

## РОЗРОБКА МОБІЛЬНОГО ДОДАТКУ ІНТЕРНЕТ-МАГАЗИНУ ЗА ДОПОМОГОЮ REACT NATIVE

Вінницький національний технічний університет

### *Анотація*

*Розроблено мобільний додаток для інтернет-магазину «Катран», метою якого є максимальна ефективність залучення відвідувачів. Проаналізовано різні підходи щодо реалізації вищевказаного додатку та обґрунтовано застосування платформи react native.*

**Ключові слова:** мобільний додаток, веб-ресурс, крос-платформність, крос-браузерність, інтерактивний додаток, React Native.

### *Abstract*

*A mobile application for the Katran online store has been developed, aiming at maximizing the effectiveness of attracting visitors. Different approaches to the implementation of the above-mentioned application are analyzed and the use of the react native platform is substantiated.*

**Key words:** mobile app, web resource, cross-platform, cross-browser, interactive applications, React Native.

Ринку сучасних мобільних додатків уже більше десяти років, однак бурхливість його розвитку в останній час набирає все більших обертів. Попит на створення мобільних додатків з боку компаній постійно зростає і він все ще помітно перевищує пропозицію, що призводить до постійного подорожчання розробок. Одним з рішень, що дозволяє знизити вартість розробки мобільних додатків є кросплатформний підхід, коли один і той же код використовується на усіх платформах.

Якщо розробники у процесі написання програми користуються прийнятою для конкретної платформи мовою програмування, такий додаток прийнято називати нативним або «рідним» [1].

Перевагами нативних додатків є [1]:

- Швидкість роботи та відгуку інтерфейсу.
- Зрозумілий і простий доступ до функцій і датчиків пристрою.
- Можливість поглибленої роботи з функціями смартфона.
- «Рідний» для платформи інтерфейс.

Але є один недолік – велика вартість розробки та підтримки. Для кожної платформи треба розробляти свій код. З розвитком ринку мобільних додатків, вартість розробки стала надто дорога. Тому, було розроблено альтернативу – створення кросплатформних додатків. Кросплатформні додатки пишуться одразу для декількох платформ на одній мові, яка відрізняється від нативної.

Є два підходи для створення кросплатформних додатків. Перший, полягає в тому, що на етапі підготовки додатка до побудови, код конвертується у нативний для деякої платформи, за допомогою транспілера. Другий підхід, полягає в тому, що до створеного коду, додається деяка обгортка, яка, працюючи вже на пристрої, на льоту транслює виклик не нативного коду, до нативних функцій системи.

До основних переваг кросплатформних додатків можна віднести такі:

- Вартість і швидкість розробки.
- У випадку наявності у штаті власного веб-розробника зникає необхідність у залученні нативних розробників, праця яких досить дорого коштує.

Основними недоліками кросплатформних додатків є такі:

- Не нативний інтерфейс або, як мінімум, необхідність роботи з інтерфейсом кожної платформи окремо.

- Проблеми у реалізації складних функцій або можливі проблеми роботи, навіть, з простими процедурами, в силу помилок самих фреймворків розробки.
- Відносно невисока швидкість функціонування.

Найпопулярнішими інструментами для створення кросплатформових додатків за методом обгортки, є Cordova, та PWA [2, 3].

Apache Cordova дозволяє створювати універсальні мобільні застосунки, що працюють на різних мобільних платформах, з використанням стандартних веб-технологій (HTML5, CSS3 і JavaScript). Використання Cordova дозволяє створювати застосунки, що функціонують на широкому спектрі мобільних платформ, включаючи Tizen, webOS, Android, Apple iOS, Blackberry, Samsung Bada і Windows Phone.

PWA також дозволяє створювати веб-додатки, але за рахунок використання певних технологій у першу чергу, це Service Worker (скрипти, що працюють у фоновому режимі), а також Web App Manifest (опис веб-додатку в зрозумілому для мобільної системи вигляді), які без обгортки з PhoneGap можуть працювати як нативні. Вони можуть встановлюватися на домашній екран в обхід магазину додатків, працювати в офлайн, працювати з пуш-повідомленнями, а також з нативними функціями.

Для створення кросплатформних додатків, за методом конвертування у нативний код, в основному використовують Xamarin, React Native та Flutter [4 – 6]. Далі коротко зупинемося на деяких особливостях цих додатків.

У Xamarin використовується стандартна для Enterprise-розробки мова програмування C#, кросплатформне середовище розробки – Visual Studio. На виході маємо нативні додатки для iOS, Android та Windows, однак, досить великого розміру.

React Native – це платформа від Facebook. Тут додатки пишуться на JavaScript і з використанням CSS-подібних стилів. Інтерфейс отримуємо рідний, а код інтерпретується вже на платформі, що надає йому потрібної гнучкості.

Flutter не перетворює вихідний код у нативний, який виконується платформою, а насправді мале вікно на екрані смартфона і промальовує усі елементи сам. У якості мови використовується «фірмовий» Dart, який Google створив як удосконалену версію JavaScript.

Основні вимоги до додатку:

- Адаптивність.
- Підтримка бази даних.
- Система авторизації.
- Можливість проведення банківських операцій.

Також доцільно зазначити, що для підвищення ефективності розроблюваного додатку можна було б скористатися підходами, запропонованими у [7].

## Висновки

Обґрунтовано актуальність розробки мобільного додатку мобільного додатку інтернет магазину «Катран» за допомогою React Native Сформульовано основні вимоги до розроблюваного додатку, враховуючі існуючі додатки аналогічного призначення.

Проаналізовано деякі технології, для створення програмного продукту та обрано React Native, оскільки завдяки своїм властивостям, ця платформа оптимізована для роботи з популярними веб-інструментами, такими як: Firebase, Google Maps тощо. Це дозволяє якісніше імплементувати дані інструменти та покращить якість роботи додатку.

## СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Нативний vs. кросплатформенний: различия подходов в разработке мобильных приложений: [Електронний ресурс]. – Режим доступу: <http://www.ironwaterstudio.com/ru/blog/nativnyj-vs-krossplatformennyj-razlichiya-podhodov-v-razrabotke-mobil-nyh-prilozhenij>
2. Progressive Web Apps: [Електронний ресурс]. – Режим доступу: <https://developers.google.com/web/progressive-web-apps/>.
3. Apache Cordova: [Електронний ресурс]. – Режим доступу: <https://cordova.apache.org/docs/en/latest/guide/overview/index.html>.

4. React Native 0.58: [Електронний ресурс]. – Режим доступу: <https://facebook.github.io/react-native/>.
5. Visual Studio Tools for Xamarin: [Електронний ресурс]. – Режим доступу: <https://visualstudio.microsoft.com/ru/xamarin/?rr=https%3A%2F%2Fru.wikipedia.org%2F>.
6. Flutter: [Електронний ресурс]. – Режим доступу: <https://flutter.dev/>.
7. Месюра В. І. Основи проектування систем штучного інтелекту. Навчальний посібник / В. І. Месюра, Л. М. Ваховська. – В.: ВДТУ, 2000. – 96 с.

*Солоний Максим Андрійович*, – ст. гр. 1КН-17мс факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, м. Вінниця, unreal9981@gmail.com

*Арсенюк Ігор Ростиславович* – к. т. н., доцент, доцент кафедри комп'ютерних наук, Вінницький національний технічний університет, м. Вінниця

*Solonyi Maksym A.* – student, 1KS-17ms, Faculty of information Technologies and Computer Engineering, Vinnytsia National Technical University, Vinnytsia, unreal9981@gmail.com

*Arsenyk Igor R.* – PhD, Faculty of Information Technologies and Computer Engineering, Vinnytsia National Technical University, Vinnytsia