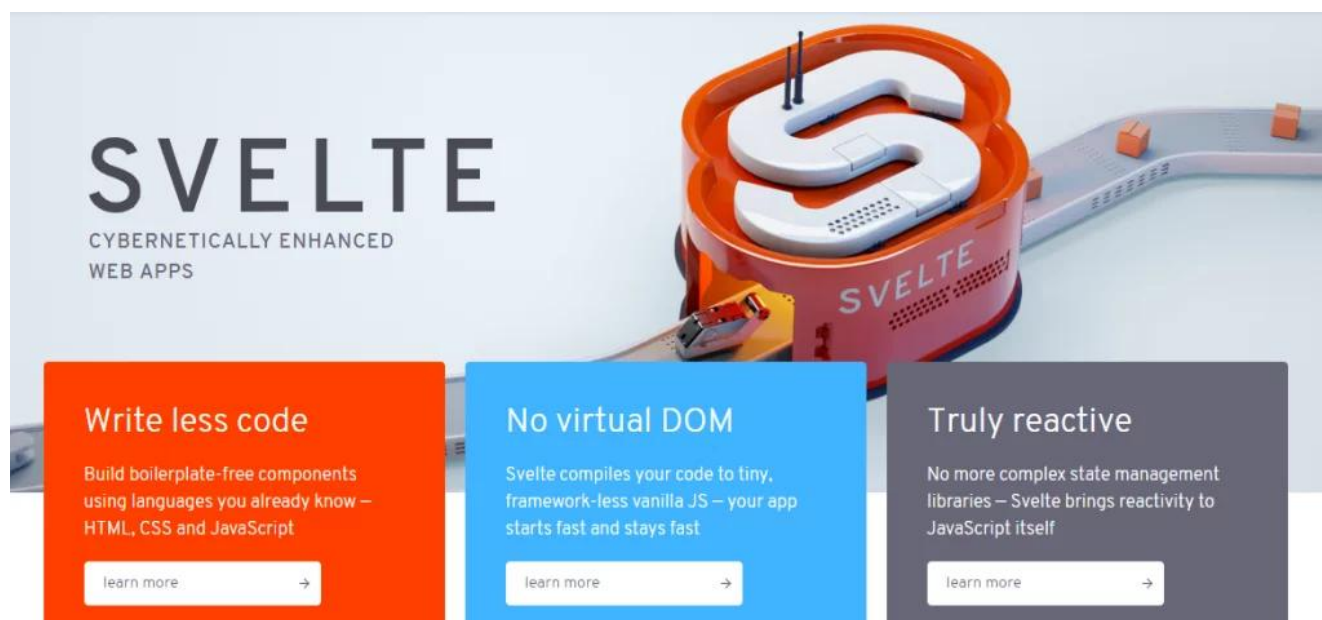


# Svelte против React: Какой JS-фреймворк выбрать в 2023 году?

30.07.2023

JavaScript является одним из самых распространенных языков программирования, поскольку на нем можно создавать приложения как на стороне клиента, так и на стороне сервера. На JavaScript существует множество фреймворков и библиотек, которые упрощают создание веб-приложений и расширяют их функциональность. Svelte и React – оба фронтенд-фреймворка, которые пользуются популярностью среди разработчиков. Какой из них выбрать разработчику? В этой статье мы познакомимся с дебатами между Svelte и React, обсудим их различия и покажем, где можно использовать каждый из них.

## Что такое Svelte?



Svelte – это онлайн-компилятор JavaScript с открытым исходным кодом. В отличие от большинства JavaScript-фреймворков, выполняющих основную часть работы в браузерах, Svelte использует свой компилятор для оптимизации кода во время сборки. Svelte был создан Ричем Харрисом в 2016 году и с тех

пор собрал большую базу пользователей. Согласно опросу Stack Overflow 2022, Svelte входит в число самых любимых JavaScript-фреймворков. Для создания Svelte необходимо сначала установить Node.js на локальной машине.

После установки NodeJS можно приступить к работе с Svelte, используя следующие команды:

```
npx degit sveltejs/template moz-todo-svelte
```

```
cd moz-todo-svelte
```

```
npm install
```

```
npm run dev
```

Вот что будет отображаться в вашем браузере:



## Особенности Svelte

- **Реактивный.** Svelte обновляет DOM во время сборки. При использовании этого фреймворка разработчикам не нужно использовать внешние библиотеки управления состоянием.
- **Используется обычный JavaScript.** Разработчики, владеющие HTML, CSS и обычным JavaScript, легко освоят Svelte.
- **Легкий вес и использование меньшего количества кода.** В Svelte можно создать простой мир hello, используя всего несколько строк кода:

```
<script>
```

```
let name = "World"

</script>

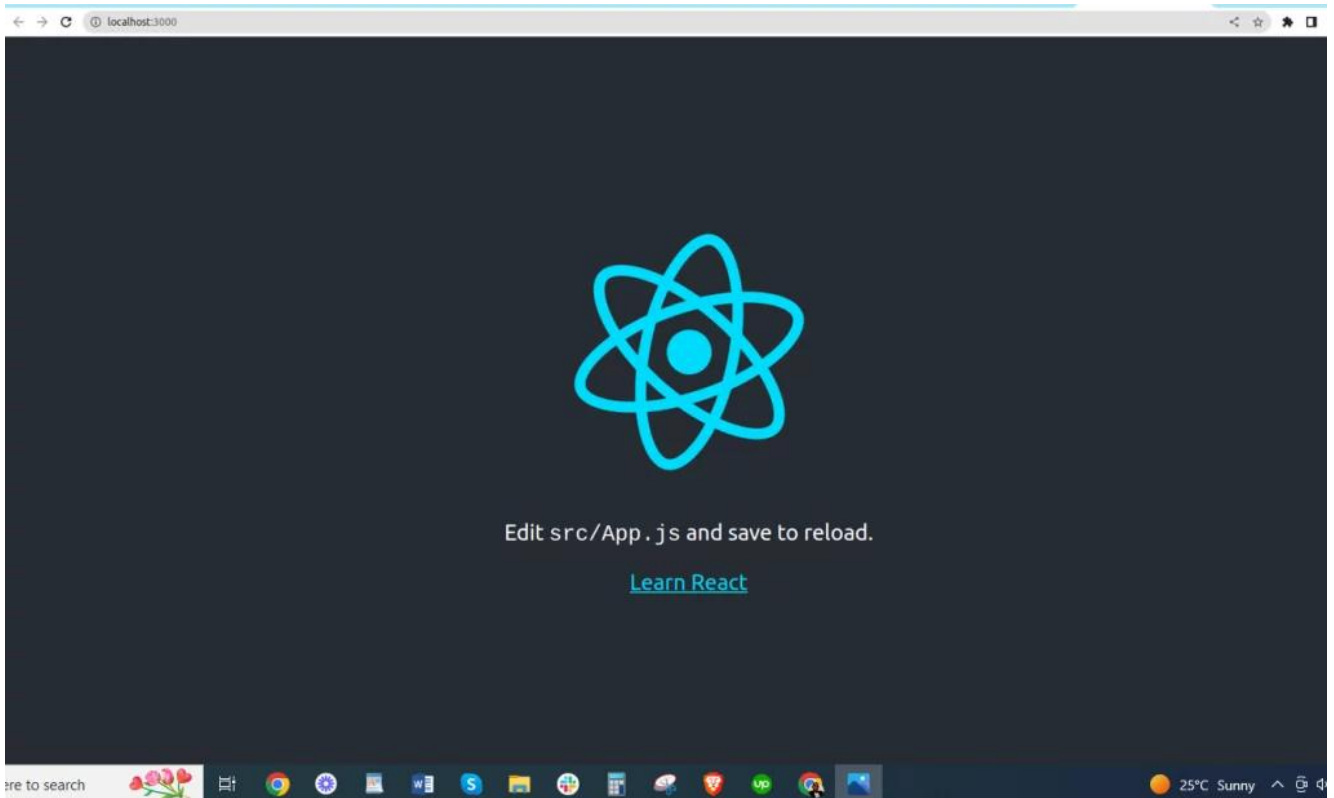
<h1>Hello {name}!</h1>
```

## Преимущества использования Svelte

- **Небольшой размер:** По сравнению с другими фреймворками, приложение Svelte имеет небольшой размер и быстро загружается.
- **Простой синтаксис:** Svelte прост в освоении благодаря простому синтаксису.
- **Не использует виртуальный DOM:** В Svelte, в отличие от большинства фреймворков, отсутствует виртуальный DOM, что приводит к быстрому рендерингу.
- **Высокая производительность:** Этот онлайн-компилятор компилирует код во время сборки, что позволяет создавать быстрые и небольшие приложения.

## Что такое React?

React – это популярная JavaScript-библиотека для создания пользовательских интерфейсов. Эта библиотека с открытым исходным кодом поддерживается компанией Meta (ранее Facebook) и используется для создания пользовательских интерфейсов таких популярных веб-приложений, как Airbnb, Facebook и Instagram.



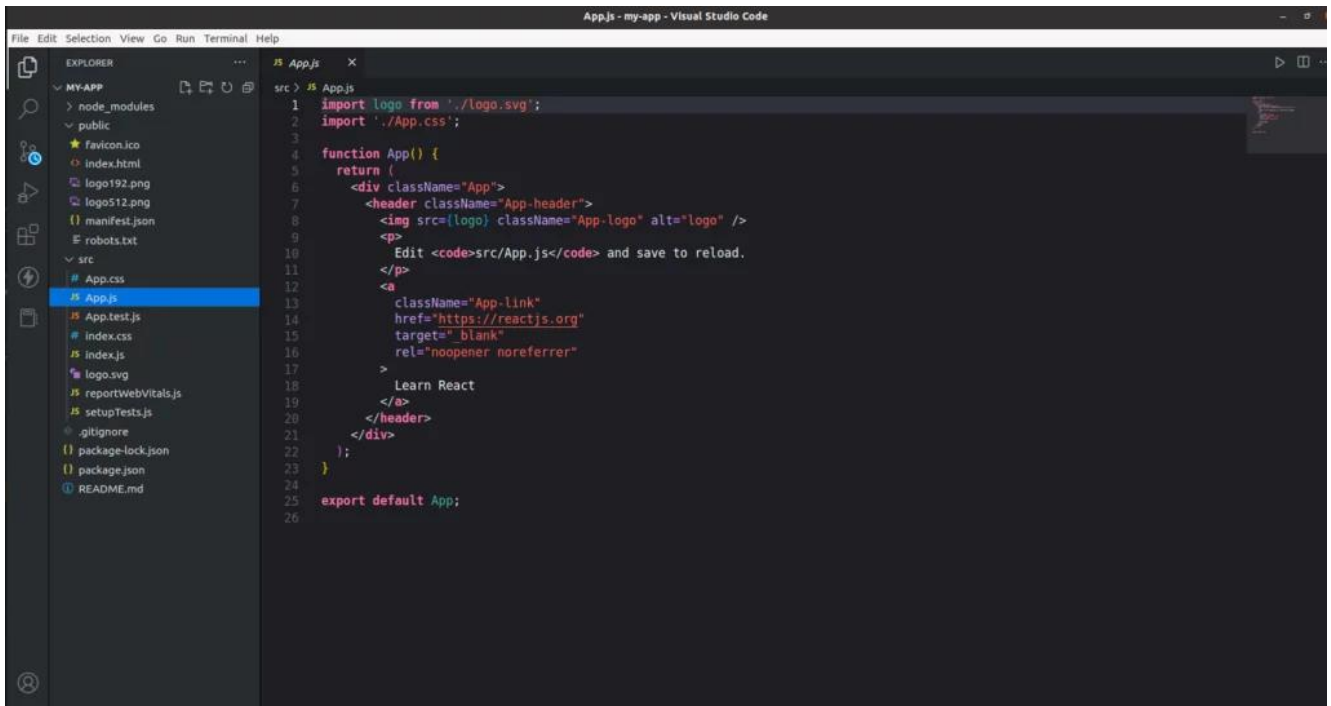
Первой компанией, использовавшей React, была Facebook в своей ленте новостей. После того как в 2013 году библиотека получила открытый исходный код, ее стали использовать все больше компаний, и она стала одной из самых используемых библиотек JavaScript в современном пространстве разработки. React также работает на NodeJS. После установки Node на компьютер выполните следующие команды для настройки приложения React:

```
npx create-react-app my-app
```

```
cd my-app
```

```
npm start
```

Вот файловая структура приложения React:

The image shows a screenshot of the Visual Studio Code editor. The Explorer sidebar on the left shows a project structure with a 'src' folder containing 'App.js'. The main editor window displays the content of 'App.js', which is a React component. The code includes imports for 'logo.svg' and 'App.css', a function 'App()' that returns a JSX element, and an export statement. The JSX element consists of a 'div' with 'className="App"', containing a 'header' with 'className="App-header"', which includes an 'img' tag for the logo and a 'p' tag with a code editor instruction. Below the header is an 'a' tag with 'className="App-link"', 'href="https://reactjs.org"', 'target="blank"', and 'rel="noopener noreferrer"', followed by the text 'Learn React'.

```
1 import logo from './logo.svg';
2 import './App.css';
3
4 function App() {
5   return (
6     <div className="App">
7       <header className="App-header">
8         <img src={logo} className="App-logo" alt="logo" />
9         <p>
10          Edit <code>src/App.js</code> and save to reload.
11        </p>
12        <a
13          className="App-link"
14          href="https://reactjs.org"
15          target="blank"
16          rel="noopener noreferrer"
17        >
18          Learn React
19        </a>
20      </header>
21    </div>
22  );
23 }
24
25 export default App;
```

## Особенности React

- **Расширение синтаксиса JavaScript (JSX):** JSX позволяет разработчикам писать HTML-код в том же файле, который содержит JavaScript-код. JSX также отличается от HTML тем, как он называет свои разделы (DIV), поскольку вместо class используется 'className' (camelCase).
- **Виртуальная объектная модель документа (VDOM):** В React реализовано облегченное представление реальной DOM с помощью виртуальной DOM. Когда состояние объекта изменяется, VDOM обновляет только этот объект в реальном DOM, вместо того чтобы обновлять весь проект.
- **Модульная архитектура:** React позволяет разработчикам писать небольшие компоненты многократного использования. Обновлять и поддерживать такие компоненты также легко.

## Преимущества использования React

- **Компонентная архитектура:** React позволяет разработчикам разбивать свой код на небольшие многократно используемые компоненты.

- **Поддержка различных библиотек:** Функциональность приложения React можно расширить с помощью различных библиотек и фреймворков для поддержки таких функций, как аутентификация.
- **Декларативность:** Создавать интерактивные пользовательские интерфейсы с помощью React очень просто. Декларативные представления в приложении React упрощают чтение и отладку кода.
- **Гибкость:** С помощью React можно создавать различные веб-приложения – от социальных сетей и развлекательных платформ до образовательных сайтов.

## Svelte против React: Сходства

Несмотря на то что Svelte и React отличаются друг от друга, у них есть и общие черты:

- **Компонентная основа.** И Svelte, и React используют компонентную архитектуру. Такой подход позволяет разработчикам разбивать кодовую базу на небольшие фрагменты.
- **Реактивность.** Оба фреймворка исключают ручное вмешательство, поскольку их приложения обновляются автоматически при изменении данных.

## Svelte vs. React: Отличия

### Размер

Размер .gzip-версии **Svelte** составляет всего 1,6 килобайта. Таким образом, вы можете быстро загрузить это приложение и быть уверенными в его высокой производительности.

Размер .gzip-версии **React** составляет 42,2 килобайта. Приложение немного больше, так как в нем используется

ReactDOM.

## Производительность

**React** использует виртуальный DOM – временное хранилище в памяти для изменений, вносимых в пользовательский интерфейс. Таким образом, React работает быстрее, чем традиционная объектная модель документа (DOM), используемая в обычном JavaScript, поскольку виртуальная DOM задерживает обновления до тех пор, пока рендеринг и обновление не будут выполнены эффективно.

**Svelte** не использует виртуальную DOM. Этот фреймворк интерпретирует свой код во время сборки. Svelte – это serverless-first фреймворк, DOM которого обновляется всякий раз, когда назначение/действие вызывает изменение на стадии компонента. Отсутствие виртуального DOM в Svelte делает приложение на Svelte быстрее, чем на React.

## Обслуживание

**Svelte** менее десяти лет, поскольку он был представлен в 2016 году. Однако этот онлайн-компилятор имеет потрясающую команду разработчиков и техподдержки.

**React** поддерживается компанией Meta, различными компаниями и отдельными разработчиками. У этой библиотеки есть преданная команда, что объясняет, почему в ней постоянно появляются новые функции. React выигрывает, когда речь идет о поддержке.

## Тестирование

В качестве фреймворка для тестирования **Svelte** использует `@testing-library/svelte`. Эта библиотека предназначена для тестирования приложений с использованием подхода, который в точности повторяет взаимодействие пользователей с приложением.

В **React** используется библиотека React Testing Library, которая тестирует компоненты с точки зрения пользователя. Также можно

использовать библиотеку Enzyme, если требуется детальный контроль над процессом тестирования.

Svelte и React имеют специальные библиотеки тестирования, помогающие разработчикам создавать функциональные приложения. Также можно использовать внешние фреймворки тестирования, такие как Mocha, для тестирования приложений как на Svelte, так и на React.

## Поддержка сообщества

**Svelte** – достаточно новый фреймворк, и его сообщество не так велико по сравнению с React. Количество разработчиков и компаний, использующих Svelte, также невелико.

**React** имеет большую поддержку сообщества, которое создает учебники, руководства, обновления и компоненты. Он входит в число наиболее используемых библиотек JavaScript и поддерживается технологическим гигантом Meta. Получить помощь от сообщества React очень просто, поскольку оно очень большое и поддерживаемое. React выигрывает за счет поддержки сообщества. Разработчики React также более востребованы, чем разработчики Svelte.

## Библиотеки

**Svelte** имеет сообщество разработчиков с открытым исходным кодом, которое предлагает дополнительные функции к React. Например, вы можете прокладывать маршруты в приложении Svelte с помощью SvelteNavigator. Svelte также имеет мощные библиотеки пользовательского интерфейса, такие как Sveltstrap и Svelte Material UI.

**React** поддерживается сообществом разработчиков с открытым исходным кодом, которые создают инструменты и библиотеки для расширения его функциональности. Например, в этой библиотеке есть библиотеки пользовательского интерфейса Material UI и React Bootstrap, а также используется React Router для



маршрутизации. Для рендеринга на стороне сервера React использует Next.js и Gatsby. Несмотря на то, что сообщество Svelte упорно работает над добавлением новых инструментов, фреймворков и библиотек, React по-прежнему далеко впереди.

## Синтаксис и простота использования

**Svelte** имеет простой синтаксис, использующий чистые HTML, JavaScript и CSS. Таким образом, любой человек, обладающий базовыми знаниями HTML, CSS и JavaScript, легко освоит Svelte.

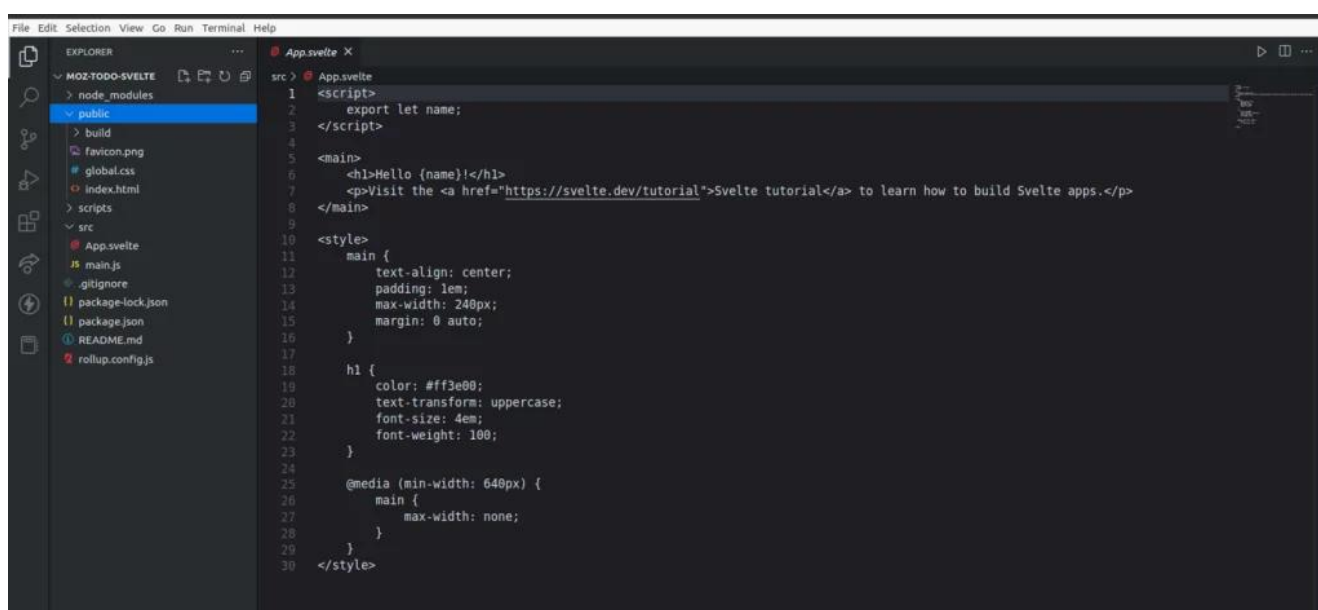
**React** имеет крутую кривую обучения, поскольку пользователям приходится изучать и осваивать новые концепции, такие как JSX и CSS-in-JS. При именовании классов в div React использует className (camelCase), что может сбить с толку человека, знакомого с HTML и CSS. Svelte выигрывает у React по простоте синтаксиса, так как его легче изучать тем, кто разбирается в ванильном CSS, HTML и CSS.

Характеристика	React	Svelte
Размер	42,2 килобайта	1,6 килобайта
Производительность	Использование виртуальной DOM	Не использует виртуальную DOM
Техническое обслуживание	Meta, индивидуальные разработчики и компании	Основная команда разработчиков под руководством Рича Харриса
Тестирование	Использование библиотеки React Testing	Используется @testing-library/svelte
Поддержка сообщества	Огромная	Продолжает расти
Синтаксис	JSX	Чистый HTML, JavaScript и HTML

# Что лучше? Svelte или React?

Svelte и React – замечательные JavaScript-библиотеки, позволяющие создавать широкий спектр приложений. Обе они имеют свои сильные и слабые стороны, и решить, какой из них лучше использовать, бывает непросто. Исходя из анализа возможностей и производительности, вы можете использовать эти фреймворки следующим образом:

## Когда использовать Svelte?

A screenshot of a code editor showing a Svelte application. The editor has a dark theme. On the left, there's an Explorer sidebar showing a file tree with folders like 'public', 'build', 'scripts', and 'src'. The main editor area shows the code for 'App.svelte'. The code includes a script tag for exporting a name, a main component with a heading and a paragraph, and a style tag with CSS for the main container and the heading. The paragraph contains a link to a Svelte tutorial.

```
1 <script>
2   export let name;
3 </script>
4
5 <main>
6   <h1>Hello {name}!</h1>
7   <p>Visit the <a href="https://svelte.dev/tutorial">Svelte tutorial</a> to learn how to build Svelte apps.</p>
8 </main>
9
10 <style>
11   main {
12     text-align: center;
13     padding: 1em;
14     max-width: 240px;
15     margin: 0 auto;
16   }
17
18   h1 {
19     color: #ff3e00;
20     text-transform: uppercase;
21     font-size: 4em;
22     font-weight: 100;
23   }
24
25   @media (min-width: 640px) {
26     main {
27       max-width: none;
28     }
29   }
30 </style>
```

- **Создание небольших проектов:** Svelte подходит, если вы хотите создать простой сайт, например, сайт-портфолио с небольшим количеством функций.
- **Вы цените производительность и оптимизированный код:** Svelte не использует Virtual DOM, что делает его более быстрым по сравнению с приложениями React.
- **Вы хотите создавать динамические пользовательские интерфейсы:** Этот компилятор компилирует код в высоко оптимизированный JavaScript, что делает его идеальным для пользовательских интерфейсов, которые часто меняются.

## Когда использовать React?

- **При создании сложных пользовательских интерфейсов:** Многократно используемые компоненты React делают его идеальным вариантом для создания приложений со сложными интерфейсами.
- **При создании больших приложений:** React обладает множеством функций, позволяющих легко создавать большие приложения.
- **Если вы ищете библиотеку с большим количеством библиотек, инструментов и отличной поддержкой:** React поддерживается большим сообществом, поэтому вы, скорее всего, быстро получите помощь.

## Заключение

Мы надеемся, что теперь вы сможете принять участие в споре Svelte vs. React и определить их сходства, различия и наилучший вариант использования. Для принятия обоснованного решения всегда необходимо определить характер приложения, которое вы хотите создать. Svelte подходит для тех, кто ценит скорость при создании небольших приложений и построении динамических интерфейсов. С другой стороны, библиотека React должна стать вашим выбором, если вы хотите создавать сложные пользовательские интерфейсы, нуждаетесь в библиотеке с большой поддержкой и при создании больших приложений.