

УДК 004.7

*Петров Д.В., здобувач,  
Єфремов Ю.М., к.т.н., доцент  
Державний університет «Житомирська політехніка»*

## **АРХІТЕКТУРА СИСТЕМИ ПЕРСОНАЛІЗАЦІЇ В ВЕБ-ПЛАТФОРМАХ ДЛЯ ЧИТАННЯ РАНОБЕ**

Читання з екрана саме по собі навантажує зір, а якщо йдеться про ранобе з сотнями розділів - це відчувається вдвічі сильніше. До того ж орієнтуватися в такому обсязі без зручних інструментів навігації буває справді незручно. Саме тому сучасна платформа для читання мусить вирішувати не лише технічні питання швидкості, а й думати про те, наскільки комфортно конкретному користувачеві взаємодіяти з інтерфейсом.

У рамках роботи було розроблено модуль інтерактивного рідера, вбудованого безпосередньо у веб-платформу. Головна ідея полягала в тому, щоб дати читачеві змогу підлаштовувати вигляд сторінки під себе - змінювати колірну схему, відстань між рядками - і при цьому не перезавантажувати сторінку щоразу. Крім цього, окремо реалізовано функцію автозбереження місця читання з можливістю синхронізації між кількома пристроями.

Під час дослідження було розглянуто різні способи зберігання користувацьких налаштувань. Результати порівняння зведено в табл. 1.

За підсумками аналізу було обрано комбінований підхід: налаштування зовнішнього вигляду зберігаються у LocalStorage браузера - це забезпечує миттєву реакцію інтерфейсу - тоді як прогрес читання та список вподобань передаються на сервер через API. Такий підхід дає змогу не переривати читання навіть у тому випадку, коли користувач переходить на іншій пристрій або очищає кеш браузера.

Окремо реалізовано функцію відновлення позиції в розділі - якщо читач не дочитав розділ до кінця і пізніше повернувся до нього, сторінка автоматично прокручується до того місця, де він зупинився. Це позбавляє від необхідності вручну шукати потрібний абзац і робить повернення до читання природним продовженням, а не черговим пошуком по тексту.

Таблиця 1

Порівняння методів зберігання користувацьких даних

Метод зберігання	Швидкість доступу	Синхронізація між пристроями	Навантаження на сервер
LocalStorage	Максимальна	Відсутня	Нульове
Cookies	Середня	Часткова	Низьке
Database	Нижч	Повна	Середнє

Технічна реалізація серверної частини базується на використанні **ASP.NET Core Web API**. Для забезпечення високої швидкості вибірки розділів із бази даних застосовано асинхронні запити через **Entity Framework Core**. Клієнтська частина використовує сучасні механізми **Tailwind CSS** для миттєвої зміни теми оформлення, що мінімізує використання ресурсів процесора на мобільних пристроях.

**Список використаних джерел:**

1. Документація ASP.NET Core Web API. [Електронний ресурс] - Режим доступу: <https://learn.microsoft.com/en-us/aspnet/core/web-api/>
2. Use of LocalStorage for User Preferences. [Електронний ресурс] - Режим доступу: <https://developer.mozilla.org/en-US/docs/Web/API/Window/localStorage>
3. Дизайн-патерни в розробці інтерфейсів для читання. [Електронний ресурс] - Режим доступу: <https://uxdesign.cc/>