

УДК 004.43

*Лобода Р.М., студент,
Курчанов В.М., к.т.н., доцент,
Черницька І.О., асистент
Полтавський національний технічний
університет імені Юрія Кондратюка*

АНАЛІЗ ПРОДУКТИВНОСТІ СУЧАСНИХ JAVASCRIPT-ФРЕЙМВОРКІВ

Анотація. В даній статі було проаналізовано ситуацію фреймворків, написаних мовою JavaScript для розробки інтернет-ресурсів. Приділено увагу проблемі перенасичення ІТ-сфери додатками та фреймворками даного типу. Виділено 5 найактуальніших JavaScript-фреймворків. Проведено детальне тестування та аналіз результатів тестування продуктивності. На основі результатів зроблено висновок, що Aurelia найпродуктивніший фреймворк 2017 року.

Ключові слова: *фреймворк, інтернет-ресурс, JavaScript, Aurelia, NPM.*

Вступ

Розповсюдження мережі Інтернет в усіх сферах соціальної, наукової, економічної, виробничої та інших діяльностей призвело до збільшення об'єму розробки web-спрямованих інформаційних ресурсів, що в свою чергу призвело до розвитку та збільшення кількості додатків для розробки інтернет-ресурсів. Якщо на початку розвитку ІТ індустрії використовувались лише HTML, CSS і JavaScript мови без бібліотек і допоміжних додатків, то зараз без бібліотеки або фреймворка не розробляється жоден інтернет-ресурс.

Фреймворк (англ. Framework, каркас, платформа, структура, інфраструктура) — інфраструктура програмних рішень, що полегшує розробку

складних систем. Спрощено дану інфраструктуру можна вважати своєрідною комплексною бібліотекою [2]. На теперішній 2017 рік по різних джерелах [3], [4], [5] нараховується близько 85 фреймворків, кількість яких продовжує збільшуватись. Але при розробці інтернет-ресурсу середньої складності використовується лиш один, а за рідким виключенням – два фреймворка, тому виникає необхідність в дослідженні і аналізі фреймворків, тестуванні їх продуктивності і виділенні найактуальніших і кращих з них.

Враховуючи кількість розроблених на даний час фреймворків, частоту виходу нових і відсутність підтримки деяких зі сторони розробника, дослідження всього переліку втратить свою актуальність вже через 1 – 2 місяці. Тому було обрано 5 фреймворків за наступними показниками:

1. Рік виходу. Фреймворк, котрий не втратив актуальності з плином часу вважається стабільним і перевіреним спільнотою розробників і такий фреймворк буде продовжувати розроблятися і підтримуватися розробниками.
2. Патерн (концепт) проектування. На теперішній час є декілька патернів проектування [6], котрі закладаються в основу реалізації фреймворків. І від того, якого концепту притримувались розробники під час проектування та розробки фреймворку залежить його швидкість і зручність.
3. Розробник. Від команди розробника залежить наявність підтримки і гарантія розвитку фреймворка та його покращення.

За перерахованими показниками було відібрано 5 фреймворків, котрі показані в Таблиці 1.

Таблиця 1. Список досліджуваних фреймворків

назва\критерії	Рік виходу	Патерн проектування	Розробник
Angular 5	2010	MVVM	Google

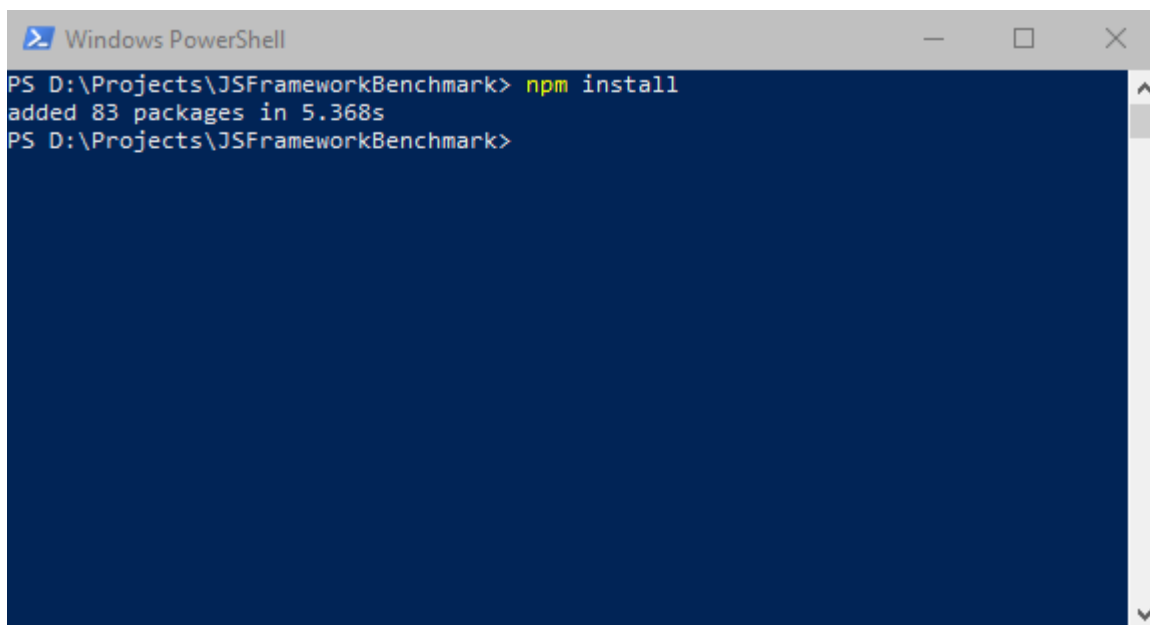
React	2013	View only	Facebook
Aurelia	2016	VM*	Blue Spire
Vue 2	2013	View only	Evan You
Knockout	2010	View only	Steve Sanderson

Налаштування середовища

Для тестування обраних фреймворків була використана бібліотека-утиліта «JS Framework Benchmark» [12] розроблена Стефаном Краусером, код якої дозволяє проводити тестування над більше 40 бібліотек різних версій. Для її використання необхідно попередньо налаштувати середовище розробки і встановити наступні бібліотеки:

1. NodeJS - платформа з відкритим кодом для виконання високопродуктивних мережевих застосунків, написаних мовою JavaScript.
2. NPM (Node Package Manager) - це менеджер пакунків для мови програмування JavaScript.
3. JDK (Java Development Kit) - безкоштовний розповсюджуваний Oracle (раніше Sun) комплект розробника застосунків на мові Java, який включає до себе компілятор Java (javac), стандартні бібліотеки класів Java, приклади, документацію, різноманітні утиліти і виконавчу систему Java (JRE).

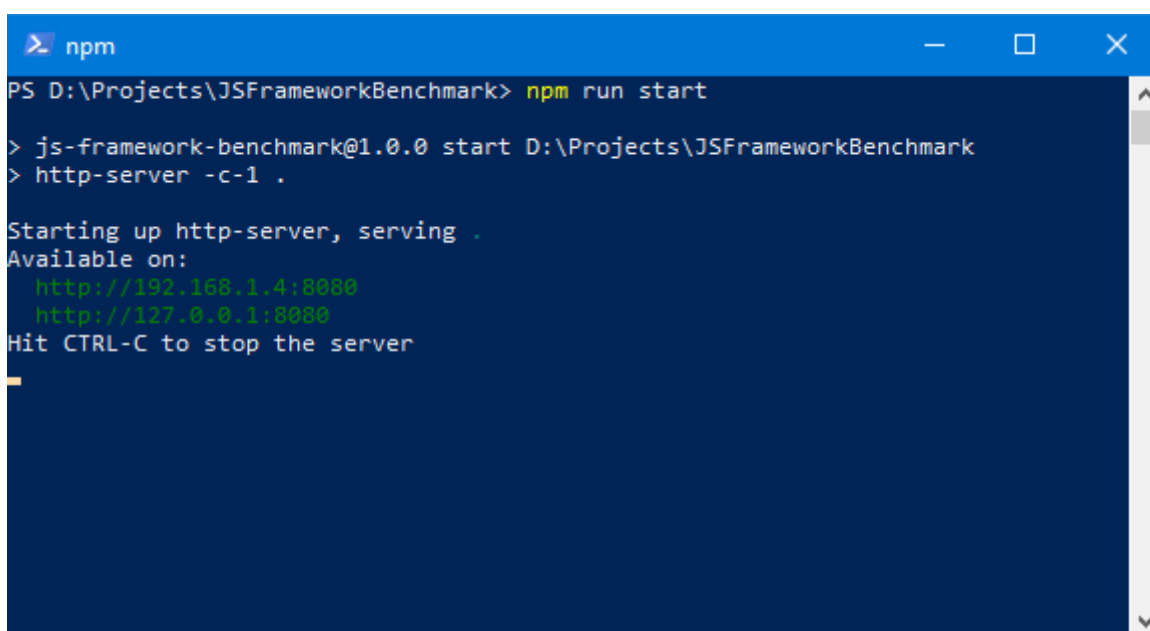
Наступним кроком виконуємо команду `npm install` в кореневій теці бібліотеки-утиліти «JS Framework Benchmark». Результатом має бути наступне повідомлення (Рис. 1).



```
Windows PowerShell
PS D:\Projects\JSFrameworkBenchmark> npm install
added 83 packages in 5.368s
PS D:\Projects\JSFrameworkBenchmark>
```

Рис. 1. Встановлення пакетів в кореневу директорію

Цю операцію також необхідно виконати в директорії кожного досліджуваного фреймворку, вказаного в Таблиці 1, попередньо перейшовши в неї за допомогою команди `cd <шлях до директорії>`. Після встановлення всіх необхідних пакетів повертаємось до кореневої директорії і виконуємо команду запуску серверу `npm run start` (Рис 2).



```
npm
PS D:\Projects\JSFrameworkBenchmark> npm run start
> js-framework-benchmark@1.0.0 start D:\Projects\JSFrameworkBenchmark
> http-server -c-1 .

Starting up http-server, serving .
Available on:
  http://192.168.1.4:8080
  http://127.0.0.1:8080
Hit CTRL-C to stop the server
```

Рис. 2. Запуск серверу бібліотеки-утиліти «JS Framework Benchmark»

Проведення тестування і аналіз результатів

На зображенні (Рис. 2) вказано посилання, по яким нам доступний запуск сервер і перейшовши по яким ми побачимо список всіх фреймворків, щоб розпочати тестування кожного по черзі. Щоб отримувати результати тестування після кожного кроку, переходимо по посиланню з назвою необхідного фреймворку (Рис. 3) і відкриваємо консоль розробника в браузері, натиснувши клавішу F12 для браузера Google Chrome (в кожного браузера своя комбінація відкривання консолі розробника) (Рис. 4).

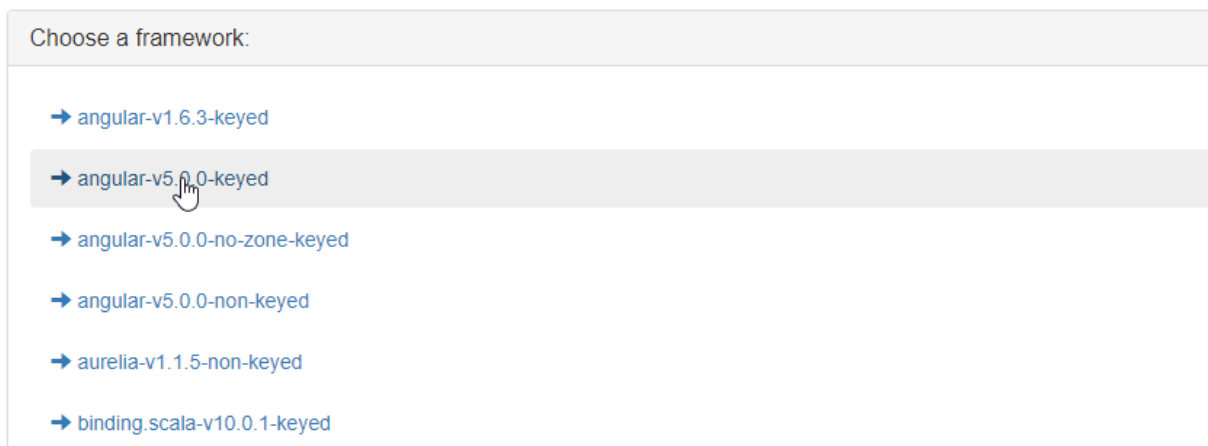


Рис. 3. Перелік доступних фреймворків для тестування

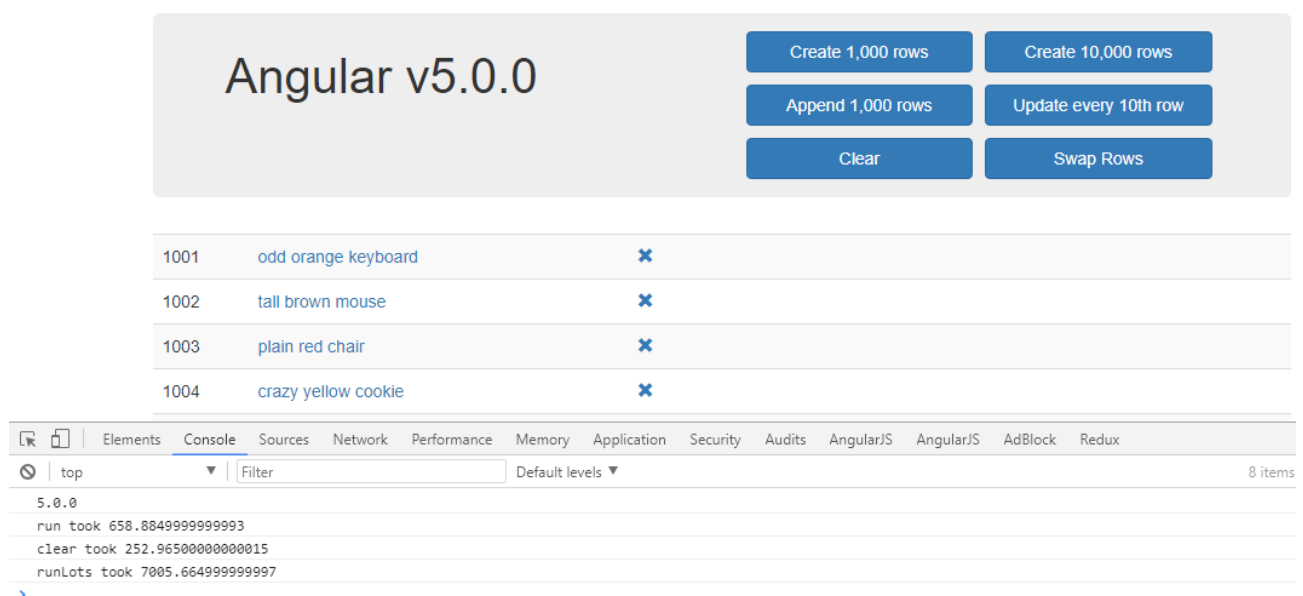


Рис. 4. Проведення тестування та збір результатів

Використовуючи кнопки створення, оновлення, змішування та видалення записів проведемо тестування і занесемо результати в Таблицю 2.

Таблиця 2. Критерії та результати тестування

Опис тесту\Ім'я	Aurelia v1	Vue v2	React v16	Angular v5	Knockout v3.4
Тривалість створення 1000 рядків після завантаження сторінки	181.7 ±4.8 мс	169.2 ±3.6 мс	187.6 ±4.3 мс	185.7 ±7.8 мс	339.8 ±11.7 мс
Тривалість оновлення 1000 рядків в таблиці	74.7 ±1.5 мс	161.8 ±3.9 мс	165.2 ±7.0 мс	179.3 ±6.5 мс	321.4 ±7.7 мс
Час на оновлення тексту в кожному 10 рядку в таблиці з 10 000 рядками.	172.1 ±129.3 мс	168.1 ±7.4 мс	93.6 ±5.6 мс	73.5 ±4.9 мс	77.9 ±4.7 мс
Тривалість підсвічування рядка після натискання по ньому курсором миші	8.3 ±2.8 мс	9.8 ±2.5 мс	12.4 ±4.1 мс	7.6 ±4.0 мс	13.0 ±3.7 мс
Тривалість заміни двох рядків в таблиці розміром 1000 рядків	17.4 ±4.7 мс	19.0 ±2.8 мс	121.8 ±4.2 мс	118.5 ±2.8 мс	125.1 ±3.9 мс
Видалення рядка	42.9 ±2.2 мс	52.5 ±1.8 мс	51.5 ±2.0 мс	46.1 ±2.6 мс	52.8 ±1.9 мс
Тривалість створення 10,000 рядків після завантаження сторінки	1785.6 ±42.1 мс	1521.4 ±55.7 мс	2033.7 ±32.0 мс	1682.0 ±53.1 мс	3102.4 ±127.0 мс

Тривалість додавання 1000 записів в таблицю з 10,000 записами	238.1 ±8.0 мс	338.4 ±10.3 мс	271.8 ±9.9 мс	257.6 ±11.1 мс	4250.2 ±78.6 мс
Тривалість очищення таблиці з 10,000 записами	264.3 ± 12.3 мс	240.9 ± 11.4 мс	224.4 ± 6.0 мс	360.3 ± 16.4 мс	564.4 ± 88.7 мс
Час завантаження, розбору та запуску	28.5 ± 0.9 мс	48.4 ± 2.4 мс	49.4 ± 0.7 мс	88.8 ± 2.9 мс	47.8 ± 0.6 мс
Середнє значення	286,1 мс	296,5 мс	326,6 мс	304,6 мс	578,9 мс

В кінці робимо підрахунок середнього часу тривалості всіх операцій і підсумовуємо, що середнє значення тривалості по всім операціям найкраще у фреймворка Aurelia. Його середнє значення складає 286,1 мілісекунд.

Висновок

У статті було проведено аналіз проблеми великої кількості і різноманітності фреймворків, написаних мовою JavaScript. Відібрано 5 найпоширеніших у використанні фреймворків та проведено детальне тестування кожного в однакових умовах. Кожен був випробуваний в різних щоденних ситуаціях, котрі трапляються при розробці інтернет-ресурсів, а саме додавання інформації, її сортування, фільтрація, обробка та видалення. Тестування проводилось з великою кількістю даних для більш чіткого і якісного відображення різниці в продуктивності кожного. Аналіз результатів тестування (Таблиця 2) приводить до висновку, що найпродуктивнішим фреймворком у 2017 році являється Aurelia, розроблена компанією Blue Spire рік тому.

Посилання

1. JavaScript бібліотека [електронний ресурс] Режим доступу: https://uk.wikipedia.org/wiki/%D0%91%D1%96%D0%B1%D0%BB%D1%96%D0%BE%D1%82%D0%B5%D0%BA%D0%B0_JavaScript
2. Фреймворк [електронний ресурс] Режим доступу: https://uk.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BD%D0%B8%D0%B9_%D0%BA%D0%B0%D1%80%D0%BA%D0%B0%D1%81
3. Alex Ivanovs. Top 23 Best Free JavaScript Frameworks for Web Developers 2017 [електронний ресурс] Режим доступу: <https://colorlib.com/wp/javascript-frameworks/>
4. Список JavaScript бібліотек [електронний ресурс] Режим доступу: https://uk.wikipedia.org/wiki/%D0%A1%D0%BF%D0%B8%D1%81%D0%BE%D0%BA_JavaScript_%D0%B1%D1%96%D0%B1%D0%BB%D1%96%D0%BE%D1%82%D0%B5%D0%BA
5. The definitive source of the best JavaScript libraries, frameworks, and plugins [електронний ресурс] Режим доступу: <https://www.javascripting.com/application-tools/frameworks/?sort=rating>
6. Denis Gladkikh. Паттерны: MVC, MVP и MVVM [електронний ресурс] Режим доступу: <https://www.outcoldman.com/ru/archive/2010/02/22/%D0%BF%D0%B0%D1%82%D1%82%D0%B5%D1%80%D0%BD%D1%8B-mvc-mvp-%D0%B8-mvvm/>
7. Angular – офіційний сайт [електронний ресурс] Режим доступу: <https://angular.io/>
8. React – офіційний сайт [електронний ресурс] Режим доступу: <https://reactjs.org/>
9. Aurelia – офіційний сайт [електронний ресурс] Режим доступу: <http://aurelia.io/>
10. VueJS – офіційний сайт [електронний ресурс] Режим доступу: <https://vuejs.org/>
11. Knockout – офіційний сайт [електронний ресурс] Режим доступу: <http://knockoutjs.com/>
12. Stefan Krause. JS Framework Benchmark [електронний ресурс] Режим доступу: <https://github.com/krausest/js-framework-benchmark>

Authors:

Ruslan Loboda, Valeriy Kurchanov, Pona Chernytska

PERFORMANCE ANALYSIS OF MODERN JAVASCRIPT-FRAMEWORKS

Abstract. In this article, the situation of the frameworks written in JavaScript language for the development of Internet resources was analyzed. The attention paid to the problem of over-saturation of IT-sphere with applications and frameworks of this type. Five most up-to-date JavaScript frameworks are selected. A detailed testing and analysis of the performance testing results was performed. Based on the results, it is concluded that Aurelia is the most productive frame of the year 2017.

Keywords: framework, internet resource, JavaScript, Aurelia, NPM.

Рецензент:

Автори:

Лобода Руслан Николаевич, Курчанов Валерий Никитич, Черницкая Илона Александровна

АНАЛИЗ ПРОИЗВОДИТЕЛЬНОСТИ СОВРЕМЕННЫХ JAVASCRIPT-ФРЕЙМВОРКОВ

Аннотация. В данной статье была проанализирована ситуация фреймворков, написанных на языке JavaScript для разработки интернет-ресурсов. Уделено внимание проблеме перенасыщения IT-сферы приложениями и фреймворками данного типа. Выделено 5 самых актуальных JavaScript-фреймворков. Проведено детальное тестирование и анализ результатов тестирования производительности. На основе результатов сделан вывод, что Aurelia самый производительный фреймворк 2017 года.

Ключевые слова: фреймворк, интернет-ресурс, JavaScript, Aurelia, NPM.