

АКТУАЛЬНІСТЬ ТА ПРАКТИЧНЕ ЗАСТОСУВАННЯ SPA-ДОДАТКІВ

Single page application (SPA) – це веб-додаток, який виконується безпосередньо на стороні клієнта у Web-браузері, зазвичай для написання такого типу веб-додатків використовується комбінація HTML, JavaScript та CSS. Додаток може отримувати доступ до структури веб-сторінки як до об'єктів DOM. Відразу хочу звернути увагу на переваги та недоліки SPA-додатків.

Переваги використання SPA:

- Першою перевагою варто відмітити той факт, що SPA-додатки відміно працюють як на стаціонарних так і на мобільних приладах. «Великі» комп'ютери, планшети, смартфони, і, навіть, звичайні телефони (деякі) можуть безперешкод працювати з сайтами, що побудовані по принципу SPA.
- Друга перевага – різноманітний користувальницький інтерфейс, що має назву User Experience. Так як веб-сторінка одна, то побудувати різноманітний, насичений користувальницький інтерфейс набагато простіше.
- Третя перевага – SPA істотно (в рази) скорочує так звані «ходіння по колу», тобто завантаження одного і того ж контенту знову.

Недоліки використання SPA:

Використання SPA-додатків для великих і трудомістких задач тягне за собою великі трудозатрати на налагодження процесу завантаження всіх модулів, так, щоб це не займало велику кількість часу, але і це можна вирішити. Як? Описано нижче.

SPA – це абстрактне поняття, це принцип архітектури додатку. З чого ж почати при розробці сайту за принципами SPA. Існує велика кількість базових бібліотек, що реалізують принцип SPA. Що ж дають ці фреймворки:

- забезпечують базові принципи для SPA розробки, мінімізують трудовитрати на вирішення універсальних задач;
- фреймворки створені спільнотами розробників, що означає використання досвіду створення сайтів.

В чому ж різниця між традиційними веб-додатками та SPA-додатками.

В традиційному веб-додатку при кожному вивозі сервера, той здійснює рендеринг нової HTML-сторінки. Це визиває оновлення сторінки браузера.

В SPA після завантаження першої сторінки вся взаємодія з сервером відбувається через AJAX-виклики. Ці AJAX-виклики повертають дані (на розмітку) – зазвичай у форматі JSON. Додатки виористовують JSON-дані для динамічного оновлення сторінки без її перезагрузки. Рис. 1 ілюструє різницю між цими двома підходами.

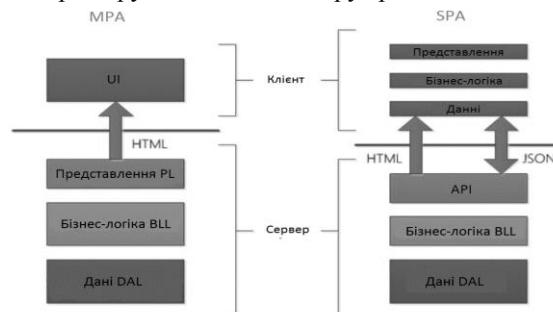


Рис. 1. Різниця між традиційним підходом та SPA

Одна з переваг SPA очевидна: додатки більш гнучкі та адаптивні, вільні від рваного ефекту перезавантаження сторінки і її рендеринга заново. Інша перевага може здатися менш очевидною і стосується того, як ви проектуєте архітектуру веб-додатку. Відправлення даних додатку як JSON забезпечує поділ між презентаційною частиною (HTML-розміткою) і прикладною логікою (AJAX-запити плюс JSON-відповіді). В SPA вся UI-взаємодія відбувається на стороні клієнта за допомогою JavaScript і CSS. Після початкового завантаження сторінки сервер діє виключно як рівень сервісів. Клієнтові потрібно просто знати, які HTTP-запити він повинен посилати серверу. Йому не важливо, як сервер реалізує свою частину. При такій архітектурі клієнт і сервіс незалежні. Ви могли б повністю замінити серверну частину, яка виконує сервіс, і, поки ви не змінюєте API, ви ніяк не порушите роботу клієнта.

Також варто згадати, що у SPA широко використовується підхід Asynchronous module definition (AMD). Asynchronous module definition – це підхід до розробки на JavaScript, який дозволяє створювати модулі таким чином щоб вони і їх залежності могли бути завантажені асинхронно. Асинхронне завантаження модулів дозволяє покращити швидкість завантаження веб-сторінок в цілому, так як модулі завантажуються одночасно з іншим контентом сайту.