iOS, Windows Phone, Windows, Linux, Mac.

Xamarin представляет собой набор инструментов для написания приложений с использованием языка программирования C# для платформ MacOS, iOS, Android.

Фреймворк состоит из нескольких основных частей:

- Xamarin.iOS – библиотека классов для C#, предоставляющая разработчику доступ к iOS SDK;
- Xamarin.Android – библиотека классов для C#, предоставляющая разработчику доступ к Android SDK;
- Компиляторы для iOS и Android;
- IDE Xamarin Studio;
- Плагин для Visual Studio.

На текущий момент технология Xamarin является серьезным инструментом для решения сложных задач в области разработки мобильных приложений. Несмотря на это, команда разработчиков не останавливается и продолжает его активное развитие и улучшение. Кроме того существуют различные фреймворки, позволяющие для создания мобильных приложений использовать HTML/XML + JavaScript (TypeScript) + CSS.

NativeScript – фреймворк для кроссплатформенной разработки, придерживающийся концепции write once – use everywhere. это JS, который запускается на JSVM, специфичной для каждой системы). NativeScript предоставляет доступ ко всем системным API.

Cordova / PhoneGap – PhoneGap изначально основывается на Cordova. Cordova/Phonegap предлагает набор Javascript API, которые позволяют подключаться к нативным функциям устройства (камера, компас, контакты, геолокации). Cordova/Phonegap позволяет создавать мобильные приложения без необходимости использовать нативный язык программирования, вместо которого можно использовать фреймворк типа jQuery Mobile. Инструмент скомпилирует ваше приложение при помощи SDK платформы, и оно будет готово к установке на любую платформу, включая iOS, Android, Windows Phone, Blackberry и Web OS.

Ionic – это мобильный фреймворк на HTML5, сосредоточенный на производительности, повышение которой достигается

за счет минимизации использования аппаратного ускорения. К тому же для работы с этим фреймворком не понадобятся сторонние JS-библиотеки. Для разработки интерактивных приложений лучше всего дополнительно использовать Angular.js. Ionic укомплектован красивым иконическим шрифтом под названием Ionicons, а также набором HTML-классов с возможностью повторного использования при разработке мобильных UI.

jQueryMobile – это продвинутый фреймворк для мобильной разработки, который позволяет создавать кросс-платформенные мобильные приложения. jQuery Mobile поддерживает множество разных платформ – от настольного и смартфона до планшета и электронных читалок типа Nook и Kindle. Подобно своему смежному проекту, jQuery UI, jQuery Mobile предлагает достаточно много пользовательских интерфейсов, которые, в данном случае, оптимизированы под мобильные и сенсорные устройства.

React Native – позволяет создавать мобильные приложения используя только JavaScript. Фреймворк использует тот же набор технологий что и React, позволяя строить богатые мобильные интерфейсы используя декларативные компоненты. При создании приложения создается не «мобильное веб-приложение», «HTML5 приложение» или «гибридное приложение», а создается действительно мобильное приложение, которое неотличимо от приложений написанных на Objective-C или Java.

По итогу в настоящее время возможно написание мобильных приложений практически на любом языке программирования. Существуют различные фреймворки, требующие от программиста различного уровня знаний и умений, что позволяет выбрать себе направление по своим склонностям и предпочтениям.