

ПОРІВНЯЛЬНИЙ АНАЛІЗ ANDROID ТА IOS

М. О. Янков, *ст. гр. CI-14,*

О. Є. Тесленко, *асистент каф. ПКСМ*

Центральноукраїнський національний технічний університет

Кожен студент-програміст після закінчення університету стикається з проблемою вибору подальшого шляху для розвитку. Даний аналіз дозволить мати уявлення про речі з якими доведеться зіткнутися в міру розвитку від стажиста до справжнього розробника в сфері створення мобільних додатків. Будуть порушені такі теми як поріг входження в розробку, основні підходи, прибутковість кожної з платформ. Ключові слова: Android - операційна система для смартфонів, планшетів, наручних годинників і інших електронних пристроїв. Заснована на ядрі Linux і власної реалізації JVM від Google. iOS - мобільна операційна система для смартфонів, планшетів, мобільних медіа програвачів. В iOS використовується ядро XNU засноване на ядрі Mach, а також код з ОС NeXTSTEP і FreeBSD. Java - сильно типізований об'єктно-орієнтована мова програмування. Програми Java транслюються в спеціальний байт-код, тому вони можуть працювати на будь-якій комп'ютерній архітектурі за допомогою JVM Kotlin - статично типізований мові програмування, що працює поверх JVM і розробляється компанією JetBrains. Компілюється також в JavaScript та інші платформи через структуру LLVM Swift - відкрита мультипарадигмальна компільована мова програмування. Сумісна з основною кодовою базою Apple написаною на Objective C. Objective C - компільована об'єктно-орієнтована мова програмування.

Основною проблемою початку розробки на iOS є наявність комп'ютера який працював би з macOS X, та хоча-б одного iOS пристрою, з підпискою на Apple Developer Program. Якщо здебільше ПК необхідний лише для запуску IDE XCode - середовища розробки продукції, що випускається Apple, то iOS пристрій необхідний для тестування і налагодження, підпису додатку та його подальшого випуску в App Store. Середня ціна ПК від компанії Apple - 60 000 грн, в той час, як ціна на їх пристрої коливається біля позначки в 20 000 грн. Це задорого, як для старту поодинці. З Android ситуація набагато простіше, не потрібно мати ніяких пристроїв, емулятор Android цілком справляється з усім необхідним, випускати додатки можна теж не маючи девайса, а ціна Developer Account by Google - всього 25\$. Більша частина витрат піде на ПК. ОС, хоч Windows, хоч macOS, хоч Linux - Android Studio, працює на цих платформах. Це було вагомим аргументом для старту, тому я і вибрав цю платформу.

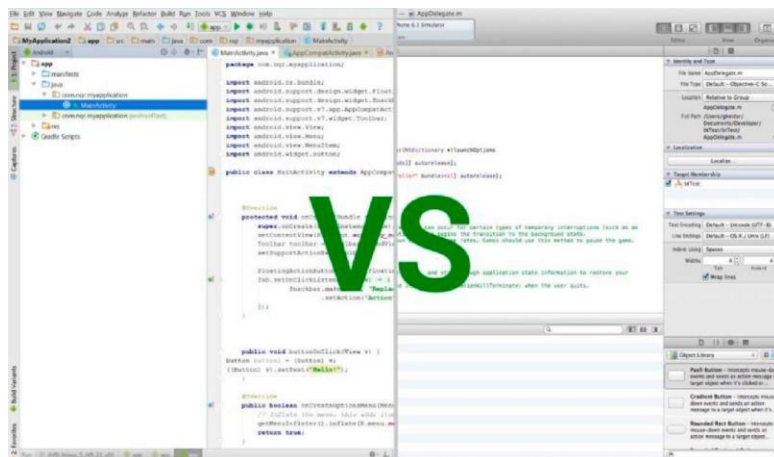


Рисунок 1 - Зовнішній вигляд IDE Android Studio vs Xcode.

IDE Android Studio vs Xcode. Починається будь-яка розробка програмного забезпечення із запуску IDE. У разі Android Studio це буде який-небудь шаблонний файл з кодом, чого не скажеш про XCode. Після екрану вітання з'явиться дуже зручний візуальний редактор, безліч всяких налаштувань в фірмовому стилі інтерфейсу від Apple, але як тільки справа починає стосуватися безпосередньо коду - XCode забирається в куток і дивиться за тим як Android Studio не залишає йому жодного шансу. Призначені для користувача настройки форматування коду, розумний автокомпліт, безліч інструментів налагодження - це все не про Apple. Так, у них теж є відладчик - який розбере вам ваш код аж до мови Асемблера і видасть пачку помилок в хексовому форматі. У перший раз це може досить сильно налякати.

Кодова база. Отже, з IDE розібралися, перед нами завдання - написати простий новинний додаток. Терміни мінімальні, бюджет урізаний, дизайн - стандартний для операційної системи. Потрібно вибрати технології які ми будемо для цього використовувати, перше з чим необхідно визначитися - мова програмування. Android підтримує розробку додатків на таких мовах: Java, Kotlin, C / C ++. Останні два необхідні лише в крайніх випадках, коли швидкість роботи критичніше від часу розробки і вартості роботи, тому вони відразу відпадають. Вибір належить робити між Java і Kotlin. Java це мова яка безліч років на ринку, вона вже довела що не дарма вважається однією з найбезпечніших і відмовостійких. У свою чергу Kotlin - це досить молода мова, яка компілюється в Java, розширює її можливості, додає функціональну парадигму, повністю з нею сумісна і має безліч нововведень, які полюбилися програмістам по всьому світу. Особисто я вперше побачив її в 2016, і поставився до неї скептично, зараз же я можу заявити що це по праву одна з кращих мов програмування. Втім, якщо ви любите C-синтаксис і маєте досить консервативні погляди - ніщо не заважає вибрати вам Java. Швидкість роботи додатків не зміниться, а на такому дрібному проєкті швидкість розробки не буде перевагою Kotlin. Що ж стосується iOS - тут ви можете вибрати Swift / Objective-C / C. Чиста Сі практично не використовується - вона необхідна лише в ситуації коли вам потрібно оновити ПЗ для, наприклад, iPhone 3GS родом з далекого 2009-го. Вибір стоїть між Swift і Objective C. Велика частина ПО екосистеми Apple написана на Objective C - дуже дорослій мові, вона розширює собою чисту C, внаслідок чого працює так само швидко, і також ненадійно. Swift в свою чергу дуже молода мова, сумісна зі своєю попередницею - Objective C, створена спеціально для того щоб прибрати головний біль всіх розробників - боротьбу з NullPointer і націлена на посилення загальної відмовостійкості. Вибір очевидний - це Swift.

Документація. Приступати до роботи не знаючи як працює платформа, все-одно що вийти у відкритий океан на плоту - якийсь час ви протримаєтесь, але вас погубить відсутність запасів ресурсів, хвилі, і інші фактори. Так як мови які ми вибрали - досить молоді і розвиваються не по роках, а по годинах - знайти хорошу літературу це велика проблема, і тут нам допоможуть самі їх творці. На офіційних сайтах Android і Apple є безліч навчальних матеріалів, роликів з конференцій, прикладів коду та іншого. Крім усього цього - Apple подбала про своїх розробників і включила повну документацію найпопулярніших фреймворків власного виробництва безпосередньо в саму XCode. Шкода що вони забули зробити її зрозумілою для пересічного користувача - таким чином доведеться витратити в рази більше часу на освоєння iOS ніж Android. Але це компенсується структурою ОС. Нижче наведені життєві цикли програм на Android і iOS відповідно.

Дизайн. Якщо особливих вимог до дизайну немає - можна користуватися стандартними елементами. Обидві платформи подбали про це. XCode з коробки має все необхідне прямо на сторінці візуального редактора - елементи, анімації, переходи - все робиться в кілька кліків. З Android ситуація трохи гірше - необхідно встановити окрему бібліотеку (якщо раптом у вас немає інтернету - вам вже нічого не допоможе), а краще відразу дві - Support Design Library v4 і Support Design Library v7. Вони додадуть вам нові елементи для візуального відображення, але боротися з анімаціями і писати логіку переходів вам доведеться самостійно. Не варто забувати про те що, роблячи додаток для iOS - ви

завжди маєте на меті останню її стабільну версію для пари-трійки пристроїв, в той час як для Android вам потрібно думати про десятки тисяч пристроїв, версії ОС на яких - різні, вендори (виробники) - різні, реалізація ОС теж різна. Не кажучи вже про різні розмірах екрану і відсутності будь-яких стандартів. Фрагментованість: У лінійці iPhone є досі потужні та зручні смартфони з невеликими екранами. Це і чотиридюймовим iPhone SE, і більш пізні iPhone 6s, 7 і 8 з діагоналлю екрана 4,7 дюйма. Знайти ж маленький і потужний Android-смартфон - це проблема. iOS - це як мінімум чотири роки оновлень операційної системи. Фанати Android ворожать, чи вийде для їх смартфонів хоча б наступна версія платформи, а власники яблучних пристроїв мають це як функцію. Наприклад, новітню iOS 11, випущену в вересні 2017 року, можна встановити навіть на iPhone 5s. А він вийшов в вересня 2013. Для порівняння - "еталонний" смартфон Google Nexus 5 того ж року випуску оновився в жовтні 2016 і на цьому все. А деякі виробники Android-смартфонів можуть взагалі ні разу не оновити платформу. Ніяких нових функцій сервісів, ніяких оновлень безпеки.

Популярність. iOS попереду всіх, Android наздоганяє. Порівняйте кількість додатків для систем: • iOS -1.2 мільйона. • Android -1.2 мільйона. За кількістю бачимо паритет, а ось що стосується якості ... iOS завжди був найзручнішою платформою для розробників, а з огляду на нелюбов Apple до піратів, не дивно, що саме тут в першу чергу виходили майбутні хіти серед ігор і додатків. І тільки потім вони переїжджали на Android. Однак Android продовжує рости і намагається взяти своє за рахунок безкоштовних додатків. Та й ексклюзивного софту для системи останнім часом виходить куди більше.

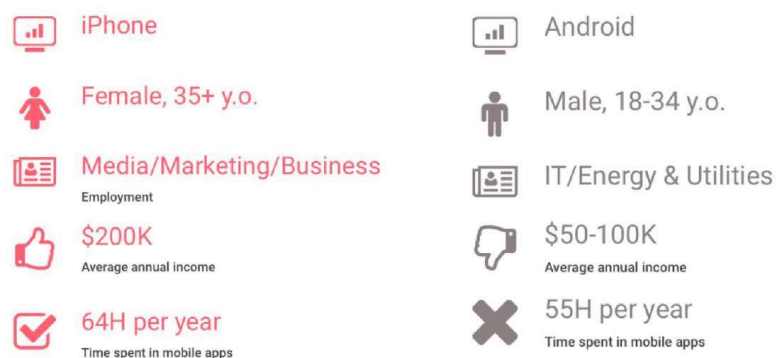


Рисунок 2 - Опосередковані споживчі дані для кожної з платформ:

Висновок. У обох платформ є безліч "за" і "проти", проте вибір доведеться робити, не варто забувати - щоб домогтися рівних умов вам необхідно працювати в компанії де немає обмеження у виборі пристроїв, а значить грошове питання не буде проблемою. На стороні Android хмарні сервіси, безліч безкоштовних додатків, лояльне ставлення з боку розробника і неймовірно широкий вибір комплектуючих. У цьому криються і його слабкості-найчастіше розрив між бюджетними телефонами і флагманами підживлює стереотип про те, що платформа «гальмує» і «Android незручний». Дивлячись на «перемоги» iOS, платформа залишає враження більш цілісної, тут значна бібліотека корисних додатків, хороша базова ПО. Також на користь Apple говорить дуже серйозне ставлення до безпеки пристроїв і оновлень прошивки. Стартувати набагато швидше вийде на iOS, ви витратите менше часу на ознайомлення з усіма її особливостями ніж на вивчення Android. Перший прибуток також вийде набагато швидше отримати розробляючи під iOS. Проте вам буде необхідно затратити в рази менше ресурсів для розробки на Android. Внаслідок чого можна отримати паритет нерівнозначних. Android перемагає як платформа для старту у розробників-одинаків. iOS ж не залишає шансів своєму суперникові як платформа для команд з уже налагодженим бюджетом.