

УДК 004.418

**Б. Равчак**

Тернопільський національний технічний університет імені Івана Пулюя

## ХАРАКТЕРИСТИКА МЕТОДОЛОГІЇ JAMSTACK

UDC 004.418

**B. Ravchak**

(Ternopil Ivan Puluj National Technical University, Ukraine)

## CHARACTERISTICS OF JAMSTACK METHODOLOGY

Архітектура веб-додатків описує взаємодію між додатками, базами даних та системами проміжного програмного забезпечення в Інтернеті. Як тільки користувач натискає кнопку переходу після введення URL-адреси, сервер надсилає файли в браузер як відповідь на зроблений запит. Потім браузер виконує ці файли, щоб показати потрібну сторінку.[1].

Нинішні веб-сайти, в основному, відносяться до базової моделі мережевого клієнт-серверного програмування з двома загальними елементами:

- сторона сервера. Сюди входить апаратне і програмне забезпечення веб-сервера, а також програмні елементи і вбудовані технології. діапазон цих технологій простягається від простих програм CGI, написаних на Perl, до комплексних багатоланкових додатків на основі PHP. Тут враховуються прикладні технології, наприклад - сервери баз даних, які можуть забезпечувати підтримку веб-сайту.

- сторона клієнта. Сторона клієнта пов'язана з веб-браузером і підтримуваними ним технологіями, такими як мови HTML, CSS і JavaScript, елементи управління ActiveX і змінні модулі Explorer, які використовуються для створення шаблону сторінки або забезпечення інтерактивних функцій.

На заміну цій моделі прийшла методологія JAMstack. Генератори статичних сайтів стають дуже поширеними, оскільки вони базуються на найсучасніших технологіях та структурах JavaScript, таких як Vue.js або React.

JAMstack - це не технологія. Натомість JAMstack - це новий спосіб створення веб-сайтів та додатків, сучасна архітектура веб-розробки, заснована на стороні клієнта, і не залежить від веб-сервера; статичний HTML-сайт, який автоматично відновлюється щоразу, коли оновлюється вміст, і розгортається безпосередньо на CDN[2].

JavaScript: будь-яке динамічне програмування під час циклу запит / відповідь обробляє JS, повністю працює на клієнті. Це може бути будь-який фронтенд фреймворк чи бібліотека, або навіть натівний JavaScript.

API: усі процеси на сервері або дії з базою даних абстрагуються в API багаторазового використання, доступ до яких здійснюється через HTTPS з JavaScript.

Markup: шаблонна розмітка повинна бути попередньо побудована під час розгортання, як правило, використовуючи генератор веб-сайтів для контентних сайтів або інструмент побудови веб-додатків.

Переваги JAMstack:

- більш швидка та краща продуктивність. Він може генерувати нові сторінки під час розгортання та обслуговувати попередньо розбудовані розмітки та активи над CDN;

- менш дорогі і простіші в масштабі. Менша складність розробки зменшує витрати, а також розміщення статичних файлів є дешевим або навіть безкоштовним;

- вища безпека. З делегуванням операцій на базі сервера та бази даних нам більше не потрібно турбуватися про вразливості;

**Література**

1. Web-application architecture definition models, types and more – Режим доступу : <https://hackr.io/blog/web-application-architecture-definition-models-types-and-more>
2. Why the JAMstack is becoming so popular– Режим доступу : <https://medium.com/notonlycss/why-the-jamstack-is-becoming-so-popular-a26133b12a30>