

СУЧАСНІ ПІДХОДИ ПРИ ПРОЕКТУВАННІ ВЕБ-ЗАСТОСУНКІВ

¹Вінницький національний технічний університет

Анотація

Дана робота присвячена порівняльному аналізу підходів при проектуванні веб-застосунків, які забезпечать легку та швидку розробку, високу продуктивність, легку підтримку та адаптивність під різні платформи.

Ключові слова: HTML, CSS, Javascript, розробка веб-застосунку, бібліотеки, фреймворки.

Abstract

This work is devoted to a comparative analysis of approaches to web application design that ensure easy and fast development, high performance, easy support, and adaptability to different platforms.

Keywords: HTML, CSS, Javascript, web application development, libraries, frameworks.

Вступ

Сучасні вимоги до веб-застосунків кардинально змінились. Користувачі вже не бажають встановлювати велику кількість застосунків на свою операційну систему, особливо, коли їх використання здійснюється вкрай рідко – це все призводить до неефективного використання пам'яті пристрою. Також з розвитком інших платформ у користувачів виникає бажання працювати з застосунком не залежно від платформи, що при класичному підході є неможливим.

У зв'язку з вище переліченими факторами великої популярності набули веб-застосунки, їх не потрібно встановлювати та вони забезпечують роботу з будь-якої платформи, де є браузер та доступ до мережі Інтернет.

Метою даної роботи є здійснення порівняльного аналізу підходів, що використовуються, при проектуванні веб-застосунків, та забезпечують легку та швидку розробку, високу продуктивність та адаптивність під різні платформи.

Аналіз підходів проектування

Веб-застосунок (іноді веб-додаток) – розподілений застосунок, в якому клієнтом виступає браузер, а сервером — веб-сервер. Браузер може бути реалізацією так званих тонких клієнтів — логіка застосунку зосереджується на сервері, а функція браузера полягає переважно у зображенні інформації, завантаженої мережею з сервера, і передачі назад даних користувача [1].

На даний момент існують наступні підходи проектування веб-застосунків: класичний, з використанням бібліотек та фреймворків та з використанням конструкторів.

Класичний підхід – підхід в розробці веб-застосунків, де використовуються базові веб-технології для розробки, такі як HTML, CSS та JavaScript.

Перевагами цього підходу є максимальна гнучкість під потреби проекту, через повний контроль розробника над кодом. Також варто відмітити малий розмір таких застосунків, через контроль коду, функції, які не потрібні при роботі застосунку не реалізуються, так як в інших, більш автоматизованих, підходах. Також веб-застосунки розроблені за цим підходом будуть підтримуватися всіма платформами.

Головним недоліком цього підходу є великий час розробки, так як цей підхід не має реалізації базових речей та не дозволяє повторно використовувати код, також ці причини ускладнюють розробку та підвищують рівень наслідків при допущенні помилок.

Мала кількість та складне використання сторонніх бібліотек – теж є недоліком цього підходу.

Підхід з використанням бібліотек та фреймворків – підхід в розробці веб-застосунків у якому розробники використовують надані бібліотеками та фреймворками інструменти, для пришвидшення циклу розробки та його спрощення за рахунок зменшення рутинної роботи, можливості повторного

використання коду та широкій базовій реалізації. Також до переваг варто віднести розвинену інфраструктуру для розробників, такі менеджери пакетів як NPM та Yarn [2] дозволяють з легкістю встановлювати та використовувати сторонні бібліотеки, що позитивно впливає на час та важкість розробки. Деякі з фреймворків та бібліотек оптимізують рендеринг інтерфейсу, наприклад React [3], який за допомогою патерну віртуального DOM-дерева підвищує продуктивність ре-рендерингу.

Основним недоліком цього підходу є відсутність підтримки деяких бібліотек та фреймворків на старих браузерах [4].

Підхід з використанням конструкторів – це підхід, де для розробки використовуються спеціальні програми-конструктори в якій будують веб-застосунок за допомогою блоків.

Основною перевагою цього підходу є непотрібність знати мови програмування, оскільки веб-застосунок будується з готових блоків та шаблонів – це робить поріг входження та час розробки дуже низьким.

Недоліком цього підходу є відсутність гнучкості під час розробки, оскільки не всі готові блоки є універсальними. Варто відмітити що застосунки матимуть обмежену функціональність, оскільки деякі функції майже неможливо реалізувати в конструкторах.

Висновки

В результаті аналізу підходів при проектуванні веб-застосунків було визначено їх переваги та недоліки, а саме:

- класичний підхід варто використовувати коли потрібно створити веб-застосунок, який буде підтримуватися на всіх платформах, повинен буди максимально гнучким під потреби проекту та продуктивним. Використання цього підходу потребує високо рівня знань від розробника;

- підхід з використанням бібліотек та фреймворків доцільно використовувати, коли підтримка старих браузерів не є пріоритетною, але потрібно реалізувати високопродуктивний та широкий за функціоналом застосунок в обмежений час;

- підхід з використанням конструкторів доцільно використовувати при реалізації нескладних проєктів за короткий час.

Отже, вибір підходу розробки веб-застосунку залежить від потреб проєкту та рівня знань розробника.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Вебзастосунок [Електронний ресурс] – Режим доступу до ресурсу: <https://uk.wikipedia.org/wiki/Вебзастосунок>.

2. Шпаргалка по Yarn [Електронний ресурс] – Режим доступу до ресурсу: <https://devzone.org.ua/post/spargalka-po-yarn>

3. JavaScript-бібліотека для створення користувацьких інтерфейсів [Електронний ресурс] – Режим доступу до ресурсу: <https://uk.legacy.reactjs.org/>

4. Can I use? [Електронний ресурс] – Режим доступу до ресурсу: <https://caniuse.com/?search=React>

Ренгач Валентин Юрійович – студент групи КН-22мс, факультет інтелектуальних інформаційних технологій та автоматизації, Вінницький національний технічний університет, Вінниця, e-mail: renhach.valentyn@gmail.com

Белзетський Руслан Станіславович – кандидат технічних наук, доцент кафедри комп'ютерних наук, Вінницький національний технічний університет, e-mail: belzetskiy@vntu.edu.ua

Renhach Valentyn Yu. – student of CS-22JS group, faculty of intellectual information technologies and automation, Vinnytsa national technical university, Vinnytsa, e-mail: renhach.valentyn@gmail.com

Belzetskiy Ruslan S. – candidate of technical sciences, associate professor of the Department of Computer Sciences, Vinnytsia National Technical University, e-mail: belzetskiy@vntu.edu.ua