

*М.А. Толстых,
магистрант,
УО БГУИР,
г. Минск, Беларусь*

ОЦЕНКА ПЕРСПЕКТИВНОСТИ ФРЕЙМВОРКОВ ДЛЯ СОЗДАНИЯ СОВРЕМЕННЫХ WEB-ПРИЛОЖЕНИЙ

Аннотация: данная статья посвящена анализу современных web-технологий и оценки их перспективности в качестве платформ для построения современных высококачественных web-сервисов.

Ключевые слова: фреймворк, web-сервисы, web-технологии, javascript, reactjs, angular, vue, svelte.

*М.А. Talstykh,
master student
BSUIR,
Minsk, Belarus*

EVALUATION OF THE PERSPECTIVE OF FRAMEWORKS FOR CREATION MODERN WEB APPLICATIONS

Abstract: this article is devoted to the analysis of modern web-technologies and assessment of their prospect as a platform for building modern high-quality web-services.

Ключевые слова: framework, web-services, web-technologies, javascript, reactjs, angular, vue, svelte.

Современные web-платформы являются эффективным средством достижения целей, поставленных бизнесом. При этом, как и с любым инструментом, важно правильно выбрать технологию, позволяющую решать задачи оптимальным способом.

Среди современных javascript-фреймворком можно выделить следующие наиболее распространенные в коммерческой разработке:

1. React.
2. Angular 2+.
3. VueJS.

В основе ReactJS лежит использование HTML-подобного синтаксиса, называемого JSX [1]. При желании его возможности могут быть расширены с помощью языка TypeScript, при этом получая новый TSX синтаксис [2]. Одним из главных преимуществ данного фреймворка является возможность использования одной кодовой базы для мобильных платформ и приложений с серверным рендерингом страниц. Это возможно благодаря поддержке React

Native и Next.JS.

Из недостатков можно выделить JSX и TSX синтаксисы, из-за которых логика и разметка страницы смешиваются. Из архитектурных недостатком можно выделить возможность разработчиков самим выбирать способ развития проекта и компоненты на основе функций, что вынуждает нарушать принципы объектно-ориентированного программирования.

Angular 2+ разработан компанией Google для создания масштабных web-приложений [3]. В качестве языка по умолчанию используется Typescript. В отличие от React, имеет строгую структуру, что позволяет не отклоняться от лучших архитектурных практик. Из коробки имеет большое количество хорошо интегрированных инструментов для эффективной разработки сложных приложений.

Главными недостатками являются все сопутствующие инструменты так как из-за этого сильно увеличивается размер итогового проекта и уменьшается скорость работы приложения [4].

VueJS не имеет за собой поддержки крупных it-гигантов. Разработка велась энтузиастами из Китая [5]. Первый релиз фреймворка состоялся в 2014 году. На его развитии положительно сказывается тот факт, что дизайн и архитектура схожа и с ReactJS, и с Angular. При этом главным преимуществом является вес фреймворка и высокая скорость работы [6].

Недостатком является сравнительно небольшое сообщество разработчиков, что иногда может приводить к проблемам при интеграции в крупные проекты из-за отсутствия наработанных временем практик.

Одним из лучшим показателем перспективности фреймворка является то как к нему относятся разработчики. Хорошая репутация будет напрямую влиять на то, как часто фреймворк выбирается в качестве основного инструмента разработки, а значит и на его популярность в данный момент. На рисунке 1 представлен рейтинг фреймворков по лояльности разработчиков за последние три года [7].

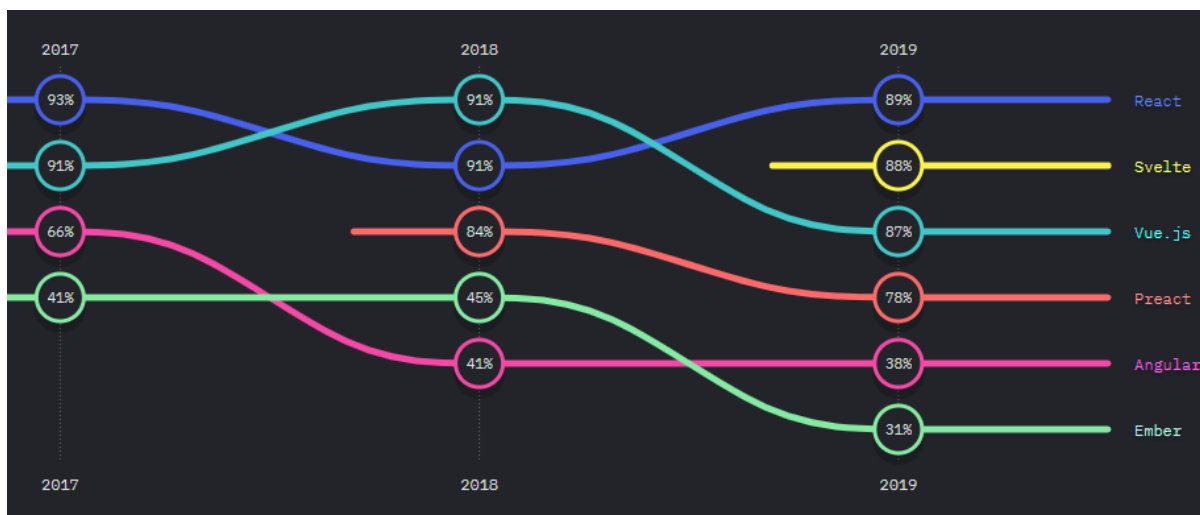


Рисунок 1 – Рейтинг фреймворков по лояльности разработчиков

Из представленных на рисунке данных можно сделать вывод, что наиболее востребованным фреймворком среди разработчиков является ReactJS. VueJS не сильно отстает, но в прошлом году его подвинул новый фреймворк Svelte. При этом его пока не получается рассматривать для коммерческой разработки из-за отсутствия сформированной вокруг инфраструктуры и сообщества разработчиков с большим опытом решения типовых задач. По этой же причине не рассматривается и фреймворк Preact. Angular же имеет традиционно невысокую позицию из-за плохой репутации ранних версий и сложности в освоении фреймворка.

Для перспективы коммерческой разработки важны такие показатели как производительность приложения, стоимость поддержки кода, легкость освоения разработчиками и легкость поиска инженеров со знанием выбранной технологии. Проведем анализ фреймворков на соответствие данным критериям.

React в силу своей скорости, простоты и большого сообщества разработчиков на данный момент является лучшим выбором для малых и средних проектов. Среди разработчиков фреймворк пользуется популярностью и высоко ценится, что значит простой поиск кадров. В то же время отсутствие строгой архитектуры может вызвать некоторые сложности при масштабировании проекта.

Строгая архитектура Angular и хорошая масштабируемость делает его хорошим выбором для крупных проектов со сложной логикой. Несмотря на недовольства разработчиков старыми версиями, последние обновления улучшают ситуацию и разница в скорости работы приложения на Angular и с его основными конкурентами VueJS и ReactJS снижается. Его популярность среди разработчиков падает, если сравнивать с новыми популярными фреймворками вроде Svelte и Preact, но Angular всё ещё остается лучшим выбором для сложных и масштабных приложений и имеет большую долю в сегменте корпоративной разработки.

VueJS всё ещё не так популярен среди крупных компаний по причине нежелания менять и так хорошо справляющиеся Angular и React. В новых проектах у него получается конкурировать с React по причине схожего фокуса на средние проекты и высокую производительность. За пару лет накопилась минимальная необходимая кодовая база и набор типовых решений, поэтому можно говорить о закреплении данного фреймворка в качестве альтернативы легковесным и быстрому React для стартапов и новых проектов.

Большая тройка фреймворков в виде ReactJS, Angular и VueJS является хорошим выбором для проектов разного калибра и под разные требования. Каждый из них занимает свою нишу в корпоративной разработке и решает поставленные перед ним задачи. На ближайшие пару лет все фреймворки можно назвать перспективными, особенно учитывая уже существующее количество написанного кода, который необходимо будет поддерживать ещё долгое время.

Список использованных источников и литературы:

[1] React – a javascript library for building user interfaces [Электронный

ресурс]. – Режим доступа: <https://reactjs.org/>

[2] JSX TypeScript [Электронный ресурс]. – Режим доступа: <https://www.typescriptlang.org/docs/handbook/jsx.html>

[3] About Angular [Электронный ресурс]. – Режим доступа: <https://www.angular.io>

[4] Angular 2 versus React [Электронный ресурс]. Режим доступа: <https://medium.com/@housecor/angular-2-versus-react-there-will-be-blood-66595faafd51>

[5] Evan You [Электронный ресурс]. – Режим доступа: <https://www.linkedin.com/in/evanyou/>

[6] Vue.JS [Электронный ресурс]. – Режим доступа: <https://vuejs.org/>

[7] The State of JavaScript 2019 [Электронный ресурс]. – Режим доступа: <https://2019.stateofjs.com/>

© М.А. Толстых, 2020