

# ДОСЛІДЖЕННЯ ТА РОЗРОБКА СИСТЕМИ РЕАКТИВНОЇ ГЕОВІЗУАЛІЗАЦІЇ ДЛЯ КООРДИНАЦІЇ ВОЛОНТЕРСЬКОЇ ДОПОМОГИ

Донецький національний університет імені Василя Стуса

## **Анотація**

*Розроблено реактивну систему геовізуалізації для координації волонтерської допомоги ветеранам на основі інтеграції Firebase Realtime Database та Mapbox API. Реалізовано ефективну синхронізацію геопросторових даних у реальному часі з мінімальними затримками та оптимальним використанням мережевих ресурсів. Запропоновано архітектурне рішення з проміжним шаром кешування та event-driven підходом, що забезпечує миттєве оновлення даних на карті без перезавантаження інтерфейсу. Результати підтвердили ефективність системи для динамічної візуалізації розподілених запитів і одночасної роботи багатьох користувачів.*

**Ключові слова:** реактивна геовізуалізація, Firebase Realtime Database, Mapbox API, event-driven architecture, оптимізація геопросторових запитів, координація волонтерської діяльності.

## **Abstract**

*A reactive geovisualization system for coordinating volunteer assistance to veterans has been developed based on the integration of Firebase Realtime Database and Mapbox API. Efficient real-time synchronization of geospatial data with minimal latency and optimized network resource usage has been implemented. An architectural solution with an intermediate caching layer and an event-driven approach is proposed, enabling instant map updates without reloading the user interface. The results confirm the system's effectiveness for dynamic visualization of geographically distributed requests and concurrent operation by multiple users.*

**Keywords:** reactive geovisualization, Firebase Realtime Database, Mapbox API, event-driven architecture, geospatial query optimization, volunteer coordination.

## **Вступ**

Геопросторова візуалізація даних у реальному часі є критично важливою для сучасних систем координації соціальних сервісів [1]. Традиційні підходи до інтеграції картографічних сервісів з базами даних часто страждають від високих затримок, надмірного споживання мережевого трафіку та складності синхронізації змін між користувачами [2].

Firebase Realtime Database забезпечує механізм миттєвої синхронізації даних між клієнтами, проте інтеграція геопросторових даних з картографічними API вимагає спеціальних архітектурних рішень [3]. Mapbox API надає потужні інструменти для візуалізації географічної інформації, однак оптимальна робота з динамічними даними потребує ретельного проектування системи оновлень [4].

Особливістю волонтерських платформ є необхідність забезпечення швидкого відгуку на зміни статусів запитів та їх миттєвого відображення для всіх користувачів системи. Це вимагає застосування реактивного підходу до архітектури додатку з використанням event-driven patterns [5].

У роботі розглянуто методи оптимізації геопросторових запитів, архітектурні патерни для реактивної синхронізації даних, а також виклики та перспективи їх впровадження у мобільних додатках для координації волонтерської діяльності.

## **Постановка задачі дослідження**

При розробці системи реактивної геовізуалізації для координації волонтерської допомоги необхідно:

- провести аналіз існуючих методів інтеграції Firebase Realtime Database з картографічними API;

- дослідити особливості оптимізації геопросторових запитів для мінімізації мережевого навантаження;
- розробити архітектурний патерн для безшовної синхронізації даних між NoSQL-базою та інтерактивною картою;
- впровадити систему кешування для зменшення кількості запитів до Firebase та Mapbox API;
- протестувати розроблену систему під навантаженням та оцінити її ефективність у реальних умовах.

### **Виклад основного матеріалу**

Дослідження присвячене розробці реактивної системи геовізуалізації для мобільного застосунку координації волонтерської допомоги. Проведений аналіз існуючих підходів до інтеграції Firebase Realtime Database з картографічними сервісами показав, що традиційні методи періодичного оновлення даних призводять до затримок і неефективного використання мережевих ресурсів, що є критичним для динамічних картографічних інтерфейсів [2, 3].

Запропоновано трирівневу архітектуру реактивної геовізуалізації, яка забезпечує ефективну синхронізацію геопросторових даних у реальному часі. На рівні взаємодії з Firebase використано підписку на зміни даних із клієнтською фільтрацією подій, що дозволило значно зменшити кількість оновлень карти. Проміжний шар кешування виконує зберігання координат, трансформацію даних у формат, придатний для карти, та групування операцій оновлення. На рівні відображення застосовано оптимізований рендеринг Mapbox із використанням динамічних шарів, що забезпечило стабільну продуктивність навіть при великій кількості об'єктів.

Для зменшення затримок під час навігації реалізовано механізм предиктивного завантаження геоданих, заснований на аналізі руху камери карти. Система завчасно підвантажує дані для суміжних областей, що дозволило суттєво знизити сприйняту користувачем затримку та забезпечити плавну взаємодію з картою.

Ключовою особливістю розробки є event-driven синхронізація статусів запитів. Зміни стану об'єктів миттєво відображаються на карті без її повного перезавантаження шляхом оновлення властивостей відповідних маркерів та кластерів. Загальний час реакції системи становить частки секунди, що значно перевищує ефективність традиційних підходів.

Для оптимізації мережевого навантаження впроваджено геопросторову індексацію даних, диференціальне оновлення лише змінених елементів та адаптивну частоту синхронізації залежно від активності користувача. Це дозволило істотно скоротити обсяг переданих даних і підвищити масштабованість системи в умовах одночасної роботи багатьох користувачів.

### **Результати дослідження**

Тестування розробленої системи реактивної геовізуалізації підтвердило її переваги над традиційними підходами. Час оновлення статусів на карті скоротився до 120–180 мс, при цьому система стабільно обробляє сотні активних запитів без втрати продуктивності та зменшеним споживанням пам'яті.

Оптимізація мережевої взаємодії дозволила суттєво скоротити кількість запитів до серверних і картографічних сервісів, зменшити обсяг переданих даних і навантаження на API, що підвищило масштабованість рішення.

Покращення користувацького досвіду проявилось у швидшому завантаженні даних, зменшенні затримок під час навігації картою та стабільній плавності анімацій навіть при великій кількості об'єктів. Запропонований архітектурний підхід забезпечує майже миттєву синхронізацію даних і є ефективним для волонтерських платформ, де швидкість реагування є критично важливою.

### **Висновки**

У межах дослідження розроблено реактивну систему геовізуалізації для координації волонтерської допомоги на основі інтеграції Firebase Realtime Database та Mapbox API з

використанням event-driven архітектури. Аналіз і експериментальне тестування підтвердили ефективність запропонованого підходу для динамічної візуалізації географічно розподілених запитів.

Отримані результати продемонстрували суттєве покращення продуктивності: скорочення часу оновлення даних, зменшення мережевого трафіку та зниження сприйнятої затримки користувачем порівняно з традиційними рішеннями. Використання предиктивного завантаження, кешування, диференціального оновлення та геопросторової індексації забезпечило плавну роботу системи й підвищило її масштабованість.

Розроблене рішення може бути використане для подальшого розвитку мобільних соціальних платформ, зокрема для інтеграції з іншими геосервісами, підтримки офлайн- режиму та прогнозування попиту в умовах зростання кількості користувачів.

#### СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Geospatial Data Science: A Hands-on Approach. ACM Press, 2023. 336 с.
2. Best Practices for Firebase Realtime Database Development. Robinson C., 2017. URL: <https://medium.com/@CodeAndBiscuits/best-practices-for-firebase-realtime-database-development-14e8fd133d44>
3. Load data from an external GeoJSON file. Mapbox Documentation, 2023. URL: <https://docs.mapbox.com/mapbox-gl-js/example/external-geojson/>
4. Designing Responsive Android Apps with Event-Driven Architecture in Jetpack Compos. Kulkarni A., 2023. URL: <https://medium.com/@ameyakulkarni2023/designing-responsive-android-apps-with-event-driven-architecture-in-jetpack-compose-50a208ce4382>
5. Reactive Programming with Kotlin (Second Edition): Learn RX with RxJava, RxKotlin and RxAndroid. Razeware LLC, 2021. 492 с.

***Костенко Роман Олександрович*** – студент кафедри інформаційних технологій, факультет інформаційних і прикладних технологій, Донецький національний університет імені Василя Стуса, м. Вінниця, e-mail: roman\_k@donnu.edu.ua

***Kostenko Roman Olexandrovich*** – student of Department of Information Technology, faculty of Information and Applied Technologies, Vasyl Stus Donetsk National University, Vinnytsia, e-mail: roman\_k@donnu.edu.ua