

# ReactJS против React Native – ключевые различия, преимущества и недостатки

14.03.2023

Веб-разработка – очень важное понятие в 21 веке. Каждый бизнес, разработчик продукта, поставщик поддержки или маркетолог нуждается в веб-сайте или веб-приложении для взаимодействия с клиентами и распространения своего бренда, продукта или услуги по всему миру. При такой высокой значимости важно знать, что такое JavaScript и как они соотносятся друг с другом. JavaScript был важной частью веб-разработки с момента ее появления. React и React Native являются одними из лучших JavaScript-фреймворков, которые взяли мир штурмом и создали несколько трендов, рабочих мест и эстетически привлекательных веб-сайтов! Очень важно, чтобы вы понимали, в чем заключается сравнение между ReactJS и React Native.

## Разница между ReactJS и React Native

Коэффициент сравнения	React.js	React Native
Процесс установки	ReactJS – это библиотека JavaScript, устанавливаемая с помощью менеджера пакетов npm.	React Native – это инструмент интерфейса командной строки, который требует установки Node.js и React Native CLI.

<p>Эффективность</p>	<p>ReactJs более эффективна с точки зрения времени разработки и возможности повторного использования кода.</p>	<p>React Native более эффективен с точки зрения производительности и использования памяти.</p>
<p>Технологическая база</p>	<p>ReactJS – это библиотека JavaScript, используемая для построения пользовательских интерфейсов. Она использует алгоритм Virtual DOM для повышения производительности.</p>	<p>React Native – это кроссплатформенная структура или фреймворк для разработки мобильных устройств, в котором в качестве строительных блоков используются не веб-компоненты, а нативные компоненты. Это позволяет создавать приложения, которые имеют более “родные” ощущения и внешний вид.</p>
<p>Целесообразность</p>	<p>ReactJS является более целесообразным вариантом для разработки веб-приложений.</p>	<p>React Native – более целесообразный вариант для разработки мобильных приложений.</p>

Компоненты	Компоненты ReactJS обычно написаны на HTML. Эти компоненты должны быть импортированы в приложение, прежде чем их можно будет использовать.	Компоненты React Native написаны на JSX, синтаксическом расширении JavaScript. Эти компоненты компилируются непосредственно в нативный код.
Совместимость	Он совместим с огромным количеством браузеров, включая Internet Explorer.	React Native – это фреймворк, используемый для разработки нативных мобильных приложений. Он совместим только с двумя основными мобильными платформами, iOS и Android.
Навигация	ReactJS использует традиционный подход на основе браузера.	React Native – лучший выбор для создания нативных мобильных приложений, чем ReactJS. React Native опирается на навигационные компоненты родной платформы.
Хранение	ReactJS – хороший выбор для проектов, требующих высокопроизводительного хранения данных, таких как динамические веб-приложения.	React Native – лучший выбор для проектов, которым требуется возможность легкого масштабирования.

<p>Дружественный к поисковым системам</p>	<p>ReactJS более дружелюбен к поисковым системам, чем React Native.</p>	<p>React Native нельзя сделать дружелюбным к поисковым системам – с помощью этого фреймворка сложнее добиться хорошего SEO.</p>
<p>Платформа</p>	<p>React – это фреймворк для создания приложений с использованием JavaScript.</p>	<p>React Native – полноценная платформа, позволяющая создавать нативные и кроссплатформенные мобильные приложения.</p>
<p>Кривая обучения</p>	<p>React.js является наиболее используемой библиотекой JavaScript для создания высокопроизводительных пользовательских интерфейсов, что делает ее легкой для изучения.</p>	<p>React Native содержит все компоненты React; таким образом, изучение одного из них полезно для обоих.</p>
<p>Рендеринг</p>	<p>Браузерный код в React рендерится через виртуальный DOM.</p>	<p>React Native использует нативные API для рендеринга всех компонентов на мобильных устройствах.</p>
<p>Синтаксис</p>	<p>React использует HTML и его синтаксический поток.</p>	<p>React Native использует синтаксис React Native.</p>

Поддержка CSS	React хорошо работает с CSS для стилизации и анимации.	React хорошо работает с CSS для стилизации и анимации
---------------	--	---

## ReactJS vs React Native

### ReactJS против React Native: Процесс установки

ReactJS – это библиотека JavaScript, используемая на любой веб-платформе.

Ее легко настроить и начать работу с ней, и она имеет большое сообщество пользователей и разработчиков. Узнать о сертификации React можно здесь.

Однако ReactJS не поддерживает мобильные устройства, поэтому для создания мобильных приложений с помощью ReactJS вам придется использовать сторонний инструментарий, например React Native или Cordova.

С другой стороны, React Native – это фреймворк для мобильных устройств, который позволяет создавать кросс-платформенные приложения. Его сложнее настроить, чем ReactJS. Тем не менее, он обеспечивает лучшую производительность и более последовательный пользовательский опыт на разных платформах.

### ReactJS против React Native: Эффективность

ReactJS считается более эффективным, чем React Native, при отображении динамического контента. Это связано с тем, что ReactJS использует виртуальную DOM (Document Object Model),

которая позволяет ускорить время рендеринга.

Напротив, React Native использует реальную DOM, что может замедлить производительность при рендеринге сложных страниц.

Кроме того, ReactJS более легкий, чем React Native, что может еще больше повысить эффективность.

В результате, если вы ищете наиболее эффективный вариант, ReactJS – это то, что вам нужно. Чтобы узнать больше об этих темах, перейдите по ссылке обучение веб-разработке.

## **ReactJS против React Native: Технологическая база**

Что касается технологической базы, то различия между ReactJS и React Native заключаются в следующем.

ReactJS основан на JavaScript, а React Native – на Objective-C или Java. Это может существенно повлиять

на производительность, поскольку нативный код быстрее интерпретируемого.

Кроме того, ReactJS использует виртуальный DOM, что может быть медленнее, чем работа с реальным DOM API.

Однако этот компромисс стоит того ради повышенной гибкости виртуального DOM.

## **ReactJS против React Native: Целесообразность**

ReactJS is a framework that is well-suited for developing web

applications. On the other hand, React Native is better suited for mobile app development.

However, both frameworks are still viable choices for application development, depending on the project requirements.

## **ReactJS vs React Native: Совместимость**

ReactJS – это библиотека JavaScript, используемая для создания пользовательских интерфейсов и совместимая с широким

спектром браузеров, включая Internet Explorer. В отличие от этого, React Native – это фреймворк, используемый для разработки

нативных мобильных приложений. Он совместим только с двумя основными мобильными платформами, iOS и Android. Он

обеспечивает лучший пользовательский опыт благодаря нативным компонентам.

## **ReactJS против React Native: Навигация**

ReactJS использует традиционный подход на основе браузера, в то время как React Native полагается на навигационные

компоненты родной платформы. Это может повлиять на общий пользовательский опыт, а также на сам процесс разработки.

ReactJS больше подходит для приложений, которые будут использоваться преимущественно в веб-браузере.

В то же время React Native – лучший выбор для создания нативных мобильных приложений.

Однако некоторые гибридные подходы сочетают в себе элементы обоих фреймворков, поэтому решение о том, какой из них использовать, лишь иногда бывает однозначным.

Лучший способ выбрать между ReactJS и React Native – оценить конкретные потребности проекта и определить, какой фреймворк обеспечит наилучший общий опыт для пользователей.

## **ReactJS vs React Native: Хранение**

ReactJS – хороший выбор для проектов, требующих высокопроизводительного хранения данных, таких как динамические веб-приложения.

Это также хороший выбор, если вам нужно хранить большие объемы данных. Однако ReactJS может быть трудно масштабировать, поэтому это не лучший выбор для больших проектов.

React Native – лучший выбор для проектов, которые должны легко масштабироваться. Он также имеет лучшую поддержку автономного хранения и кроссплатформенную совместимость.

## **ReactJS против React Native: Удобство для поисковых систем**

ReactJS более дружелюбен к поисковым системам, чем React Native. Это связано с несколькими факторами, в том числе с тем,

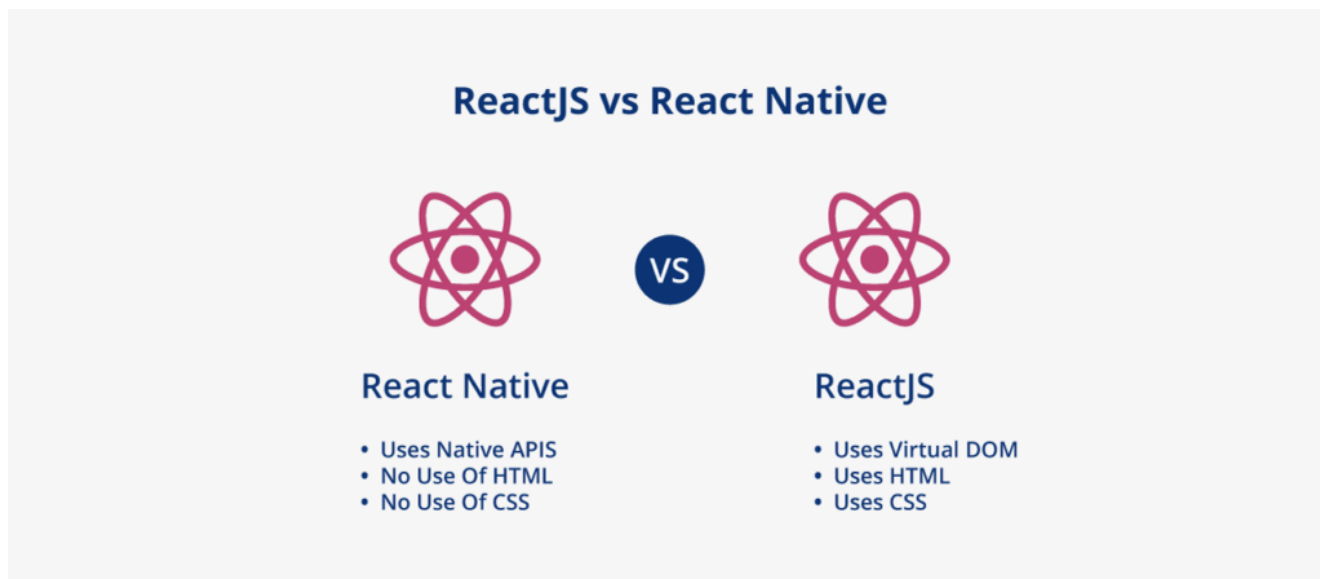
что ReactJS использует стандартные HTML-теги, которые легче индексируются поисковыми системами.

Код ReactJS, как правило, более организован и легче читается, чем код React Native,



что облегчает поисковым системам его просмотр и индексацию.

Однако это не означает, что React Native нельзя сделать дружественным к поисковым системам – добиться хорошего SEO с помощью этого фреймворка сложнее.



## Что такое ReactJS?

ReactJS – это библиотека JavaScript, которая сочетает в себе скорость JavaScript и новый способ рендеринга веб-страниц, делая их очень динамичными и чутко реагирующими на ввод пользователя. В результате этот продукт существенно изменил подход Facebook к разработке. После того как в 2013 году библиотека была выпущена в качестве инструмента JavaScript с открытым исходным кодом, она получила широкое распространение благодаря своему революционному подходу к программированию пользовательского интерфейса, что привело к длительному сотрудничеству между React и Angular, другим популярным инструментом веб-разработки.

## Что такое React Native?

Через два года после выпуска ReactJS в качестве инструмента с

открытым исходным кодом инженеры Facebook представили React Native, гибридный фреймворк для разработки приложений для iOS и Android. Инструмент был основан на тех же принципах, что и ReactJS, и вскоре был принят инженерным сообществом и компаниями, придерживающимися стратегии mobile-first.

React Native – это кроссплатформенный мобильный фреймворк, который использует ReactJs для создания уникальных веб-сайтов и приложений. Он помогает разработчикам создавать мобильные приложения путем компиляции нативных компонентов приложений в JavaScript, что позволяет запускать их на различных платформах, таких как Android, Windows, iOS.

Самое интересное, что React Native создает фреймворк Reactjs под собственным фреймворком, а Reactjs помогает создавать компоненты с помощью React Native.

## Введение в React Native

Многие люди используют React Native, от новичков, которые только начали свою карьеру в программировании, до продвинутых разработчиков iOS. React Native – это фреймворк, разработанный Facebook для создания приложений в стиле native для iOS и Android на общем языке JavaScript. Facebook впервые представил React Native только для поддержки своей системы iOS. Однако с недавней (20 января 2021 года) поддержкой операционной системы Android, библиотека теперь может предложить мобильные пользовательские интерфейсы для обеих платформ.

## Особенности React Native

**Платформено-специфичный код:** Одним из значительных преимуществ использования React Native является написание кода, специфичного для конкретной платформы, для разработки пользовательских бизнес-приложений и веб-сайтов. Как фреймворк, React Native автоматически создает приложение,

определяя платформу, на которой оно работает. Это помогает вашему бизнесу создавать правильный код для нужной платформы, тем самым улучшая процесс разработки.

**Ускоренная разработка приложений:** Большая часть кода, доступного в React Native, – это JavaScript, что помогает front-end разработчикам быстрее создавать приложения. С таким фреймворком для веб-разработки разработчикам достаточно изучить API платформы, дизайн, ориентированный на платформу, некоторые элементы пользовательского интерфейса и JavaScript. Даже если вы не являетесь великим разработчиком, вы можете быстро начать работу с React Native и воплотить свою бизнес-идею в реальность.

**Функция “горячей перезагрузки”:** С React Native ваша команда разработчиков получит возможность беспрепятственно перезагружать мобильное приложение. Это дает вашему бизнесу достаточно свободы для увеличения времени загрузки и улучшения пользовательского опыта.

**Интуитивно понятный пользовательский интерфейс:** React Native Framework состоит из библиотек пользовательского интерфейса компонентов, которые предприятия могут использовать для обеспечения бесшовного пользовательского интерфейса для своих пользователей. Совместное использование библиотек пользовательского интерфейса, таких как Expo, Shout On и т.д., экономит много времени на их создание с самого начала. Использование React Native позволяет полностью сосредоточиться на инновациях. Вместо того чтобы создавать библиотеки с нуля, вы можете сосредоточиться на развитии своего бизнеса.

## Компоненты React

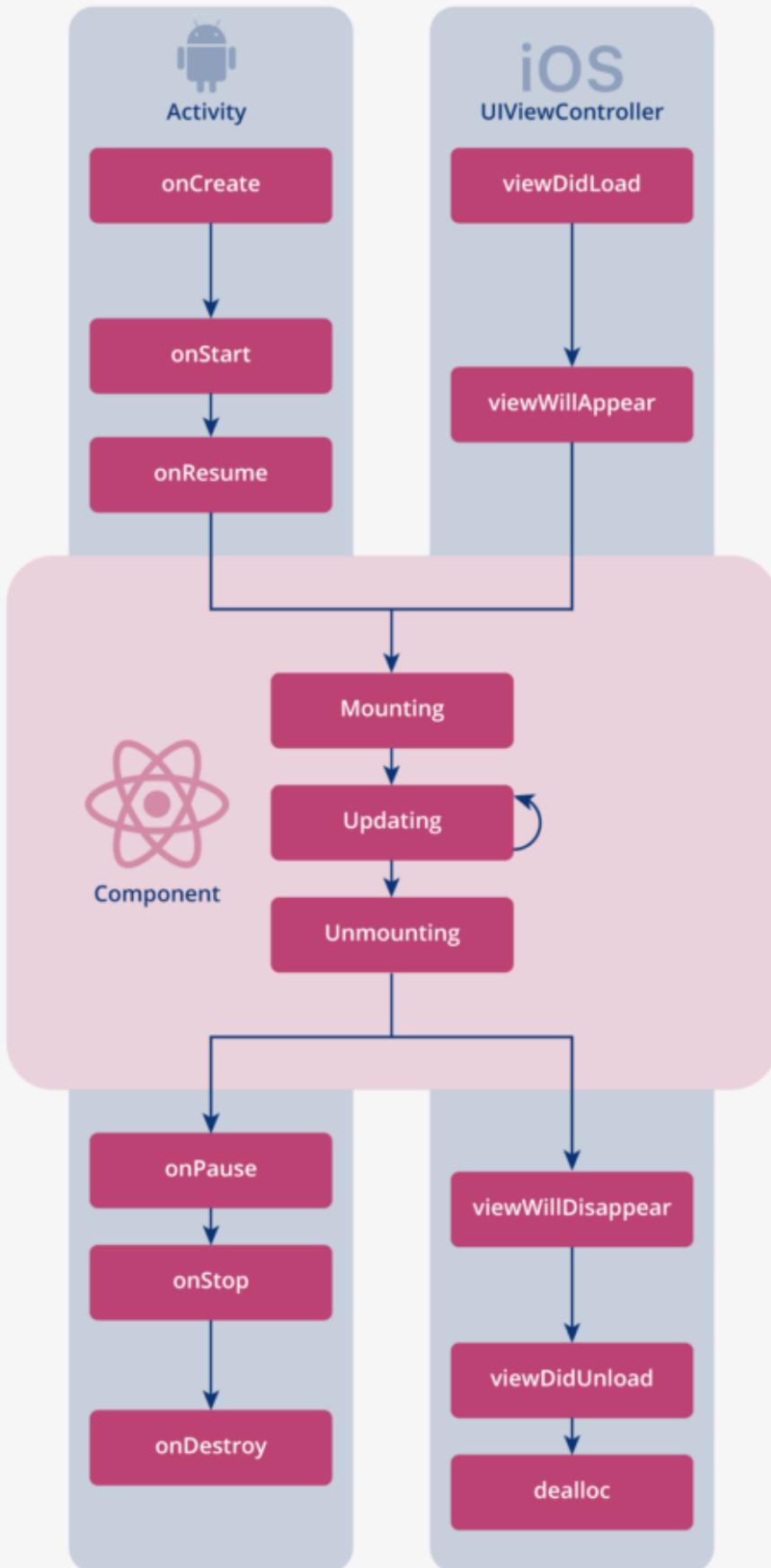
React позволяет повторно использовать компоненты кода на разных уровнях, экономя время. Дизайнеры постоянно повторно используют одни и те же активы. Если бы они этого не делали,

им пришлось бы снова и снова создавать, например, логотип компании. Все ясно: повторное использование компонентов – это эффективность дизайна. В программировании этот процесс немного сложнее. Обновление системы часто становится головной болью, поскольку каждое изменение может повлиять на работу других компонентов системы.

Разработчикам проще управлять обновлениями, поскольку все компоненты ReactJS изолированы, и изменения, внесенные в один из них, не влияют на другие. Это позволяет повторно использовать компоненты, которые не меняются, чтобы сделать программирование более точным, эргономичным и удобным для разработчиков.



# React Component-Lifecycle



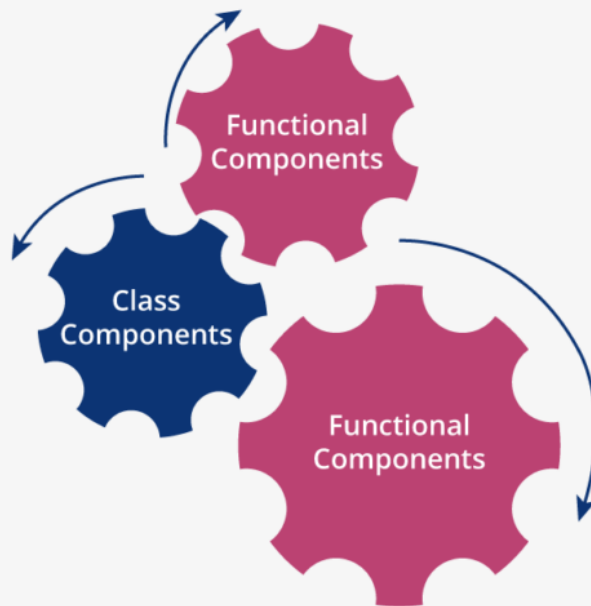
# Работа с данными в ReactJS

ReactJS позволяет работать с другими компонентами напрямую и использует обратную связь данных для того, чтобы дочерние структуры не влияли на свои родители. Это делает код стабильным.

Самые сложные системы моделей представления JS имеют важный, но понятный недостаток – структуру потока данных. Например, в системе модели представления дочерние элементы могут влиять на родительские, если их изменить. Facebook устранил эти проблемы в React JS, сделав их только видимыми.

Вместо использования явного связывания данных, ReactJS использует поток данных в одном направлении – вниз, то есть структура дочерних элементов не может повлиять на родительские элементы. Таким образом, для изменения объекта только разработчику необходимо изменить его состояние и применить обновления. Соответственно, обновляться будут только авторизованные компоненты.

## React JS Components



## Использование React Native

Приложения, разработанные на React Native, отлично работают с системами Android и iOS. React Native обеспечивает кроссплатформенную доступность, и это самое лучшее. Давайте рассмотрим еще несколько причин, почему людям нравится использовать React Native:

### 1) Живая перезагрузка

Одной из интересных особенностей React Native является наличие живой перезагрузки. Любые изменения, которые вы вносите в код приложения, будут заменены в процессе разработки. Любые изменения в логике записываются на экране, и вы можете увидеть эти изменения практически сразу.

### 2) Высокая производительность

React Native в значительной степени отвечает потребностям разработчиков, работающих над мобильными приложениями. Он



обеспечивает основу для повышения производительности с помощью графического процессора (GPU).

### **3) Удобство для бюджета и экономия средств**

Как и React JS, React Native также может повторно использовать компоненты кода. Повторное использование означает, что это экономит много времени и денег, необходимых для разработки приложения с нуля.

### **4) Управляемый сообществом**

React Native был разработан в ходе хакатона сообщества разработчиков. Это означает, что фреймворк постоянно развивается и поддерживается сообществом. Наличие команды помогает структурировать опыт разных умов.

## **Expo и React Native**

Expo – это универсальная платформа для создания приложений на основе React, которая помогает быстро разрабатывать, создавать, развертывать и итерировать мобильные приложения. Она предоставляет ряд инструментов, которые упрощают разработку с использованием React Native. Хотя для создания приложений React Native Expo не требуется, она помогает разработчикам избавиться от необходимости владения базовыми знаниями нативного языка.

- Expo Go – это приложение, которое вы можете загрузить на свой телефон, чтобы “увидеть” свое приложение в процессе разработки.
- Expo CLI – это инструмент для создания, управления и разработки приложений.
- Expo SDK – это набор модульных пакетов для доступа к родным API, таким как камера и уведомления.
- Expo Snack относится к веб-площадке. Вы можете писать и

запускать фрагменты React Native в браузере.

Иногда вы увидите упоминание Expo в документации по React Native. Это самый простой способ начать создание приложения, и здесь нет никакой привязки. Начав работу с Expo, вы всегда сможете вернуться к React Native.

## Как это работает?

В каждом приложении React Native работают два важных потока. Один из них – главный поток, который работает в каждом стандартном нативном приложении. Он отображает элементы пользовательского интерфейса и обрабатывает жесты пользователя.

Второй поток специфичен для React Native. Его функция заключается в выполнении кода JavaScript в отдельном движке JavaScript. JavaScript используется для работы с бизнес-логикой приложения. Он также определяет структуру и функциональность пользовательского интерфейса. Эти два потока никогда не общаются напрямую и не блокируют друг друга.

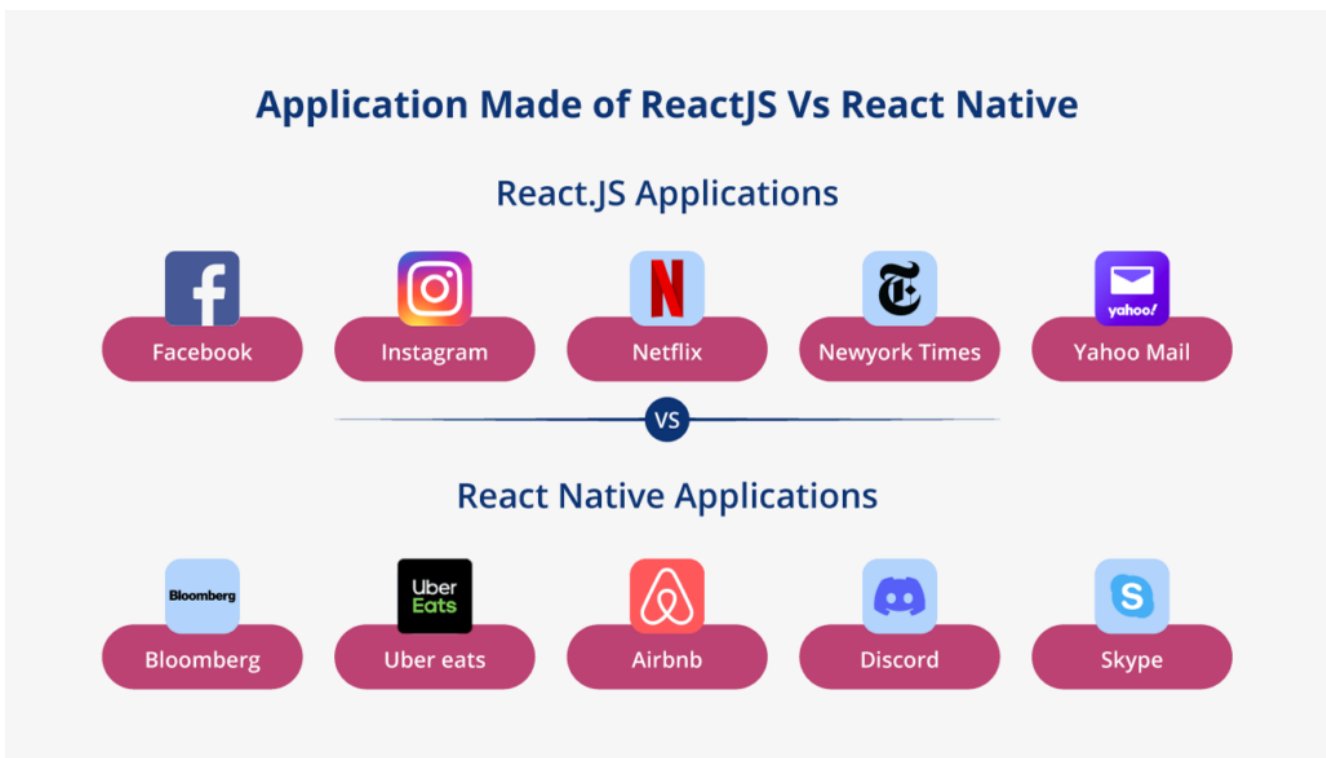
## Преимущества использования React Native

- Чтобы избежать отдельной разработки, логический слой может быть использован повторно для создания одного и того же приложения как для Android, так и для iOS.
- React Native визуализирует некоторые компоненты кода с помощью нативных API.
- Если вы умеете работать с JavaScript, то вы легко сможете работать с React Native.
- Чтобы восстановить старые приложения, достаточно добавить компоненты пользовательского интерфейса React Native, и нет необходимости переписывать их.

- React Native добавляет React в приложения, позволяя легко использовать его в различных инструментах.
- Это помогает создать эффективный пользовательский опыт без особых усилий.

# React Native VS React.js: Приложения

React Native VS React.js: Приложения



Статистика

## Statistics of React JS Vs React Native

Stars	ReactJS 174,633	React Native 19,166
Yearly Download	ReactJS 10,721,886	React Native 196,133
Weekly Download	ReactJS 11,542,466	React Native 168,613
Version	ReactJS 17.0.2	React Native 0.17.1
Forks	ReactJS 35.2K	React Native 21.2k

## Чем они похожи?

ReactJS и React Native – два популярных варианта создания пользовательских интерфейсов. Оба фреймворка основаны на React, библиотеке JavaScript, созданной Facebook. В результате между ReactJS и React Native существует много сходств. Чтобы узнать больше о сходствах или различиях, нажмите на KnowledgeHut React Certification. Вот шесть основных способов, с помощью которых эти два фреймворка похожи:

- ReactJS и React Native используют декларативный синтаксис, что делает код легким для чтения и понимания.
- Оба фреймворка поддерживают использование компонентов, которые можно повторно использовать во всем приложении.
- Оба фреймворка предлагают виртуальный DOM, который повышает производительность за счет пакетного обновления.
- ReactJS и React Native имеют открытый исходный код, что означает, что их можно использовать без затрат.
- Оба фреймворка имеют большие экосистемы с множеством

доступных плагинов и инструментов.

- Наконец, и ReactJS, и React Native поддерживаются сильным сообществом.

## **React Native для мира, ориентированного на мобильные устройства**

Основными причинами, по которым стоит рассмотреть возможность использования технологии React Native, являются низкая стоимость разработки приложений и потенциал для быстрой разработки приложений. React Native хорошо работает, если вы хотите создать одно мобильное приложение для двух платформ, таких как iOS и Android, где скорость работы не важна.

Поскольку React Native позволяет быстро разрабатывать приложения, он также является хорошим инструментом для создания прототипов и быстрого вывода идеи на рынок для проверки бизнеса.

Еще одно преимущество этого фреймворка в том, что он позволяет быстро предоставлять новую функциональность на поддерживаемых платформах.

React Native использует популярный язык JavaScript и фреймворк ReactJS, а это значит, что количество опытных разработчиков программного обеспечения, доступных на рынке, огромно.