

ОБЗОР АКТУАЛЬНЫХ ФРЕЙМВОРКОВ ДЛЯ РАЗРАБОТКИ КЛИЕНТСКОЙ ЧАСТИ СОВРЕМЕННЫХ ВЕБ-ПРИЛОЖЕНИЙ

*В.Г.Ротарь,
М.Ткачев, А.Е.Трофимова,
Томский политехнический университет
E-mail: mihail1@tpu.ru*

Введение

В следствии стремительного роста пользователей, интернет становится главной платформой для бизнеса и сфер услуг. Поэтому множеству компаний требуется создание своего представительства в интернете. Широко используемые инструменты программирования, такие как фреймворки, позволяют оптимизировать создание информационного ресурса, обладающего современным функционалом, высокой скоростью загрузки, адаптированным к мобильным устройствам и т.п [1]. Поэтому важным аспектом в веб-разработке является правильный выбор технологии создания клиентской части веб-приложения.

По результатам исследования ресурса Stack Overflow 2020 и рейтингу крупнейшей платформы для хостинга проектов и их совместной разработки GitHub, JavaScript является наиболее часто используемым языком уже 8-й год подряд, и его выбрали 67,7% разработчиков. На сегодняшний день среди JavaScript-фреймворков самыми востребованными являются Angular2, React и Vue. Эти инструменты решают задачу синхронизации пользовательского интерфейса и состояния системы, а также позволяют ускорить рабочий процесс, повысить производительность системы без ущерба для качества или функциональности, сохраняя при этом гибкость в разработке. [2].

Особенности Angular2

В настоящее время общая производительность пользовательских устройств увеличивалась, поэтому стало возможным выполнение логики приложения в браузере. Это привело к подходу создания одностраничных приложений или SPA, для разработки которых назначался AngularJS. Он также предоставлял возможность заменить низко производительный jQuery, предлагая разработчикам такие функции, как двусторонняя привязка данных и возможность организации модулей для импорта внешних скриптов. Затем AngularJS был полностью переписан под Angular2, соответственно изменилось множество основных концепций фреймворка. В Angular2 отсутствуют контроллеры или модели представления, вместо этого есть компоненты, которые состоят из шаблона, классов и метаданных. В то время как AngularJS был сосредоточен на шаблоне MVC, Angular2 полностью полагается на иерархию компонентов [3].

Важным нововведением фреймворка является TypeScript, который добавляет в проект строгую типизацию, а также различный новый функционал, такой как декораторы, еnumы и т.д. Впоследствии TypeScript компилируется в JavaScript для работы в браузере [3].

Angular2, также как React и Vue, основан на компонентах, которые отображают информацию и шаблоны, выполняют действия с данными и состоят из трех отдельных файлов: HTML-файл для шаблона, CSS-файл для стилизации и TS-файл для контроля. Следуя этому подходу, реализуется принцип разделения проблем, а также код становится более организованным и структурированным. Внутри компонентов обмен данными между представлением (шаблоном HTML) и моделью (файлом TypeScript) происходит за счет привязки данных и событий. В Angular2 реализована собственная система маршрутизации для удобства использования SPA. Определив маршруты в отдельном файле или модуле, Angular2 обрабатывает логику того, какой компонент должен отображаться в зависимости от текущего активного URL-пути [4].

Ключевые возможности React

React – это внешняя библиотека/фреймворк JavaScript для создания пользовательских интерфейсов веб-приложений. Самыми популярными приложениями, использующие React, являются Instagram и WhatsApp. React часто используется в веб-разработке, однако его ядро является отдельной библиотекой и используется в различных применениях, включая нативные приложения iOS и Android [4].

Поскольку React отвечает только за часть представления приложения, процесс разработки требует таких технологий, как компилятор Babel, модули и соответствующий загрузчик для структуры приложения, процесс сборки, маршрутизация и управление состоянием. С помощью компилятора могут быть использованы новые языковые функции еще не реализованных спецификаций ECMAScript.

Что касается модулей и сборки, то Webpack - одно из лучших решений. Для управления состоянием используется Redux либо Mobx. Маршрутизация может выполняться с помощью React Router, основанном на тех же принципах, что и Ember-Router. Основной целью фреймворка является преобразование определенного состояния приложения в представление, которое может отображаться в браузере [5].

Одной из особенностей React, является возможность использовать расширение JSX, чтобы дополнить код JavaScript фрагментами HTML. Также одной из основных функций фреймворка является построение собственной виртуальной модели DOM-дерева. С помощью элементов ReactElements React повторно визуализирует полный пользовательский интерфейс компонента и генерирует операции, которые выполняются в реальном DOM. Так как операции запроса к реальному DOM-дереву HTML-документа очень медленные и ресурсоемкие, React позволяет минимизировать нагрузки на нативную DOM, что является основой его быстродействия [6].

Обзор функционала Vue

Vue описывают, как прогрессивный фреймворк, принципы проектирования которого частично взяты из паттерна Model-View-ViewModel. Согласно официальному сайту, фреймворк можно использовать для разработки SPA и небольших проектов в сочетании с другими библиотеками, аналогично React. Все компоненты Vue очень похожи на Custom Elements, которые являются частью спецификации веб-компонентов и позволяют создавать новые теги HTML или изменять существующие, дополняя их функциями с использованием JavaScript, HTML и CSS [7].

Масштабируемость – одно из главных достоинств данного фреймворка среди других технологий. Vue надежно работает во всех поддерживаемых браузерах. Кроме того, компоненты фреймворка добавляют функциональность, которую не могут обеспечить нативные элементы, например, передачу настраиваемых событий и межкомпонентный поток данных [6].

Структурирование Vue предполагает, что каждый компонент должен иметь единственный корневой элемент. Функции Vue отображают Virtual DOM. Фреймворк устанавливает минимальный набор компонентов для необходимого количества манипуляций с DOM-деревом.

Фреймворк предоставляет официальный интерфейс командной строки для улучшения разработки, особенностью которого является тот факт, что команда разработчиков предлагает набор предварительно настроенных шаблонов для различных вариантов использования [7].

Заключение

В настоящее время JavaScript-фреймворки предлагают полный стек полезных функций и отвечают за архитектурные решения в процессе разработки, включают в себя стратегии для маршрутизации URL-адресов в приложении, управления состоянием, объединения и другое.

Фреймворки React и Vue, использующие Virtual DOM и компонентную структуру, сосредоточены на создании пользовательского интерфейса. Angular2 используется в крупных коммерческих приложениях с большой функциональной нагрузкой, но в сравнении с React и Vue проявляет меньшую гибкость в разработке, предлагая четко следовать своим правилам. React превосходит Vue в разнообразии доступных средств отрисовки интерфейса пользователя, в то время как Vue в большей мере основывается на классических веб-технологиях.

Список использованных источников

1. Пьюривал С. Основы разработки веб-приложений //СПб.: Питер. – 2015. – 272 с.
2. Елисеева Е. С., Хаханова А. Д., Учанева А. А. Применение Javascript-фреймворков при разработке интерактивных образовательных веб-приложений //Современное образование: традиции и инновации. – 2020. – №. 2. – 240-243 с.
3. Кургасов В. В., Лапшова А. Г. Javascript фреймворки //Центральный научный вестник. – 2018. – Т. 3. – №. 15-16. – 40-41 с.
4. Будаев Е. С., Димова А. В. Обзор и сравнительный анализ библиотек и фреймворков JavaScript //Вестник современных исследований. – 2019. – №. 6.2. – 33-37 с.
5. Акулов А. А. Исследование популярных javascript-фреймворков для разработки клиентской части веб-приложения //Цифровизация экономики: направления, методы, инструменты. – 2019. – 358-362 с.
6. Сукиасян В. М., Придиус Е. С. Современные принципы и подходы к frontend архитектуре веб-приложений //Наука, техника и образование. – 2019. – №. 10 (63). – 54-57 с.
7. Бондаренко Ю. В. Почему стоит использовать Vue. JS //наука в современном обществе: закономерности и тенденции развития. – 2018. – 22 с.