

УДК-004.422.81

ТЕХНОЛОГИЯ РАЗРАБОТКИ ОДНОСТРАНИЧНОГО ВЕБ-ПРИЛОЖЕНИЯ НА ПЛАТФОРМЕ ANGULAR 8

А. В. Сулыз, А. Н. Панфилов

Южно-Российский государственный политехнический университет (НПИ) имени М. И. Платова
(г. Новочеркасск, Российская Федерация)

Приводится пример разработки одностраничного веб-приложения на платформе Angular 8. Рассмотрены различные аспекты в разработке одностраничных веб-приложений, а также преимущества и недостатки их применения. Описаны наиболее популярные JavaScript фреймворки для создания одностраничных веб-приложений. Разработана диаграмма развертывания приложения, разработанного по архитектуре REST с применением фреймворка Spring и СУБД PostgreSQL. На основании проведенной работы сделаны выводы о целесообразности применения рассматриваемой технологии.

Ключевые слова: веб-программирование, UML, диаграмма развертывания.

UDC-004.422.81

TECHNOLOGY OF DEVELOPING A SINGLE-PAGE WEB APPLICATION ON THE ANGULAR 8 PLATFORM

A. V. Sulyz, A. N. Panfilov

Platov South-Russian State Polytechnic University (NPI), Novocherkassk, Russian Federation

This paper gives an example of developing a single-page web application on the Angular 8 platform. Various aspects in the development of single-page web applications, as well as the advantages and disadvantages of their application, are considered. Various JavaScript frameworks are considered, which are the most popular for creating single-page web applications. A deployment diagram has been developed for a REST architecture application using the Spring framework and PostgreSQL DBMS. Based on the work done, conclusions are drawn on the appropriateness of using the technology in question.

Keywords: web development, UML, deployment diagram.

Введение. В настоящее время все большую популярность при веб-разработке набирают одностраничные приложения Single Page Application (SPA). За последние несколько лет SPA получили широкое распространение для создания панелей инструментов приложений, реализованных по форме Software as a service (SaaS), а также для интернет-сервисов в целом. Данные веб-приложения имеют как ряд преимуществ, так и недостатков.

Из-за ряда новых технологий, применяемых в SPA, изменяется и архитектура разрабатываемого веб-приложения. Понимание структуры и протоколов связи компонентов такого сайта может стать трудной задачей для впервые столкнувшегося с ней разработчика. В данной работе рассматривается структура компонентов SPA на примере диаграммы развертывания UML с применением конкретных технологий.

Преимущества и недостатки SPA. Основными преимуществами данных приложений являются [1]:

– высокое быстродействие за счет того, что большинство ресурсов таких как HTML, CSS, JavaScript загружаются только один раз, изменяются только данные;

– упрощен поиск ошибок, поскольку почти каждым веб-браузером предоставляются инструменты для отслеживания сетевых операций, исследования элементов страницы и их данных;

– существующие фреймворки для создания SPA позволяют использовать один и тот же код для создания кроссплатформенных приложений;

– SPA позволяет эффективно кешировать данные в локальном хранилище, что позволяет использовать данные даже в автономном режиме.

Однако наряду с указанными преимуществами есть и недостатки:

– проблемная поисковая оптимизация веб-приложения (SEO) за счет того, что все данные страницы загружаются с помощью AJAX;

– по сравнению с традиционными веб-приложениями SPA менее безопасны;

– утечки памяти движка JavaScript могут привести к замедлению работы системы.

Популярные реализации SPA. Несмотря на вышеприведенные недостатки фреймворки для разработки SPA развиваются и пользуются большой популярностью. Среди наиболее популярных веб-фреймворков можно выделить следующие [2]:

– Angular является старейшим фреймворком для создания SPA. Разработан компанией Google в 2010 году и до 2016 носил название Angular.js. В 2016 был выпущен Angular.js 2, который по сути являлся совершенно новым фреймворком и, чтобы избежать путаницы, его переименовали в Angular;

– React является фреймворком разработанным компанией Facebook в 2013 году. Он пользуется популярностью и поддержкой сообщества и имеет множество сторонних библиотек и расширений;

– Vue.js является самым молодым фреймворком, разработанным бывшим сотрудником Google в 2014 году и, в настоящий момент, поддерживается лишь группой энтузиастов. Тем не менее, по качеству реализации он держится на уровне своих конкурентов и догоняет их по популярности.

Пример реализации SPA. SPA обычно взаимодействуют с приложениями на стороне веб-сервера по архитектуре Representational State Transfer (REST). REST — это архитектурный стиль взаимодействия компонентов приложения в сети. На сегодняшний день он практически вытеснил все остальные стили, в том числе Simple Object Access Protocol (SOAP) и Web Services Description Language (WSDL). Рассмотрим пример использования технологии SPA в связке с Spring Boot и СУБД PostgreSQL.

Spring — это фреймворк для создания веб-приложений на Java. Он обеспечивает комплексную модель разработки и конфигурации для современных бизнес-приложений для любых платформ. Его основными возможностями являются внедрение зависимостей, аспектно-ориентированное программирование, создание RESTful веб-сервисов и многое другое [3].

Для реализации технологии SPA может быть выбран фреймворк Angular 8. Основные возможности Angular: двустороннее связывание, позволяющее динамически изменять данные в одном месте интерфейса при изменении данных модели в другом, шаблоны, маршрутизация и так далее. Одной из ключевых особенностей Angular является то, что в качестве языка программирования используется TypeScript. Несмотря на то, что другие фреймворки пользуются большей популярностью, разработчики компаний все равно продолжают использовать его в своих проектах [4].

Из диаграммы развертывания, приведенной на рис. 1 видно, что для развертывания SPA в браузере пользователю достаточно иметь поддержку JavaScript, которая присутствует во всех веб-

браузерах, однако, он может быть отключен пользователем. В этом случае SPA не будет работать. В другом случае, при использовании SPA в качестве мобильного приложения, будет создано полноценное прикладное приложение, основной компонентой которого будет среда NativeScript или, в случае использования React, — ReactNative. При этом стоит отметить, что Angular при запуске его в контейнере NodeJs компилирует код из TypeScript в JavaScript. Такой подход позволяет программисту писать на TypeScript код, который не имеет привязки к определенной версии JavaScript, что позволяет единожды написать программу и компилировать под нужную версию языка. Данная задача обусловлена тем, что различные веб-браузеры могут не поддерживать новые стандарты JavaScript.

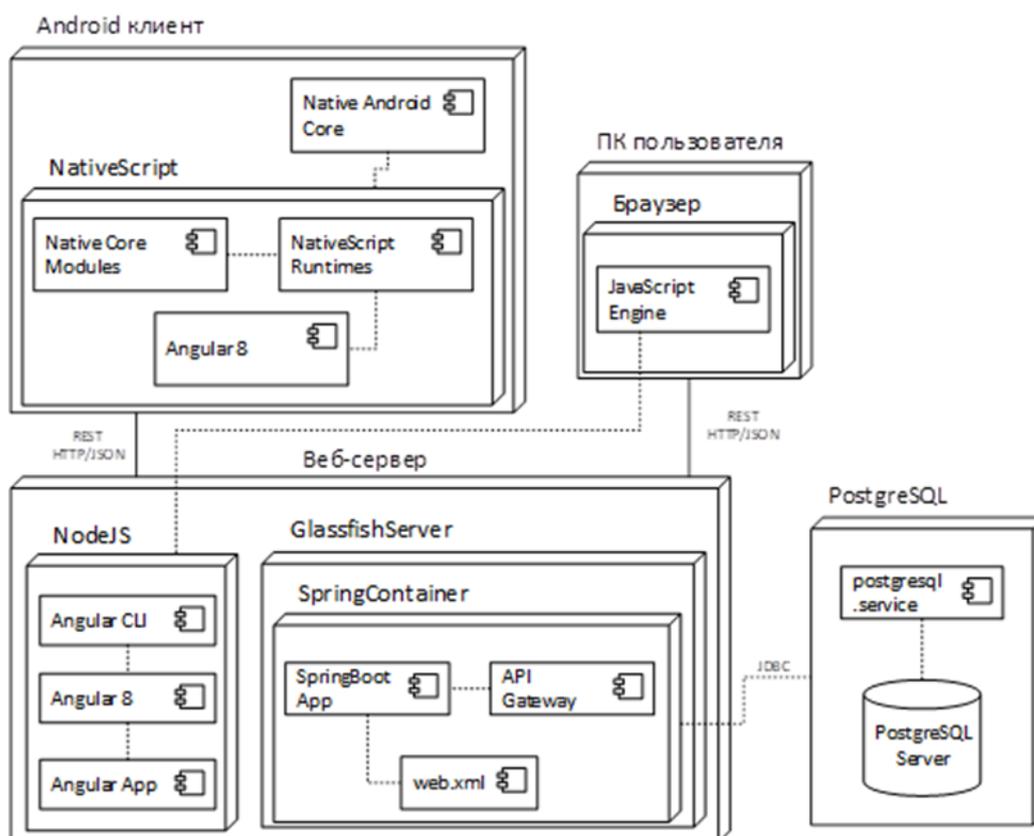


Рис. 1. Диаграмма развертывания приложения на Angular 8

Заключение. Таким образом можно сделать следующие заключения:

- SPA отличаются быстродействием и кроссплатформенностью, но, в тоже время, имеют низкий уровень безопасности;
- Angular 8 является наиболее популярным фреймворком на текущий момент для реализации SPA, однако, в скором времени его могут заменить конкуренты, пользующиеся большей поддержкой сообщества.

Результаты данной работы могут пригодится для понимания структуры современного SPA, основных рисках и преимуществах, связанных с их применением, а также получения представления о современной ситуации в веб-разработке.

Библиографический список

1. Single-page application vs. multiple-page application [Электронный ресурс] / Medium. — Режим доступа: <https://medium.com/@NeotericEU/single-page-application-vs-multiple-page-application-2591588efe58> (дата обращения : 10.01.2020).

2. React vs Angular vs Vue.js — What to choose in 2019? (updated) [Электронный ресурс] / Medium. — Режим доступа : <https://medium.com/@TechMagic/reactjs-vs-angular5-vs-vue-js-what-to-choose-in-2018-b91e028fa91d> (дата обращения : 10.01.2020).

3. Building a RESTful Web Service [Электронный ресурс] / Spring. — Режим доступа : <https://spring.io/guides/gs/rest-service/> (дата обращения : 10.01.2020).

4. Architecture overview [Электронный ресурс] / Angular. — Режим доступа : <https://angular.io/guide/architecture> (дата обращения : 10.01.2020).

Об авторах:

Сулыз Андрей Викторович, студент Южно-Российского государственного политехнического университета (НПИ) имени М.И. Платова (РФ, г. Новочеркасск, ул. Просвещения, 132), sulyz2012@mail.ru

Панфилов Александр Николаевич, доцент кафедры «Информационные и измерительные системы и технологии» Южно-Российского государственного политехнического университета (НПИ) имени М.И. Платова (РФ, г. Новочеркасск, ул. Просвещения, 132), кандидат технических наук, доцент, panfiloff@rambler.ru

Authors:

Sulyz Andrey Viktorovich, Student, Platov South-Russian State Polytechnic University (NPI), (Russian Federation, Novocherkassk, Prosveshcheniya street 132), sulyz2012@mail.ru

Panfilov Alexander Nikolaevich, Associate professor of the Department of Information and Measurement Systems and Technologies, Platov South-Russian State Polytechnic University (NPI), (Russian Federation, Novocherkassk, Prosveshcheniya street 132), Candidate of technical sciences, Associate professor, panfiloff@rambler.ru