

Колеснікова Н. С.

Крупельницький Л. В.

## Засоби розробки кросплатформенної інформаційно-пошукової системи лікарів Вінниці

### Анотація

*В роботі проведено дослідження засобів кросплатформенної розробки додатків, елементів архітектури, інструментів, мов програмування, фреймворків.*

**Ключові слова:** Javascript, Expo, React, Android.

### Abstract

*The work includes research on cross-platform application development, elements of architecture, tools, programming languages, frameworks.*

**Keywords:** Javascript, React, Expo, Android.

### Вступ

Сьогодні вже існує багато платформ, бібліотек і систем у сфері розробки мобільних додатків. Opensource відкриває безмежні можливості для програмування. Це робить розробку програм та веб-сайтів легкою та цікавою. Однак постає проблема в кросплатформенності та в універсальності коду. Це означає, що той самий код може використовуватися на всіх платформах (веб, iOS, android). Кожна платформа використовує власну мову та бібліотеку, які не завжди стандартизовані та повністю функціональні. Тому для вирішення цієї проблеми була створена система та платформа на мові програмування Javascript під назвою Expo. Використовуючи цей фреймворк, можна одночасно створювати глобальні додатки для всіх платформ. Він компілює Javascript код в код, який необхідний для кожної з платформ. Фундаментально це заощаджує дуже багато ресурсів на розробку продукту. Швидкість цього додатку буде високою через використання бібліотеки React для відображення даних.

### Розробка на фреймворку Expo

Javascript – динамічна, об'єктноорієнтовна, прототипна мова програмування. Може працювати як на стороні сервера (express js, koa js) так і на стороні клієнта. JavaScript класифікують як прототипну (підмножина об'єктно-орієнтованої), скриптову мову програмування з динамічною типізацією. Окрім прототипної, JavaScript також частково підтримує і інші парадигми програмування (імперативну та частково функціональну) і деякі відповідні архітектурні властивості, зокрема: динамічна та слабка типізація, автоматичне керування пам'яттю, прототипне наслідування, функції як об'єкти першого класу.

React – відкрита JavaScript-бібліотека для створення інтерфейсів користувача, яка покликана вирішувати проблеми часткового оновлення вмісту веб-сторінки, з якими стикаються в розробці односторінкових застосунків. React дозволяє розробникам створювати великі веб-застосунки, які використовують дані, що змінюються з часом без перезавантаження сторінки. Його мета полягає в тому, щоб бути швидким, простим, масштабованим.

Фреймворк Expo - це платформа для універсальних програм, заснованих на JavaScript-бібліотеці React. Цей фреймворк складається з набору готових інструментів та сервісів на основі React Native та native платформ, які допомагають розробляти, створювати, розгортати та швидко виконувати ітерації на iOS, Android та веб-додатках з однієї і тієї ж кодової бази JavaScript/TypeScript. У даному фреймворку існує два основних підходи до створення програм керований (managed) і простий (bare) підхід. За допомогою керованого підходу пишуться тільки інструменти та сервіси JavaScript/TypeScript, а все інше фреймворк Expo зробить сам.

У керованому підході Expo намагається полегшити складність створення мобільних додатків. Розробники в даному підході не використовують Xcode або Android Studio, вони просто пишуть код JavaScript і керують налаштуванням таких речей, як значок програми і заставка через app.json. У Expo SDK представлений більш повний набір API-інтерфейсів, які дають можливість доступу до таких можливостей пристрою як камера, біометрична автентифікація, файлова система, тактильні пристрої тощо.

Розмір керованої програми Expo на iOS становить близько 20 МБ, а Android – близько 15 МБ. Це пов'язано з тим, що керований підхід включає безліч API-інтерфейсів незалежно від того, використовуються вони в проекті чи ні. Використання API дозволяє надсилати оновлення через бездротову мережу для використання нових API-інтерфейсів. Деякі з увімкнених API-інтерфейсів пов'язані з послугами, які, швидше за все, не будуть використовуватися в проекті, наприклад, Facebook Mobile SDK включений для підтримки входу в Facebook та реклами Facebook, а також Google Mobile SDK з аналогічних причин. Якщо необхідно максимально зменшити розмір програми, керований підхід може виявитися не кращим вибором – власні бібліотеки для інтеграції з пропрієтарними сервісами зазвичай не включаються до SDK. Отже, SDK не включає всіх інструментів, які могли знадобитися під час розробки. У цих випадках розробники використовують простий підхід та додати потрібні елементи вручну – єдиний підтримуваний сервіс push-повідомлень усередині фреймворку Expo. У випадку, якщо API Expo Push-повідомлень не підходить для розробки даного проекту, слід додати іншу службу push-повідомлень, наприклад OneSignal використовуючи простий підхід – мінімальні підтримувані версії ОС: Android 5+ та iOS 10+. Якщо ж потрібно підтримувати більш старі версії, не слід використовувати керований підхід – безкоштовні складання іноді можуть бути поставлені в чергу. У фреймворку Expo є можливість створити свою програму для відправки в Apple App Store і Google Play Store, навіть не встановлюючи Xcode або Android Studio. Це можна зробити, використовуючи безкоштовну автономну службу складання, але іноді вона має чергу в залежності від того, скільки інших розробників збирають інсталяційний файл у цей час. У простому підході розробник отримує повний доступ до базових нативних проектів та будь-якого власного коду. У результаті створюється чистий проект із React Native та одним або декількома пакетами з Expo SDK.

## Висновки

Досліджено інноваційний на сьогодні інструмент для розробки додатку для всіх платформ та його ключові елементи. Проаналізувавши всі платформи було встановлено, що на сьогодні нативні додатки мають перевагу у продуктивності, проте швидкість розробки і універсальність коду має Expo. Прогнозується, що у майбутньому продуктивність цих додатків буде однаковою та процес створення додатків буде швидший. Досліджені всі переваги Expo. Встановлено, що для створення додатку для всіх платформ необхідно знати лише мову програмування Javascript в той час для розробки додатків без Expo необхідно знати як мінімум 3 мови (Kotlin/Java, Swift, Javascript). Також, без Expo витратити мінімум у 3 рази більше часу та в 3 рази більше репозитаріїв для зберігання коду. Підсумовуючи можна сказати що Expo вперше дає нам можливість створювати додатки на всіх платформах і не витрачати на це багато часу.

## СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Java (programming language): [Електронний ресурс].

Режим доступу до ресурсу: [https://en.wikipedia.org/wiki/Java\\_\(programming\\_language\)](https://en.wikipedia.org/wiki/Java_(programming_language)).

2. Нативная или кроссплатформенная разработка – что лучше?: [Електронний ресурс].

Режим доступу до ресурсу: <http://wnfx.ru/nativnaya-ili-krossplatformennaya-razrabotka-chtoluchshe/>.

3. Офіційний сайт Expo.[Електронний ресурс].

Режим доступу: <https://docs.expo.io>

4. Nasrullah Patel . Top 5 Databases For React Native App Development. [Електронний ресурс].

Режим доступу: <https://www.business2community.com/tech-gadgets/top-5-databasesfor-react-native-app-development-02383205>

Колеснікова Наталія Сергіївна - студентка групи ІКІ-18б, факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, Вінниця, e-mail: [tixikl62@gmail.com](mailto:tixikl62@gmail.com)

Крупельницький Леонід Віталійович – доцент кафедри обчислювальної техніки Вінницького національного технічного університету, Вінниця, e-mail: [krupost@gmail.com](mailto:krupost@gmail.com)

Kolesnikova Natalia — student of the 1KI-18b group, Faculty of Information Technologies and Computer Engineering, Vinnytsa National Technical University, Vinnytsa, e-mail: [tixikl62@gmail.com](mailto:tixikl62@gmail.com)

Krupelnytsky Leonid - Associate Professor of the Department of Computer Science, Vinnytsia National Technical University , Vinnitsia, e-mail: [krupost@gmail.com](mailto:krupost@gmail.com)