***Гнатченко Дмитро Дмитрович****, старший викладач*

*Кафедра інженерії програмного забезпечення та кібербезпеки*

*Державний торговельно-економічний університет, м. Київ*

*ORCID: 0000-0002-6584-4525*

***Цівка Влада Назарівна****, здобувач вищої освіти*

*Факультет інформаційних технологій*

*Державний торговельно-економічний університет, м. Київ*

**ПОРІВНЯННЯ FLUTTER ТА KOTLIN MULTIPLATFORM: ПЕРЕВАГИ ТА НЕДОЛІКИ ДЛЯ РОЗРОБКИ МОБІЛЬНИХ ДОДАТКІВ НА JAVA**

В сучасному цифровому світі розробка мобільних додатків стала ключовим напрямком технологічного розвитку. Із зростанням популярності смартфонів та планшетів розробники активно вивчають різні платформи та інструменти для створення якісних та функціональних додатків. Один із основних виборів, з яким стикаються розробники – це вибір між нативними технологіями та кросплатформними рішеннями. Ми розглянемо два потужні інструменти для розробки мобільних додатків: Flutter та Kotlin Multiplatform, звертаючи особливу увагу на їхні переваги та недоліки для розробки додатків на мові програмування Java.

Flutter – це фреймворк інтерфейсу користувача, який був створений Google у 2017 році. Він має відкритий вихідний код і пропонує чудову документацію, а також багато посібників і відео, що доводить, що його створено з урахуванням розробників. Ключовою особливістю архітектури Flutter є власний механізм рендерингу на основі Skia. Для створення програм у Flutter потрібен Dart – мова зі статичною типізацією, яка пропонує численні функції сучасної мови програмування [1].

Kotlin Mutiplaftom – SDK, розроблений JetBrains. Вперше він був випущений у 2016 році, але все ще швидко розвивається, і до кожної версії додається багато функцій.SDK базується на мові Kotlin (також розробленій JetBrains). Kotlin сучасна статично типізована мова, яку можна скомпілювати до рідного коду платформи, Java Script або коду віртуальної машини Java. Більшість розробників Android користуються такою мовою [1].

Обидві ці технології є відносно молодими інструментами розробки, створеними для підтримки кросплатформних проектів. Кожне рішення має свій унікальний спосіб створення кросплатформних програм. Розглянемо переваги та недоліки обох підходів наведені в табл. 1.

**Таблиця 1**

Таблиця порівняння між Flutter і Kotlin Multiplatform для розробки мобільних додатків на Java

|  |  |  |
| --- | --- | --- |
| **Критерій** | **Flutter** | **Kotlin Multiplatform** |
| Мова програмування | Dart | Kotlin |
| Архітектура | Фреймворк зі своїм власним движком (C++) | Багатоплатформені бібліотеки Kotlin |
| Підтримувані платформи | Android, iOS, веб, desktop, Fuchsia | Android, iOS, backend (в тому числі Java), веб |
| Інтеграція Java | Доступ до Java через плагіни та методи моста | Повна підтримка Java на бекенді та Android |
| Взаємодія з нативним кодом | Легка взаємодія з нативними модулями через каналы | Взаємодія з нативним кодом складніша, але можлива |
| Інструментарій розробки | Інтегроване середовище розробки Android Studio, IntelliJ IDEA | Інтегроване середовище розробки IntelliJ IDEA |
| Швидкість розробки | Швидка розробка завдяки готовим бібліотекам та віджетам | Швидка розробка завдяки спільному коду на різних платформах |
| Продуктивність | Деякі обмеження щодо продуктивності та швидкості через вбудований движок | Вища продуктивність і швидкість завдяки використанню нативного коду |
| Розмір додатків | Додатки можуть бути трохи важчими через вбудований движок Flutter | Додатки можуть бути меншими та швидше завдяки використанню нативного коду |
| Підтримка платформи | Один код для багатьох платформ (Android, iOS, веб і т. д.) | Один код для Android та iOS, інші платформи потребують окремого коду |

*Джерело: розроблено автором на основі [2* , ст. 24-25, 34-37*]*

Обираючи між Flutter і Kotlin Multiplatform для розробки мобільних додатків на Java, розробники повинні враховувати свої потреби, ресурси та вимоги проєкту. Обидва підходи мають свої переваги та недоліки, і вибір залежить від конкретного контексту розробки. Важливо враховувати, що обидва ці підходи пропонують засоби для створення якісних мобільних додатків на Java, і кожен може бути вигідним для певних випадків використання.

**Література**

1. Lewandowski M. Kotlin Multiplatform vs Flutter. Which Is Better for Your App? 2023. URL: <https://www.netguru.com/blog/kotlin-multiplatform-vs-flutter> (дата звернення: 27.10.2023).

2. Будник В. В. Методи та засоби розробки мобільних застосунків. 2021. С. 24-25, 34-37. URL: <https://openarchive.nure.ua/server/api/core/bitstreams/5558deb6-3327-4940-9231-6d82b5e713f9/content> (дата звернення: 27.10.2023).