

Міністерство освіти і науки України
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «КИЄВО-МОГИЛЯНСЬКА АКАДЕМІЯ»
Кафедра мультимедійних систем факультету інформатики

Розробка Android-додатку для тренування та вдосконалення пам'яті

Текстова частина до курсової роботи

за спеціальністю „Комп'ютерні науки” 121

Керівник курсової роботи

Старший викладач

Борозенний С.О.

(підпис)

“ ____ ” _____ 2021 р.

Виконав студент БП КН-4

Ємельянов О.А

“ ____ ” _____ 2021 р.

Міністерство освіти і науки України
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «КИЄВО-МОГИЛЯНСЬКА АКАДЕМІЯ»
Кафедра мультимедійних систем факультету інформатики

ЗАТВЕРДЖУЮ

Зав. кафедри мультимедійних систем

Доцент., к. ф.-м. н. О.П. Жежерун

(підпис)

“ ____ ” _____ 2021 р.

ІНДИВІДУАЛЬНЕ ЗАВДАННЯ

на курсову роботу

студенту Ємельянову Олександрю Анатолійовичу факультету інформатики 4-го курсу

ТЕМА: Розробка Android-додатку для тренування та вдосконалення пам'яті

Зміст ТЧ до курсової роботи:

Дата видачі “ ____ ” _____ 2020 р. Керівник _____

(підпис)

Завдання отримав _____

(підпис)

Календарний план виконання роботи

№	Назва етапу курсової	Термін виконання	Примітка
1.	Отримання теми курсової роботи	05.10.2020	
2.	Ознайомлення з предметною областю	12.10.2020	
3.	Пошук корисної літератури	19.10.2020	
4.	Ознайомлення з літературою	22.10.2020	
5.	Планування архітектури практичної частини роботи	19.01.2021	
6.	Розробка зручного дизайну застосунку	01.02.2021	
7.	Розробка «тренажера» для тренування	15.02.2021	
8.	Створення функціоналу авторизації	25.02.2021	
9.	Створення функціоналу налаштувань застосунку	03.03.2021	
10.	Створення функціоналу можливості push-повідомлень	12.03.2021	
11.	Рефакторинг коду	16.03.2021	
12.	Написання текстової частини курсової роботи	29.03.2021	
13.	Створення презентації та доповіді до неї	07.04.2021	
14.	Здача курсової роботи на перевірку	12.04.2021	

Зміст

Анотація.....	5
Вступ.....	6
Розділ 1. Аналіз предметної області та постановка задачі.....	8
1.1 Аналіз проблеми тренування мозку та вдосконалення пам'яті.....	8
1.2 Аналіз наявних додатків, які допомагають тренувати пам'ять.....	9
Розділ 2. Теоретична інформація про використані технології.....	11
2.1 Особливості розробки додатків для ОС Android.....	11
2.2 Загальні відомості про мову програмування Kotlin.....	17
Розділ 3. Реалізація застосунку.....	19
3.1 Опис та огляд технічного завдання.....	19
3.2 Обґрунтування вибраного середовища розробки.....	19
3.3 Обґрунтування та реалізація гри «Додавання».....	22
3.4 Екран налаштувань.....	24
3.5 Загальний опис додатку.....	26
Висновок та можливість подальшого вдосконалення додатку.....	28
Список використаних джерел.....	29

Анотація

Метою даної курсової роботи є створення зручного мобільного застосунку для тренування і вдосконалення роботи мозку та пам'яті для людей, котрі не хочуть або не можуть витратити на це багато часу.

Проведено аналіз схожих мобільних застосунків та досліджено використання мобільних пристроїв людьми. Також було враховано основні проблеми людей та побажання потенційних користувачів щодо інтерфейсу та функцій майбутнього мобільного застосунку.

Застосунок розроблений за допомогою інтегрованого середовища розробки Android Studio, мова програмування: Kotlin

Ключові слова: розробка застосунку, мобільні технології, швидкість, ефективність, Android, Kotlin

Вступ

Актуальність теми:

Мобільні телефони вже дуже багато років відіграють велику роль в нашому житті. Вони перестали бути чимось незвичним та відмінно справляються зі своєю головною функцією – завжди залишатись на зв'язку. Але не тільки цією функцією обмежується використання смартфона. Наразі із впевненістю можна сказати, що мобільні телефони – це міні-комп'ютери, які завжди можна носити з собою. Вони настільки функціональні, що важко уявити, що вони ще не вміють робити. Зараз смартфон – це і фотоапарат, і навігатор, і музичний плеєр, і можливість доступу до Інтернет-ресурсів, тощо.

За даними компанії Google, кількість активних користувачів Android-пристроїв становить більше 2,5 мільярдів[1], що значно більше ніж користувачів смартфонів на базі iOS, котрих налічується 1 мільярд. Така популярність зумовлена тим, що телефони із операційною системою Android виробляються різними виробниками і відповідно цінова політика теж різна. Іншою причиною популярності є те, що Android дуже відкритий в плані засобів розробки програмного забезпечення. Засоби розробки для Android є безкоштовними. Окрім всього вищеперерахованого для зручної та повноцінної розробки Android-застосунків є велика кількість безкоштовних бібліотек, які полегшують роботу зі сторонніми ресурсами – Retrofit, Room, RxJava, Dagger, Android KTX, Timber, ThreeTen[10], в той час як для Windows Phone Mobile такі бібліотеки взагалі не розповсюджені.

Згідно з даними різних досліджень середньостатистична людина у віці від 15 до 30 років заглядає у свій смартфон кожні 4-7 хвилин, або ж 150-300 разів на день[11]. З цього можна зробити висновок, що люди досить залежні від своїх гаджетів, вони не пропускають можливість перевірити свою електронну пошту, погортати стрічку в тій чи іншій соціальній мережі, тощо.

Проте з іншого боку збільшення кількості гаджетів позитивно впливає на їх популяризацію, що в свою чергу пришвидшує появу нових технологій та застосунків. Компанії стараються розробити все нові і нові застосунки та привернути увагу якнайбільшої кількості користувачів саме до своїх програмних продуктів

Мета та завдання дослідження:

Спроекувати та розробити мобільний додаток для ОС Android, який допоможе людям вдосконалювати та розвивати пам'ять не витрачаючи на це багато часу.

Об'єкт дослідження:

Операційна система – Android, мова програмування – Kotlin, мова розмітки – XML, інтегроване середовище розробки – Android Studio.

Предмет дослідження:

Функції та можливості, які надають мова програмування Kotlin та ОС Android під час реалізації та розробки додатку.

Логіка досліджень зумовила таку структуру роботи: анотація, вступ, 3 розділи, висновок, список використаної літератури та практична частина у вигляді застосунку. Перший розділ – аналіз предметної області, розгляд можливих варіантів вирішення проблеми тренування та вдосконалення пам'яті. Другий розділ включає в себе опис технологій, які були використані при розробці застосунку. Третій розділ містить опис роботи з готовим ПП та інструкція користувача.

Розділ 1. Аналіз предметної області та постановка задачі

1.1 Аналіз проблеми тренування мозку та вдосконалення пам'яті

Мозок потребує регулярного тренування не менше ніж наші м'язи. Чим старішою стає людина, тим важливіше тримати мозок в тонусі. Розумове навантаження стимулює роботу мозку та допомагає його розвивати та тренувати, збільшує ймовірність краще запам'ятовувати інформацію. Також тренування мозку зміцнює нейронні зв'язки[12]. Поєднання розумового та фізичного навантаження допомагає збільшити та вдосконалити здібності людського мозку.

Існує багато різних вправ та завдань, котрі допомагають покращити та вдосконалити когнітивні навички мозку. Деякі з них це розвиваючі ігри, гра на музичних інструментах, спілкування, тощо. Ще один хороший спосіб це різні мобільні застосунки, які тренують пам'ять і реакцію.

Втрата пам'яті та її порушення призвели до появи нових способів її вдосконалення, лікування та покращення. Проблеми із запам'ятовуванням інформації, швидкістю реакції може проявлятися в будь-якому віці. Інколи це дрібниці наприклад, людина може забути ім'я нової людини в компанії чи номер свого автомобіля, а інколи це проявляється в тому, що хтось забуває продукти, які потрібно купити в супермаркеті чи справи, котрі потрібно виконати наступного дня.

В наш час кількість та різноманітність засобів для тренування та вдосконалення пам'яті справді вражає. Це і різні онлайн-курси, персональні заняття, освітні програми, ресурси, тощо. Проте це все займає багато часу, а інколи і коштів. В час, коли кожна хвилина на вагу золота, слід мати під рукою зручний засіб, котрий допоможе потренувати пам'ять та швидкість реакції навіть якщо витратити на це 5 хвилин. Тому застосунок в телефоні є найкращим рішенням такої проблеми.

1.2 Аналіз наявних додатків, які допомагають тренувати пам'ять

Щоб краще зрозуміти шлях та можливість вирішення проблеми потрібно проаналізувати вже наявні рішення. Розуміючи, які смартфони популярні та доступні, а також те, що майже кожен в наш час має гаджет, велика кількість компаній починають розроблювати додатки, які тренують мозок. Смартфони досить компактні і майже всі користувачі носять їх з собою цілий день. Це дозволяє гаджетам бути корисними і відкрити улюблений додаток для в будь-який момент, коли користувач має час.

Наразі в Google Play Market існує багато різних застосунків для покращення пам'яті[13]. Вони відрізняються іграми, завданнями, дизайном. Є платні, також є і безкоштовні. Наприклад, NeuroNation розроблений для покращення пам'яті, інтелектуальних здібностей, концентрації, уваги та логічного мислення. Цей додаток дозволяє в ігровій формі перевірити рівень своїх здібностей, і на його основі пропонує вправи та персоналізовані уроки. Також він аналізує слабкі та сильні сторони користувача і показує графіки прогресу.

У додатку Brain Wars можна тренувати мозок у дуелях, які відбуваються в реальному часі з іншими користувачами з усього світу. Короткі ігри націлені на розвиток уважності, пам'яті, навичок з рахування тощо. Після кожного раунду додаток показує, які когнітивні здібності користувача розвинені краще, а над якими ще треба попрацювати. Додаток має систему балів, заохочень і бонусів для переможців. Також він дає можливість спостерігати за статистикою власних перемог та розвитку. А дуелі можна влаштовувати не тільки з випадковими користувачами, але й зі своїми друзями.[2]

Elevate – Brain Training тренує розумові та математичні здібності, концентрацію, швидкість мислення та навіть ораторські навички. До речі, додаток буде корисний для тих, хто погано запам'ятовує імена людей або ж має проблеми з концентрацією під час розмови. Кожен користувач отримує

індивідуальну програму тренувань залежно від рівня та бажаного результату. Щоразу додаток пропонує різноманітні тренування та поступово їх ускладнює. А після кожного з них показує аналітику у вигляді графіку.[2]

Розробники переконують, що в користувачів, які займаються з додатком хоча б 3 рази на тиждень, значно покращуються розумові здібності та підвищується впевненість у своїх силах. Втім, аби використовувати його ефективно, потрібно добре володіти англійською.[2]

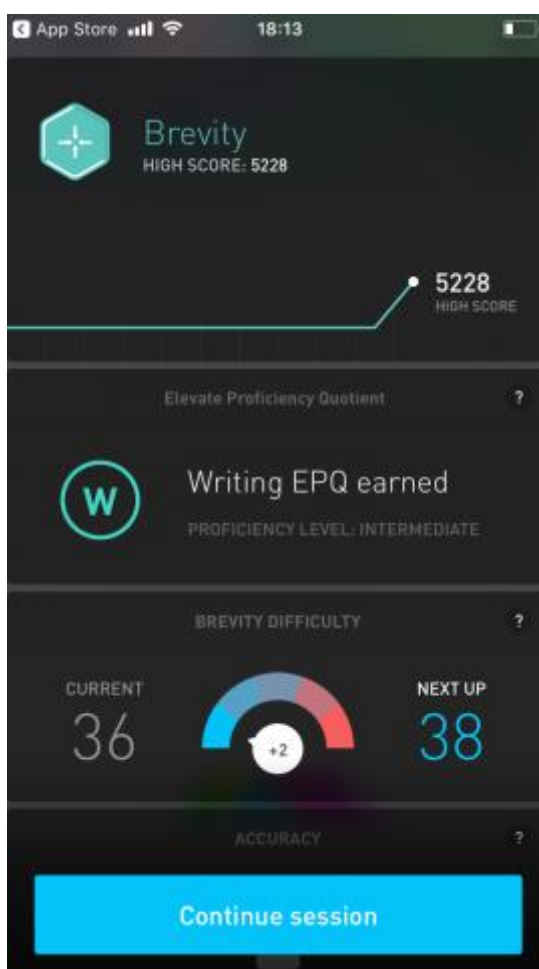


Рис. 1

Розділ 2. Теоретична інформація про використані технології

2.1 Особливості розробки додатків для ОС Android

Мобільний додаток – це програмне забезпечення, яке розробляється для роботи в умовах мобільної операційної системи. Вони можуть працювати як тільки на одній ОС (Android, iOS, тощо), так і на декількох відразу, такі застосунки називаються кросплатформними. Немає сумніву, що Android сьогодні є найпопулярнішою та найпоширенішою операційною системою для мобільних пристроїв. Адже більше 71% телефонів у всьому світі працюють саме на цій операційній системі[14]. ОС Android оточує нас всюди – це наші телефони і годинники, програмне забезпечення в автомобілі і в камерах спостереження, вдома і в офісі.

Зазвичай виділяють 2 види мобільних додатків:

- гібридні – це мобільні застосунки, які в більшості випадків зберігають свої дані на сервері і отримують доступ до API операційної системи гаджета. Цей додаток зберігає залежність від основного сайту, але з іншого боку він може використовувати можливості смартфона та бути розміщеним в Play Market чи App Store.
- нативні – це додатки, які розробляються виключно для певної мобільної платформи і можуть використовувати всі її можливості. Розробка такого програмного продукту вимагає більше витрат, як фінансових так і технічних та розумових. Але на виході замовник отримує готовий продукт, який більш продуктивний, швидкий та має добре продуманий інтерфейс користувача.

Залежно від вимог замовника чи користувачів розробник вже обирає, який застосунок створювати. Чи це гібридний додаток, котрий буде базуватись на певному сайті, чи це нативний додаток, який буде розроблений з нуля і виконуватиме всі, потрібні користувачам, функції.

Переваги розробки мобільних додатків для ОС Android:

- **доступність вихідних кодів.** Багато ресурсів для розробки Android-додатків знаходяться у вільному доступі. Будь-хто за необхідністю може завантажити собі фрагмент певного коду і почати розбиратись як працює та чи інша функція, модуль, тощо.
- **відносно низька вартість пристроїв.** Розробляти додатки для Android можна на будь-якій ОС: Windows, Linux чи macOS. Також самі смартфони коштують досить дешево і якщо розробнику для тестування свого програмного продукту потрібні девайси з різним розширенням екрану, діагоналлю чи версією Android він за невеликі кошти може купити декілька телефонів з різними характеристиками.
- **завантаження готового продукту в Play Market.** Відразу після натиснення кнопки «Опублікувати» в Google Play Market програмний продукт стає видимим для всіх користувачів, чиї девайси підходять під вимоги застосунку. В той час як публікація готового ПП в AppStore займає декілька тижнів. Також вартість публікації в Play Market є досить демократичною. Розробник платить всього 25\$ за реєстрацію. AppStore же вимагає 100\$ щомісяця за розміщений додаток.
- **технології та мова розробки.** Ще декілька років тому головною мовою для розробки Android-застосунків була Java, проте в 2011 році JetBrains представила нову мову Kotlin, котра зараз є основною мовою розробки

додатків для ОС Android. Kotlin цілком сумісний з Java, що дає можливість впроваджувати його в додатки, які написані мовою Java. На мові Kotlin пишуть не тільки мобільні додатки. Цією мовою можна розробляти клієнт-серверні застосунки, Kotlin також може компілюватись в JavaScript, що є його великим плюсом. Стек технологій, які потрібні для початку роботи в сфері розробки Android майже не змінюється від проєкту до проєкту[3]. За необхідності розробник має можливість без великих проблем змінити напрям роботи, так як було вищесказано Kotlin і Java використовуються не тільки для розробки мобільних додатків. В той час як Swift та Object-C – мови, які використовуються для розробки тільки iOS продуктів.

- **сфера розробки.** Програмісти, які працюють на посаді Android-розробник створюють застосунки не тільки для смартфонів чи планшетів. Вони можуть розробляти продукти для SmartTV, автомобілів, розумних годинників і всіх пристроїв, які працюють на ОС Android, а також працювати з AV та VR технологіями:

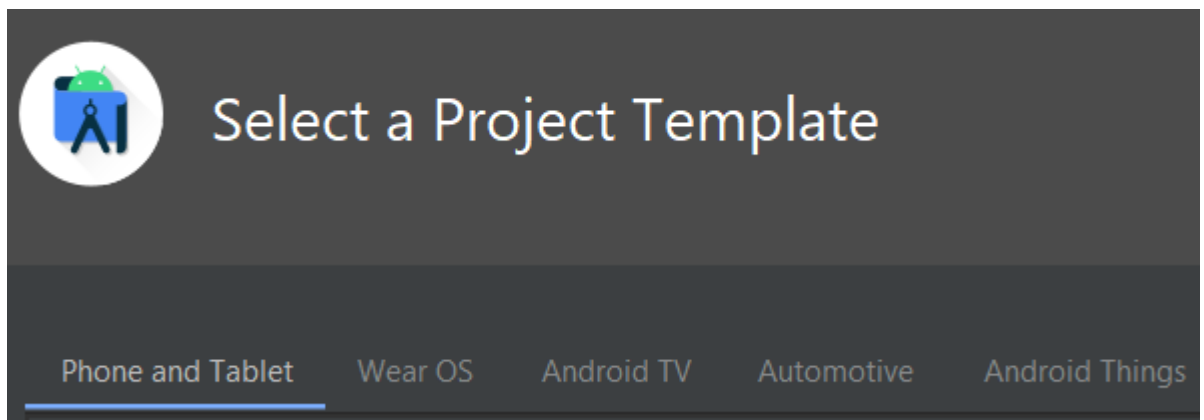


Рис.2

Не зважаючи на всі перераховані переваги та плюси розробки застосунків розробка для Android має свої недоліки:

- **велика кількість пристроїв.** Вище було згадано, що багато простоїв – це плюс, але якщо поглянути з іншого боку, то такий великий вибір досить різних пристроїв є недоліком. В цьому випадку з’являється необхідність підтримувати додаток для дуже великою кількості девайсів з різними характеристиками і що складніше від різних виробників. Адже смартфони різних виробників відрізняються своєю внутрішньою системою. Може бути таке, що на не дуже якісних телефонах, які вироблені, наприклад в Китаї, додаток може не працювати не через помилку в коді, а через проблеми в самому телефоні.
- **значна кількість версій ОС Android.** Ще одним недоліком великої кількості різних девайсів є те, що вони працюють на різних версіях Android. Google не оновлює операційні системи на всіх смартфонах, так як це робить Apple: iOS оновлюється кожного року і попередні версії не отримують підтримки. Актуальними версіями залишаються передостання та остання. ОС Android оновлюють виробники телефонів, Google тільки розробляє та постачає ці оновлення. Таким чином актуальними версіями залишаються відразу декілька. Це спричиняє багато труднощів для розробників, адже версії відрізняються між собою і не у всіх версіях одна і та ж функція буде працювати так як вона задумувалась від самого початку. Бувають випадки, коли в певній версії взагалі немає потрібної функції.

ANDROID PLATFORM VERSION	API LEVEL	CUMULATIVE DISTRIBUTION
4.0 Ice Cream Sandwich	15	
4.1 Jelly Bean	16	99,8%
4.2 Jelly Bean	17	99,2%
4.3 Jelly Bean	18	98,4%
4.4 KitKat	19	98,1%
5.0 Lollipop	21	94,1%
5.1 Lollipop	22	92,3%
6.0 Marshmallow	23	84,9%
7.0 Nougat	24	73,7%
7.1 Nougat	25	66,2%
8.0 Oreo	26	60,8%
8.1 Oreo	27	53,5%
9.0 Pie	28	39,5%
10. Android 10	29	8,2%

*Рис. 3.
Версії Android та % смартфонів, які їх підтримують*

Виходячи з вищесказаного можна зробити висновок, що Android ще довго буде найпоширенішою операційною системою серед смартфонів та інших гаджетів, а розробка застосунків для цієї системи буде важливою та великою нішою в сфері ІТ.

Починаючи розробляти програмні продукти для ОС Android потрібно визначитись з архітектурою. Адже вибрати правильну архітектуру програмного продукту перед початком розробки означає запобігти багатьом помилкам та проблемам в майбутньому.

Android-додатки мають складну структуру. Вони мають багато компонентів самої програми це фрагменти, різні сервіси, велика кількість розроблених екранів та переходів між ними. Всі ці компоненти зберігаються

маніфесті розроблюваного застосунку. Потім ОС Android використовує цей файл, щоб вирішити, як інтегрувати вашу програму в загальний досвід роботи пристрою. Враховуючи, що правильно написаний додаток для Android містить декілька компонентів і що користувачі часто взаємодіють з декількома програмами за короткий проміжок часу, додатки повинні адаптуватися до різних типів робочих процесів та завдань, керованих користувачами.[4]

Найбільш вживані архітектури для Android-додатків є: MVC, MVP, MVVM та MVI. MVC (MODEL – VIEW – CONTROLLER) є однією з найбільш використовуваних архітектур для розробки Android-застосунків. Вона має сильне розмежування між моделлю – як структурувати дані, видом – як ці дані представляти та контролером – як обробляти взаємодію з користувачем.

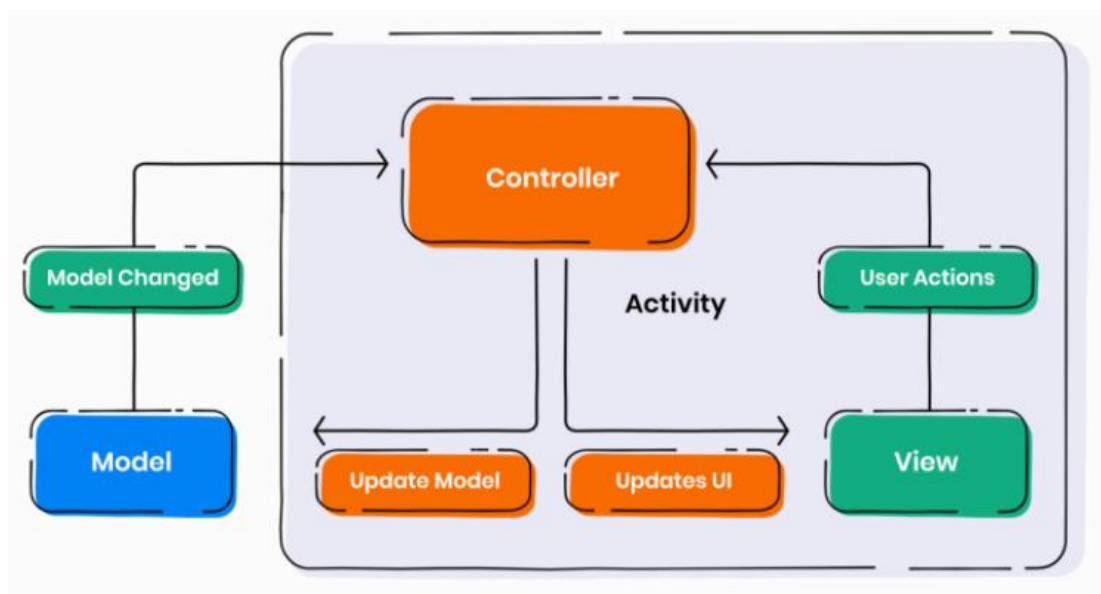


Рис. 4

Більшість додатків для Android розроблені з дотриманням вимог архітектури MVC. Проте один недолік цієї архітектури все-таки є. Проблема полягає в тому, що Activity – це і View, і контролер, але це вже порушує принцип єдиної відповідальності, що є основним в цій архітектурі.[5]

MVP (MODEL – VIEW – PRESENTER) є архітектурою, яка все частіше використовується розробниками мобільних застосунків для ОС Android. Тут бізнес-логіка визначається всередині класів, які називаються Presenter. Activity передасть дані у View і буде взаємодіяти з Presenter. Цей алгоритм дій дасть можливість проінформувати Presenter, що користувач виконав певну дію.

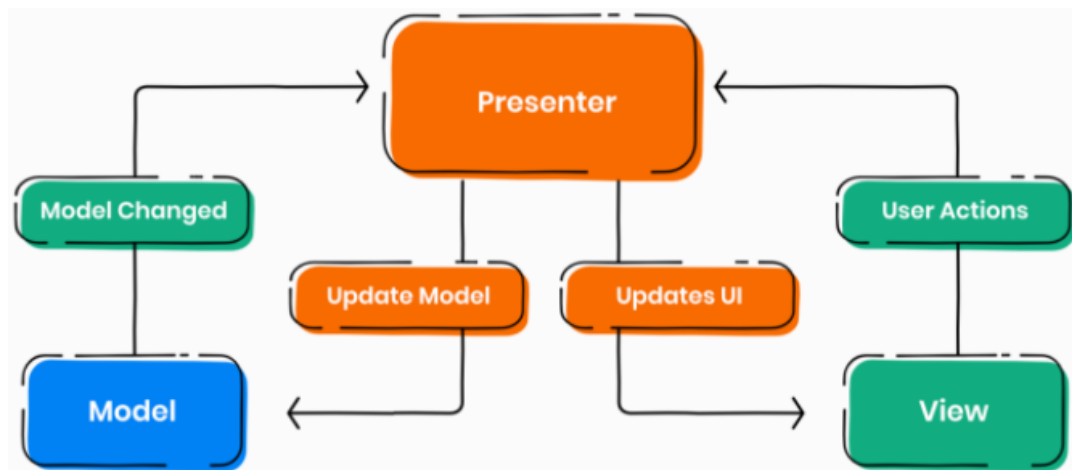


Рис. 5

Ця архітектура виявилась досить успішною та ефективною, оскільки бізнес-логіка добре ізольована і View може бути змінене незалежно від Presenter.[5]

2.2 Загальні відомості про мову програмування Kotlin

Kotlin – відносно молода мова програмування від компанії JetBrains. Ця мова була представлена широкому загалу в 2011 році, а на конференції Google I/O 2017 команда розробників Android оголосила, що Kotlin отримав офіційну підтримку для розробки Android-застосунків.[6]

Kotlin – це статично типізована мова програмування, яка підтримує не тільки об'єктно орієнтоване програмування, але й процедурне. Він орієнтується на безпеку, зрозумілість, сумісність та підтримку різних інструментів.[7] На

перший погляд Kotlin виглядає ніби більш лаконічна і спрощена Java(див Рис. 6 та Рис. 7).

```
StringBuilder sb = new StringBuilder();
```

Рис. 6. Фрагмент коду, написаний на Java

```
val sb = StringBuilder()
```

Рис. 7. Фрагмент коду, написаний на Kotlin

За приблизною оцінкою кількість рядків, у порівнянні Kotlin і Java, зменшилася на 40% на користь першої.[8] Більше того, Kotlin – безпечніший в плані типу даних. В ньому не допускається оголошення змінної, яка матиме значення NULL. Це робить код менш уразливим для помилки NullPointerException.

В цій мові програмування функції визначаються ключовим словом “fun”, вкінці кожного рядка, не потрібно ставити крапку з комою. Ключове слово “val” визначає перемінну, яка не може бути змінена в подальшій роботі програми. Для того, щоб визначити змінну, яку потім можна буде змінити потрібно використовувати ключове слово “var”. Проте Kotlin залишається строго типізованою мовою.

Розділ 3. Реалізація застосунку

3.1 Опис та огляд технічного завдання

Технічне завдання являє собою розробку програмного продукту для телефонів з операційною системою Android для людей, які бажають вдосконалювати та розвивати роботу свого мозку. Проаналізувавши предметну область та знайшовши головні проблеми, було поставлено задачу розробити інтуїтивно зрозумілий додаток, після запуску якого можна відразу приступати до тренування мозку.

Дизайн повинен бути зрозумілим та не перевантаженим зайвими іконками, кнопками, тощо. Потрібно розробити можливість задавати час на виконання завдань, щоб користувач міг самостійно вибирати скільки часу він може витратити на тренування. Додатковою опцією повинна бути можливість надсилання push-повідомлень, щоб нагадати користувачам про необхідність регулярного виконання завдань для досягнення кращого результату.

Додатковим завданням було створити можливість зберігати найкращі результати виконаних завдань. Так користувач буде більш мотивованим перевершити свої попередні показники. А також розробити можливість перегляду всіх попередніх результатів та час, витрачений на тренування.

3.2 Обґрунтування вибраного середовища розробки

Для розробки було обрано середовище Android Studio – це інтегроване середовище розробки.[9] Це середовище розробки замінило плагін Android Development Tools для середовища Eclipse, який перестав підтримуватись. Android Studio надає всі необхідні інструменти для зручної розробки програмних

продуктів. Дане середовище розробки – це звичайне одновіконне IDE, яке можна налаштувати під різні вимоги та потреби розробника(див. Рис. 8).

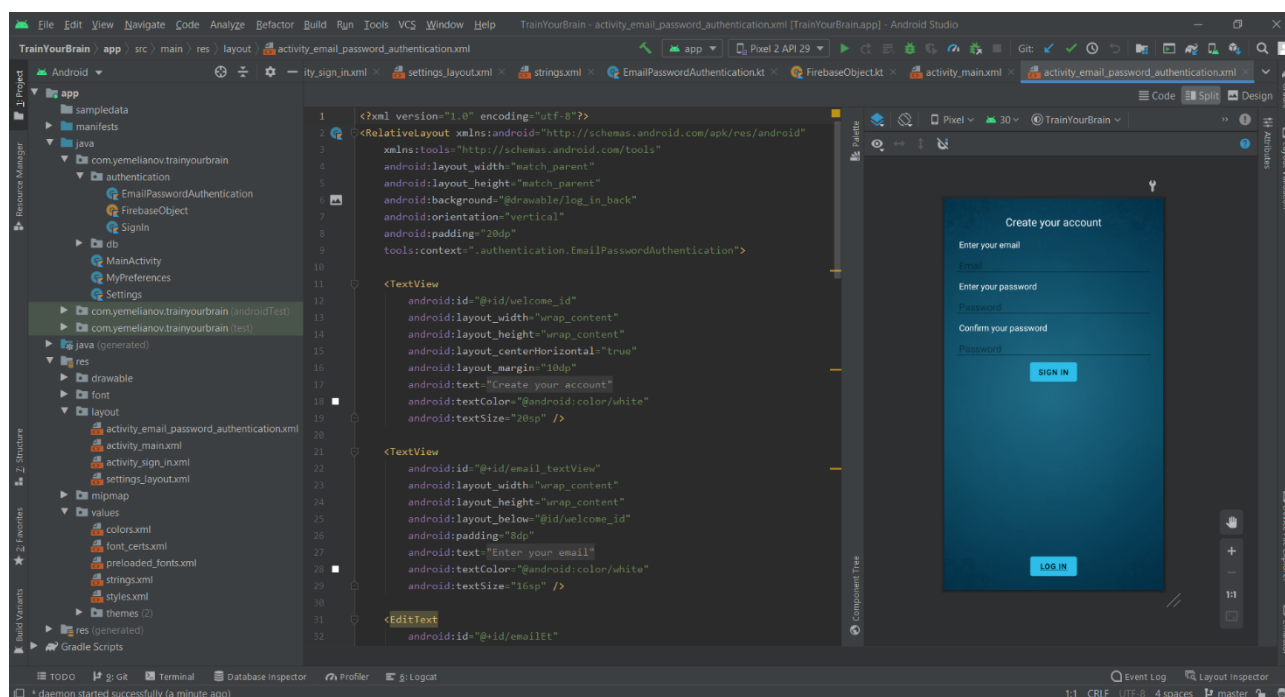


Рис. 8

За бажанням розробника певні вікна можна приховати, щоб не перенавантажувати себе непотрібними даними.

Android Studio, як і всі популярні середовища розробки має прямий доступ до систем контролю версіями, в тому числі Git. Робити коміти, або повертатись до попередніх версій проекту можна за декілька кроків. Під час творення коміту розробник може подивитись де були зроблені зміни та які нові файли будуть завантажені. Перед завантаженням змін на GitHub система покаже розробнику, якщо в коді є помилки або попередження та запропонує їх виправити. Також всі дії пов'язані з Git можна робити і в консолі(див. Рис. 9). Це буває швидше та простіше в певних

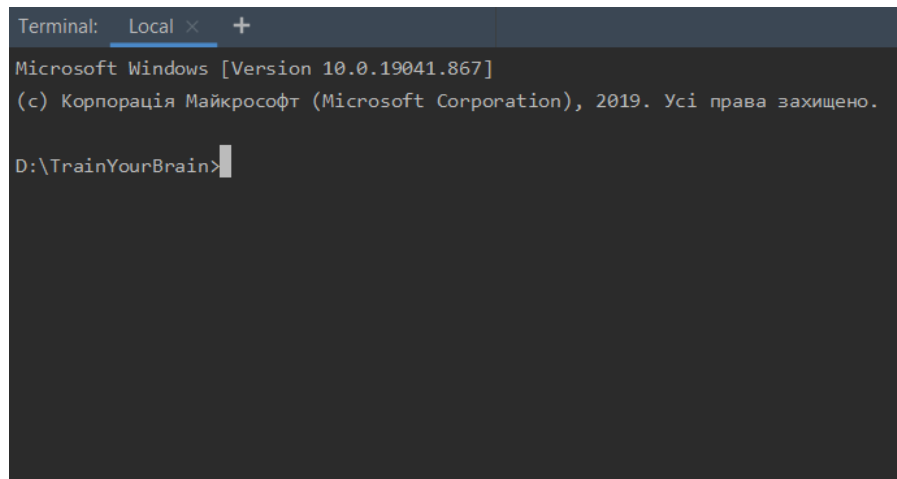


Рис. 9

Всі ці функції полегшують роботу програмісту з системою контролю версій та дають можливість не використовувати сторонні програмні продукти.

В Android Studio існує 2 варіанти тестування створеного застосунку. Це через фізичний девайс, який підключається до комп'ютера чи ноутбука, чи через створення симулятора смартфона в самому редакторі коду. Щоб створити симулятор девайсу потрібно вибрати відповідну категорію, розмір та розширення екрану і версію Android, яка буде встановлена на цей симулятор (див. Рис. 10 та Рис. 11)

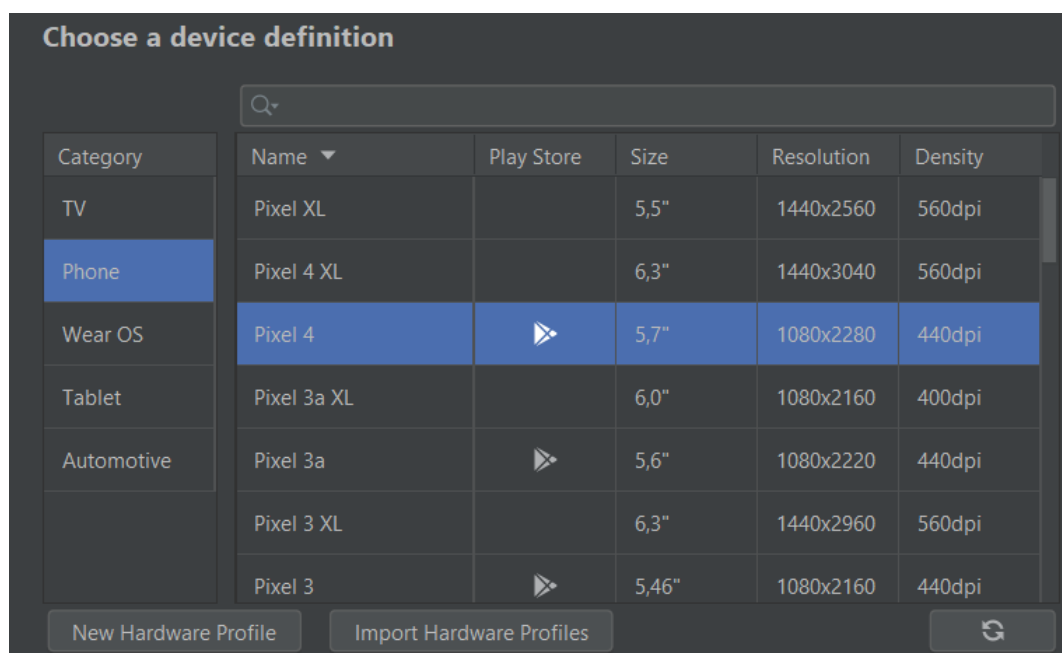
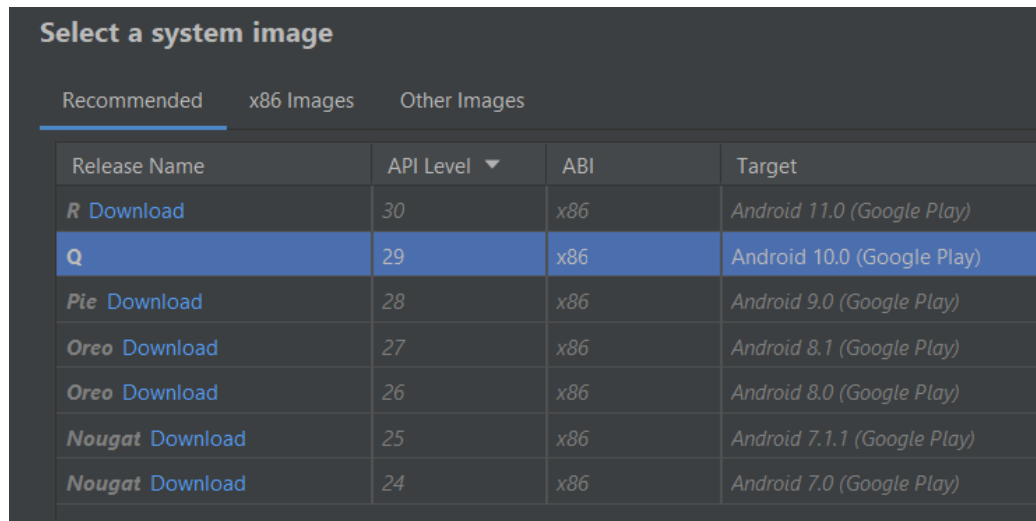


Рис. 10



Release Name	API Level ▼	ABI	Target
R Download	30	x86	Android 11.0 (Google Play)
Q	29	x86	Android 10.0 (Google Play)
Pie Download	28	x86	Android 9.0 (Google Play)
Oreo Download	27	x86	Android 8.1 (Google Play)
Oreo Download	26	x86	Android 8.0 (Google Play)
Nougat Download	25	x86	Android 7.1.1 (Google Play)
Nougat Download	24	x86	Android 7.0 (Google Play)

Рис. 11

Симулятор потрібен у випадку, якщо розробник не має фізичного девайсу з ОС Android, або хоче протестувати роботу додатку на пристрої, який має інше розширення екрану чи іншу версію операційної системи. Всі створені в симуляторі смартфони нічим не відрізняються від фізичних і на них можна виконувати всі дії, що і з реальними пристроями.

3.3 Обґрунтування та реалізація гри «Додавання»

Для тренування роботи мозку, реакції та пам'яті була вибрана гра «Додавання». Ця проста та звичайна на перший погляд гра дозволяє досить добре розвивати реакцію та стимулювати роботу мозку. Протягом всієї гри мозок користувача тримається в напрузі, щоб виконати якнайбільше задач за відведений проміжок часу.

В застосунку гра реалізована таким чином: на екран виводиться завдання, наприклад «34 + 17», а під ним розміщені 4 варіанта відповіді, з яких тільки 1 правильний. Користувач повинен вибрати число, яке він вважає правильним і натиснути на нього. Якщо відповідь правильна – користувач здобуває 1 бал, який зображується в нижній частині екрану разом з усією кількістю вирішених завдань і виводиться наступне завдання. А якщо ж відповідь не вірна –

користувач не отримує жодного балу і знову ж виводиться наступне завдання. Після того як завершиться відведений час гра закінчується, а результат гри заноситься в базу даних, а також порівнюється з попереднім найкращим результатом і якщо користувач перевершив свій результат на екран виводиться відповідне повідомлення.

```
private fun generateQuestion() {
    val intent = intent
    x = intent.getIntExtra( name: "x", defaultValue: 21)
    val rand = Random()
    val a = rand.nextInt(x)
    val b = rand.nextInt(x)
    sumTV!!.text = "{a} + {b}"
    locationOfCorrectAnswers = rand.nextInt( bound: 4)
    answers.clear()
    for (i in 0..3) {
        var incorrectAnswer: Int
        if (i == locationOfCorrectAnswers) {
            answers.add(a + b)
        } else {
            incorrectAnswer = rand.nextInt( bound: x + 20)
            while (incorrectAnswer == a + b) {
                incorrectAnswer = rand.nextInt( bound: x + 20)
            }
            answers.add(incorrectAnswer)
        }
    }
    button3!!.text = answers[0].toString()
    button4!!.text = answers[1].toString()
    button5!!.text = answers[2].toString()
    button6!!.text = answers[3].toString()
}
```

Рис. 12. Генерація завдання



Рис. 13. Виведене завдання та варіанти відповідей

3.4 Екран налаштувань

Для переходу на екран налаштування застосунку потрібно натиснути на відповідний значок в лівому верхньому куті екрану. Налаштування були створені для зручності, щоб кожен користувач міг налаштувати застосунок згідно своїх вподобань.

В налаштуваннях можна вибрати рівень складності. Користувачу доступні 3 рівні складності: легкий, середній та складний. Вони відрізняються між собою величиною чисел, з яких будуть генеруватись завдання. В легкому рівні числа не перевищують значення «21», в середньому рівні – «41» та в складному рівні – «61».

Користувач може вказати час в секундах, протягом якого він хоче вирішувати завдання. За замовчуванням, встановлено 10 секунд для однієї спроби. Все це користувач може змінити в меню налаштувань. Ще в налаштуваннях можна змінити тему застосунку(див. Рис. 14).

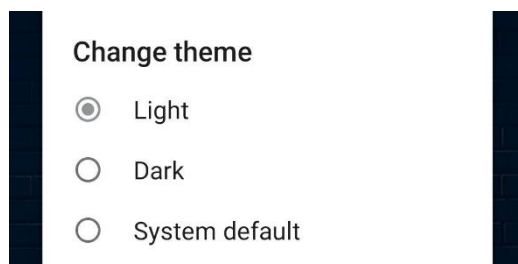


Рис. 14

Також користувач має можливість авторизуватись в додатку ввівши адресу своєї електронної пошти, яка слугує за логін та пароль(див Рис. 15)

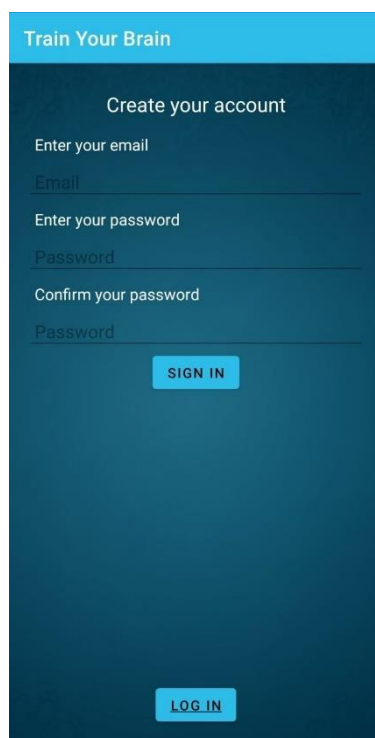


Рис. 15

3.5 Загальний опис додатку

Початок роботи над застосунком почався з розробки його назви та логотипу. Назва повинна бути лаконічною, але така, що запам'ятовується і відразу передає суть та мету додатку. Затвердженою назвою стала: TrainYouBrian. Логотип також повинен відповідати вищезгаданим критеріям. Таким був вибраний логотип, наведений нижче (див. Рис. 16). З його вигляду відразу зрозуміло, що це за додаток і для чого він призначений.





Рис. 16

Реєстрація користувача була реалізована за допомогою функції Authentication, яку надає платформа Firebase[15]. Дані всіх зареєстрованих користувачів зберігаються на сервері Firebase. Також додати або видалити користувача можна вручну на сайті, ввівши його дані або ж видалити відповідно. Push-повідомлення також реалізовані за через Firebase за допомогою функції Cloud Messaging. Тут дуже зручно створювати повідомлення для користувачів додатку і відслідковувати статистику. Є можливість надсилати push-повідомлення тільки вибраним користувачам.

Create experiment

New notification

Notification	Status ?	Platform	Start / Send	End	Sends	Opens
<div>➤</div> <div>Test notification 2! Train Your Brain!!!!</div>	✓ Completed		Apr 4, 2021 10:30 PM	—	<1000	0%
<div>➤</div> <div>Test notification Hello, world! Train Your Brain!</div>	✓ Completed		Apr 4, 2021 10:27 PM	—	<1000	0%

↺

0/10 Recurring notifications ?

Рис. 16

Висновок та можливість подальшого вдосконалення додатку

Розробка Android-застосунків має надзвичайно великі можливості та великий потенціал росту. Компанії можуть значно полегшувати життя, створюючи все нові та нові додатки, якими будуть користуватись мільйони людей. Ця сфера ще не скоро заповниться всіма потрібними програмними продуктами, які задовільнять всі потреби користувачів. Тим паче в час, коли технології розвиваються надзвичайно швидко.

Отже, закінчивши розробку додатку TrainYouBrain для тренування та вдосконалення роботи мозку було використано та вивчено можливості платформи Firebase, архітектури, які використовуються у розробці мобільних застосунків. Вдосконалено знання з розробки мобільних додатків для ОС Android. Всі цілі, які були поставлені напочатку роботи – виконані і застосунок повністю готовий до використання та публікування в Google Play Market.

Під час розробки додатку виникали певні значні та не дуже труднощі, які були виправлені. Наприклад, пошук найкращого рішення для реалізації push-повідомлень під час всіх станів додатку(вимкнений, активний та звернутий стани) та можливості простої але водночас ефективної реєстрації в додатку.

Застосунок має великі можливості для росту та вдосконалення. Планується розробити нові ігри та завдання, котрі будуть розвивати та тренувати мозок різними методами, додати можливість ведення статистики та автоматичних порад користувачам. Також проведення додаткових опитувань наявних та потенційних користувачів на предмет того, що вони ще хочуть побачити в даному застосунку.

Список використаних джерел

1. Bigger than Windows, bigger than iOS: Google now has 2.5 billion active Android devices. – [Електронний ресурс]. – Доступно з:
<https://www.zdnet.com/article/bigger-than-windows-bigger-than-ios-google-now-has-2-5-billion-active-android-devices-after-10-years/>
2. Додатки тижня: для покращення пам'яті, уваги та концентрації – [Електронний ресурс]. – Доступно з: <https://inspired.com.ua/ideas/dodatky-tyzhnya-dlya-pokrashhennya-pamyati-uvagy-ta-kontsentratsiyi/>
3. Вакансії із сайту dou.ua – [Електронний ресурс]. – Доступно з:
<https://jobs.dou.ua/vacancies/?search=android>
4. Developers Android – [Електронний ресурс]. – Доступно з:
<https://developer.android.com/jetpack/guide>
5. A GUIDE TO GOOGLE'S RECOMMENDED ARCHITECTURE FOR ANDROID APPS – [Електронний ресурс]. – Доступно з:
<https://www.scalablepath.com/blog/recommended-architecture-for-android-apps/>
6. Kotlin Programming Language – [Електронний ресурс]. – Доступно з:
<https://kotlinlang.org>
7. What is Kotlin? The Java alternative explained – [Електронний ресурс]. – Доступно з: <https://www.infoworld.com/article/3224868/what-is-kotlin-the-java-alternative-explained.html>
8. FAQ | Kotlin – [Електронний ресурс]. – Доступно з:
<https://kotlinlang.org/docs/faq.html#is-kotlin-an-object-oriented-language-or-a-functional-one>

9. Developers Android – [Електронний ресурс]. – Доступно з:

<https://developer.android.com/studio>

10. 14 Best Android Frameworks for App Development in 2020 – [Електронний

ресурс]. – Доступно з: <https://www.intellectsoft.net/blog/best-android-frameworks/>

11. Скільки часу діти проводять перед екраном – [Електронний ресурс]. –

Доступно з:

https://www.bbc.com/ukrainian/science/2015/03/150329_children_screens_she

12. Як тренувати мозок і покращувати пам'ять – [Електронний ресурс]. –

Доступно з: <https://moz.gov.ua/article/health/jak-trenuvati-mozok-i-pokraschuvati-pamjat>

13. Google Play Market – [Електронний ресурс]. – Доступно з:

play.google.com/store

14. Mobile operating systems' market share worldwide from January 2012 to January 2021 – [Електронний ресурс]. – Доступно з:

<https://www.statista.com/statistics/272698/global-market-share-held-by-mobile-operating-systems-since-2009/>

15. Firebase helps you build and run successful apps – [Електронний ресурс]. –

Доступно з: <https://firebase.google.com>