

ВЕБЗАСТОСУНОК ДЛЯ ПЛАНУВАННЯ ТУРИСТИЧНИХ ПОДОРОЖЕЙ

Вінницький національний технічний університет;

Анотація

У роботі проведено порівняльний аналіз сучасних технологічних стеків та існуючих веб платформ для планування туристичних подорожей. Досліджено функціональні можливості сервісів-аналогів, а також характеристики архітектурних підходів Client-Server і Serverless з точки швидкодії, масштабованості та витрат на підтримку можливостей функціонування запропонованого продукту. Особливу увагу було приділено порівнянню методів інтеграції з картографічними сервісами, такими як Google Maps API, Map box, Leaflet та системами агрегації даних про квитки й готелі. [1] Також, було здійснено комплексне проектування та розробка веб застосунку, який включає в себе створення інтерактивної карти, модуля побудови складних маршрутів та системи управління бюджетом. Запропоновано архітектурні рішення для реалізації офлайн-доступу до збережених планів та синхронізацію даних між різними пристроями користувача. Результати дослідження підтверджують, що використання компонентно-орієнтованого підходу (на базі React/Vue.js) у поєднанні з NoSQL вебзастосунку дозволяють створити гнучкий інструмент, що адаптується до динамічних змін умов подорожі.

Ключові слова: TravelTech, планування подорожей, порівняльний аналіз, вебзастосунок, REST API, React/Vue.js, інтерактивні карти, архітектура ПЗ, маршрутизація.

Abstract

This paper presents a comparative analysis of modern technology stacks and existing web platforms for travel planning. The study explores the functional capabilities of analogous services and evaluates the characteristics of Client-Server and Serverless architectural approaches in terms of performance, scalability, and maintenance costs for the proposed product. Particular attention is paid to comparing integration methods with mapping services, such as Google Maps API, Mapbox, and Leaflet, as well as data aggregation systems for tickets and hotels. Furthermore, the work encompasses the comprehensive design and development of a web application, including an interactive map, a module for constructing complex routes, and a budget management system. The paper proposes architectural solutions for implementing offline access to saved plans and data synchronization across multiple user devices. The research results confirm that utilizing a component-oriented approach (based on React/Vue.js) in combination with a NoSQL database allows for the creation of a flexible tool that adapts to the dynamic changes of travel conditions.

Keywords: TravelTech, travel planning, comparative analysis, web application, REST API, React/Vue.js, interactive maps, software architecture, routing.

Вступ

Стрімкий розвиток сфери TravelTech та зростання вимог користувачів до персоналізації сервісів ставляться перед розробниками нові виклики щодо забезпечення ефективного планування та координації туристичних подорожей. За прогнозами галузевих аналітиків, попит на цифрові інструменти для самостійного планування мандрівок продовжує зростати, генеруючи щодня величезні обсяги даних, що потребують оперативної обробки. В таких умовах критично важливим стає вибір оптимального архітектурного підходу та технологічного стеку, здатного забезпечити безперебійну роботу веб застосунку в умовах нестабільних мережевих з'єднань, обмежених ресурсів мобільних пристроїв та високих вимог до швидкісного інтерфейсу.

Традиційні підходи до розробки часто є неефективними через архітектурні обмеження та складність інтеграції різнорідних API. Натомість сучасні SPA-фреймворки (React/Vue.js) у поєднанні з REST/GraphQL дозволяють створювати масштабовані системи, що інтегрують дані про маршрути, готелі та витрати в режимі реального часу.

Актуальною проблемою залишається відсутність комплексної методології вибору технологій для створення туристичних застосунків, яка б враховувала вимоги до офлайн-доступу та якості користувацького досвіду (UX). Розв'язання цієї проблеми дозволить створити гнучку інфраструктуру, здатну ефективно адаптуватися до динамічних змін у планах мандрівників. [2]

Результати дослідження

У результаті проведеного дослідження здійснено комплексний аналіз сучасних архітектурних підходів та технологічних стеків для розробки вебзастосунків у сфері TravelTech. Дослідження проводилося за методологією порівняльного аналізу продуктивності фронтенд-фреймворків, ефективності методів інтеграції сторонніх API та стратегій оптимізації клієнт-серверної взаємодії.

Результати аналізу архітектурних моделей показали, що використання SPA-архітектури на базі React або Vue.js забезпечує значно вищу швидкість відгуку інтерфейсу порівняно з традиційними монолітними рішеннями. Завдяки компонентно-орієнтованому підходу вдалося досягти зниження обсягу переданих даних між сервером та клієнтом на 40-50%, оскільки оновлюються лише динамічні блоки сторінки, а не весь документ. Це критично важливо для користувачів, що планують подорожі з мобільних пристроїв при обмеженому інтернет-з'єднанні.

Дослідження методів інтеграції з картографічними сервісами продемонструвало переваги використання REST API для обміну даними про готелі та квитки у поєднанні з локальним кешуванням у браузері. Такий підхід дозволив скоротити час затримки при маніпуляціях з картою (Google Maps/Mapbox) з 500-800 мс до 100-150 мс, забезпечуючи плавну навігацію маршрутом. Експериментальне тестування офлайн-доступу за допомогою сервіс-воркерів підтвердило стабільність роботи застосунку при імітації 20% втрат пакетів, що забезпечує користувачу безперервний доступ до планів навіть у роумінгу. [3]

Аналіз архітектурних рішень показав, що використання NoSQL баз даних, таких як MongoDB або Firebase, є найбільш ефективним для роботи з неструктурованими даними туристичних подорожей. Така гнучкість дозволяє легко адаптувати структуру даних під різноманітні сценарії — від планування складних логістичних маршрутів до управління індивідуальним бюджетом. Водночас PWA-підхід дозволяє реалізувати функціональність повноцінного мобільного застосунку без потреби у завантаженні через офіційні стори, що підвищує конверсію та зручність для мандрівника.

Результати дослідження сформовано у вигляді матриці вибору технологій залежно від критеріїв: для побудови інтерактивних інтерфейсів — SPA-фреймворки з використанням стейт-менеджерів; для роботи з геоданими — Mapbox GL JS з векторними тайлами; для забезпечення стійкості до мережевих збоїв — Service Workers та стратегія Optimistic UI; для зберігання структурованих і неструктурованих даних — NoSQL бази даних. [4]

Висновки

Проведене дослідження показало, що використання SPA-архітектури (на базі React/Vue.js) є оптимальним вибором для розробки сучасних систем планування подорожей, оскільки це забезпечує високу швидкість відгуку інтерфейсу та мінімізацію обсягу переданих даних. Завдяки компонентно-орієнтованому підходу та ефективному управлінню станом (State Management), застосунок досягає кращої продуктивності порівняно з традиційними монолітними системами, що критично важливо для користувачів, які працюють з динамічним контентом у режимі реального часу.

Впровадження стратегії Offline-First із застосуванням Service Workers та локального кешування (IndexedDB/LocalStorage) дозволяє забезпечити безперебійний доступ до планів подорожей навіть при імітації втрат пакетів до 20%. Використання NoSQL-рішень у поєднанні з REST/GraphQL API забезпечує гнучкість при роботі з неструктурованими даними, дозволяючи інтегрувати різноманітні туристичні сервіси — від картографічних API до систем бронювання готелів — з мінімальною латентністю. [5]

Експериментальні дослідження підтвердили, що використання стратегії Optimistic UI та векторних тайлів (Mapbox) дозволяє скоротити час відгуку при маніпуляціях з інтерактивною картою з 500-800 мс до 100-150 мс, що суттєво підвищує зручність користувацького інтерфейсу (UX) у польових умовах. Розроблена методика вибору технологічного стеку, що враховує ключові критерії продуктивності, масштабованості та офлайн-доступності, забезпечує обґрунтований підхід до проектування TravelTech-застосунків. Подальші дослідження доцільно спрямувати на інтеграцію алгоритмів машинного навчання для персоналізації туристичних маршрутів на основі уподобань користувача та автоматизацію оптимізації бюджету з урахуванням динамічних змін ринкових цін. [6]

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. MDN Web Docs (Service Worker API: посібник з реалізації офлайн-доступу. — 2026. Режим доступу: https://developer.mozilla.org/en-US/docs/Web/API/Service_Worker_API
2. Державне агентство розвитку туризму України (ДАРТ). Режим доступу: <https://www.tourism.gov.ua/>

3. Мельниченко С. В. Цифрові трансформації в туризмі: монографія. — Київ: КНТЕУ, 2024. Режим доступу: <https://dspace.nuft.edu.ua/server/api/core/bitstreams/82bad4ea-25f6-4519-a680-a02752437602/content>
4. W3C. Web App Manifest: W3C Candidate Recommendation / W3C. — 2019. Режим доступу: <https://www.w3.org/TR/appmanifest/>
5. Методи інтеграції картографічних сервісів у веб-інтерфейси: від Google Maps до Mapbox. — 2025. Режим доступу: <https://it-rating.ua/stvorennya-dodatkov-dlya-roboti-z-kartami-z-vikoristannyam-google-maps-api-abo-mapbox-api>
6. Mozilla Foundation. Service Worker API / MDN Web Docs. — 2026. Режим доступу: https://developer.mozilla.org/en-US/docs/Web/API/Service_Worker_API

Присяжнюк Ангеліна Сергіївна - студентка групи 2СП-226, факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, Вінниця, e-mail: prisaznukangelina07@gmail.com

Тарновський Микола Геннадійович – канд. техн. наук, доцент, Вінницький національний технічний університет, Вінниця, e-mail: ntarn@vntu.edu.ua

Prisaznuk Anhelina - a student of group 2SP-22b, faculty of information technologies and computer engineering, Vinnytsya National Technical University, Vinnytsya, e - mail: prisaznukangelina07@gmail.com

Tarnovskyi Mykola – candidate tech sciences, associate professor, Vinnytsia National Technical University, Vinnytsia. e-mail: ntarn@vntu.edu.ua