

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ОДЕСЬКИЙ ДЕРЖАВНИЙ ЕКОЛОГІЧНИЙ УНІВЕРСИТЕТ

Факультет комп'ютерних наук,
управління та адміністрування
Кафедра інформаційних технологій

МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА

на тему: Реалізація Android-застосунку для
запису дзвінків

Виконав студент 2 курсу групи МІС-21
спеціальності 122 Комп'ютерні науки
Осадченко Олександр Олександрович

Керівник к.т.н., доцент
Фразе-Фразенко Олексій Олексійович

Рецензент Начальник ЦІТ ОНЕУ
к.т.н., Домаскін О.М.

Одеса 2022

АНОТАЦІЯ

на магістерську кваліфікаційну роботу
«Реалізація Android-застосунку для запису дзвінків»,
студента Осадченко Олександра Олександровича

Актуальність теми магістерської кваліфікаційної роботи обумовлюється необхідністю реалізацією сучасного застосунку задля запису дзвінків з відкритим кодом.

Мета роботи – аналіз засобів, технологій та методів розробки застосунку для пристроїв під ОС Android.

Об’єкт дослідження – сучасні середовища та засоби розробки графічних застосунків для запису дзвінків.

Предмет дослідження – технології створення застосунку запису дзвінків для пристроїв під ОС Android.

Методи дослідження – емпіричний метод із використанням експертних оцінок, сучасних програмних продуктів та технологій.

Перший розділ присвячений аналізу та опису основних причин використання функцій запису дзвінків. В даному розділі, представлено переваги використання даної функції та її масштаб використання як у кол-центрі компанії так і на пристрої звичайного користувача. У другому розділі проведено аналіз найпопулярніших застосунків, які представлені на маркетплейсах під операційну систему Android. Також, в даному розділі наведено невеликий опис по кожному із застосунків, а також представлена порівняльна таблиця проаналізованих інструментів / мобільних застосунків. Третій розділ основні етапи на які ділиться розробка застосунку. В даному розділі наведено Основні кроки відповідно до життєвого циклу розробки програмного забезпечення, яких було притримано безпосередньо під час розробки даного застосунку. В четвертому розділі, описується хід проведення розробки застосунку під операційну систему Android для запису дзвінків. Для початку розглядається процес створення

нового проекту, після чого описано та представлено уривки коду які були написані задля розробки мобільного застосунку.

Результатом є розроблений Android-застосунок для запису дзвінків. Розробка була в середовищі Android Studio Dolphin з використанням мови програмування Kotlin.

Магістерська кваліфікаційна робота містить 70 сторінок, 18 рисунків, 1 таблиці та 8 джерел посилань.

Ключові слова: ANDROID STUDIO DOLPHIN, DATABASE, GOOGLE PLAY, GUI, KOTLIN, OS, RECORD CALL, SQLITE.

SUMMARY

for a master's degree

" Implementation of an Android application for recording calls",

of the student Osadchenko Oleksandr

The relevance of the topic of the master's qualification work is determined by the need to implement a modern application for recording calls with open source code.

The purpose of the work is the analysis of the means, technologies and methods of developing an application for Android devices.

The object of research is modern environments and tools for developing graphical applications for call recording.

The subject of research is the technology of creating an application for recording calls for Android devices.

Research methods – an empirical method using expert assessments, modern software products and technologies.

The first chapter is dedicated to the analysis and description of the main reasons for using the call recording function. This chapter presents the advantages of using this feature and its large-scale use both in the company's call center and on the device of an ordinary user. The second chapter analyzes the most popular applications that are presented on the marketplaces for the Android operating system. Also, this section provides a short description for each of the applications, as well as a comparison table of the analyzed tools / mobile applications. The third chapter of the main stages into which the development of the application is divided. This chapter lists the main steps according to the software development life cycle that were followed during the development of this application. The fourth chapter describes the development of an application for the Android operating system for recording calls. To begin with, the process of creating a new project is considered, after which the code snippets that were written for the development of a mobile application are described and presented.

The result is a developed Android application for recording calls. Development was in the Android Studio Dolphin environment using the Kotlin programming language.

The master's thesis contains 70 pages, 18 figures, 1 table and 8 sources.

Keywords: ANDROID STUDIO DOLPHIN, DATABASE, GOOGLE PLAY, GUI, KOTLIN, OS, RECORD CALL, SQLITE.

ЗМІСТ

Терміни і скорочення	9
Вступ.....	11
1 Аналіз та опис причин використання функції запису дзвінків у масштабному форматі	13
2 Аналіз і опис найпопулярніших застосунків з функцією запису дзвінків..	22
2.1 Розгляд та аналіз застосунків для запису розмови на безоплатній основі під ОС Android.....	23
2.1.1 Опис застосунку Another Call Recorder	24
2.1.2 Опис застосунку TRUECALLER.....	25
2.1.3 Опис застосунку RMC: ANDROID CALL RECORDER	27
2.1.4 Опис застосунку Smart Call Recorder – SCR	29
2.1.5 Опис застосунку Cube ACR	30
2.1.6 Опис застосунку Automatic Call Recorder callX.....	32
2.1.7 Опис застосунку Auto Call Recording	34
2.1.8 Опис застосунку Voice Call Recorder.....	36
2.1.9 Опис застосунку CryOk Call recorder.....	38
2.1.10 Опис застосунку Callbox	40
3 Опис основних етапів при розробці застосунку для запису дзвінків	46
3.1 Етапи розробки програмного забезпечення	47
4 Розробка Android-застосунку запису дзвінків.....	51
Висновки	68
Перелік джерел посилання	70

ТЕРМІНИ І СКОРОЧЕННЯ

База даних – сукупність даних, організованих відповідно до концепції, яка описує характеристику цих даних і взаємозв'язки між їх елементами; ця сукупність підтримує щонайменше одну з областей застосування (за стандартом ISO/IEC 2382:2015). В загальному випадку база даних містить схеми, таблиці, подання, збережені процедури та інші об'єкти. Дані у базі організують відповідно до моделі організації даних. Таким чином, сучасна база даних, крім самих даних, містить їх опис та може містити засоби для їх обробки.

Життєвий цикл розробки програмного забезпечення або SDLC – це процес чіткого визначення цілей і етапів побудови програмного рішення.

Кешування – соблива швидкісна пам'ять або частина оперативної пам'яті, де зберігаються копії часто використовуваних даних.

Об'єкти доступу до даних – об'єкт що надає абстрактний інтерфейс до деяких видів баз даних чи механізмів персистентності реалізуючи певні операції без розкриття деталей бази даних.

Операційна система – це базовий комплекс програм, що виконує керування апаратною складовою комп'ютера або віртуальної машини; забезпечує керування обчислювальним процесом і організовує взаємодію з користувачем. Операційна система звичайно складається з ядра операційної системи та базового набору прикладних програм.

Програмне забезпечення – сукупність програм системи оброблення інформації та програмних документів, необхідних для експлуатації цих програм. Це набір інструкцій, які розповідають комп'ютеру, як працювати. Це на відміну від апаратного забезпечення, з якого побудована система і фактично виконує роботу.

Програмний продукт – програмне забезпечення, розроблене для вирішення задачі масового попиту та призначене для постачання користувачам.

App Store – платформа цифрової дистрибуції, що розроблена і підтримується Apple Inc. для комп'ютерних програм та мобільних застосунків на своїх операційних системах.

Google Play – крамниця застосунків від Google, що дозволяє власникам пристроїв з мобільною операційною системою Android та іншими завантажувати і купувати різні застосунки, книги, фільми і музику.

GUI – програмна оболонка, яка надає користувачеві зручний інтерфейс для роботи з операційною системою. GUI візуалізує багато компонентів у вигляді графічних об'єктів, наприклад, кнопки, меню, стрілки тощо.

SQLite – полегшена реляційна система керування базами даних. Втілена у вигляді бібліотеки, де реалізовано багато зі стандарту SQL-92.

- БД – база даних.
- ЗБТ – закрите бета-тестування.
- КТІ – комп'ютерно-телефонна інтеграція.
- ОС – операційна система.
- ПЗ – програмне забезпечення.
- ПП – програмний продукт.
- API – Application Programming Interface.
- CRM – Customer Relationship Management.
- DAO – Data Access Object.
- GUI – Graphical User Interface.
- SDLC – Software development life cycle

ВСТУП

Розробка простого, зручного у використанні та на безоплатній основі застосунку для запису дзвінків завжди буде користуватися попитом у користувачів. Запис дзвінків є частиною успішного ведення бізнесу. Запис дзвінків – це процес збору аудіо (і іноді активності екрана), пов’язаного з телефонними дзвінками. Компанія, яка записує телефонні дзвінки, може збирати розмови та використовувати інформацію, отриману всередині, для покращення взаємодії з клієнтами та прийняття розумних бізнес-рішень.

Актуальність магістерської кваліфікаційної роботи обумовлюється необхідністю реалізацією сучасного застосунку задля запису дзвінків з відкритим кодом.

Метою кваліфікаційної роботи є аналіз засобів, технологій та методів розробки застосунку для пристроїв під ОС Android.

Рішення для збору даних і супутні закони про запис дзвінків служби підтримки клієнтів існують уже більше століття. Записи почалися з пристрою воскового диска, створеного в 1903 році, а перший магнітофон з’явився в 1930-х роках. З тих пір з’явилося багато технологій для спрощення та вдосконалення процесу запису. Рішення для запису дзвінків часто додаються до хмарних контакт-центрів, аналітики на основі штучного інтелекту та навіть систем навчання агентів. Вони можуть бути важливою частиною управління якістю, допомагаючи керівникам зрозуміти, чи правильно агенти стежать за потоками викликів і перехресними продажами продуктів.

Найкращі у своєму класі системи запису розмов сьогодні є повністю цифровими. Вони інтегруються з існуючим контакт-центром і платформою бізнес-телефонії, використовуючи хмару для зберігання. Сучасна система запису телефонних розмов для користувачів малого бізнесу може навіть мати додаткові функції. Деякі рішення оснащені технологією захоплення екранної та відеоінформації, а також аудіо. Інші мають передові технології, вбудовані в програмне забезпечення для підтримки відповідності PCI та GDPR.

Система запису дзвінків — це не нова концепція на сучасному ринку. Однак, оскільки клієнти продовжують вимагати більшої конфіденційності та захисту своїх конфіденційних даних, багато хто часто запитує, наскільки законним є запис розмов. Важливо відзначити, що система запису телефонних розмов для малого бізнесу – це не те саме, що людина, яка сама записує розмови.

Компанії в багатьох галузях часто зобов'язані записувати дзвінки з метою захисту даних і відповідності, а також для того, щоб вони могли захистити себе від суперечок у суді. Однак запроваджені правила, такі як GDPR, означають, що організації також повинні переконатися, що їхні клієнти знають про запис і дозволять це зробити.

Дана кваліфікаційна робота магістра, складається з 70 сторінок, 18 рисунків, 1 таблиця та 12 джерел посилання.

1 АНАЛІЗ ТА ОПИС ПРИЧИН ВИКОРИСТАННЯ ФУНКЦІЇ ЗАПИСУ ДЗВІНКІВ У МАСШТАБНОМУ ФОРМАТІ

Якщо одним із головних аспектів бізнесу є дзвінки клієнтів, але при цьому володілець ще не почав їх активно записувати, то втрачаються способи підвищити ефективність роботи агентів, покращити клієнтський досвід і, таким чином, значно збільшити доходи.

В повсякденному житті перед певними дзвінками часто повідомляють, що «розмову буде записано з метою навчання та оцінки якості». Це твердження може здатися зрозумілим для більшості людей. Однак для тих, хто працює в контакт-центрі, запис дзвінків і голосу, звичайно, означає набагато більше.

Щоб спростити все, можна перерахувати 10 переваг запису дзвінків у кол-центрі:

- визначення проблемної області;
- введення детального обліку;
- відновлення пропущених деталей;
- забезпечення якості;
- постійне вдосконалення;
- надання цінних відгуків;
- покращення надання послуг та/або продукту;
- вирішення суперечок між оператор/компанією та клієнтом;
- удосконалення маркетингової кампанії;
- дотримання правових норм.

Точно знати, що оператори компанії говорять клієнтам по телефону, а також те, як вони це говорять, – це найкращий спосіб визначити потенційні проблеми зв'язку у телефонній підтримці як на індивідуальному, так і на груповому рівнях. Усвідомлення того, як саме клієнти реагують на певні фрази чи стилі спілкування, може бути корисним для модифікації сценаріїв, які викори-

стовуюють оператори, що, зрештою, підвищить задоволеність клієнтів. Наприклад, замість того, щоб запитувати абонентів: "Яка проблема у вас зараз?" було б краще, щоб запитали: "Чим я можу вам допомогти сьогодні?" Те, як оператори навчають пропонувати певні продукти чи послуги, також можна переглянути, якщо буде виявлено, що на них підписується дуже мало клієнтів.

Записані дзвінки, заархівовані у кол-центрі CRM або іншій інформаційній системі, можуть бути корисними для перевірки збереженої інформації про клієнта. Завдяки КНІ записані дзвінки можна автоматично зберігати у відповідному файлі клієнта, не вимагаючи від оператора, який прийняв дзвінок, шукати в системі, щоб вручну зберегти звуковий файл у потрібному місці. Це також може допомогти в кампаніях з продажу, дозволяючи агентам повертатися до точних деталей попередніх .

Можливість повернутися назад і прослухати заархівовані дзвінки певного клієнта може бути надзвичайно цінною, якщо колись виникне ситуація, пов'язана з плутаниною, наприклад, щодо покупки клієнта. Припустімо, керівник кол-центр обслуговує клієнтів у компанії, яка займається басейном, і клієнт наполягає на тому, що фахівець на місці встановив неправильний тип насоса для фільтрації води. Хоча записи можуть вказувати, що це саме той продукт, який замовили клієнт, менеджер служби підтримки може повернутися та прослухати всі попередні обміни між клієнтом і останнім агентом, щоб знайти точне джерело непорозуміння, щоб легше знайти рішення.

Записані дзвінки дають змогу оцінити загальний рівень задоволеності клієнтів і оцінити якість допомоги, яку надає служба підтримки по телефону. Це, у свою чергу, дозволить ефективно адаптувати привітання IVR та повідомлення про очікування , а також сценарії для операторів у всіх відділах. Інформування абонентів про те, що їхні дзвінки записуються, також сприяє відчуттю прозорості, запевняючи клієнтів, що їхня особиста інформація використовується лише в їхніх інтересах.

Запис дзвінків є дуже ефективним способом «виміряти температуру» телефонної підтримки, постійно тримаючи в курсі рівня задоволеності клієнтів,

постійних проблем, а також сильних і слабких сторін вашого персоналу. Наймання найкращих працівників – це лише перший крок до надання чудових послуг. Необхідно встановити системи для легкого відстеження та визначення ключових областей, які потребують покращення, раз за разом. Який би шлях до покращення ви не вибрали, використання записів дзвінків може допомогти переконатися, що вжиті кроки мають значний вплив на ефективність роботи кол-центру.

Навчання співробітників, відповідальних за обробку дзвінків, буде набагато ефективнішим, якщо і керівник, і оператор зможуть разом прослухати попередній дзвінок і детально обговорити його позитивні та негативні сторони. Агенти, які можуть відчувати труднощі, можуть постійно прослуховувати власні дзвінки та навчитися розпізнавати потенційні підводні камені (наприклад, уникати хвилювання, коли абонент підвищує голос), а також те, що вони роблять правильно (наприклад, запитувати, чи є будь-що інше, з чим вони можуть допомогти, перш ніж покласти трубку). Нові оператори, які вміють прослуховувати дзвінки своїх найефективніших колег, можуть отримати цінну інформацію про методи холодних дзвінків, а також підвищити ціну лояльним клієнтам.

Якщо десятки або навіть сотні абонентів щодня висловлюють подібні скарги службі підтримки клієнтів, це означає, що компанії потрібно вирішити проблему якомога швидше. Використовуйте записані дзвінки, щоб дізнатися, чи є певні аспекти продукту чи послуги, які потрібно перевірити, і якщо всі загалом не задоволені цим, подумайте про те, щоб повністю вилучити його зі своїх пропозицій.

Повернутися назад і знайти джерело розбіжностей може бути особливо цінним для їх вирішення. Оператор, який допустив ескалацію гарячої дискусії, можливо, доведеться відірвати від телефонів на деякий час, поки він пройде додаткове навчання або оцінку. І навіть якщо в такому випадку провина лежить на клієнті, прослуховування суперечливих розмов може допомогти ва-

шому керівництву вирішити, чи краще врятувати професійні стосунки – пропонуючи знижки чи безкоштовне оновлення – чи повністю розірвати відносини з клієнтом.

Маркетингова команда, хоча й не має прямого контакту з клієнтами, отримує величезну користь від думок ваших клієнтів. Записуючи розмови з клієнтами, колеги з маркетингу та комунікацій можуть переглядати їх, щоб з'ясувати недоліки та позитивні сторони вашого ідеального клієнта та знайти креативні способи вирішення їх у своїх маркетингових кампаніях. Маркетологи також можуть уважно прослухати записи дзвінків, щоб точно визначити ключові слова, які клієнти згадують (чи ні), щоб краще оптимізувати свої публікації та онлайн-рекламу. Перегляд записів дзвінків може допомогти уникнути здогадів під час створення зіркових маркетингових інструментів, зокрема цільових персонажів і ключових ліній для залучення потрібної аудиторії [1].

Хоча в деяких країнах це може бути не рекомендовано або навіть незаконно, деякі інші можуть вимагати за законом, щоб записувати дзвінки компанії та зберігали ці записи протягом певного часу. Якщо судовий процес або інший вид судового розгляду виникне через щось, сказане одним із ваших агентів по телефону, від вашої компанії можуть вимагати опублікувати запис розмови як доказ. І навіть якщо країна не вимагає від вас цього, найкраще записувати всі дзвінки, якщо це дозволено як засіб захисту в потенційних судових спорах. Незалежно від того, чи працюєте ви в бізнесі, де працює лише кілька сумісників або працюйте 1000 людей у колл-центрі з кількома вузлами, телефонна діяльність, ймовірно, є важливим аспектом того, як спілкуєтеся зі своєю клієнтською базою, а це означає, що керівник компанії несе відповідальність за те, щоб це було якнайкраще.

Можна виділити 5 основних причини використання функції запису дзвінків під час розмови (рис. 1)



Рисунок 1 – Схематичне відображення 5 основних причин використання функції запису дзвінків під час розмови

За допомогою програмного забезпечення кол-центру супервайзери можуть стежити за голосовим реєстратором, щоб знайти конкретні приклади та змусити представників зрозуміти їхні недоліки. Це допомагає ефективно вирішувати ключові проблеми, а також визначати напрямки покращення. Це, безумовно, призведе до покращення якості дзвінків. Можна сміливо зауважити, що запис дзвінків допомагає точно визначити першопричину проблем із продуктивністю, щоб їх можна було виправити. З іншого боку, гарна розмова з клієнтом може стати чудовим прикладом для інших агентів щодо того, як завоювати клієнта та досягти бажаних результатів.

Активне прослуховування має бути перевагою кожного представника колл-центру. Однак у багатьох випадках представникам може бути важко слухати та запам'ятовувати всю розмову, під час отримання даних клієнтів і ведення нотаток. Крім того, необхідно враховувати якість голосу дзвінка. Це конкретні ситуації, коли запис дзвінків стає в нагоді; де представники можуть відтворювати цілі записані дзвінки або їх частини, щоб ніхто не пропустив

щось важливе. Це також дозволить представникам бути повністю присутніми з клієнтами під час дзвінків.

Контакт-центри – це все про ваших клієнтів, а не про продукт. Тому у інтересах компанії створити середовище, яке сприятиме більшій задоволеності клієнтів. Першим кроком до досягнення цієї мети є відстеження будь-якої інформації про клієнта. Простіше кажучи, запис розмов значною мірою допоможе отримати достатню кількість даних про клієнтську базу. Крім того, поділившись записами розмов із певними командами у контакт-центрі, можна краще зрозуміти, як клієнти використовують той чи інший продукт, що їм у ньому подобається та що можна було б зробити краще.

Сучасний клієнт має досить високі очікування щодо комунікацій колл-центру. По суті, клієнти не хочуть взаємодії за сценарієм. Вони хочуть, щоб представники спілкувалися з ними таким чином, щоб вони відчували себе особливими та піклувалися про них. Вони хочуть, щоб їх запевнили і переконали, що з боку компанії буде зроблено все необхідне, щоб зробити їх щасливими. Визначивши деякі важливі моменти розмови, можна розробити індивідуальні плани та продукти для клієнтів. У результаті компанія вже буде готова з кращою пропозицією ще до того, як клієнти її попросять.

У деяких конкретних випадках запис дзвінків і голосу може служити усним договором. Це саме означає, що записані дзвінки можуть допомогти у вирішенні суперечок клієнтів. Він також може прийти на допомогу, коли доведеться захищати компанію від судових процесів з боку розлючених клієнтів. Відповідно до звіту за 2014 рік про витрати на відповідність федеральним нормам, компанії з менш ніж 20 співробітниками платять у середньому майже 11 000 доларів США на одного працівника в якості нормативних витрат [2]. Компанії з 500 і більше співробітників платять в середньому 7755 доларів США на одного працівника на нормативні витрати. Запис голосу може допомогти заощадити гроші, а також юридичні головні болі та, таким чином, захистити репутацію вашої компанії.

Розміщені рішення для управління та запису дзвінків на основі хмари забезпечують більш гнучкий комунікаційний стек для сучасних компаній. Немає жодних витрат на обладнання на місці, про які варто турбуватися, а оскільки рішення є повністю розміщеним, немає обмежень щодо кількості записів, які ви можете проводити одночасно.

Хмарні та хостингові рішення для запису дзвінків стають все більш популярними в епоху гнучкого бізнесу. За допомогою цієї технології можна підтримувати роботу агентів кол-центру з будь-якого місця. Крім того, простота хмари дає змогу розгортати в той же день і легко адаптуватися для нових членів команди. Оскільки готові інформаційні панелі та скляні інтерфейси з єдиною панеллю настільки інтуїтивно зрозумілі та прості у використанні, будь-хто може почати миттєво.

Хмарні системи запису дзвінків також мають можливість доступу до вбудованої інтеграції як із існуючими, так і з майбутніми технологіями. В свою чергу можна поєднати запис розмов із коучингом агента та системами CRM або ввімкнути можливості на основі штучного інтелекту в майбутньому.

Функція запису дзвінків працює за допомогою апаратного та програмного забезпечення, яке підключається до телефонних ліній і записує розмови. Цифровий файл зберігається для відтворення або створення транскрипції. Потреби визначатимуть тип рішення для запису дзвінків, яке працюватиме найкраще. Змінні включають спосіб виклику, кількість користувачів і те, чи знаходяться вони в одному місці чи в кількох. Запис може бути налаштований на автоматичний початок або його буде ініційовано вручну агентом або співробітником. У більшості варіантів вибраний інтерфейс дозволить активувати функцію запису. На ринку є багато рішень для запису дзвінків, і не всі вони будуть встановлені.

Є дуже небагато компаній, які не шукають постійно способів покращити свою роботу, незалежно від того, чи це вдосконалення включає збільшення продажів, спрощення командної співпраці, утримання клієнтів або просто забезпечення оптимальної роботи кожного рівня. Незалежно від того, які області

кол-центрів компанія хоче вдосконалити, запис дзвінків дуже ефективний спосіб побачити зміни на краще в усіх сферах діяльності. Від навчання представників служби обслуговування клієнтів тому, як краще вирішувати складні ситуації, з якими можна зіткнутися по телефону, до того, щоб дати новим агентам почути, як ветерани продають ціни цінним клієнтам, і точно визначити, де саме криється незадоволення серед клієнтів, запис розмов може значною мірою допомогти збільшити прибуток, мінімізувати плинність кадрів і зберегти лояльність клієнтів.

Голоси по телефону записуються вже більше століття, починаючи з пристрою воскового диска, запатентованого Теодором і Карлом Фрізами в 1903 році, який служив механізмом відповіді на дзвінки для відсутніх одержувачів. Цей перший «автовідповідач» так і не набув звичного статусу, оскільки його було важко використовувати та він мав обмежену здатність запису, але намір його створення спонукав до нових розробок протягом багатьох років. У 1930-х роках оригінальний восковий дисковий магнітофон був замінений магнітофонами, які використовували магнітну стрічку для створення та зберігання записаного звуку. Протягом наступних десятиліть було розроблено численні версії автовідповідачів, які записували голоси абонентів на плівку – кожна була меншою, зручнішою для користувача та з більшою можливістю запису, ніж попередня. Вони широко використовувалися в кол-центрах аж до кінця 20-го століття, коли цифровий запис викликів почав переважати. Цей тип запису розмов все ще розробляється донині, використовуючи такі можливості, як КТІ (комп'ютерно-телефонна інтеграція), яка синхронізує інформацію в комп'ютерній системі з телекомунікаційною системою компанії, і зберігання значно більшого обсягу даних протягом більш тривалого часу.

Можливість прослуховування та перегляду дзвінків пропонує ряд переваг для бізнесу:

- вирішення проблем: записи про взаємодію між співробітниками та клієнтами допомагають вирішити конфлікти;

- якісний сервіс: якби користувач мобільним пристроєм знав, що хтось прислухається до кожного слова, користувач поводитиметься більш професійно (так само записи розмов заохочують співробітників залишатися професійними, навіть якщо клієнти звертаються до них з ворожою поведінкою);
- задоволеність клієнтів: реєстрація та реєстрація дзвінків, які стосуються одного клієнта, допомагає компанії звертатися до розмов, про які йдеться, будь то для роз'яснення чи правового захисту;
- практичне навчання: новий персонал часто потребує моніторингу, щоб забезпечити єдину філософію бренду (під час навчання контрольні дзвінки дозволяють керівникам зареєструватися та допомогти/виправити слухачів, якщо це необхідно);
- відповідність законодавству: багатьом підприємствам необхідно узгодити законодавство штату, і, записуючи дзвінки, можна буде захистити компанію в разі виникнення суперечки.

Більшість підприємств отримують вигоду від використання функцій запису дзвінків із зазначених вище причин [3]. Для деяких це необхідний інструмент для надання якісних послуг (див.рис.2)

урядові організації та організації громадської безпеки використовують дану функцію для запису інцидентів і реагування;

банки, страхові компанії, торгові центри та громадські організації зобов'язані відповідати нормам, а запис дзвінків не підлягає обговоренню;

контактні та кол-центри повинні використовувати запис розмов для контролю якості та збору даних.

Рисунок 2 – Схематичне відображення переліку органів

2 АНАЛІЗ І ОПИС НАЙПОПУЛЯРНІШИХ ЗАСТОСУНКІВ З ФУНКЦІЄЮ ЗАПИСУ ДЗВІНКІВ

Кожен дзвінок, який надходить до компанії є важливим. На сучасному ринку, коли клієнти мають такий широкий вибір покупок, а конкуренція є жорсткою, те, як бізнес обробляє дзвінки клієнтів, може бути різницею між отриманням і збереженням бізнесу абонента або втратою його одному з конкурентів. У цьому середовищі моніторинг того, наскільки ефективно співробітники компанії спілкуються з абонентами, є життєво важливим. Найефективнішим способом зробити це є запис розмов.

Запис дзвінків дозволяє аналізувати навички телефонування співробітників. Регулярне прослуховування цих дзвінків дає змогу скласти уявлення про те, як відбувається взаємодія з клієнтами.

Розвиток персоналу – записи дзвінків можна використовувати для розвитку персоналу. Коли керівник компанії дізнається, як добре чи погано звучить виклик для бізнесу, він може використовувати дзвінки, щоб проілюструвати це своєму персоналу.

Процеси – керівник або управляючий компанії може використовувати запис розмов, щоб допомогти вдосконалити процеси під час роботи. Якщо поточні процеси обробки викликів не працюють – виникає необхідність їх змінити і використовувати записи, щоб побачити результати.

Обслуговування клієнтів, – один задоволений клієнт може принести п'ятьох нових до потенційного бізнесу, але один незадоволений може поширити свою думку сотням людей, особливо через онлайн-огляди та сайти соціальних мереж. Використовуючи запис дзвінків, можна відстежувати, як ставляться до клієнтів і як обробляються їхні скарги. Цю інформацію можна використати для покращення обслуговування клієнтів. Слід перетворювати незадоволених клієнтів на щасливих.

Сезонні рекламні акції, тут мається на увазі якщо команда рекламує спеціальні пропозиції компанії або сезонні акції абонентам, які дзвонять по телефону. Запис дзвінків дозволяє контролювати це.

Запити про продажі, якщо у кол-центрі є велика кількість вхідних телефонних запитів, які не перекладаються на зустрічі чи продажі, запис розмов допоможе з'ясувати причину.

Генерація потенційних клієнтів, у разі якщо команда компанії з продажів здійснює вихідні дзвінки для пошуку потенційних клієнтів, можна визначити якість (або її відсутність) потенційних клієнтів, які вони генерують.

Вирішення суперечок, тобто запис дзвінків можна використовувати як допоміжний засіб для вирішення суперечок. Також можна перевірити, що було домовлено, і використати інформацію для вирішення проблеми. При цьому можна почути, як співробітники справлялися з розлюченими абонентами та чи потребують вони додаткового навчання, щоб допомогти їм впоратися зі складними ситуаціями [4].

Загалом, запис дзвінків може принести користь будь-якому бізнесу кількома способами: для якості та навчання, вирішення суперечок, а також для цілей відповідності. За допомогою правильного рішення можна змінити бізнес в кращу сторону.

Щоб вибрати оптимальну безкоштовну програму для запису дзвінків на Android, варто розглянути найбільш популярні [5].

2.1 Розгляд та аналіз застосунків для запису розмови на безоплатній основі під ОС Android

Щоб вибрати оптимальну програму на безоплатній основі для запису дзвінків на Android, варто розглянути та проаналізувати переваги і недоліки найбільш популярних.

Найкраще всього скачати програму для запису дзвінків з Play Market, що гарантує стабільну роботу та захист від шахраїв. Цей офіційний спосіб найпростіший у використанні та гарантує високий рівень безпеки.

2.1.1 Опис застосунку Another Call Recorder

Another Call Recorder (ACR) один із найпопулярніших на сьогодні додатків. Запускається на будь-якому пристрої під керуванням ОС Android. Завдяки продуманому, і в той же час дуже простому інтерфейсу, воно зручне у використанні (рис.3).



Рисунок 3 – Логотип застосунку Another Call Recorder

ACR (Another Call Recorder) – безкоштовна програма для запису дзвінків. Це один із найкращих і найдосконаліших реєстраторів дзвінків у Play Store і пропонує багато розширених функцій, таких як:

- список записів, зіставляючи номери телефонів із контактами;
- пошук за номером телефону, іменем контакту та приміткою;
- корзина для легкого відновлення видалених записів.

Безпека починається з розуміння того, як розробники збирають і передають дані. Практики конфіденційності та безпеки даних можуть відрізнятися залежно від використання, регіону та віку. Розробник надає інформацію та може оновлювати її з часом.

До переваг даного застосунку можна віднести :

- запускається у фоновому режимі;
- при необхідності можна вимкнути всі повідомлення та відображення статусу роботи, – це дозволяє вести запис непомітно;
- усі дані зберігаються на пристрої, можна задати період, протягом якого зберігатимуться записи;
- після видалення все зберігається в кошику на період, що задається користувачем;
- можливість завантажувати аудіофайли відразу в хмарне сховище, наприклад Google Drive, Gmail, DropBox та інші.

До недоліків слід віднести:

- під час роботи з деякими моделями телефонів запис ведеться некоректно;
- для отримання доступу до розширених функцій доведеться придбати PRO версію.

2.1.2 Опис застосунку TRUECALLER

Truercaller – шведська компанія, заснована в 2009 році в Стокгольмі, Швеція, Намі Заррінгалам і Аланом Мамеді. Програма почалася, коли співзасновники були просто студентами, які хотіли створити службу, яка б легко ідентифікувала вхідні дзвінки з невідомих номерів.

Сьогодні Truercaller (рис.4) користуються понад 330 мільйонів активних користувачів у всьому світі, і це програма для ідентифікації абонента та блокування спаму.



Рисунок 4 – Логотип застосунку Truecaller

Цей софт також можна зустріти на просторах Play Market, і він має гарний рейтинг. Програма для запису телефонних дзвінків дійсно користується великою популярністю. Щоправда, останні версії програми не можуть виконувати поставлену функцію. TRUECALLER спочатку був розроблений як визначник номера та блокатор реклами, дзвінків шахраїв. Запис розмов був додатковою функцією. В Android, починаючи з версії 9.0 на системному рівні почали забороняти запис розмов. Отже, TRUECALLER також відмовився від цієї функції. Тому, як найкращий додаток для запису дзвінків, його розглядати не варто, але впоратися зі своїм завданням може, якщо встановити стару версію програми.

Переваги використання даного софту:

- база даних номерів телефонів та імен контактів, що постійно оновлюється;
- захист від спаму та небажаних дзвінків;
- блокування sms-повідомлень зі спамом;
- власний захищений месенджер, де можна спілкуватися з друзями;
- простий та зрозумілий інтерфейс.

До недоліків варто віднести:

- на Android версії 9.0 та вище заблоковано можливість запису викликів;
- для отримання доступу до всіх функцій потрібна платна передплата.

2.1.3 Опис застосунку RMC: ANDROID CALL RECORDER

На відміну від вище розглянутого прикладу, Record My Call був створений для запису розмов. Але, знову ж таки, внутрішня політика розробників Android внесла корективи і до його роботи. На рівні операційної системи запис вести не є можливим, тому розробники перейшли до радикальнішого рішення. Для запису програма використовує безпосередньо мікрофон пристрою. Це означає, що за звичайної розмови запис буде поганої якості. Щоб отримати результат, під час розмови необхідно включити гучномовець. Тільки тоді звук буде записано. Такий функціонал досить незручний. Тому RMC можна використовувати за відсутності альтернатив (див.рис.5).



Рисунок 5 – Логотип застосунку Record My Call

Переваги використання застосунку:

- запис дзвінків ведеться на будь-яких операційних системах;
- простий та зручний інтерфейс;
- підтримка найпоширеніших форматів аудіофайлів;
- автоматичне завантаження треків у хмарні сховища;
- можна налаштувати сортування записів у декілька папок.

Недоліки використання застосунку:

- запис лише з гучного зв'язку;
- відсутність українськомовного інтерфейсу.

RMC: Android Call Recorder – це безкоштовна службова програма, розроблена Cosonatech для мобільних пристроїв. Ця комунікаційна програма може допомогти користувачам записувати та зберігати свої розмови під час телефонних дзвінків. Він підтримує як вхідні, так і вихідні дзвінки, а також має кнопку ручного запису, яку можна легко активувати.

Подібно до Automatic Call Recorder, RMC: Android Call Recorder дозволяє будь-кому легко отримати копію своїх телефонних розмов для різноманітних цілей, наприклад для безпеки чи архівування. Він підтримує кілька аудіоформатів і пропонує різні параметри керування файлами та налаштування.

RMC: Android Call Recorder – це програма для запису вхідних і вихідних телефонних дзвінків. Він може автоматично записувати всі дзвінки, або можна перемкнути кнопку ручного запису, щоб запустити функцію за потреби. Для файлів можна вибрати формати MP3, AMR, MP4, 3GP або WAV. Користувач навіть може регулювати різні аудіопараметри, такі як збільшення гучності, бітрейт та багато інших.

Після збереження записів програма пропонує налаштування керування файлами. Ви отримуєте стандартні, як-от параметри перейменування та фільтрування файлів, а також сортування їх за двома категоріями: Несортовані або Важливі. Є навіть функція резервного копіювання та відновлення, яка спрощує збереження даних під час перемикання пристроїв. Крім того, можна завантажити все в хмарні служби зберігання, Dropbox і Google Drive.

Більше того, є можливість встановити 4-значний пароль для безпеки та безпеки, а також можливість приховати програму на своєму телефоні, щоб ніхто інший не бачив і не відкривав її. Одне застереження, про яке необхідно знати, полягає в тому, що ця програма може працювати лише з телефонними дзвінками через гучний зв'язок, оскільки вона використовує вбудований мікрофон вашого пристрою для захоплення звуку [6].

2.1.4 Опис застосунку Smart Call Recorder – SCR

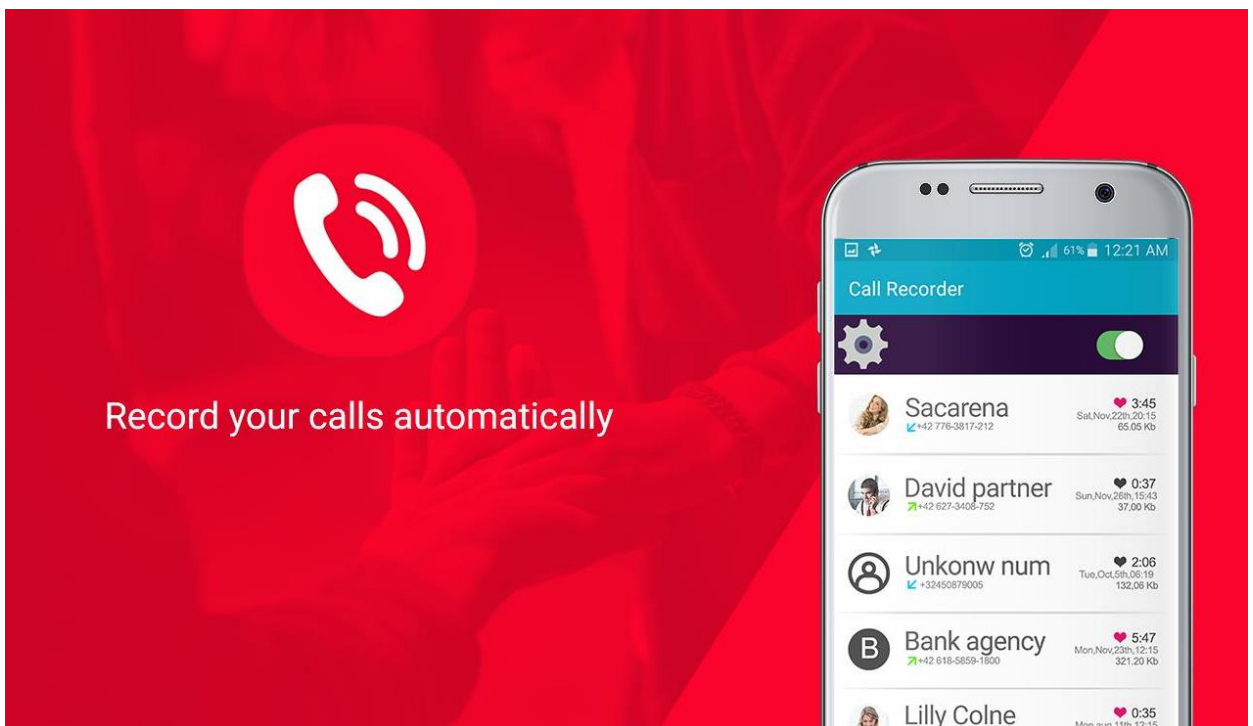


Рисунок 6 – Логотип застосунку SCR

Smart Call Recoder – це безкоштовний додаток (рис.6) для запису дзвінків. Функціонал відповідає поставленому завданню. За допомогою Smart Call Recorder (SCR) можна записувати всі вхідні та вихідні дзвінки. Гнучкі налаштування програми дозволяють все під налаштувати відповідно до вимог конкретного користувача.

До переваг застосунку відноситься:

- постійний запис розмов без додаткових дій;
- вбудовані інструменти для керування аудіо файлами;
- можливість пересилати фрагменти розмов із програми;
- функція затримки початку запису;
- усі функції безкоштовні.

До недоліків:

- збої у роботі на деяких нових пристроях;
- підпис файлів у форматі номера телефону, а не імені телефону.

SCR – це зручна розробка для здійснення запису телефонних розмов.

Програма спеціально спроектована для широкого кола користувачів, а тому має зручну панель керування та цілком зрозумілу систему налаштувань, щоб з процесом запису зміг розібратися будь-хто. Користувач зможе налаштувати якість запису та формат кінцевого файлу, вибрати шлях збереження, вказавши SD-карту, внутрішнє або інтернет-сховище. Є можливість чітко вказати тип розмов, які підлягають запису – всі дзвінки, лише вхідні, вихідні, і навіть розмови з конкретним абонентом чи певний час доби. Залежно від вибраної якості, змінюватиметься розмір отриманого файлу. Є можливість вказати максимальний обсяг запису та кількості файлів у сховищі, що дозволить автоматично видаляти ранні збереження. Всі сучасні власники пристроїв отримають можливість провести масу додаткових налаштувань.

2.1.5 Опис застосунку Cube ACR

Найбільш стабільний і функціональний додаток з аналогів, особливу увагу потрібно звернути на Cube ACR. Характерною особливістю цього рішення є можливість запису як телефонних дзвінків, і викликів VoIP (Viber, WhatsApp, Telegram). Це безперечний плюс, тому що месенджери з кожним днем все більше входять у наше життя. Як і в більшості програм цього типу, деякі нові пристрої можуть працювати некоректно. Однак розробники Cube

ACR подбали про це, забезпечивши зворотний зв'язок та якісну техпідтримку своїх клієнтів (рис.7).

Переваги використання застосунку:

- постійне оновлення;
- хороша техпідтримка;
- підтримка VoIP; гнучка настройка;
- інструменти для керування аудіо файлами.

Недоліки використання застосунку:

- проблеми із записом дзвінків на нових пристроях;
- для отримання доступу до розширених функцій необхідно придбати PRO-версію.



Рисунок 7 – Логотип застосунку Cube ACR

Найкращий додаток для запису дзвінків для Android та iOS. Cube ACR записує телефонні дзвінки та VoIP-розмови на вашому пристрої Android, а також дозволяє записувати телефонні дзвінки та створювати голосові нагадування на iPhone.

Cube ACR для Android дозволяє записувати дзвінки на мобільний телефон, записувати дзвінки WhatsApp і розмови в інших VoIP-додатках і месенджерах, таких як LINE, Viber, Skype, WeChat і багато інших.

Можна записувати вхідні та вихідні дзвінки в найкращій якості за допомогою Cube Call Recorder. Для цього необхідно обрати із багатьох варіантів запису та джерел, щоб знайти той, який найбільше підходить.

2.1.6 Опис застосунку Automatic Call Recorder callX

Розробники callX зробили спробу створити кращу програму для запису дзвінків на Android, в якій також вбудований блокувальник реклами та визначник номера. Більше того, у них це вийшло і досить непогано. Automatic Call Recorder (див.рис.8) дозволяє автоматично записувати всі дзвінки, а також захистити себе від шахраїв та спаму. Те, що програма бере до уваги номер, з якого здійснюється дзвінок, а не записує всі дзвінки поспіль, дозволяє оптимізувати його роботу. Можна задати номери, дзвінки з яких не будуть збережені. Або навпаки, створити список імен, які мають бути записані в обов'язковому порядку. Інші дзвінки збережені не будуть.

Переваги використання даного застосунку:

- гнучка настройка запису;
- можливість завантажувати файли у хмарне сховище;
- приємний та інтуїтивний інтерфейс;
- можна поставити, які розмови будуть записані, а які ні;
- вбудований плеєр аудіофайлів.

Недоліки використання застосунку:

- проблеми у роботі на нових пристроях;
- можливі перебої в записі під час використання гарнітури.



Рисунок 8 – Логотип застосунку Automatic Call Recorder callX

CallX автоматично записує всі вхідні та вихідні дзвінки. Automatic Call Recorder пропонує унікальний набір функцій, які дозволять автоматично записувати всі вхідні та вихідні дзвінки. Крім того, можна просто струснути пристрій під час розмови, щоб миттєво почати запис. Користувач може будь-коли прослухати записаний дзвінок у форматі MP3 High Audio Quality або автоматично синхронізувати його зі зв'язаним обліковим записом Dropbox.

Auto Call Recorder – це найкраща програма для автоматичного запису дзвінків, яка дозволяє автоматично записувати дзвінки на пристрій Android. Додаток також містить ідентифікатор абонента, який допоможе вам ідентифікувати телефонні дзвінки та уникнути спаму.

Якщо потрібне найкраще рішення для запису телефонних розмов на пристрої Android, слід взяти під контроль телефонні дзвінки та обрати застосунок callX, найкращу програму для запису дзвінків у 2020 році з красивим і сучасним дизайном в останній версії. Доступна нова версія на 2020 рік.

Ключові функції запису дзвінків:

- записуйте дзвінки за номером телефону або іменем контакту;
- виключити запис за номером телефону або ім'ям контакту;

- ідентифікатор абонента визначає невідомі телефонні номери;
- ручний і автоматичний запис голосу обох сторін;
- виберіть між високоякісними аудіоформатами HD MP3 і WAV;
- завантажте розмови дзвінків на Dropbox або Google Drive;
- відтворення аудіозаписів розмов;
- параметр фільтра запису дозволяє записувати всі дзвінки, контакти або лише невідомі номери;
- відтворення через динамік або навушник на пристрої;
- Запис вхідних і вихідних дзвінків.

За допомогою застосунку можна поділитися файлами з використанням наступних сервісів:

- Dropbox;
- Диск Google;
- СМС;
- Whatsapp;
- Viber;
- Skype.

2.1.7 Опис застосунку Auto Call Recording

Одна з найменш відомих та поширених на сьогоднішній день додатків. Пов'язано це з тим, що проект є досить новим, і знаходиться на стадії розробки. Це не означає, що це рішення не варте уваги, оскільки воно щодня вдосконалюється та оновлюється. Тисячі користувачів щодня висловлюють свої побажання розробникам, які досить успішно реалізують. Самі ж творці закликають перед покупкою PRO-версії перевірити працездатність безкоштовної версії і лише тоді перейти до версії на платній основі (рис.9).

Переваги використання Auto Call Recording:

- постійне вдосконалення та доопрацювання програми;
- приємний дружній інтерфейс;

- хороша техпідтримка та зв'язок з розробниками;
- можливість підпису файлів іменами контактів, що полегшує пошук;
- синхронізація з Google Drive та DropBox.

Недоліки використання вищеописаного застосунку:

- нестабільна робота на початкових етапах після встановлення;
- для отримання доступу до розширених функцій доведеться придбати підписку PRO.



Рисунок 9 – Логотип застосунку Auto Call Recording

Цей інструмент дозволяє вибрати, які дзвінки зберегти, а які видалити. Коли розмову буде збережено, ви можете додати примітки, щоб уточнити будь-яку інформацію, яку ви не хочете забути, прослуховуючи її знову. Користувач також може синхронізувати ці розмови з хмарою, щоб зберегти їх у безпеці, якщо щось трапиться з вашим телефоном.

Користувач також може вибрати папку призначення для всіх записів і спосіб їх запису. Є три можливі налаштування, але за умовчанням програма записуватиме всі телефонні розмови, незалежно від того, надсилаєте ви їх чи

отримуєте. Існує також функція «ігнорувати всіх», щоб дзвінки не записувалися, і ще одна, щоб ігнорувати певних людей, тому програма записуватиме лише розмови з людьми у вашому списку.

Особливості використання застосунку Auto Call Recording:

- запис дзвінків є автоматичним і дуже надійним;
- це дозволяє встановити, які дзвінки записуються в білий список, а які ігноруються;
- можна зберігати записи на SD-карту;
- є можливість запису всіх телефонних розмов;
- відтворення аудіозаписів розмов;
- позначення записів як важливих.

2.1.8 Опис застосунку Voice Call Recorder

Розробники з lovekara створили зручний інструмент фіксації дзвінків. Грунтуючись на відгуках користувачів, можна сказати, що програма досить стабільно працює і чудово справляється зі своїм завданням. Простий у використанні, без безлічі непотрібних налаштувань, Voice Call Recorder підійде для виконання покладеної функції (рис.10).

Переваги функцій у застосунку :

- можливість зберігати файли на картці пам'яті, не завантажуючи пам'ять телефону;
- простота пошуку потрібного файлу завдяки різним варіантам сортування;
- інтуїтивно зрозумілий інтерфейс;
- вбудований аудіо плеєр;
- усі дзвінки записуються автоматично.

Недоліки функцій у застосунку:

- можливі проблеми з пристроями на базі Android 9.0 або вище;
- назви запису формуються довільно.

Call Recorder

lovekara

Contains ads

4.0★
442K reviews

10M+
Downloads

E
Everyone



Install

You don't have any devices

You can share this with your family. [Learn more about Family Library](#)



Рисунок 10 – Логотип застосунку Voice Call Recorder з Google Play

Call Recorder від Lovekara – це програма, яку можна встановити на свій телефон Android, як забажаєте, для легкого запису дзвінків. Найпростіший користувальницький інтерфейс спрощує роботу з ним для користувачів усіх вікових груп.

Додаток допоможе записати всі дзвінки з хорошою якістю звуку. Після запису ви можете відтворювати їх у форматі mp3 і також зберігати на SD-карті. На відміну від інших додатків, цей додаток є абсолютно безкоштовним і без жодних прихованих платежів.

Даний застосунок має досить багато особливостей, а саме:

- запишіть усі аудіо в mp3, що може заощадити багато місця (зі старим можна було записувати лише години голосу, а з новим – сотні годин);


- сім різних варіантів якості аудіо, користувач може вибрати аудіо високої якості, щоб отримати кращий звук, або вибрати низьку якість, щоб заощадити набагато більше місця.
- запис аудіо можливий у фоновому режимі, щоб користувач міг робити інші речі під час запису.
- легко керувати всіма аудіозаписами, можна відтворювати, ділитися, видаляти та переглядати деталі записів дуже простим способом.

Додаток для запису дзвінків – Lovekara, що найважливіше, безкоштовний додаток. Це ставить його вище всіх інших програм для запису дзвінків для пристроїв Android. Додаток має простий інтерфейс, який робить його зручним для розуміння функцій.

Сортування записів викликів також полегшить пошук конкретного запису. Хоча відсутність інтеграції резервного копіювання в хмару робить це певним фоном. Інша особливість, про яку повідомляють численні відгуки користувачів, полягає в тому, що програма повинна працювати для сповіщень про зберігання. Користувач досягнете максимального ліміту пам'яті для записів дзвінків.

2.1.9 Опис застосунку CryOk Call recorder

Застосунок Самописець від CRYOK SIA можна віднести до списку найбільш ефективних рішень для запису викликів. Без зайвих вбудованих функцій, застосунок призначено виключно для запису, прослуховування та зберігання аудіофайлів з розмовами. Інтегруючись в ОС Android, Самописець виконує якісний запис голосових викликів. Окрім цього, він захищений від несанкціонованого доступу, дозволяє задати пароль (див. рис.11).



Call recorder

CRYOK SIA Tools ★★★★★ 59,720

3+

Contains Ads · Offers in-app purchases

i This app is compatible with some of your devices.

[Add to Wishlist](#) [Install](#)

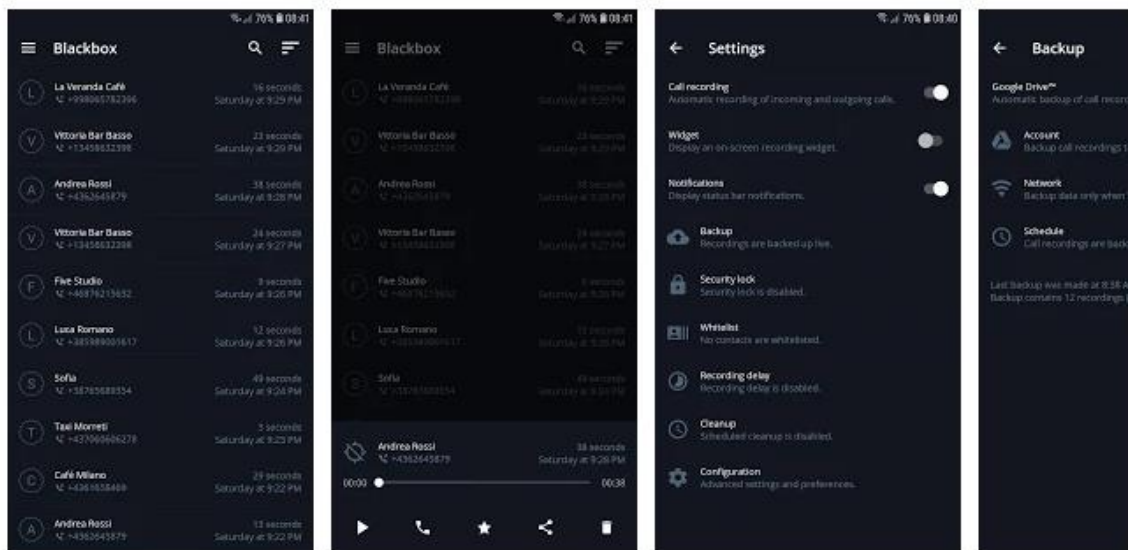


Рисунок 11 – Логотип застосунку Call Recorder з Google Play

Преваги використання Call Recorder:

- додаток розроблений спеціально для запису дзвінків, що робить його найбільш надійним рішенням;
- захист від несанкціонованого доступу;
- приємний інтерфейс;
- гнучка настройка;
- вбудований аудіоплеєр для відтворення розмов.

Недоліки використання Call Recorder:

- проблеми із роботою на нових пристроях;
- порушення стабільності після виходу оновлень.

Даний застосунок не підтримує VoIP-дзвінки за допомогою WhatsApp, Messenger, Skype, Viber тощо, а також він не сумісний з іншими програмами запису розмов.

Застосунок Call Recorder, має особливості, а саме:

- захист від несанкціонованого доступу;
- зберігання та відновлення даних з Google Drive;
- пошук, сортування та додавання записів у вибране;
- приховування ваших записів від інших програм;
- білий список контактів;
- видалення старих записів за автоматичним розкладом;
- відтворення вхідних та вихідних записів;
- запис вручну з використанням віджету;
- сортування записів за датою, часом та тривалістю;
- додавання важливих записів до вибраного;
- підтримка датчика наближення під час відтворення;
- підтримка Bluetooth;
- відображення фото, ініціалів або імені контакту;
- вибір папки для збереження файлів;
- підтримка пристроїв Dual SIM;
- підтримка високої якості або компактності при виборі формату файлів;
- масовий вибір та видалення записів;
- вибір папки для файлів даних ;
- затримка запису вихідних дзвінків.

2.1.10 Опис застосунку Callbox

Рішення від SMSROBOT LTD є перевіреним часом програмою для збереження історії телефонних розмов. З його допомогою всі дзвінки будуть записані та збережені. Куди зберігати файли, вирішує користувач, вибираючи

між локальним зберіганням, або ж надсиланням у хмарні сховища, такі як DropBox, Google Drive та подібні до них (рис.12).

До переваг використання застосунку можна віднести:

- зручний функціонал, орієнтовані під конкретне завдання;
- гнучкі налаштування;
- можливість увімкнути збереження запису залежно від тривалості розмови;
- можна зберігати розмови в хмарних сховищах;
- програма стабільно оновлюється для коректної роботи на нових пристроях.

До недоліків в свою чергу слід віднести:

- збої у роботі на Android версії вище 9.0;
- під час розмови через навушники або гарнітуру запис може бути неправильним.

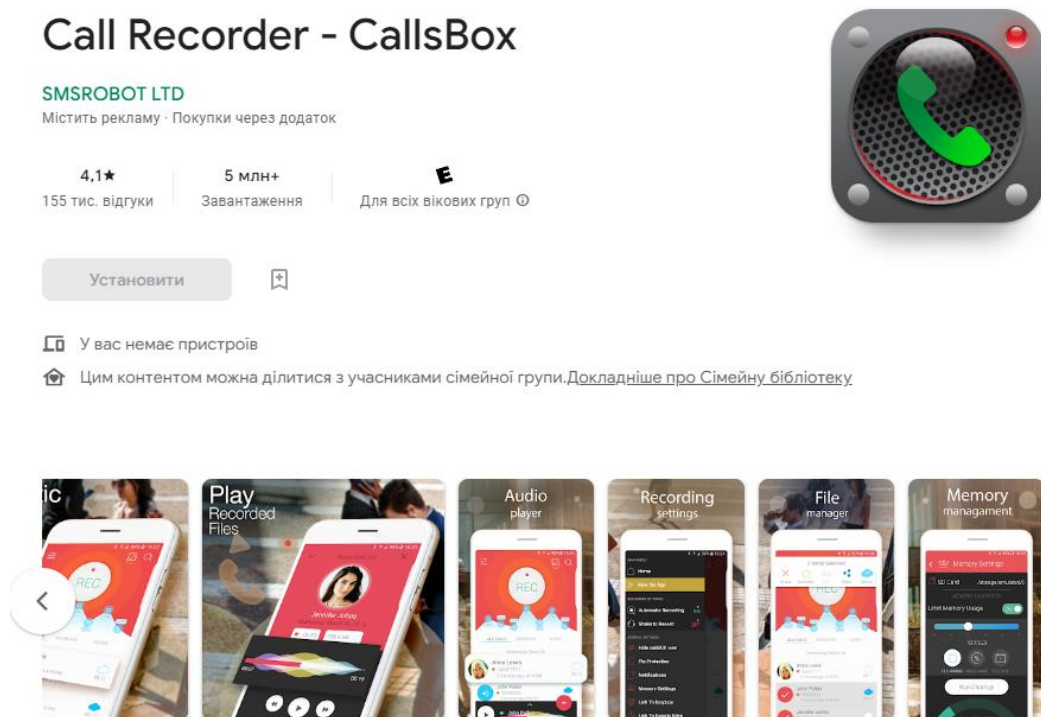


Рисунок 12 – Логотип застосунку Callbox з Google Play

Call Recorder – найкраща програма автоматичного запису дзвінків, яка дозволяє автоматично записувати дзвінки на ваш пристрій Android. Додаток також має ідентифікатор абонента, який допоможе вам ідентифікувати телефонні дзвінки та уникнути спаму.

Автоматичний реєстратор дзвінків – дає можливість приховати піктограму програми та запису дзвінків за допомогою callBOX PRO. Реєстратор виликів (callBOX PRO) – найбільш повний додаток для запису дзвінків у Google Play Store у 2018 році. Всі опції PRO тепер повністю безкоштовні, включаючи приховування піктограми програми, автоматичну синхронізацію з Dropbox та Google Drive, струшування пристрою для запису та багато інших опцій. Користувач також може записувати файли у форматі MP3 із чотирма різними рівнями якості. Програму можна захистити пін-кодом для збереження конфіденційності.

Безліч функцій для запису дзвінків, все, що потрібно користувачу, – це програма для запису дзвінків

До функцій даного застосунку можна віднести:

- оптимізовано для нових пристроїв, таких як Samsung galaxy S9 та S9 plus;
- автоматичний запис дзвінків;
- запис розмови під час дзвінка;
- запис вхідного дзвінка;
- запис Вихідний дзвінок;
- запис усіх телефонних розмов;
- записувати дзвінки під час розмови автоматично.

Автоматичний запис викликів ніколи не був простішим, ніж з Callbox. З цього моменту користувачу не потрібно турбуватися про необхідність робити нотатки або запам'ятовувати важливу інформацію під час розмови. Розробники додали до застосунку дуже багато безкоштовних опцій, зробивши Callbox найбільш повним додатком Автоматичного реєстратора дзвінків у 2018 році.

До особливостей слід віднести наступне:

- запис розмов;
- відтворення записаних аудіо;
- запис телефонних розмов;
- диктофон для запису телефонних розмов;
- записати телефонну розмову;
- записує всі телефонні дзвінки.

У застосунку передбачено ділитися файлами за допомогою наступних інструментів:

- Dropbox;
- Google;
- СМС;
- Skype;
- Facebook.

Загальна порівняльна таблиця (табл.1) містить узагальнюючі дані щодо переваг та недоліків інструментів/застосунків з функцією запису дзвінків.

Таблиця 1 – Переваги та недоліки найпопулярніших застосунків з функцією запису дзвінків

№	Назва застосунку	+	-
1	2	3	4
1	Another Call Recorder	<ul style="list-style-type: none"> – запускається у фоновому режимі; – при необхідності можна вимкнути всі повідомлення та відображення статусу роботи, – це дозволяє вести запис непомітно; – усі дані зберігаються на пристрої, можна задати період, протягом якого зберігатимуться записи; – після видалення все зберігається в кошику на період, що задається користувачем; – можливість завантажувати аудіофайли відразу в хмарне сховище, наприклад Google Drive, Gmail, DropBox та інші. 	<ul style="list-style-type: none"> – під час роботи з деякими моделями телефонів запис ведеться некоректно; – для отримання доступу до розширених функцій доведеться придбати PRO версію.

Продовження таблиці 1

1	2	3	4
2	TRUE-CALLER	<ul style="list-style-type: none"> – база даних номерів телефонів та імен контактів, що постійно оновлюється; – захист від спаму та небажаних дзвінків; – блокування sms-повідомлень зі спамом; – власний захищений месенджер, де можна спілкуватися з друзями; – простий та зрозумілий інтерфейс. 	<ul style="list-style-type: none"> – на Android версії 9.0 та вище заблоковано можливість запису викликів; – для отримання доступу до всіх функцій потрібна платна передплата.
3	Record My Call	<ul style="list-style-type: none"> – запис дзвінків ведеться на будь-яких операційних системах; – простий та зручний інтерфейс; – підтримка найпоширеніших форматів аудіофайлів; – автоматичне завантаження треків у хмарні сховища; – можна налаштувати сортування записів у декілька папок. 	<ul style="list-style-type: none"> – запис лише з гучного зв'язку; – відсутність українськомовного інтерфейсу.
4	Smart Call Recorder	<ul style="list-style-type: none"> – постійний запис розмов без додаткових дій; – вбудовані інструменти для керування аудіо файлами; – можливість пересилати фрагменти розмов із програми; – функція затримки початку запису; – усі функції безкоштовні. 	<ul style="list-style-type: none"> – збої у роботі на деяких нових пристроях; – підпис файлів у форматі номера телефону, а не імені телефону.
5	Cube ACR	<ul style="list-style-type: none"> – постійне оновлення; – хороша техпідтримка; – підтримка VoIP; гнучка настройка; – інструменти для керування аудіо файлами. 	<ul style="list-style-type: none"> – проблеми із записом дзвінків на нових пристроях; – для отримання доступу до розширених функцій необхідно придбати PRO-версію.
6	Automatic Call Recorder callX	<ul style="list-style-type: none"> – гнучка настройка запису; – можливість завантажувати файли у хмарне сховище; – приємний та інтуїтивний інтерфейс; – можна поставити, які розмови будуть записані, а які ні; – вбудований плеєр аудіофайлів. 	<ul style="list-style-type: none"> – проблеми у роботі на нових пристроях; – можливі перебої в записі під час використання гарнітури.
7	Auto Call Recording	<ul style="list-style-type: none"> – постійне вдосконалення та доопрацювання програми; – приємний дружній інтерфейс; – хороша техпідтримка та зв'язок з розробниками; – можливість підпису файлів іменами контактів, що полегшує пошук; – синхронізація з Google Drive та DropBox. 	<ul style="list-style-type: none"> – нестабільна робота на початкових етапах після встановлення; – для отримання доступу до розширених функцій доведеться придбати підписку PRO.

Продовження таблиці 1

1	2	3	4
8	Voice Call Recorder	<ul style="list-style-type: none"> – можливість зберігати файли на картці пам'яті, не завантажуючи пам'ять телефону; – простота пошуку потрібного файлу завдяки різним варіантам сортування; – інтуїтивно зрозумілий інтерфейс; – вбудований аудіо плеєр; – усі дзвінки записуються автоматично. 	<ul style="list-style-type: none"> – можливі проблеми з пристроями на базі Android 9.0 або вище; – назви запису формуються довільно.
9	Call Recorder	<ul style="list-style-type: none"> – додаток розроблений спеціально для запису дзвінків, що робить його найбільш надійним рішенням; – захист від несанкціонованого доступу; – приємний інтерфейс; – гнучка настройка; – вбудований аудіоплеєр для відтворення розмов. 	<ul style="list-style-type: none"> – проблеми із роботою на нових пристроях; – порушення стабільності після виходу оновлень.
10	Callbox	<ul style="list-style-type: none"> – зручний функціонал, орієнтовані під конкретне завдання; – гнучкі налаштування; – можливість увімкнути збереження запису залежно від тривалості розмови; – можна зберігати розмови в хмарних сховищах; – програма стабільно оновлюється для коректної роботи на нових пристроях. 	<ul style="list-style-type: none"> – збої у роботі на Android версії вище 9.0; – під час розмови через навушники або гарнітуру запис може бути неправильним.

3 ОПИС ОСНОВНИХ ЕТАПІВ ПРИ РОЗРОБЦІ ЗАСТОСУНКУ ДЛЯ ЗАПISУ ДЗВІНКІВ

Життєвий цикл розробки програмного забезпечення або SDLC – це процес чіткого визначення цілей і етапів побудови програмного рішення. Він включає в себе кілька основних кроків:

- визначення проблем;
- технічні характеристики;
- дизайн;
- розробка та кодування;
- тестування;
- розгортання;
- обслуговування.

Основною причиною проходження всіх етапів SDLC є задоволення всіх вимог і очікувань клієнтів з урахуванням термінів і початкової оцінки проекту. Розробка будь-якого індивідуального програмного рішення вимагає не тільки сильної технічної частини. Кожен клієнт повинен бути в курсі нових етапів розробки, змін або оновлень.

Перед початком розробки необхідно визначити основні завдання та структурувати їх у логічну послідовність. SDLC допомагає зробити його легшим і організованим завдяки своїм простим етапам, які містять певні завдання. Серед основних переваг розгляду всіх етапів SDLC є:

- надання клієнтам ясності щодо процесу розробки;
- відстеження всіх завдань і термінів;
- точні прогнози доставки;
- допомогти бути в рамках бюджету;
- суворе дотримання всіх вимог клієнтів.

Життєвий цикл розробки програмного забезпечення – це неминучий і корисний набір правил, який допомагає випускати цілісне та багатофункціо-

нальне програмне забезпечення, яке відповідає всім вимогам клієнтів і перетворює їхні ідеї на завдання. Це також посібник для членів команди розробників, які займаються технікою та не займаються технікою. Загалом, SDLC керує всіма, хто бере участь у процесі розробки, показуючи їм чіткий план, як досягти цілей і які ресурси для цього потрібні.

Етапи процесу розробки програмного забезпечення (ПЗ) є загальними, але їх зміст залежить від обраної моделі SDLC.

Життєвий цикл розробки програмного забезпечення підвищує ефективність і якість послуг шляхом організації робочого процесу та дотримання його етапів. Це також допомагає зменшити або усунути непотрібні витрати та залишатися в межах бюджету. Коли справа доходить до процесу розробки, першим кроком є визначення проблем, з якими кожен день стикається замовник, і визначення основних цілей впровадження програмного рішення. Цей етап є ключовим, оскільки він визначає наступні кроки розвитку. Планування є ключем до успіху в усіх сферах, особливо коли важливі деталі. Це знижує ймовірність підводних каменів і помилок програмного рішення.

Після аналізу вимог і створення планів і завдань починається власне етап розробки, який включає частину дизайнерського рішення, кодування та тестування. Частина тестування вимагає високого рівня відповідальності та уваги з боку QA спеціалістів. Цей етап необхідний, щоб виявити всі баги та помилки в програмному рішенні та усунути їх до випуску, щоб забезпечити відмінний UX. Етап тестування допомагає заощадити робочий час і кошти, оскільки він показує слабкі сторони, коли ще не пізно.

3.1 Етапи розробки програмного забезпечення

Перший етап розробки – планування. Жоден проект не може розпочатися й успішно розвиватися без плану догляду. Усі цілі, завдання та ризики рішення мають бути ретельно визначені та структуровані з урахуванням будь-яких деталей або потенційних проблем, які можуть виникнути під час процесу.

Звичайно, неможливо згадати всі, але досвідчений бізнес-аналітик проаналізує конкретну сферу бізнесу і визначить основні ризики. Планування – це початковий крок кожного SDLC, який допомагає виявити основні труднощі, які будуть вирішені за допомогою програмного забезпечення, і створити завдання для досягнення цілей. Цей етап також визначає шляхи розвитку, які будуть відповідати всім запитам бізнесу з урахуванням часу, бюджету, кількості спеціалістів тощо.

Ініціативи, епіки та історії користувачів – це другий етап. Цей етап процесу розробки програмного забезпечення можна назвати ключовим. Важливо зібрати всю інформацію та бізнес-вимоги від власників бізнесу, щоб побачити повну картину. Це завдання є відповідальністю бізнес-аналітика, і воно може бути ідеально виконано лише за умови постійного тісного спілкування з клієнтами. Усі потреби та вимоги бізнесу згодом трансформуються у завдання для розробників. Після визначення основних вимог до програмного рішення фахівці створюють план необхідних функцій, які необхідно реалізувати, щоб задовольнити всі вимоги. Усі цілі розділені на три частини розробки: ініціативи, епопеї та історія користувача.

Ініціативна група є найбільшою і включає епіки та історії користувачів. Епопеї – це менша частина розвитку, яка йде після ініціативи. Зазвичай він включає основні функції розробки, свого роду заповнення розчину. Історія користувача містить детальну інформацію про функціонал, це конкретне і точне завдання в епопеї. На цьому етапі також передбачається оцінка MVP продукту з урахуванням годин роботи кожного розробника.

Наступний етап, створення дизайну. Цей етап розробки програмного забезпечення передбачає створення концепції рішення. На цьому етапі замовники можуть побачити всю конструкцію та остаточний прототип своїх розробок. Це означає, що потрібно зібрати всі документи і поставити всі завдання на відповідальних розробників. Це останній етап перед фактичним початком розробки.

Далі, етап розробки (кодування), мова йде про написання коду та прихованих епосів та історій користувачів у справжнє програмне забезпечення. Цей етап SDLC вважається найтривалішим і вимагає прискіпливої уваги до кожної деталі. Код має відповідати всім вимогам і специфікаціям програмного забезпечення, які були визначені раніше.

Етап тестування отримує найвищий ранг відповідальності та необхідності після етапу планування. Тестування є важливим етапом життєвого циклу розробки програмного забезпечення. У міру створення програмного рішення його повинні протестувати інженери з контролю якості, щоб виявити будь-яку помилку або помилку під час використання рішення. Існують різні види тестування та способи виконання цього етапу. Зазвичай спеціалісти з контролю якості виконують ряд тестів, які включають тестування функціональності, системної інтеграції, сумісності, прийняття користувачами тощо. Інженери QA проводять різні тести для додатка, щоб відстежити його роботу під час перевантаження та виявити баги та помилки. Після перевірки та перевірки програмного рішення його можна випустити або впровадити. Тестування допомагає уникнути низьких відгуків перших користувачів і робить додаток більш кваліфікованим. Це останній етап перед випуском програми.

Завершальний етап розробки – випуск або інтеграція готового рішення (розгортання). Процес реалізації також має план, який пишеться на етапі планування. Після успішного завершення всіх етапів програмне забезпечення переходить до виробництва, що залежить від складності розробки. Прямий випуск призначений для простого рішення, поступовий випуск означає поетапне виробництво, якщо рішення складне. І тепер рішення готове випробувати кінцеві користувачі. Існують спеціальні політики та набори правил, яких слід дотримуватися, щоб подати програму на такі майданчики як App Store і Google Play [7].

Виготовлення рішення недостатньо, оскільки воно вимагає постійних оновлень і вдосконалень. Це пов'язано з відгуками користувачів або потребами змін у технічній частині рішення. Етап [8] технічного обслуговування

допомагає тримати розроблене рішення попереду конкурентів, додавати нові опції, залишатися сучасним і вчасно оновлюватися (рис. 13).

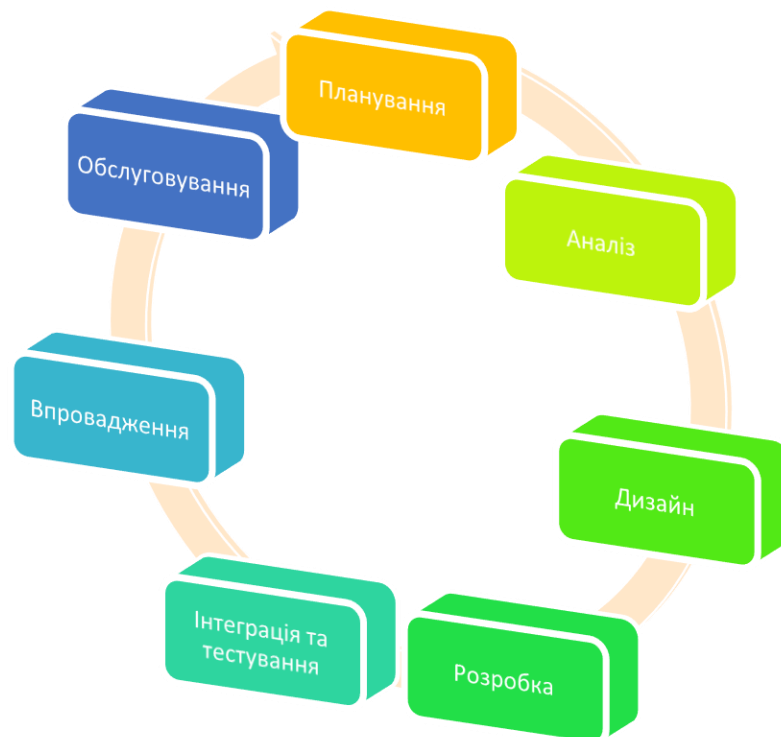


Рисунок 13 – Етапи розробки програмного продукту

4 РОЗРОБКА ANDROID-ЗАСТОСУНКУ ЗАПИСУ ДЗВІНКІВ

Сьогодні для розробки застосунку використовується та розробляється багато бібліотек. Наприклад, бібліотеки підтримки Android до прийняття Kotlin як основної мови для рідної Android і впровадження Android Jetpack, Google постійно прагне зробити розробку програм для Android легшою, швидшою та більш масштабованою.

З нещодавньої появою Jetpack Compose створювати програму для Android з нуля, використовуючи сучасні бібліотеки, найкращі практики та архітектурні моделі. Хоча збереження старого коду Android – це одне, а почати з чистого аркуша – це зовсім інше. Переглядаючи потенційні програми для створення автоматичного запису дзвінків. На жаль, це виявилось неможливим через обмеження Google на запис дзвінків у нових версіях Android.

Android Studio – це офіційне інтегроване середовище розробки (IDE) для розробки додатків Android на основі IntelliJ IDEA. Крім потужного редактора коду IntelliJ та інструментів розробника, Android Studio пропонує ще більше функцій, які підвищують вашу продуктивність під час створення програм для Android, такі як

- гнучка система збирання на основі Gradle;
- швидкий і багатофункціональний емулятор;
- уніфіковане середовище, де можна розробляти для всіх пристроїв Android;
- застосуйте зміни до push-коду та змін ресурсів до запущеної програми без перезапуску програми;
- шаблони коду та інтеграція GitHub, які допоможуть вам створювати загальні функції програми та імпортувати зразок коду;
- широкі інструменти та фреймворки тестування;
- інструменти Lint для визначення продуктивності, зручності використання, сумісності версій та інших проблем;
- підтримка C++ і NDK;

- вбудована підтримка Google Cloud Platform, що полегшує інтеграцію Google Cloud Messaging і App Engine.

По-перше варто налаштувати проект та розробити простого та зручний інтерфейсу користувача (UI) для застосунку. Слід розпочати з перевірки можливості запуску програми Android із підтримкою Compose. Для цього варто оновити версію Android Studio Arctic Fox або новіший дистрибутив.

Це забезпечить користь від функцій розумного редактора та попереднього перегляду створення. Далі це створення нового проекту із підтримкою Compose. Для цього необхідно увійти у вікно New Project і обрати «Empty Compose Activity» (рис. 14).

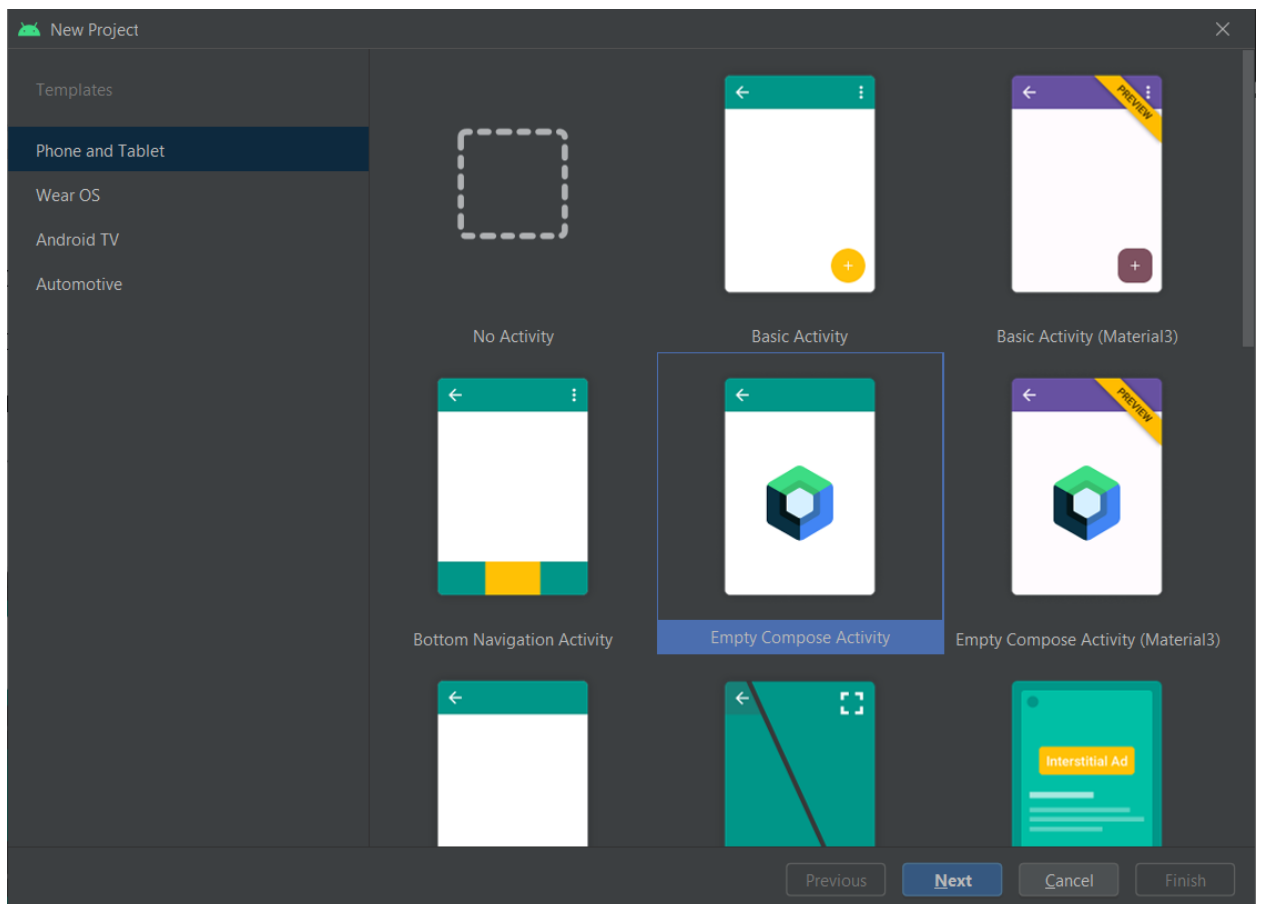


Рисунок 14 – Головний екран середовища Android Studio

Android Studio Dolphin додає підтримку для створення пустої програми Compose. Далі слід заповнити стандартні поля (назва програми, назва пакета)

і встановіть для мінімального SDK значення API 21:Android 5.0 (Lollipop) і натисніть «Готово» (див.рис.15).

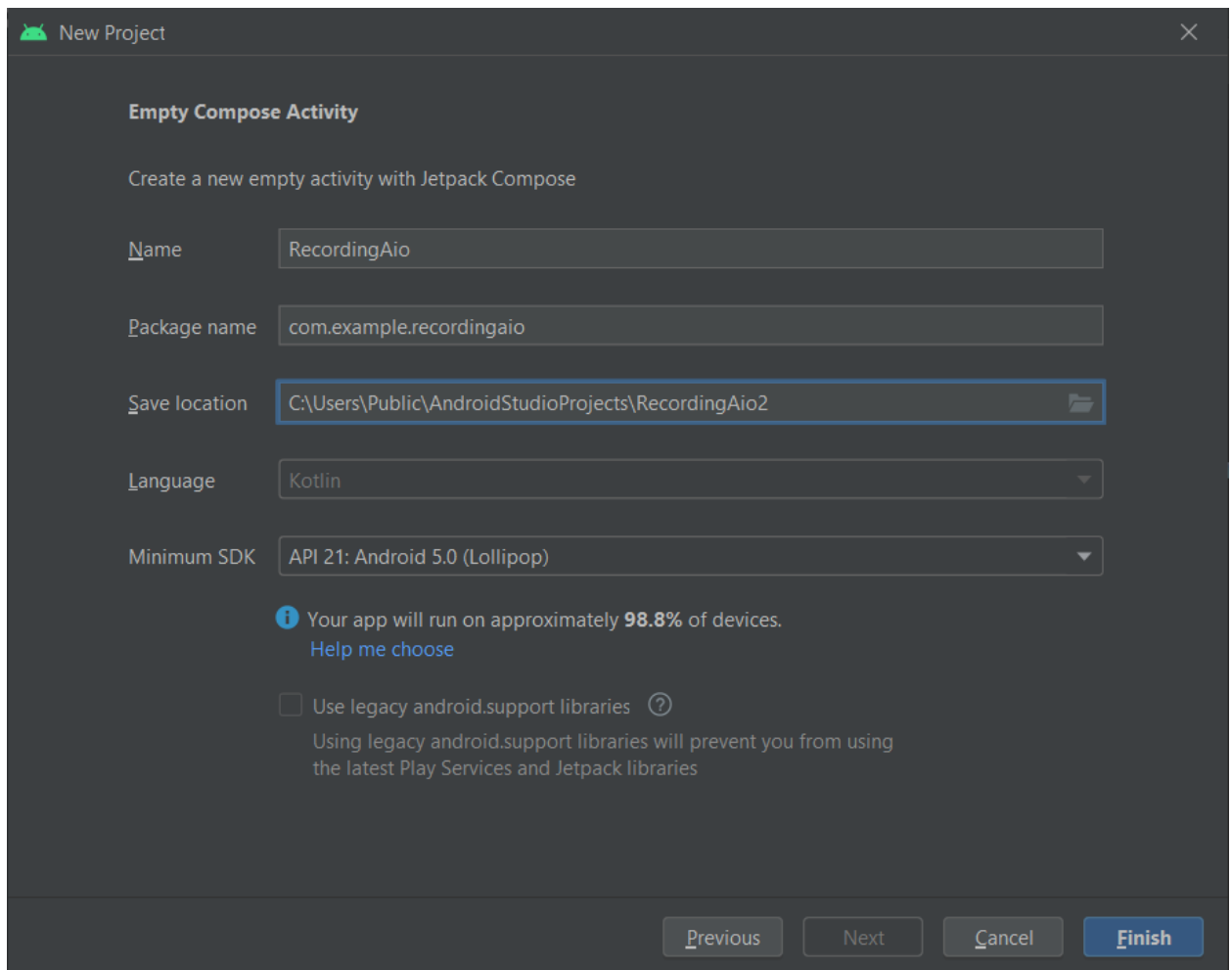


Рисунок 15 – Екран створення нового проекту
в середовищі Android Studio

Оскільки нові проекти створюються за допомогою Gradle 7 як системи збірки, також варто переконатися, що у для проекту встановлено принаймні Java 11. Щоб переконатися, що проект використовує цю версію, можна приблизно дотримуватися наступного контрольного списку:

- переконатися, що на рівні програми build.gradle(розташований за адресою) і встановлено значення (вони розташовані під); якщо у встановлена інша сумісна версія, можна використовувати її замість неї

(наприклад, також буде працювати) `.app/build.gradle` source `CompatibilitytargetCompatibilityJavaVersion.VERSION_11` android -> `compileOptionsJavaVersion.VERSION_12`;

- встановити сумісний Gradle JDK у параметрах Gradle; можна отримати доступ до них із вікна «Структура проекту» в розділі «Розташування SDK» (`Ctrl+Alt+Shift+S` у Windows/Linux , `⌘`; у macOS).

```
// ...
android {
    // ...
    compileOptions {
        sourceCompatibility JavaVersion.VERSION_11
        targetCompatibility JavaVersion.VERSION_11
    }
    // ...
}
```

Слід встановити для Gradle JDK сумісну версію. Коли Android Studio завершує створення проекту, на екрані відобразиться `MainActivity.kt`, головна точка входу для програми.

Jetpack Compose вже випущений у стабільній версії 1.0, що означає, що він більше не перебуває в бета-версії та вважається готовим до виробництва. Jetpack Compose – це сучасний набір інструментів для створення рідного інтерфейсу Android. Jetpack Compose спрощує та прискорює розробку інтерфейсу користувача на Android за допомогою меншого коду, потужних інструментів та інтуїтивно зрозумілих API Kotlin.

По-перше, важливо зазначити, що Jetpack Compose it Kotlin Only . Основна причина полягає в тому, що Compose активно використовує співпрограми Kotlin.

Якщо подивитися на основну діяльність, яку створила для Studio, то виходить.

```
class MainActivity : ComponentActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
    }
}
```

```

        setContent {
            RecordingAioTheme {
                // A surface container using the 'background' color from the theme
                Surface(color = MaterialTheme.colors.background) {
                    Greeting("Android")
                }
            }
        }
    }

    @Composable
    fun Greeting(nameObj: String) {
        Text(text = "Hi $nameObj!")
    }

    @Preview(showBackground = true)
    @Composable
    fun DefaultPreview() {
        RecordingAioTheme {
            Greeting("Android")
        }
    }
}

```

`MainActivity` успадковує від `ComponentActivity`, а не традиційне `AppCompatActivity`. Це робиться для того, щоб можна було використовувати новий `setContent` метод розширення, який приймає `@Composable` блокування.

Також надано дві функції верхнього рівня, це `@Composables`: `Greeting(name)` і `DefaultPreview`. `Greeting` створює простий `Text` складний елемент, який є еквівалентом `Compose` для традиційного `TextView`.

Метод активності `onCreate` викликає новий `setContent` метод розширення `Compose`, який відповідає за створення інтерфейсу користувача `Compose`. Він складається з двох блоків: `RecordingAioTheme` (назва походить від того, яку назву має програма `RecordingAio`), який складається з `Greeting`.

Дивлячись на `DefaultPreview`, він побудований з тієї ж композиції елементів, що й `setContent` блок, і поки не використовується. Єдина відмінність полягає в тому, що `DefaultPreview` має примітку `@Preview`, що дозволяє переглядати його в `Android Studio`, подібно до попереднього перегляду макета в макетах XML.

@Preview анотації можна застосовувати лише до функцій із анотаціями @Composable, які не отримують аргументів. Це означає, що попередній перегляд composable повинен надавати значення за замовчуванням для будь-яких аргументів, які він приймає.

Наступний етап, це створення основного інтерфейсу користувача для головного екрана. Почати слід зі створення базового інтерфейсу користувача для проекту. Перш ніж почати розробку / написання коду, важливо звернути увагу на наступні ключові моменти:

- традиційно Android використовує дії та фрагменти для розділення екранів і незалежних потоків (у Compose програми розроблено для роботи з архітектурою однієї дії. Це полегшує впровадження складних потоків навігації за допомогою Compose Navigation , а вся навігаційна логіка обробляється середовищем виконання Compose);
- це також означає, що стандартні життєві цикли активності та фрагментів тут не застосовуються (натомість Compose визначає різні побічні ефекти, які можуть діяти як зворотні виклики життєвого циклу).

У подальшому можлива розробка моделі даних з використанням бібліотек збереження місця для збереження та отримання записів дзвінків, але зараз було вирішено визначити просту модель даних для взаємодії з інтерфейсом користувача:

```
enum class CallType {
    INCOMING,
    OUTGOING,
    MISSED
}

data class RegisteredCall(
    val id: Int,
    val source: String,
    val callType: CallType,
    val duration: Duration
)
```

Також використано простий метод створення фіктивних даних лише для відображення:

```
fun callFromId(id: Int): RegisteredCall {
    return RegisteredCall(
        id,
        "+1-202-555-0108",
        CallType.values()[abs(id) % CallType.values().size],
        Duration.ofMinutes(5).plus(Duration.ofSeconds(23))
    )
}

fun dummyCalls(size: Int): List<RegisteredCall> {
    return (0 until size).map { callFromId(it) }
}
```

Головний екран додатку буде досить простим, лише з панеллю інструментів із назвою та списком записів викликів, який можна прокручувати. Якщо натиснути на який конкретний запис, відкриється екран деталей. Для початку необхідно додати кілька залежностей до рівня програми build.gradle:

```
android {
    // ...
    compileOptions {
        coreLibraryDesugaringEnabled = true
        // ...
    }
    // ...
}

dependencies {
    // ...

    coreLibraryDesugaring("com.android.tools:desugar_jdk_libs:1.1.5")
        implementation "androidx.core:core-ktx:1.6.0"
        implementation "androidx.appcompat:appcompat:1.3.1"
        implementation "com.google.android.material:material:1.4.0"
        implementation "androidx.compose.ui:ui:$compose_version"
        implementation
"androidx.compose.material:material:$compose_version"
        implementation "androidx.compose.material:material-icons-extended:$compose_version"
        implementation "androidx.compose.ui:ui-tooling-preview:$compose_version"
```

```

        implementation "androidx.activity:activity-compose:1.3.1"
        debugImplementation "androidx.compose.ui:ui-tooling:$compose_version"
    }

```

`coreLibraryDesugaringEnabled` і `desugar_jdk_libs` необхідні для того, щоб увімкнути Java 8+ API без мінімального рівня Android API. Інші залежності стосуються підтримки Jetpack Compose, розширень Kotlin (`core-ktx`) і бібліотек підтримки.

Далі варто додати змінну до файлу `compose_version` верхнього рівня `:build.gradle`

```

buildscript {
    ext {
        compose_version = '1.0.1'
    }
    // ...
}

```

Після слід перейти до написання `Composable`. Варто пам'ятати, що в Compose елементи інтерфейсу є функціями, позначеними `@Composable`, і керують власним локальним станом. Нижче предствлено, як виглядатиме список записів у застосунку:

```

@Composable
fun CallList(
    recordings: List<RegisteredCall>,
    onClick: (index: Int, recording: RegisteredCall) -> Unit,
) {
    val scrollState = rememberLazyListState()

    LazyColumn(state = scrollState) {
        itemsIndexed(recordings) { index, recording ->
            CallItem(recording) { onClick(index, recording) }
        }
    }
}

```


CallL отримує список RegisteredCalls і onClick слухача, який оброблятиме події клацання для елементів списку.

LazyColumncomposable використовується як еквівалент системи RecyclerView перегляду Android. LazyColumn подібний до елемента, головна відмінність полягає в Column тому, що перший завантажує елементи в списку лише тоді, коли вони з'являються на екрані.

Оскільки завантаження елементів на екран відбувається ліниво, LazyColumn не приймає складовий блок. Натомість він визначає LazyListScope DSL, який дозволяє описати, як створюються елементи. У даному випадку використовується itemsIndexed метод, який отримує список елементів і випускає composables, які знають свій індекс у списку.

Нарешті, список також має пам'ятати про стан прокручування. Composable завжди можна перебудувати, тому список не знає, який був його останній стан прокручування, кожне перемальовування composable призведе до того, що інтерфейс користувача «стрибне» на початок списку. З цієї причини використовуватиметься rememberLazyListState(), який створює об'єкт стану, який в подальшому можна передати в відкладений список.

Тепер необхідно визначити, як виглядає окремий елемент запису в списку, CallItemcomposable:

```
@Composable
fun CallItem(
    registeredCall: RegisteredCall,
    onClick: () -> Unit = {},
) {
    Card(
        modifier = Modifier
            .fillMaxWidth()
            .clickable(onClick = onClick)
    ) {
        Row(
            modifier = Modifier.padding(horizontal = 16.dp,
vertical = 8.dp),
            verticalAlignment = Alignment.CenterVertically
        ) {
```

```

Box(
    modifier = Modifier
        .size(48.dp)
        .clip(CircleShape)
        .background(color
MaterialTheme.colors.background),
    contentAlignment = Alignment.Center
) {
    registeredCall.callType.Icon()
}
Column(modifier = Modifier.padding(8.dp)) {
    Text(text = registeredCall.source)
    Text(
        text
registeredCall.duration.humanReadable(),
        style = MaterialTheme.typography.caption
    )
}
}
}
}
}

```

Модифікатори – це дуже зручний спосіб додати додаткову поведінку елементам інтерфейсу Compose. Додаткова поведінка, це може бути розмір, форма, семантика, межі (заповнення) і рух елементів. Його також можна використовувати для приєднання різних дій користувача, таких як натискання, фокусування тощо. Варто також зазначити, що інтерфейс модифікатора вільний. Це означає лише те, що інтерфейс підтримує ланцюжок методів, оскільки кожен метод повертає екземпляр, у якому він був викликаний (`this`). Наприклад, розглянемо модифікатор, який надається Card елементу верхнього рівня: `Modifier.fillMaxWidth().clickable(onClick = onClick)`. Цей модифікатор означає «Зроби цей елемент заповнювачем максимальної ширини екрана, а також зроби його доступним для кліків за допомогою прослухувача `onClick` кліків».

Якщо говорити про елементи, то тут використовується Card елемент із бібліотеки Compose Material, який дає картку Material Design. Також використовується зрозумілий Row елемент для горизонтального розташування, а також елементи Box і Column. Елемент Box містить піктограму, яка вказує на

тип записаного дзвінка, а також Column деяку текстову інформацію (номер телефону/ім'я контакту, пов'язане з дзвінком, і тривалість запису). В застосунку також визначаються деякі методи розширення, щоб зробити код більш читабельним і детальнішим. Тема застосунку, варто приділити увагу style атрибут, що передається Text елементу і дозволяє контролювати вигляд і відчуття тексту. MaterialTheme зберігає тему для всієї програми та дає доступ до стилів для макетів, типографіки та кольорів.

До попереднього перегляду інтерфейсу користувача можна перейти після закінчення з основними частинами інтерфейсу користувача та переглянути, як виглядають компоненти. Compose надає цю опцію за допомогою @Preview анотації. Визначаючи компонент попереднього перегляду, визначається @Composable функція без параметрів. Таким чином, всі вихідні параметри повинні бути відомі заздалегідь. Можна переглянути CallItem елемент, передавши фіктивний запис за допомогою функцій, які були створені раніше:

```
@Preview(showBackground = true)
@Composable
fun CallItemPreview() {
    CallItem(
        registeredCall = callFromId(0)
    )
}
```

Створюючи проект і дивлячись на екран дизайну в Android Studio, повинно відобразитися наступне на екрані (рис.16)

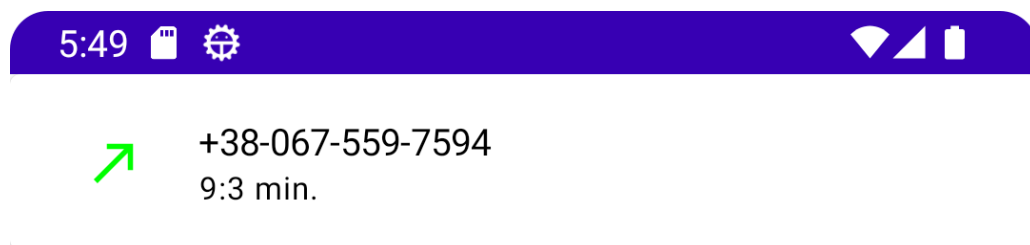


Рисунок 16 – Відображення запису елемента

Таким чином, можна отримати наступний код, переписавши основну дію. Для цього слід додали TopBar елемент, який відображає лише заголовок, і ініціалізували його CallList100 фіктивними викликами. Коли натискається елемент запису, відображається снєк-бар перевірки працездатності. Макет інкапсульовано за допомогою Surface, який відповідає за застосування кольору фону до програми, Scaffold, що в свою чергу обробляє структурований вигляд і відчуття сторінки Material Design, і RecordingAioTheme, який MaterialTheme генерується AndroidStudio для програми.

Після запуску застосунку, відображається наступне (рис. 17)

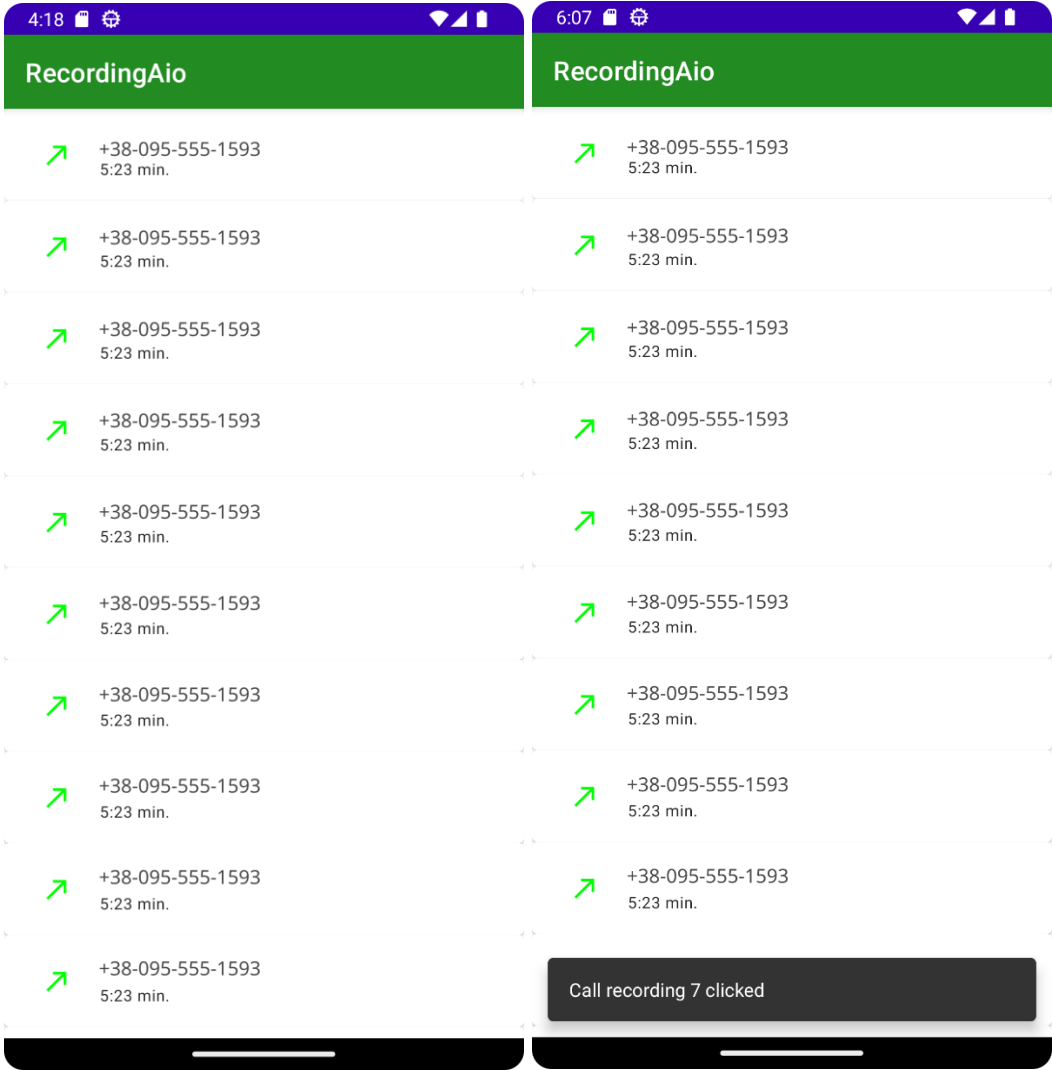


Рисунок 17 – Відображення головного екрану застосунку

Програми, які обробляють нетривіальні обсяги структурованих даних, можуть отримати значну користь від збереження цих даних локально. Найпоширенішим випадком використання є кешування відповідних фрагментів даних, щоб, коли пристрій не може отримати доступ до мережі, користувач усе ще міг переглядати цей вміст у режимі офлайн.

Бібліотека стійкості Room забезпечує рівень абстракції над SQLite, щоб забезпечити вільний доступ до бази даних, одночасно використовуючи всю потужність SQLite. Зокрема, Room надає такі переваги:

- перевірка SQL-запитів під час компіляції;
- зручні анотації, які зводять до мінімуму повторюваний і схильний до помилок шаблонний код;
- оптимізовані шляхи міграції бази даних.

Щоб використовувати Room у даному застосунку, необхідно додати наступні залежності до `build.gradle` файлу програми:

```
dependencies {
    val room_version = "2.4.3"
    implementation("androidx.room:room-runtime:$room_version")
    annotationProcessor("androidx.room:room-
compiler:$room_version")
    // To use Kotlin annotation processing tool (kapt)
    kapt("androidx.room:room-compiler:$room_version")
    // To use Kotlin Symbol Processing (KSP)
    ksp("androidx.room:room-compiler:$room_version")
    // optional - Kotlin Extensions and Coroutines support for Room
    implementation("androidx.room:room-ktx:$room_version")
    // optional - RxJava2 support for Room
    implementation("androidx.room:room-rxjava2:$room_version")
    // optional - RxJava3 support for Room
    implementation("androidx.room:room-rxjava3:$room_version")
    // optional - Guava support for Room, including Optional and
    ListenableFuture
    implementation("androidx.room:room-guava:$room_version")
    // optional - Test helpers
    testImplementation("androidx.room:room-
testing:$room_version")
    // optional - Paging 3 Integration
    implementation("androidx.room:room-paging:$room_version")
}
```

Room складається з трьох основних компонентів:

- клас бази даних , який містить базу даних і служить основною точкою доступу для підключення до збережених даних вашої програми;
- сутності даних , які представляють таблиці в базі даних вашої програми;
- об'єкти доступу до даних (DAO) , які надають методи, які ваша програма може використовувати для запити, оновлення, вставки та видалення даних у базі даних.

Клас бази даних надає програмі екземпляри DAO, пов'язані з цією базою даних. У свою чергу, програма може використовувати DAO для отримання даних із бази даних як екземплярів пов'язаних об'єктів сутності даних. Програма також може використовувати визначені сутності даних для оновлення рядків із відповідних таблиць або для створення нових рядків для вставки. На рис.18 проілюстровано взаємозв'язок між різними компонентами Room.

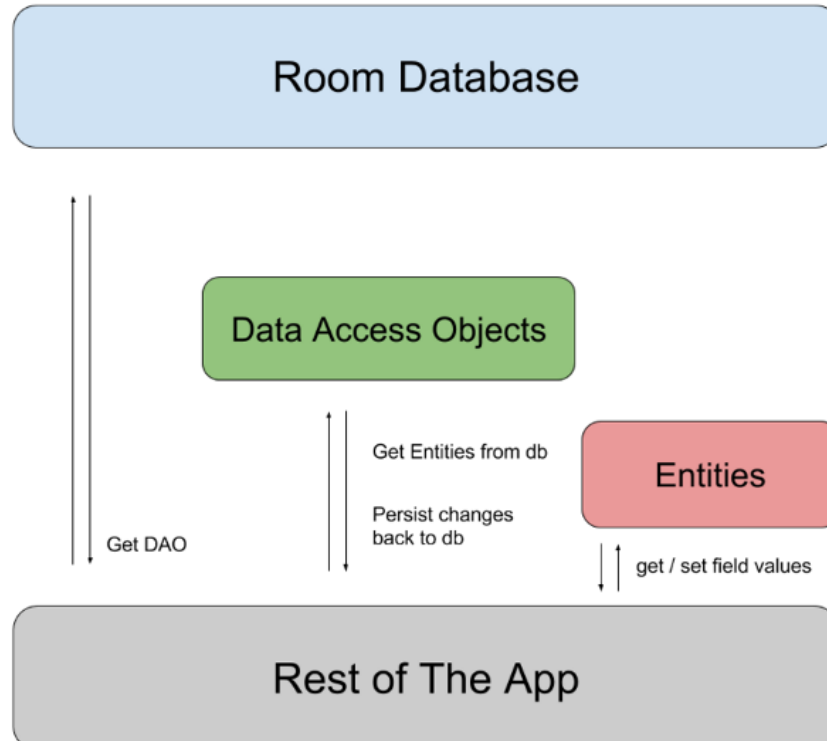


Рисунок 18 – Схема архітектури бібліотеки Room

Наступний код визначає User як сутність даних. Кожен екземпляр User представляє рядок у user таблиці в базі даних програми.

```
@Entity
data class User(
    @PrimaryKey val uid: Int,
    @ColumnInfo(name = "first_name") val firstName: String?,
    @ColumnInfo(name = "last_name") val lastName: String?
)
```

Наступний код визначає DAO (об'єкт доступу до даних) під назвою UserDao. UserDao надає методи, які решта програми використовує для взаємодії з даними в user таблиці.

```
@Dao
interface UserDao {
    @Query("SELECT * FROM user")
    fun getAll(): List<User>

    @Query("SELECT * FROM user WHERE uid IN (:userIds)")
    fun loadAllByIds(userIds: IntArray): List<User>

    @Query("SELECT * FROM user WHERE first_name LIKE :first AND "+
        "last_name LIKE :last LIMIT 1")
    fun findByName(first: String, last: String): User

    @Insert
    fun insertAll(vararg users: User)

    @Delete
    fun delete(user: User)
}
```

Наступний код визначає базу даних, `AppDatabase` клас для зберігання бази даних. `AppDatabase` визначає конфігурацію бази даних і служить основною точкою доступу програми до збережених даних. Клас бази даних повинен задовольняти наступним умовам:

- клас повинен бути анотований `@Database` анотацією, яка включає `entities` масив, який перераховує всі сутності даних, пов'язані з базою даних;
- клас має бути абстрактним класом, який розширює `RoomDatabase`;
- для кожного класу DAO, пов'язаного з базою даних, клас бази даних повинен визначати абстрактний метод, який має нуль аргументів і повертає екземпляр класу DAO.

```
@Database(entities = [User::class], version = 1)
abstract class AppDatabase : RoomDatabase() {
    abstract fun userDao(): UserDao
}
```

Слід зазначити, якщо програма працює в одному процесі, необхідно дотримуватися шаблону дизайну `singleton` під час створення екземпляра `AppDatabase` об'єкта. Кожен `RoomDatabase` екземпляр досить дорогий, і частіше всього, тільки у деяких випадках потрібен доступ до кількох екземплярів в рамках одного процесу.

Якщо ж програма виконується в кількох процесах, варто додати `enableMultiInstanceInvalidation()` до конструктора виклику бази даних. Таким чином, коли є екземпляр `AppDatabase` у кожному процесі, можна зробити недейсним спільний файл бази даних в одному процесі, і це визнання недейсним автоматично поширюється на екземпляри `AppDatabase` в інших процесах.

Після того, як визначено сутність даних, DAO та об'єкт бази даних, можна використовувати наступний код, щоб створити екземпляр бази даних:


```
val db = Room.databaseBuilder(  
    applicationContext,  
    AppDatabase::class.java, "database-name"  
).build()
```

Потім є необхідність використовувати абстрактні методи з, AppDatabase щоб отримати примірник DAO. У свою чергу, можна використовувати методи з екземпляра DAO для взаємодії з базою даних:

```
val userDao = db.userDao()  
val users: List<User> = userDao.getAll()
```

ВИСНОВКИ

Запис дзвінків – це не просто «надбудова» для системи телефонії. Правильне рішення може стати потужним інструментом для будь-якого бізнесу, якому необхідно забезпечити відповідність, захист і розвиток. Системи запису дзвінків із відповідними функціями для організації можуть допомогти дотримуватися вказівок правил, які діють у відповідному секторі. Правильна система також може дати змогу досягти кращих результатів.

За допомогою даних, зібраних із системи запису телефонних розмов, компанії можуть отримати корисну інформацію про шляхи та вподобання клієнтів. А також користувач може використовувати тенденції у своїх розмовах, щоб визначити, які стратегії найкраще працюють для співробітників.

Програма для запису дзвінків допоможе фіксувати все, сказане особою яка телефонує та її співрозмовником. Популярність таких програм пов'язана тим, що мобільні телефони перетворилися із засобів зв'язку на повноцінний комп'ютер, що міститься в кишені. Незважаючи на те, що практично у всіх телефонах вже є вбудований диктофон, він часто не дозволяє записувати дзвінки. Це пов'язано з політикою компанії виробника, законністю проведення такого запису та багатьма іншими факторами.

Сьогодні запис дзвінків – це не просто позначка в аркуші відповідності; це потужний спосіб гарантувати, що бізнес може як вижити, так і процвітати.

При розробці застосунку, програмного забезпечення слід велику увагу приділити саме життєвому циклу розробки. Етапи життєвого циклу розробки програмного забезпечення відрізняються залежно від вимог конкретної компанії та галузі бізнесу. SDLC (ЖЦП) можна змінювати або розширювати відповідно до потреб клієнтів. Це допомагає розподілити завдання та відповідальність за всіма необхідними функціями та виконувати їх за основним планом. Тим не менш, для успішного запуску продукту потрібно враховувати кожен крок і виконувати його акуратно.

Дана кваліфікаційна робота магістра передбачає розробку розроблений Android-застосунок для запису дзвінків. Дана робота передбачає поетапний опис ходу проведення розробки застосунку під операційну систему Android для запису дзвінків. Для початку розглядається процес створення нового проєкту, після чого описано та представлено уривки коду які були написані задля розробки мобільного застосунку. Розробка була в середовищі Android Studio Dolphin з використанням мови програмування Kotlin.

Під час розробки використовувався інструмент Jetpack Compose. Jetpack Compose для Android – це потужний і універсальний інструментарій для сучасного дизайну інтерфейсу користувача, який дозволяє створити простий, але дуже гарний інтерфейс із кількома сотнями рядків коду в Kotlin.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

- [1] 10 Ways Call Recordings Can Help Your Business. URL: <https://www.ringover.com/blog/benefits-of-call-recordings> (дата звернення 18.08.2022)
- [2] 5 Reasons Why Call Recording is Imperative for Your Contact Center. URL: <https://www.ameyo.com/blog/5-reasons-why-call-recording-is-imperative-for-your-contact-center/> (дата звернення 05.09.2022)
- [3] What is call recording? URL: <https://www.mitel.com/features-benefits/call-recording> (дата звернення 12.09.2022)
- [4] What are the benefits of Call Recording? URL: <https://www.bdm-voice.co.uk/what-are-the-benefits-of-call-recording/> (дата звернення 26.09.2022)
- [5] Рейтинг 10 лучших бесплатных приложений для записи телефонных звонков Android. URL: <https://geekhow.ru/obzory/luchshie-besplatnye-prilozhenij-dlya-zapisi-telefonnyh-zvonkov-android/> (дата звернення 03.10.2022)
- [6] RMC: Android Call Recorder. URL: <https://record-my-call.en.softonic.com/android> (дата звернення 17.10.2022)
- [7] What is Software Development Life Cycle? URL: <https://www.altamira.ai/software-development-life-cycle/> (дата звернення 04.11.2022)
- [8] How to Develop Audio Recorder Application. URL: <https://www.altamira.ai/how-to-develop-audio-recorder-application/> (дата звернення 25.09.2022)