

УДК 004

СРАВНИТЕЛЬНЫЙ АНАЛИЗ ПЛАТФОРМ ДЛЯ РАЗРАБОТКИ МОБИЛЬНЫХ ПРИЛОЖЕНИЙ

Ашлапов А.М. студент гр. ИТб-182, IV курс

Научный руководитель: Алексеева Г.А., старший преподаватель
Кузбасский государственный технический университет
имени Т.Ф. Горбачева
г. Кемерово

Деятельность современного предприятия невозможно представить без применения информационных и автоматизированных систем. Программное обеспечение таких систем может быть реализовано на различных платформах, в том числе на мобильных устройствах.

Современные мобильные устройства представляют из себя многофункциональный компьютер с множеством дополнительного оборудования (WIFI и Bluetooth адаптеры, NFC модуль, фото и видео цифровую камеру и многое другое), позволяющего их использовать для решения различных задач, в том числе и на производстве. Приложения для мобильных устройств прошли в своей эволюции от небольших программ для создания отчетов о выполненной работе до полноценного модуля информационной системы предприятия, которое может быть использовано в качестве основного или альтернативного рабочего места.

Существует достаточно большое количество программных сред (платформ) для разработки ИС для мобильных устройств. Предпочтительнее использовать для создания мобильного приложения кроссплатформенные среды разработки, которые позволяют обеспечить работу на нескольких платформах (Android, IOS, Windows Phone) и при этом снизят временные затраты на разработку, так как не потребуется реализовывать ИС на каждую платформу в отдельности. На рисунке 1 представлена диаграмма, демонстрирующая статистику использования различных мобильных фреймворков разработчиками в 2019 и 2020 годах.

Для сравнительного анализа выбираем следующие платформы: React Native, Cordova, Xamarin, Flutter.

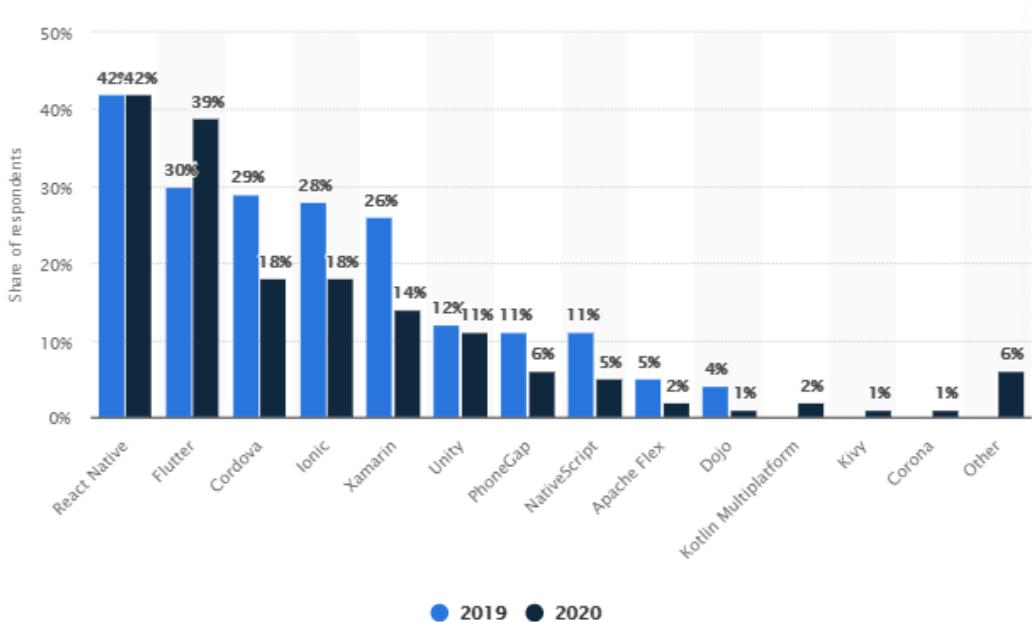


Рисунок 1

React Native – это платформа для разработки приложений, созданная Facebook. В качестве языков программирования используется JavaScript и TypeScript. React Native имеет широкий спектр используемых платформ: Android, Android TV, iOS, macOS, Apple tvOS, Web, Windows и UWP. Отличительная особенность React Native – это разрабатывать интерфейсы приложений без веб-технологий, позволяя использовать свои собственные элементы интерфейса. Как и многие популярные платформы React Native может быть использован в Visual Studio. На рисунке 2 представлена архитектура React Native.

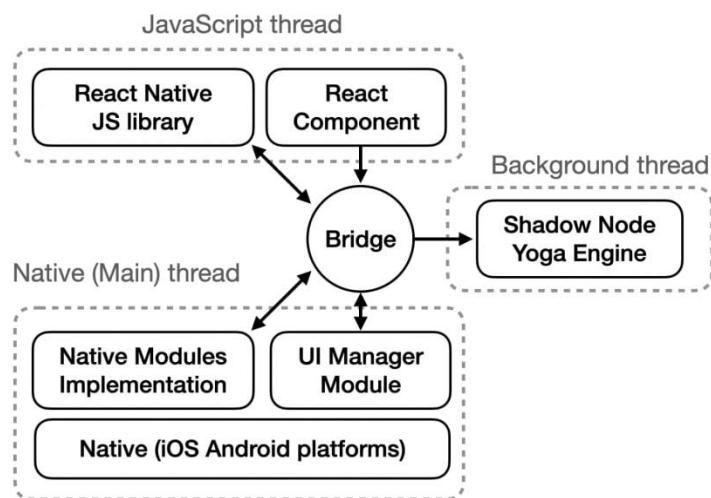


Рисунок 2

React Native одна из молодых и набравших большую популярность платформ, а все благодаря тому что ее используют компаниями-гигантами, такие

как Facebook, Tesla, Uber и многие другие. Писать программы на React Native несложно из-за использования языка JavaScript. Еще одна полезная функция – это Hot Reloading. Он позволяет вносить изменения прямо во время выполнения работы программы. Все это делает React Native хорошим кандидатом для использования.

Cordova – это среда для разработки мобильных кроссплатформенных приложений, использующая HTML5, CSS3 и JavaScript. Для работы доступны сторонние плагины, что помогает при создании приложений с большим количеством функций. Поддерживаемые платформы Cordova - Android, IOS, Windows Phone. Cordova является программным обеспечением с открытым кодом и бесплатным использованием. В дополнении ко всему, пакеты Cordova могут быть установлены в Visual Studio. На рисунке 3 представлена архитектура Cordova.

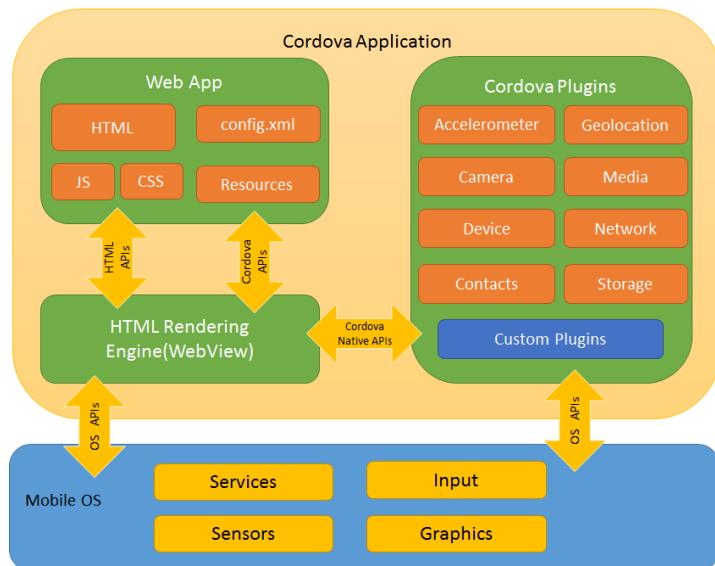


Рисунок 3

Все это делает Cordova хорошим решением, чтобы рассматривать кроссплатформенную разработку в ходе курса направления ИТ. Бесплатность подкрепляет данный вариант.

Xamarin (рисунок 4) – это платформа для создания мобильных приложений. Ее владельцем является Microsoft. Платформа поддерживает разработку для Android, IOS и Windows. Отличительной особенностью Xamarin является использование языка C# и Xaml. Приложения создаются с помощью инструментов Xamarin.Forms и Xamarin.Android/iOS. В первом случае есть возможность написания кода полностью для всех платформ, что сокращает временные затраты на разработку, а при использовании второго варианта повышается производительность работы приложения.

Xamarin.Forms

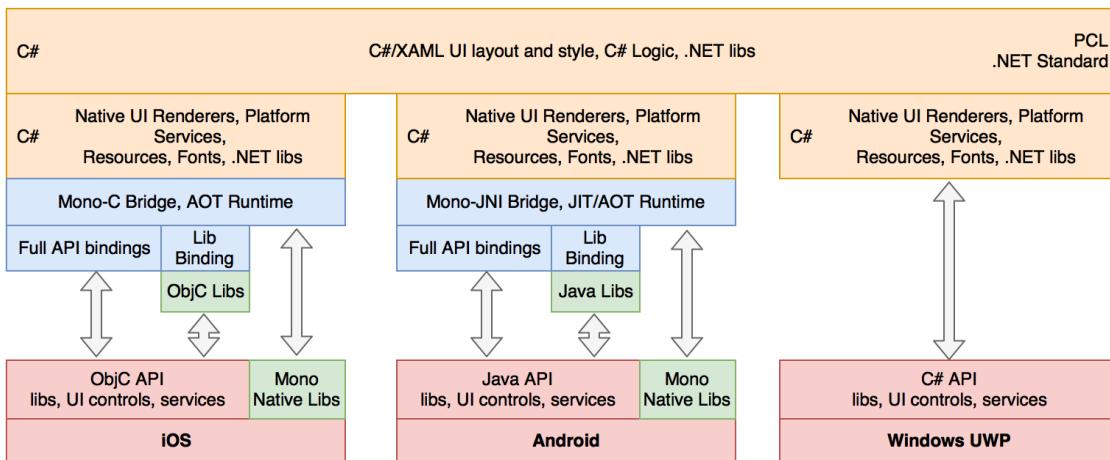


Рисунок 4

Все это делает платформу вариативной для разработчика, позволяя ему создавать приложения с уклоном в требуемую область: сокращение времени написания или повышение производительности, если требуется многофункциональное приложение. Использование C# делает Xamarin одним из лучших возможных платформ для мобильных разработок на курсе.

Flutter (рисунок 5) – платформа для разработки кроссплатформенных приложений, разработанное Google. Flutter может быть использован для создания приложений для iOS и Android, или настольных приложений на Windows, macOS и Linux. Отличительная особенность Flutter – это наличие собственного движка для рендеринга, что ускоряет производительность, а также использует язык Dart. Язык является по своей сути усовершенствованным JavaScript, что может создать некоторые трудности.

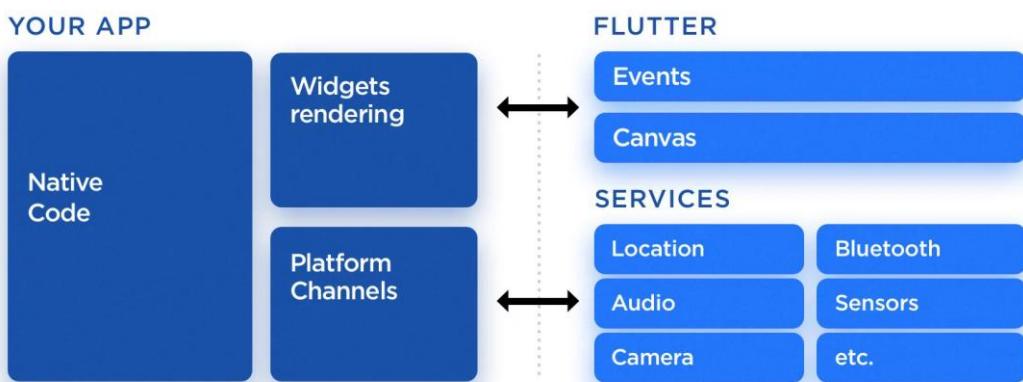


Рисунок 5

Flutter пользуется большой популярностью среди разработчиков. Он удобен для создания приложений и имеет интересную архитектуру, благодаря ко-

торой достигается хорошая производительность. Имеется несколько недостатков – изучение нового языка, установка и разбор новых редакторов исходного кода на подобие Visual Studio Code.

Все рассмотренные кроссплатформенные приложения обладают своими плюсами и минусами. По своей сути при разработке приложения стоит выбирать платформу либо ориентируясь на штат разработчиков и язык программирования, который для них является предпочтительным, либо в зависимости от того, какие требования должны быть реализованы при создании приложения (скорость разработки, быстродействие или дизайн).

Исходя из сравнительного анализа наиболее предпочтительным является Xamarin. Открытость кода, бесплатное распространение, и удобство в использовании обеспечивают ему популярность среди других платформ.

Список литературы:

1. Архитектурный обзор платформы Cordova [Электронный ресурс]. – URL: <https://cordova.apache.org/docs/en/11.x/guide/overview/index.html> (Дата обращения: 28.03.2022)
2. Свежее дополнение к Visual Studio для создания кроссплатформенных приложений на HTML и JavaScript [Электронный ресурс]. – URL: <https://habr.com/ru/company/microsoft/blog/224999/> (Дата обращения: 28.03.2022)
3. Understanding React Native Architecture [Электронный ресурс]. – URL: <https://dev.to/goodpic/understanding-react-native-architecture-22hh> (Дата обращения: 28.03.2022)
4. Core Components and Native Components [Электронный ресурс]. – URL: <https://reactnative.dev/docs/intro-react-native-components> (Дата обращения: 28.03.2022)
5. Architecture-Xamarin [Электронный ресурс]. – URL: <https://docs.microsoft.com/ru-ru/xamarin/android/internals/architecture> (Дата обращения: 29.03.2022)
6. На чём писать мобильные кроссплатформенные приложения – Лайв Тайпинг [Электронный ресурс]. – URL: <https://livetyping.com/ru/blog/nachem-pisat-krossplatformennye-prilozhenija> (Дата обращения: 29.03.2022)
7. Flutter – Википедия [Электронный ресурс]. – URL: <https://ru.wikipedia.org/wiki/Flutter> (Дата обращения: 29.03.2022)