Что такое React Native?

React Native (также известный как RN) - популярная платформа мобильных приложений на основе JavaScript, которая позволяет создавать отрисованные мобильные приложения для iOS и Android. Фреймворк позволяет создавать приложения для разных платформ с использованием одной и той же кодовой базы.

45

React Native история

Когда Facebook впервые решил сделать свой сервис доступным на мобильных устройствах, вместо того, чтобы создавать собственное приложение, как многие ведущие технологические игроки того времени, они решили использовать мобильную веб-страницу на основе HTML5. Однако решение не выдержало проверки временем, оставив достаточно места для улучшения пользовательского интерфейса и производительности. Фактически, в 2012 году Марк Цукерберг признал, что «самая большая ошибка, которую мы сделали как компания, заключалась в том, что мы слишком полагались на HTML, а не на нативный».

Вскоре после этого, в 2013 году, разработчик Facebook Джордан Уолк совершил революционное открытие - он нашел способ создавать элементы пользовательского интерфейса для приложений iOS с помощью JavaScript. Это вызвало пожар, и был создан специальный хакатон, чтобы узнать еще больше о том, как много можно сделать для разработки мобильных приложений с использованием (пока что традиционно веб-ориентированных) решений JavaScript.

Так родился React Native. Первоначально разработанный только для iOS, Facebook быстро последовал за Android и поддержал его, прежде чем выпустить фреймворк в 2015 году. Всего три года спустя React Native уже стал вторым по величине проектом на GitHub с точки зрения участников. В 2019 году он занял шестое место с более чем 9100 записями.

React vs. React Native

React.js - это библиотека JavaScript, которая помогает компаниям создавать красивые пользовательские интерфейсы. Одна из основных особенностей React заключается в том, что он может работать на стороне клиента и отрисовывать на стороне сервера, и эти части могут взаимодействовать друг с другом. Поэтому он широко используется для создания высокопроизводительных веб-приложений и пользовательских интерфейсов.

React Native - это среда разработки мобильных приложений, которая создает мобильные приложения с использованием React.js. Он позволяет создавать богатый мобильный пользовательский интерфейс из декларативных компонентов. React Native предлагает такие функции, как Hot Reloading, чтобы разрабатывать приложения быстрее и с меньшими усилиями. Библиотеки React Native были выпущены Facebook в 2015 году, давая архитектуру React приложениям для Android, iOS и Windows.

React.js использует виртуальную модель DOM для создания лучшего UX. Построение модели DOM требует времени, потому что страницы сегодня большие. Но React.js выполняет свою работу быстрее, используя виртуальную модель DOM. Таким образом, React.js использует абстрактную копию объектной модели документа и предоставляет изменения для одного компонента, не затрагивая остальную часть пользовательского интерфейса. Это то, что делает React.js одним из лучших решений для быстрого обновления и создания динамических пользовательских интерфейсов.

React Native делает это на шаг раньше. Он использует собственные API-интерфейсы для визуализации частей пользовательского интерфейса, которые можно повторно использовать на платформах iOS и Android. А также Java API для рендеринга компонентов Android и Objective-C API для написания компонентов iOS. Он также использует JavaScript для остальной части кода и настройки приложения для каждой платформы. Это дает React Native возможность повторно использовать компоненты и коды.

REACT Native - отличный выбор для разработчиков с опытом работы с JavaScript, так как нет необходимости изучать Java для Android или Swift для iOS.

Еще одна причина, по которой фреймворк REACT Native приобрел популярность в последние годы, заключается в том, что он помогает одновременно обслуживать обе платформы и положить конец священной войне вариантов разработки для iOS или Android для мобильных разработчиков.

Как это устроено?

В каждом приложении React Native работают два важных потока.

Один из них - это основной поток, который также выполняется в каждом стандартном собственном приложении. Он обрабатывает отображение элементов пользовательского интерфейса и обрабатывает пользовательские жесты.

Другой характерен для React Native. Его задача - выполнять код JavaScript в отдельном движке JavaScript. JavaScript имеет дело с бизнес-логикой приложения. Он также определяет структуру и функциональность пользовательского интерфейса.

Эти два потока никогда не взаимодействуют напрямую и никогда не блокируют друг друга.

Как взаимодействуют потоки?

Между этими двумя потоками находится так называемый мост, который является ядром React Native. Мост имеет три важных характеристики.

Асинхронный. Это обеспечивает асинхронную связь между потоками. Это гарантирует, что они никогда не блокируют друг друга.

Сдан. Он передает сообщения из одного потока в другой оптимизированным способом.

Сериализуемый. Два потока никогда не используют одни и те же данные и не работают с ними. Вместо этого они обмениваются сериализованными сообщениями.

Как выглядит процесс разработки?

Было бы интересно посмотреть, как выглядит процесс разработки.

Здесь у нас есть компонент с именем Greeting, который отображает только текст "Hello!"

Чтобы запустить приложение для iOS, нам нужно запустить ios, поддерживающую реакцию, из командной строки. Это запустит приложение на симуляторе или на реальном устройстве, если оно подключено к компьютеру.

Чтобы запустить наше приложение для Android, нам нужно запустить run-android. Для этого мы можем полностью повторно использовать наш компонент Greeting. Это связано с тем, что компонент не содержит кода для конкретной платформы. React Native заботится о предоставлении Android TextViews вместо iOS UIViews. Возможность повторного использования кода - одна из сильных сторон этой технологии.

Преимущества React Native

Мы обсудили продукты, созданные с использованием React Native, поэтому давайте перейдем к преимуществам разработки React Native и почему вы должны выбрать его в качестве решения для создания своего мобильного приложения.

- Повторное использование кода - кроссплатформенная разработка

- Большое сообщество разработчиков.

- Экономическая эффективность.

это потому, что разработчики могут использовать один и тот же код для создания приложений для iOS и Android.

Это означает, что вам не нужно нанимать две отдельные группы разработчиков iOS и Android для завершения проекта.

- Быстрое обновление.

Быстрое обновление позволяет разработчикам запускать приложение, обновляя его до новых версий и изменяя пользовательский интерфейс. Изменения сразу видны, и разработчик избавляется от необходимости перестраивать все приложение.

- Простой интерфейс

Разработка на React Native использует React JavaScript для создания интерфейса приложения, что делает его более отзывчивым и быстрым с уменьшенным временем загрузки, что улучшает общее впечатление пользователя.

- Быстрые приложения

React Native и Swift - оба достигли аналогичных результатов по производительности.

- На будущее

Учитывая скорость, с которой фреймворк захватил рынок, и его простой подход к решению проблем разработки, будущее React Native для кроссплатформенных приложений выглядит радужным.

React Native: риски и недостатки

Вот четыре основных потенциальных недостатка, о которых вам нужно знать, прежде чем вы решите разработать приложение React Native.

- Отсутствие некоторых кастомных модулей

Хотя React Native существует уже несколько лет, некоторые настраиваемые модули либо оставляют место для улучшения, либо полностью отсутствуют. Это означает, что вам может потребоваться запустить три отдельные базы кода (для React Native, iOS и Android) вместо одной.

- Проблемы совместимости и отладки

Хотя это может показаться неожиданным - React Native все-таки используется ведущими техническими игроками - он все еще находится в стадии бета-тестирования. Ваши разработчики могут столкнуться с различными проблемами совместимости с пакетами или инструментами отладки. Если ваши разработчики не владеют React Native, это может негативно повлиять на вашу разработку, поскольку они тратят время на длительное устранение неполадок.

- масштабируемость

В большинстве случаев React Native будет работать для вас очень хорошо, даже если ваше приложение со временем превратится в очень сложное и сложное решение. В конце концов, такие компании, как Facebook и Skype, добились больших успехов с фреймворком и постоянно используют его в течение многих лет. Тем не менее, некоторые компании решили прекратить использование React Native.

Airbnb, например, решил использовать фреймворк для своего мобильного приложения еще тогда, когда компания была только начинающим стартапом. Однако со временем React Native оказался неподходящим для планов роста компании, и Airbnb прибегла к разработке двух собственных приложений.

- Нужна помощь от разработчиков.

Помните функцию «моста», о которой мы упоминали ранее в этом посте? Как показано, React Native соединяет JavaScript с собственным мобильным кодом. Это означает, что если вы возьмете разработчика, который не знает нативной мобильной разработки, ему будет трудно включить нативный код в кодовую базу RN.

React Native - универсальная кроссплатформенная среда разработки.

React Native - отличное решение для создания приложений, которые работают без сбоев независимо от платформы или системы, на которой они работают.

Многие фантастические продукты были созданы с использованием React Native, включая Skype, Facebook, Pinterest и UberEats, что еще раз доказывает, что это фреймворк, который стоит рассмотреть при разработке приложений. Хотя собственное решение для iOS и Android по-прежнему является лучшим вариантом для проектов, требующих бесшовного пользовательского интерфейса и высочайшей производительности, React Native - отличный вариант, если у вас ограниченный бюджет и достаточно удовлетворительного пользовательского опыта.