

## СИСТЕМА КОНТРОЛЮ ЛОКАЛЬНОГО СТАНУ КЛІЄНТСЬКОГО ВЕБЗАСТОСУНКУ

Заполовський М.Й., Юшко В.Р.

*Національний технічний університет*

*"Харківський політехнічний інститут", м. Харків*

При створенні вебзастосунків ключову роль відіграє вибір каркасу для розробки. Від цього вибору залежить подальший можливий розвиток проєкту. Здебільшого каркаси розробки поділяють на дві групи: клієнтські і серверні. Серверні готують увесь вміст, що має відобразитися у користувача. На сервері повертають лише результат. Клієнтські – завантажують увесь потенційний статичний функціонал одразу на сторону клієнта і згодом лише отримують додаткові дані з сервера. Якщо для серверних застосунків локальне сховище даних непотрібне, оскільки можна звертатися до бази даних напямучу, то для клієнтських застосунків спосіб збереження даних є важливим і впливає на швидкодію. Для вирішення цього питання створюються спеціальні клієнтські програмні компоненти, які пропонують спосіб збереження даних на клієнтській стороні.

Метою доповіді є аналіз вимог до інструментів для збереження локального стану клієнтських застосунків. У доповіді наводяться результати розробки програмного рішення, яке дозволяє завдяки початковим налаштуванням зменшити кількість запитів до сервера, не виключаючи потенційну актуалізацію стану різними способами. Цей засіб повинен працювати незалежно від каркасу, який використовується для відображення користувацького інтерфейсу.

Для створення проєкту були залучені `prpm` – пакетний керівник з підвищеною швидкістю, та `tsup` – перетворювач, що транслює код, написаний мовою TypeScript[1], на JavaScript[2], який може виконуватися в браузері. Також для розробки були використані інструменти `biomejs` – для статичного аналізу та перевірки написаного коду, та `changesets` – для контролю версіювання розробленого компонента. Сам програмний продукт написаний мовою TypeScript[1] для полегшення створення типів для розробників, які його використовуватимуть. Для роботи потрібен допоміжний модуль `eventemitter3`, код якого був заздалегідь вивчений та перевірений, що допомагає створювати генератори подій.

Запропонований інструмент, який розроблений за допомогою сучасних технологій, дозволяє швидко, надійно та з мінімальними зусиллями взаємодіяти з сервером.

### Література:

1. The TypeScript Handbook. URL: <https://www.typescriptlang.org/docs/handbook/intro.html> (дата звернення 21.04.2025).
2. ECMAScript® 2026 Language Specification. URL: <https://tc39.es/ecma262/multipage/> (дата звернення 21.04.2025).