

ОГЛЯД МОЖЛИВОСТЕЙ KOTLIN ТА JAVA ДЛЯ РОЗРОБКИ ANDROID ДОДАТКУ

Вінницький національний технічний університет

Анотація

В роботі проведено аналіз мов програмування Kotlin та Java для розробки Android додатку. Наведено основні переваги та недоліки даних мов програмування для Android розробки.

Ключові слова: Kotlin, Java, мови програмування, Android, API, мобільний додаток

Abstract

In the article was spent the analysis of main programming languages Kotlin and Java for developing Android application. It's given the basic advantages and disadvantages of Kotlin and Java for Android development.

Keywords: Kotlin, Java, programming languages, Android, API, mobile application.

Наразі мобільні пристрої (смартфони, планшети) стають найбільш перспективним каналом комунікації та засобом оптимізації бізнес-процесів. Згідно з останніми дослідженнями Strategy Analytics, у третьому кварталі 2016 року обсяг поставок світових смартфонів досягав 375 мільйонів одиниць. Ринкова частка Android останні два роки тримається у межах 80–88% [1]. Іншими словами, приблизно дев'ять з десяти пристроїв використовують саме цю ОС, тому актуальним для розробника програмного забезпечення під Android стає обрання інструментів.

Java – сильно типізована об'єктно-орієнтована мова програмування, розроблена компанією Sun Microsystems. Програми Java зазвичай транслюються в спеціальний байт-код, тому вони можуть працювати на будь-якої комп'ютерної архітектурі за допомогою віртуальної Java-машини [2]. Більшість розробників надають перевагу Java при написанні Android додатків, основується на рекомендаціях Google та на тому, що сам Android був реалізований на Java. Дана мова програмування має як свої переваги, так і недоліки. До переваг можемо віднести простоту та зрозумілість; кроссплатформеність; наявність у Android великої кількості вбудованих бібліотек Java; прискорена побудова за допомогою Gradle. До недоліків можна віднести конфліктність між Java та архітектурою Android API; ймовірність виникнення великої кількості багів за рахунок докладності мови (verbose language Java); Java використовує багато пам'яті, тому потребує більше часу для виконання.

Kotlin – статично типізована мова програмування, що працює поверх JVM і розроблений компанією JetBrains[3]. Kotlin можна компілювати в JavaScript або в Native для запуску на iOS платформі. Основними перевагами мови Kotlin є її бурхливий розвиток; легкий перехід з Java на Kotlin – потрібно просто встановити плагін Kotlin та їх сумісність; наявність extension functions для розробки чистих API; наявність null в системі типів; Kotlin лаконічна, що зменшує кількість помилок. Проте існують і певні недоліки: Kotlin має меншу швидкість компілювання; середовища розробки (наприклад, Android Studio) працює повільніше із Kotlin.

Продуктивність є одним з найважливіших аспектів, що необхідно враховувати при виборі мови програмування. Структура байт-коду Kotlin майже ідентична структурі Java, що робить додатки так само швидкими. Однак, Kotlin може підтримувати вбудовані функції, які дозволяють коду, який містить лямбда вирази, працювати швидше, ніж той, який написано на Java. Клас, що складається з 50 рядків коду при використанні Java, може бути написаний тільки одним рядком Kotlin [4].

Наведені вище переваги та недоліки демонструють ефективність використання Kotlin, яка вимагає меншої кількості ліній коду, що робить її використання безпечнішим в порівнянні із Java. Завдяки цьому покращується читабельність коду, ефективність його обробки, знижується ймовірність виникнення помилок при написанні коду ще на етапі компіляції. Для демонстрації роботи розроблено

модулі програмне забезпечення для API «Відображення криптовалют». У таблиці 1 зображено блоки програмного коду на Java та Kotlin .

Таблиця 1 – Приклад програмного коду модуля «Отримання даних із сервера»

Програмний код мовою Java	Програмний код мовою Kotlin
<pre> @SerializedName("id") @Expose private String id; @SerializedName("name") @Expose private String name; @SerializedName("symbol") @Expose private String symbol; @SerializedName("rank") @Expose private String rank; @SerializedName("priceEth") @Expose private String priceEth; @SerializedName("priceBtc") @Expose private String priceBtc; @SerializedName("volume") @Expose private String volume; public String getId() { return id;} public void setId(String id) { this.id = id;} public String getName() { return name;} public void setName(String name) { this.name = name;} public String getSymbol() { return symbol;} public void setSymbol(String symbol) { this.symbol = symbol;} public String getRank() { return rank;} public void setRank(String rank) { this.rank = rank;} public String getPriceEth() { return priceEth;} public void setPriceEth(String priceEth) { this.priceEth = priceEth;} public String getPriceBtc() { return priceBtc;} public void setPriceBtc(String priceBtc) { this.priceBtc = priceBtc;} public String getVolume() { return volume;} public void setVolume(String volume) { this.volume = volume;} </pre>	<pre> data class Item(@SerializedName("id") @Expose var id: String?, @SerializedName("name") @Expose var name: String?, @SerializedName("symbol") @Expose var symbol: String?, @SerializedName("rank") @Expose var rank: String?, @SerializedName("priceEth") @Expose var priceEth: String?, @SerializedName("priceBtc") @Expose var priceBtc: String?, @SerializedName("volume") @Expose var volume: String?, </pre>

Висновки. У більшості ситуацій різниця між цими двома мовами є очевидною. Kotlin здатна забезпечити ефективність розробки та знизити ризики, усуваючи непотрібні рядки коду. У сучасному IT-середовищі компанії постійно шукають шляхи підвищення ефективності, але часто вони не беруть до уваги перехід на нову мову програмування, як засіб для досягнення цієї мети. Хоча Kotlin пропонує багато переваг, які Java не може реалізувати, все одно існують деякі недоліки, проте він має величезний потенціал, як засіб розробки під Android. Так, на конференції Google I/O 2017 команда Android оголосила про офіційну підтримку мови Kotlin для розробки Android-додатків.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Загальна частка ринку мобільних ОС у продажах кінцевим споживачам з 1 кварталу 2009 року до 2-го кварталу 2017 року [Електронний ресурс] – Режим доступу до ресурсу: <https://www.statista.com/statistics/266136/global-market-share-held-by-smartphoneoperating-systems/>.
2. Сьерра К. Изучаем Java; пер. с англ. / Кэти Сьерра, Берт Бэйтс. – М. : Эксмо, 2013. – 720 с.
3. Жемеров Д., Исакова С. Kotlin в действии. / пер. с англ. / Киселев А. Н. – М.: ДМК Пресс, 2018. – 402с.
4. Kotlin vs Java: What to Choose for Android Development [Електронний ресурс]. Режим доступу : <https://medium.com/@skywellssoftware/kotlin-vs-java-what-to-choose-for-android-development-89674760988a>

Науковий керівник: **Лісовенко Анна Ігорівна** – к.т.н., асистент каф. АІТ, факультет комп'ютерних систем та автоматики, Вінницький національний технічний університет, м. Вінниця.

Вовк Максим Федорович – студент групи 2СІ-15б, факультет комп'ютерних систем та автоматики, Вінницький національний технічний університет, м. Вінниця, makswowk3d@gmail.com.

Supervisor: **Lisovenko Anna I.** – Ph'd, assistant to department of Automation and Intellectual Information Technologies, Faculty for Computer Systems and Automation, Vinnytsia National Technical University, Vinnytsia.

Vovk Maksym F. - student of group 2SI-15b of Computer Systems and Automation Faculty, Vinnitsa National Technical University, makswowk3d@gmail.com.