

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА**

Факультет комп'ютерних наук та кібернетики
Кафедра інтелектуальних програмних систем

**Кваліфікаційна робота
на здобуття ступеня бакалавра**

за спеціальністю 121 Інженерія програмного забезпечення

на тему:

**МОБІЛЬНИЙ ДОДАТОК НА ANDROID ДЛЯ ПІДВИЩЕННЯ
ЕФЕКТИВНОСТІ РОБОЧОГО ЧАСУ ЗА ДОПОМОГОЮ ТЕХНІКИ
POMODORO ТА ІНТЕГРАЦІЇ ЗІ СТРІМІНГОВИМИ СЕРВІСАМИ**

Виконала студентка 4-го курсу

Яна СКІРДА



(підпис)

Науковий керівник:


доцент, кандидат фізико-математичних наук

Оксана ШКІЛЬНЯК

(підпис)

Засвідчую, що в цій роботі немає запозичень
з праць інших авторів без відповідних
посилань.

Студент



(підпис)

Роботу розглянуто й допущено до захисту на
засіданні кафедри інтелектуальних
програмних систем

«25» травня 2022 р.,

протокол № 10

Завідувач кафедри

Олександр ПРОВОТАР

(підпис)

Київ – 2022

РЕФЕРАТ

Обсяг роботи 39 сторінок, 6 ілюстрацій, 14 джерел посилань.

ОПТИМІЗАЦІЯ РОБОЧОГО ЧАСУ, ТАЙМЕР, ІНТЕГРАЦІЯ, СТРІМІНГОВІ СЕРВІСИ, МОБІЛЬНИЙ ДОДАТОК, ТЕХНІКА POMODORO, ПРОГРАВАЧІ ВІДЕО ТА АУДІО, БАЗА ДАНИХ SQLITE, КОРИСТУВАЦЬКИЙ ІНТЕРФЕЙС, СИСТЕМА РЕКОМЕНДАЦІЙ.

Об'єктом роботи є процес оптимізації робочого часу за допомогою техніки Pomodoro та допоміжних методів, які потребують інтеграції зі стрімінговими відео- та аудіоплатформами. Предметом роботи є мобільний додаток на Android, що реалізовує в собі інструментарій для зручного використання технік, спрямованих на покращення ефективності робочого часу.

Метою роботи є аналіз існуючих способів, які допомагають оптимізувати продуктивність, їх доступних програмних імплементацій та створення на основі проведеного аналізу зручного мобільного додатку на Android, що буде надавати можливість об'єднання різних обраних методик та буде надавати рекомендації щодо найпродуктивніших налаштувань відносно кожного окремого користувача.

Методи розроблення: Android SDK Platform 31, обгортка над YouTubePlayer API - android-youtube-player від PierfrancescoSoffritti, Android SDK від Spotify, що надає доступ до аудіопрогравача на основі Spotify, мова програмування Java, середовище розробки Android Studio.

Результати роботи: проведено дослідження існуючих методів покращення ефективності робочого часу, детальніше розглянуто техніку Pomodoro, її відомі реалізації на різних сайтах та додатках, виокремлено недоліки і переваги, розроблено мобільний додаток на основі проведеного дослідження, обґрунтовано корисність розширення базового функціоналу та описано, як розширення було виконано.

Мобільний додаток може використовуватись для покращення продуктивності при виконанні різних задач, зокрема коли треба підвищити концентрацію при навчанні.

ЗМІСТ

ВСТУП	4
РОЗДІЛ 1 ДОСЛІДЖЕННЯ ІСНУЮЧИХ ПРАКТИК ДЛЯ ОПТИМІЗАЦІЇ ПРОДУКТИВНОСТІ РОБОЧОГО ЧАСУ	7
1.1 Опис проблеми	7
1.2 Існуючі ефективні способи для покращення продуктивності	9
1.3 Техніка Pomodoro	10
1.4 Огляд існуючих додатків, що використовують техніку Pomodoro	12
РОЗДІЛ 2 ДЕТАЛІ ІМПЛЕМЕНТАЦІЇ ДОДАТКУ	17
2.1 Ідея додатку	17
2.2 Імплементация таймера	18
2.3 Збереження користувацьких налаштувань	21
2.4 Система рекомендацій налаштувань	23
РОЗДІЛ 3 РОЗШИРЕННЯ ДО СТАНДАРТНОГО ВИКОРИСТАННЯ ТЕХНІКИ POMODORO	27
3.1 Опис потенційно корисних розширень	27
3.2 Вибір відеострімінгового сервісу для інтеграції	28
3.3 Вибір музичного стрімінгового сервісу для інтеграції	30
3.3.1 Музичний сервіс SoundCloud	31
3.3.2 Музичний сервіс YouTube Music	32
3.3.3 Музичний сервіс Spotify	33
ВИСНОВКИ	36
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	38

ВСТУП

Оцінка сучасного стану об'єкта розробки. У сучасних реаліях, коли щоденний об'єм інформації, яку поглинає людина, значно збільшився, велика кількість людей стикається з погіршеною концентрацією та, як наслідок, низькою продуктивністю.

Оскільки це надзвичайно актуальна проблема, особливо для нового покоління, є багато різних методів, що допомагають в подоланні подібних труднощів. На основі цих методів уже існують розроблені додатки, які частково надають потрібний функціонал. Але зазвичай вони доволі примітивні та не мають достатньо можливостей, щоб їх було зручно використовувати в довгостроковій перспективі.

Актуальність роботи та підстави її виконання. Проблема з концентрацією має тенденцію до збільшення, зокрема через цифрову залежність, яка з'являється у молодого покоління змалечку. Доволі сильно вплинула на поширення проблеми епідемія COVID-19, через яку багато людей почали працювати з дому, не вміючи правильно організувати свій час та ресурси.

Існуючі платформи для вирішення цієї проблеми не надають достатнього функціоналу, як вже було сказано вище. Тому створення додатку, який би надавав можливість використовувати популярні та нескладні сучасні способи для оптимізації робочого часу та дозволяв налаштовувати їх під індивідуальні потреби залежно від виду завдань є надзвичайно актуальною задачею.

Мета й завдання роботи. Метою кваліфікаційної роботи є створення мобільного додатку на Android, що буде надавати можливість об'єднання різних методів оптимізації продуктивності та буде надавати рекомендації

щодо найпродуктивніших налаштувань відносно кожного окремого користувача. Для досягнення цієї мети поставлено такі завдання:

- проаналізувати проблему втрати концентрації та її причини;
- дослідити наявні методи покращення концентрації та ефективності роботи;
- розглянути існуючі додатки, що вирішують цю проблему та визначити їх переваги та недоліки;
- розробити базовий функціонал додатку та детально описати процес створення;
- додати розширення можливостей додатку на основі інтеграцій із відео- та аудіострімінговими платформами.

Об'єкти, методи й засоби розроблення.

Об'єктом розробки додатку є процес оптимізації робочого часу за допомогою техніки Pomodoro та допоміжних методів, які потребують інтеграції зі стрімінговими відео- та аудіоплатформами.

Офіційний набір інструментів від Android, а саме Android SDK 31, надає широкі можливості для створення додатків з різними функціями. Це дає змогу зробити основний функціонал, який необхідний в додатку, без сторонніх бібліотек. Основний функціонал включає в себе таймер, що може бути виконаний на основі елемента інтерфейсу ProgressBar, та об'єкту CountdownTimer; базу даних SQLite, яка створюється за допомогою SQLiteOpenHelper; відстеження відволікання від завдань з використанням акселерометра, доступ до якого надає SensorManager.

Для розширеного функціоналу, що буде надавати можливість прослуховувати музичні композиції та відтворювати відео на фоні є прикладний програмний інтерфейс від YouTube та Spotify, що дають доступ для створення програвачів у власному додатку.

Можливі сфери застосування. Додаток можна застосовувати для задач довільного типу, зокрема при вивченні нового матеріалу чи виконанні навчальних завдань.

РОЗДІЛ 1

ДОСЛІДЖЕННЯ ІСНУЮЧИХ ПРАКТИК ДЛЯ ОПТИМІЗАЦІЇ ПРОДУКТИВНОСТІ РОБОЧОГО ЧАСУ

1.1 Опис проблеми

В останні декілька десятиріч у зв'язку з надзвичайно швидким ростом технологій значно збільшився об'єм інформації, яку споживає людина щоденно. Дослідження повідомлять, що в середньому, людина споживає 74 гігабайти інформації в день, і кожен рік ця цифра збільшується на 5 відсотків [1]. Немає сумніву, що доступність та кількість інформації постійно збільшується, але крім цього покращуються і технології, які спрямовані на те, щоб людина поглинала якомога більше інформації, використовуючи їх ресурс. За останні роки з'явилися алгоритми штучного інтелекту, що досконало визначають, яку інформацію ми хочемо поглинати, базуючись на наших пошукових запитах, вподобаннях в соціальних мережах та навіть кількості часу, що ми витрачаємо, переглядаючи певний допис, фото чи відео. На перший погляд, це дуже зручно та корисно, оскільки не потрібно витрачати більше часу, щоб знайти певну інформацію, яка тобі подобається, чи певний продукт, який ти хочеш придбати. Але насправді, наш мозок не може так швидко прилаштуватись до тієї кількості інформації, яку забезпечує розвиток технологій.

Таким чином, значна частина молодого покоління має проблеми з концентрацією уваги, невмінням виконувати довготривалі проекти та часто страждає вигоранням. Ця проблема тільки погіршилась в зв'язку з епідемією COVID-19, необхідністю самоізолюватись та почати працювати з дому.

Людський мозок перебуває в стані постійного відволікання. Ці нескінченні відволікання надходять з інформаційного потоку, що постійно оновлюється, насамперед з онлайн-комунікацій та соціальних мереж. Навіть не задумуючись ми реагуємо на сповіщення в телефоні, новий лист чи просто при першій складності у виконанні завдання – беремо в руки телефон. Це погано не тільки тому, що фізично забирає наш час, поки ми відволікаємось, а ще й тому, що для повторного входження в задачу, якою ми займаємось, потрібно витратити часові та мозкові ресурси. Простіше кажучи, від кожного такого переключення ми значно втомлюємось і в результаті робимо менше роботи гіршої якості.

Дуже актуальною є ця проблема не тільки при роботі, але й при навчанні. Вигорання у студентів має настільки ж негативний та шкідливий характер, що й у робочому світі. Студенти, які стикаються з екстремальними вимогами, обмеженим часом та енергією, стають виснаженими, цинічними та менш ефективними [2].

Дехто, хто навіть і усвідомлює проблему, через недостатню інформованість та відсутність знань щодо правильних підходів вирішення наявних труднощів, вдається до згубних рішень. Яскравим прикладом такого рішення є багатозадачність, оскільки такий підхід не просто не допомагає бути ефективнішим, він створює абсолютно протилежний ефект – людина витрачає більше часу на завдання та отримує гірший результат, ніж при послідовному виконанні, а на додачу ще й посилює проблеми з концентрацією та виснаженням навіть від невеликих задач.

Саме тому дуже важливим є поширення інформації про правильні підходи боротьби зі зниженою концентрацією та створення інструментів, які дозволять зручно та просто використовувати певні техніки для покращення продуктивності та зниження постійно наростаючого стресового навантаження.

1.2 Існуючі ефективні способи для покращення продуктивності

Одним з найважливіших пунктів для уникнення перевантаження та покращення робочої продуктивності є вміння правильного управління часом. Частою помилкою є плутання понять зайнятості і продуктивності. Якщо людина постійно зайнята цілою купою різних задач, які не показують жодних ознак зменшення та не дають значних результатів, – це яскравий приклад поганого управління часом. Оскільки насправді продуктивність – це досягнення бажаних результатів з меншими зусиллями та часом, а не як багато хто думає, постійна зайнятість [3].

Навчитись правильно управляти своїм часом – це процес довготривалий та не зовсім простий, оскільки кожна людина має індивідуально зрозуміти, як саме виконувати їй завдання зручніше та ефективніше. Тобто потрібно не просто спробувати різні підходи, але й вміти результати таких спроб проаналізувати.

Першим кроком до правильного розподілу свого часу може бути групування завдань на основі результату. Наприклад, завдання, які потребують креативної роботи та якихось активних процесів, краще групувати разом перед тими завданнями, які мають більш механічний та монотонний характер [4].

Підвищуючи ефективність за допомогою навичок управління часом, ви зменшуєте стрес, а також виконуєте більше справ, створюючи цикл, який дає вам змогу досягати ще більше.

Наступним важливим фактором є вміння не відволікатись на електронні гаджети та соціальні мережі. Оскільки кожне таке відволікання це переключення фокусу зі своєї задачі, а повторне включення в контекст займає значні ресурси та виснажує. До того ж, такі невеличкі перевірки телефону чи

сповіщення на комп'ютері, якими маленькими вони не були, з кожним разом тільки погіршують ситуацію з проблемою фокусування.

Також необхідним є правильне використання перерв та загалом достатня їх наявність. Іноді може здаватись, що через велику кількість справ, краще буде попрацювати довше з меншою кількістю перерв - це категорично неправильний підхід. Перерви є надзвичайно важливою частиною робочого процесу. Якщо людина нехтує перервами, це в результаті призведе до того, що працювати вона буде зі значно меншою ефективністю, при цьому навіть цього не помічаючи. Перерви часто необхідні як спосіб відволікання від фіксації на неефективному рішенні.

Психологічна відстороненість – термін, розроблений Сабіною Зоннентаг — це процес «вимкнення» з робочого режиму, щоб мозок і тіло мали змогу розслабитися і відновилися від стресових вимог роботи [5]. Ця відстороненість виникає лише від фізичного відокремлення себе від робочих думок і завдань. Такі перерви допомагають нашому мозку відновитися від наростаючої втоми, що в результаті дає змогу легше знайти рішення до різних робочих проблем.

1.3 Техніка Pomodoro

На мою думку, гарним рішенням, яке об'єднує всі перелічені вище поради щодо покращення своєї робочої ефективності – є техніка Pomodoro.

Техніка Pomodoro була розроблена наприкінці 1980-х років студентом Франческо Чірілло. Чірілло стикнувся з проблемою виконання своїх завдань та намагався зосередитися на навчанні. Він не міг ніяк приступити до виконання роботи, через що його настрій все більше падав і бажання сідати за навчання відповідно не збільшувалось. Тому він вирішив, що треба почати з малого і зважився виділити хоча б 10 хвилин зосередженому навчанню.

Підбадьорений своїм же викликом, він знайшов кухонний таймер у формі помідора (pomodoro італійською), і саме так народилася техніка Pomodoro [6].

Одна з переваг цього методу – це його простота у використанні. Стандартний спосіб використання наступний:

1. Налаштувати таймер на 25 хвилин. За цей час людина повинна зосередитись на одному завданні та не відволікатись на інші фактори.
2. По закінченню часу потрібно взяти перерву в 5 хвилин.
3. Після чотирьох таких сесій рекомендується взяти велику перерву довжиною 15-30 хвилин.

Також рекомендацією до використання цього методу є створення списку задач, які потрібно зробити. Це дає змогу відразу почати займатись наступним завданням зі списку, якщо ви встигли зробити своє перше завдання раніше, ніж вийшов час.

У разі неможливості уникнення ситуації, коли необхідно зробити паузу під час робочої сесії – рекомендується зробити п'ятихвилинну перерву раніше і продовжити на звичайним графіком. Також Чірілло у свої книзі про техніка Pomodoro, рекомендує записувати такі екстрені перерви та їх причини, щоб мати змогу уникнути їх в майбутньому.

Завдяки своїй простоті почати використовувати цей метод дуже легко. Це одна з причин, чому він працює. Зазвичай прокрастинація мало пов'язана з лінню або відсутністю самоконтролю. Більшою мірою, ми просто відкладаємо справи, які можуть викликати нам незручності та складнощі. Коли ми просто плануємо зробити якесь велике завдання, це завжди приносить певний дискомфорт, через, як нам здається, неосяжний обсяг роботи. Тому ефективний спосіб вирватися з такого циклу уникнення: скоротити все, що ми відкладаєте, до крихітного першого кроку, який не буде лячним для нас через його простоту. Зробити щось невелике протягом короткого періоду часу набагато легше, ніж намагатися взятися за великий проект відразу.

Більше того, техніка Pomodoro допомагає протистояти бажанню постійно відволікатись на сторонні фактори, такі як, наприклад, нове сповіщення на телефоні, і заново тренує ваш мозок фокусуватися. Кожен Pomodoro присвячений певному завданню, і кожна перерва — це шанс відволіктись та відпочити, а після цього знову повернути увагу до того, над чим ви повинні працювати. Тому що ви завжди знаєте, що через певний час отримаєте перерву, в яку можете відпочити та перевірити всі речі, на які так хотілось відволіктись під час робочої сесії: це дає змогу стриматись від переключення фокусу на сторонні речі.

1.4 Огляд існуючих додатків, що використовують техніку Pomodoro

Техніка Pomodoro з часу свого розроблення набрала велику популярність, тому існує чимало додатків, які надають функціональність техніки.

Але звичайно найпростіший спосіб використання – це використовувати справжній або електронний таймер. Це спочатку може здатись доволі зручною опцією, але якщо використовувати цю техніку на постійній основі, то відразу буде помітно, що процес зі звичайним таймером потрібно автоматизувати. Оскільки часто за день, навіть якщо працюєш всього декілька годин, то маєш вручну заводити таймер на різну кількість часу (час для фокусування, час для маленької та великої перерви) по 4 рази за годину.

До того ж, постійно треба пам'ятати, скільки раундів вже пройшло, щоб знати, на скільки хвилин ставити таймер. Це доволі незручно і не дає відслідковувати свій прогрес протягом дня та загалом в довгостроковій перспективі.

Наступний спосіб, де можна працювати з технікою Pomodoro, – це використовувати веб-сайти, спеціально зроблені з цією метою.

Я розглянула два найпопулярніші веб-сайти: pomofocus.io (рис. 1) та pomodoro-tracker.com (рис. 2).

Обидва таймера доволі схожі: вони можуть налаштовувати тривалість сесії (її робочої частини, маленької перерви, а також частоту великої перерви та її тривалість). Також можна додавати завдання, які заплановано зробити під час роботи з технікою. По замовчуванню, в кінці кожної сесії короткий звук повідомляє про її завершення, продовжити потрібно знову натиснувши кнопку, щоб почався відлік для перерви.

Також обидва сайти надають можливість подивитись статистику використання (лише за умови авторизації в сервіс).

З мінусів можна зазначити те, що не можна скинути час поточного раунду, тобто якщо випадково натиснути на кнопку та почати робочу частину сесії, то не можна її скинути і почати заново, потрібно пропустити її, потім запустити перерву і натиснути кнопку пропуску (якщо не натиснути кнопку старту, то кнопка пропуску недоступна). Така проблема також наявна в обох сервісах.

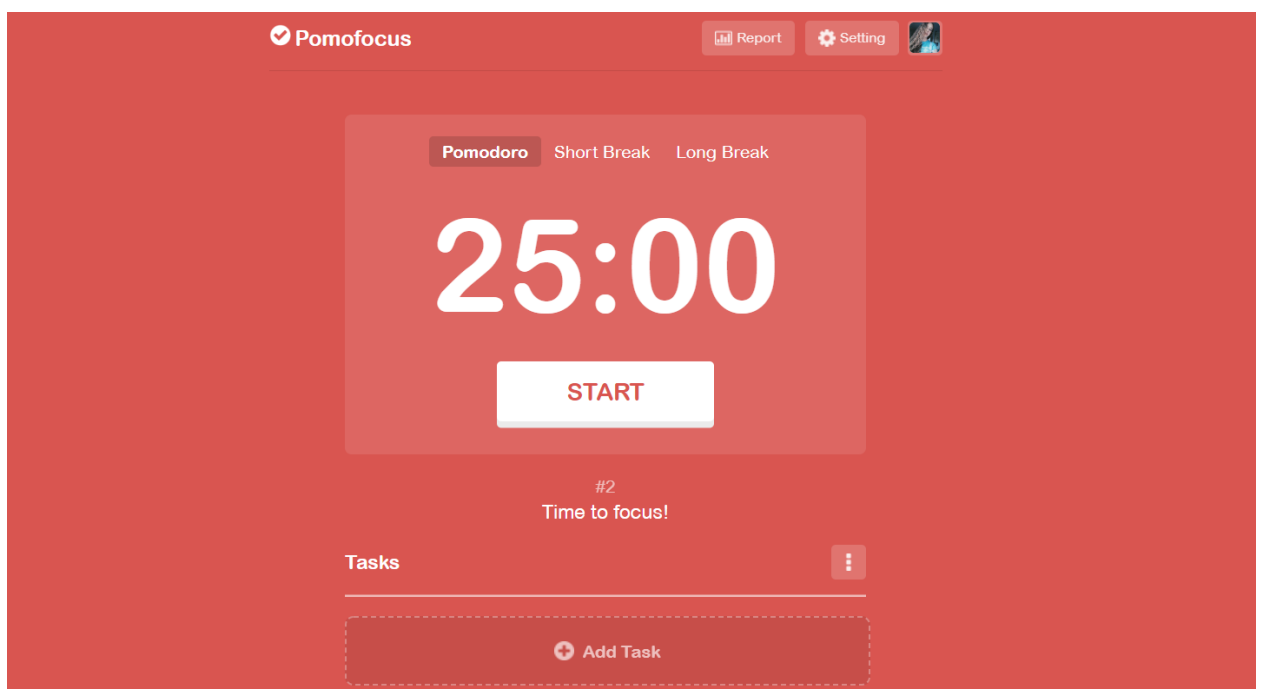


Рис. 1. Вигляд сайту pomofocus.io

Другий сервіс відрізняється лише зовнішнім виглядом, який є застрілим по міркам сучасних додатків. Але в другому сайті є одна особливість: для кожного завдання можна зазначати категорію при створенні. На жаль, категорій лише дві – особисті завдання та робочі завдання, і можливості створити нові немає.

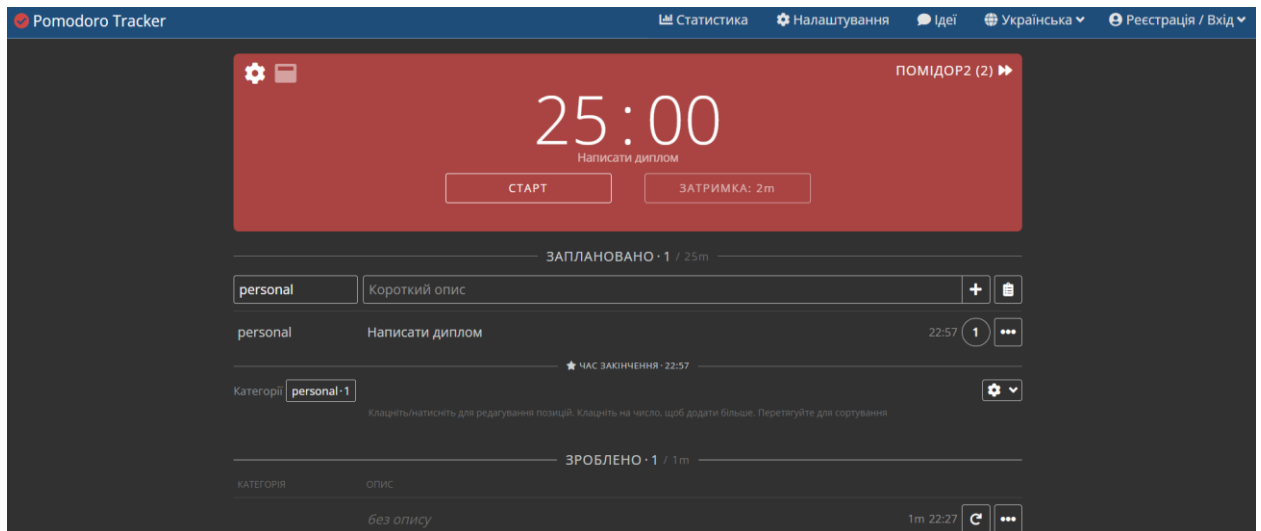


Рис. 2. Вигляд сайту pomodoro-tracker.com

Далі розглянемо мобільні додатки для роботи з технікою Pomodoro. На мою думку, саме мобільні додатки найзручніше використовувати для цієї мети.

Розглянемо два найпопулярніші додатки, які мають більше мільйона завантаження у PlayMarket – Pomodoro та Focus To-Do (рис. 3).

Додаток Pomodoro виконано у мінімалістичному стилі. Є налаштування тривалості кожної частини сесії. Але немає налаштування кількості раундів, вони просто будуть йти нескінченно, поки користувач не зупинить цей процес. До того ж, який зараз раунд від початку додаток не вказує, що доволі незручно, оскільки потрібно самому рахувати, скільки робочого часу вже пройшло.

Також у цьому додатку відсутні які-небудь види статистики, які були присутні на веб-сайтах.

Але з плюсів можна зазначити те, що в мобільному додатку виправлений недолік, який був помітний на веб-сайтах: можна відновити таймер на початок, якщо він вже почався.

Додаток Focus To-Do містить Pomodoro таймер як одну з декількох частин функціоналу. В основному в додатку приділено увагу створенню завдань та їх організації. А таймер йде як можливість ці завдання виконати, що доволі зручно, оскільки дозволяє працювати з технікою Pomodoro вже з готовим списком задач, що є однією з рекомендацій від автора самої техніки.

Мінусом використання цього додатку є те, що не можна задати кількість сесій, а також неможливість змінити тривалість перерви. Тобто, використовувати техніку якимось по-іншому, окрім стандартної схеми, не вдасться.

Але попри все, це єдиний додаток, який дозволяє декілька додаткових цікавих можливостей. По-перше, в ньому можна ввімкнути на фон одну зі стандартних музичних композицій, що добре сприяє концентрації. По-друге, додаток дозволяє включити «суворий» режим – тобто заблокувати телефон, поки йде робоча частина сесії.

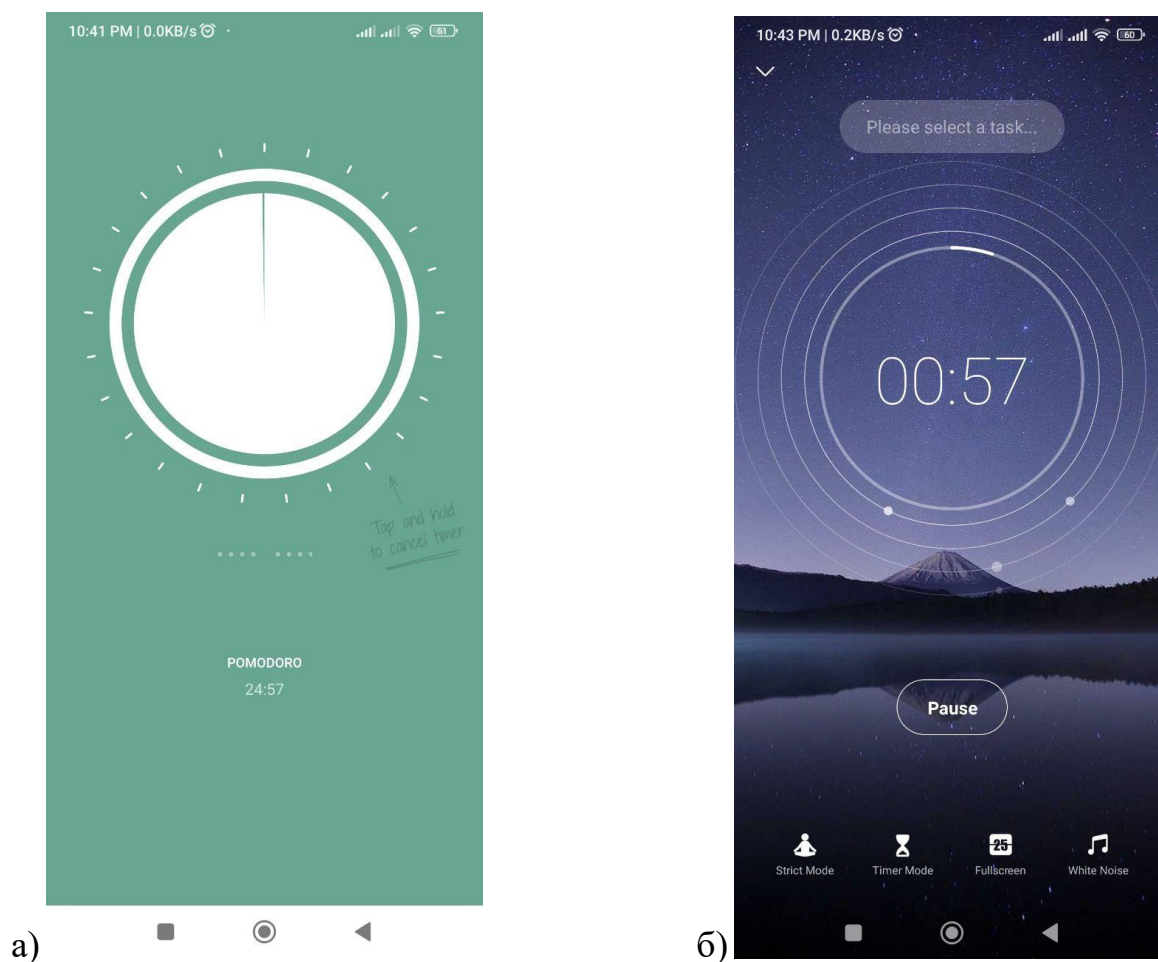


Рис. 3. Вигляд мобільних аналогів
а – додаток Pomodoro; б – додаток Focus To-Do

Загалом можна сказати, що у існуючих додатках та сайтах є цікавий функціонал, але немає одного додатку, яку містив би в собі увесь перелік можливостей, що на мою думку, допомогли б зробити використання техніки Pomodoro максимально корисним.

РОЗДІЛ 2

ДЕТАЛІ ІМПЛЕМЕНТАЦІЇ ДОДАТКУ

2.1 Ідея додатку

Метою створення окремого додатку на Android було бажання зробити техніку Pomodoro ще більш зручною та простою у використанні, додати можливість користувачу самостійно встановлювати всі можливі налаштування залежно від типу роботи та особистісних уподобань, а також додати інші фактори, які допомагають для концентрації.

Можливість самостійно налаштовувати різні параметри є дуже важливою частиною додатку, оскільки техніка Pomodoro – це не просто набір правил, яким треба безпосередньо слідувати, а більше збірка рекомендацій та загальна схема того, як покращити продуктивність.

Наприклад, є певні задачі, які потребують більшого часу робочої сесії для ефективного виконання. Для таких цілей є доволі популярна варіація від стандартних налаштувань техніки. Вона полягає в тому, що час для фокусу триває 50 хвилин, а відпочинок 10 хвилин. І кількість таких раундів зазвичай 2 або 4. Але під кожне окреме завдання можуть бути потрібні зовсім різні налаштування. Тому додаток буде мати змогу налаштовувати наступні параметри:

1. кількість раундів;
2. тривалість часу для фокусування на роботі;
3. тривалість короткого відпочинку;
4. наявність великих відпочинків;
5. частота великих відпочинків;
6. тривалість великих відпочинків.

Більше того, телефон в більшості випадків є основним фактором, що призводить до відволікання від основної роботи. Тому те, що додаток буде саме на телефоні, має ряд переваг.

По-перше, це дозволяє бути невеликою перешкодою для того, що перевірити телефон і втратити фокус, оскільки кожного разу при такому бажанні, додаток з таймером буде нагадувати, що зараз не час для цього, і через який час це можна буде зробити не завадивши робочому процесу.

По-друге, це дає можливість додатку відслідковувати кожен раз, коли користувач хоче відволіктись, що свідчить про те, що умови для фокусу недостатньо, отже, налаштування сесії підібрані неправильно. На основі таких даних додаток буде рекомендувати користувачу, з якими налаштуваннями йому найефективніше використовувати додаток.

2.2 Імплементация таймера

Реалізація основного функціоналу техніки Pomodoro буде доволі простою, вся складність буде полягати в інтеграції.

Для імплементации таймера в додатку можна використати стандартний елемент `ProgressBar`. Для атрибуту `background`, що буде потрібен для кращого відображення прогресу таймера, створимо ресурс, що бути мати форму кола. А для атрибуту `progressDrawable`, який буде безпосередньо відображати відсоток часу, що вже пройшов, від заданого, створимо інший ресурс, що буде контрастувати за кольором з фоновим ресурсом та буде мати меншу товщину. Посередині таймера відобразимо час, що залишився до його завершення. (рис. 4)

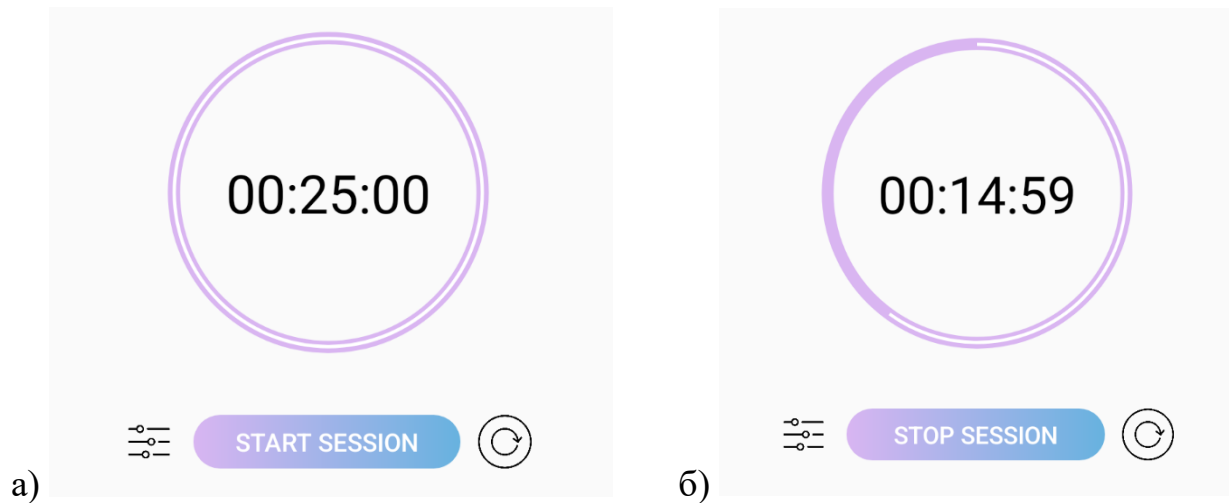


Рис. 4. Робота та вигляд таймера
 а – таймер до запуску: б – таймер під час роботи

Для відображення прогресу програмно після запуску таймера, потрібно у інтерфейсному елементі `ProgressBar` змінювати атрибут `progress`. Це робиться за допомогою створення об'єкту `CountDownTimer` при натисканні кнопки запуску, в який задається тривалість таймера в мілісекундах та імплементується метод `onTick`, в який додано зміну часу, який залишився та оновлення візуальної шкали на таймері.

Також таймер має містити в собі налаштування `Pomodoro` техніки, щоб автоматично після робочої частини сесії запускати таймер для відпочинку, а також перевіряти частоту великих перерв та запускати її коли потрібно. Статус, яка зараз частина сесії обов'язково має відображатись в додатку, щоб користувач правильно орієнтувався в техніці.

Більше того, має відображатись номер поточного раунду та загальна кількість раундів задана користувачем. Ця зміна є надзвичайно важливою, для роботи техніки та відслідковування прогресу користувача, яка була пропущена практично у всіх додатках та веб-сервісах, що були розглянуті в першому розділі. (рис. 5)

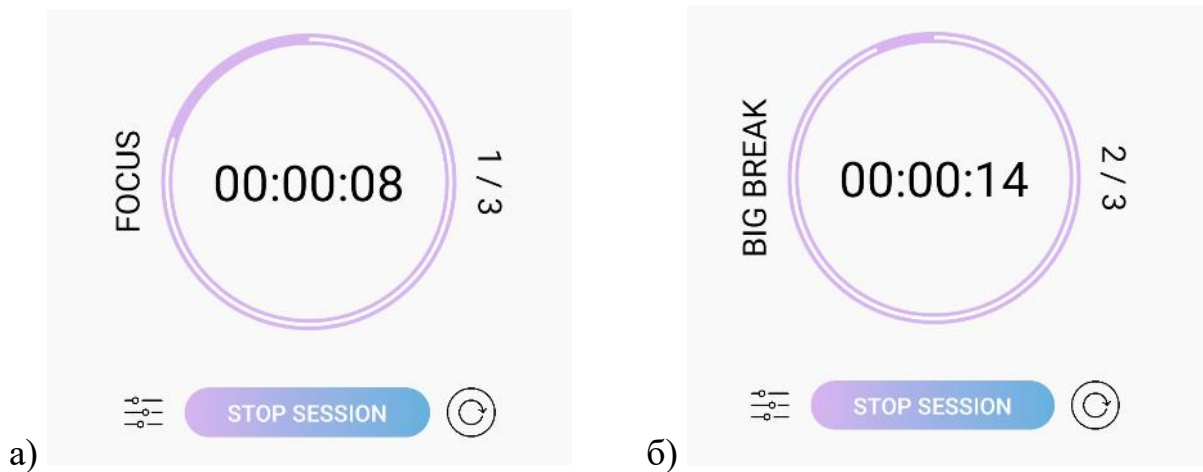


Рис. 5. Відображення статусу сесії та кількість раундів
 а – робоча частина першого раунду; б – велика перерва другого раунду

Також, при запуску таймера має автоматично вмикатись потрібний та введений завчасно користувачем відеоконтент та аудіосупровід (детальніше імплементацію та інтеграцію цих частин описано та обґрунтовано у наступному розділі). Ця частина має виконуватись відповідно до користувацьких налаштувань та тільки якщо ці функції включені для поточної сесії. (рис. 6)

При запуску, зупинці та скиданні налаштувань таймера стани відеосервісу та аудіосервісу мають відповідно змінюватись. Потрібно враховувати те, що одночасно не може бути запущений відеоряд зі звуком та відтворюватись музична композиція. Тому така опція не має бути доступна користувачу і про це має повідомлятися при виборі.

Більше того, зупинка відео чи музики не має впливати на роботу таймера, щоб користувач міг змінювати умови прямо під час робочої сесії, якщо йому щось заважає. Та має бути можливість приховати відео, оскільки цей функціонал може заважати деяким користувачам, навіть наявністю віджета.

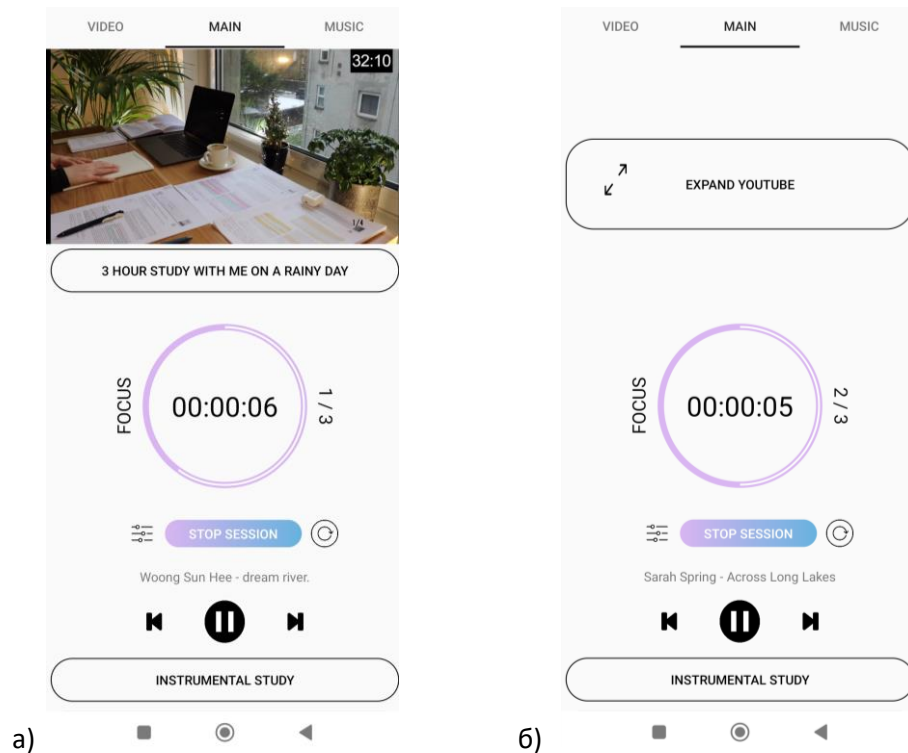


Рис. 6. Робота додатку

а – робота зі всіма інтеграціями; б – робота тільки з музикою та прихованим відеосервісом

2.3 Збереження користувацьких налаштувань

Додаток має зберігати інформацію про збережені налаштування користувача та успішно відображати їх при наступному сеансі. Тому для реалізації цього потрібно під'єднати базу даних.

Оскільки даних, що потрібно зберігати, небагато і вони не містять складних залежностей, було вирішено використовувати легку в роботі базу даних SQLite.

Для початку роботи з цією базою даних не потрібно навіть нічого додаткового під'єднувати, оскільки вона входить в основний інструментарій для Android в пакеті `android.database.sqlite` [7].

Далі потрібно створити сутності, які будуть відповідати даним, що нам потрібно зберегти. У нас таких сутностей буде три, тому розглянемо їх детальніше.

Для сутності, яка відображає налаштування безпосередньо Pomodoro сесії нам потрібні наступні дані:

1. ідентифікатор налаштування – типу INTEGER PRIMARY KEY;
2. кількість раундів в сесії – типу INTEGER;
3. тривалість робочої частини сесії – типу INTEGER;
4. тривалість стандартної перерви – типу INTEGER;
5. частота великих перерв – типу INTEGER, що відображає раз у скільки раундів замість маленької перерви буде велика;
6. тривалість великої перерви – типу INTEGER;
7. оцінка відволікань – типу REAL, яка буде відображати наскільки часто людина відволікалась під час цієї сесії, це буде показником наскільки рекомендовано використовувати саме це налаштування.

Для сутності, яка відображає збережі відео нам потрібні наступні дані:

1. ідентифікатор опції – типу INTEGER PRIMARY KEY;
2. ідентифікатор відео – типу TEXT;
3. оцінка відволікань – типу REAL, яка відображає те ж саме значення, що і для налаштувань сесії, але саме для відео.

Для сутності, яка відображає збережену музичну добірку нам потрібні наступні дані:

1. ідентифікатор опції – типу INTEGER PRIMARY KEY;
2. ідентифікатор музичної добірки – типу TEXT;
3. оцінка відволікань – типу REAL, яка відображає те ж саме значення, що і для налаштувань сесії, але саме для музичної добірки.

Для створення бази даних, нам потрібно створити окремий клас, що буде наслідуватись від `SQLiteOpenHelper` та перевезначати методи `onCreate` та `onUpgrade`.

В методі `onCreate` мають задаватись запити на створення кожної окремо таблиці у синтаксисі `SQLite`. А потім викликатись уже імплементований для бази даних метод – `execSQL`, що приймає параметром запит, який ми вже написали вище. Приклад такого запиту для однієї з таблиць: «`CREATE TABLE sessions (id INTEGER PRIMARY KEY, round_count INTEGER, focus_time INTEGER, break_time INTEGER, big_break_frequency INTEGER, bit_break_time INTEGER, distraction_rate REAL)`».

Метод `onUpgrade` потрібен для зміни версії бази даних, і стратегія для його імплементации наступна – спочатку потрібно видалити всі існуючі таблиці, а потім заново їх створити.

Далі робота з базою не стає складнішою – щоб звертатись до неї потрібно викликати або `getWritableDatabase()`, або `getReadableDatabase()`, які є частиною `SQLiteOpenHelper`. Перший потрібно викликати, коли ми хочемо змінити дані в базі, а другий коли лише зчитати їх. Щоб виконати ці дії можна або передати запит за допомогою методу `query`, або ж викликати один з методів `insert`, `update`, `delete`.

Тримати відкритим з'єднання з базою рекомендується якомога довше, щоб не перепід'єднуватись, тому закривати його будемо лише на методі, коли закінчує роботу наш основний клас – зазвичай при завершенні роботи додатку.

2.4 Система рекомендацій налаштувань

Оскільки користувач може налаштовувати усі деталі для техніки `Pomodoro`, в результаті може виникнути така ситуація, що обрані

налаштування будуть дуже неефективними саме для цього користувача, і він цього навіть не помічатиме.

Тому для додатку варто реалізувати певну систему, що буде вираховувати продуктивність користувача та за результатами цього рекомендувати налаштування.

Одним з найпростіших варіантів є створення певного списку задач, який користувач може заповнювати для кожної сесії та актуалізувати їх стан по завершенню. Це доволі легко реалізується програмно та навіть було наявно в уже існуючих додатках, що організують роботу з технікою Pomodoro, але без системи рекомендації – ця інформація використовується лише для статистики.

Але, на мою думку, це не є гарним показником того, що ці налаштування добре працюють. Оскільки задачі можуть бути зовсім різними за тривалістю, складністю, типом роботи, тощо. Тому у нашому додатку можна скористатись тим, що це саме мобільний додаток, та вираховувати продуктивність користувача за тим фактором, як часто він бере телефон, щоб відволіктись або перевірити, скільки часу залишилось до кінця. Такий фактор буде показувати, що концентрація на задачі недостатня та показувати це відносно результатів інших сесій.

Для цього ми можемо використовувати акселерометр телефону. Робота для такої базової логіки, що потрібна для виконання заданих завдань, доволі проста, оскільки Android має зручний функціонал для роботи з акселерометром.

Для початку роботи, потрібно в головному вікні, де відбувається сесія, підключити `SensorManager`. Це робиться за допомогою стандартного методу `getSystemService`, який приймає параметром назву сервісу, що потрібно повернути. У нашому випадку потрібно вказати `Context.SENSOR_SERVICE`.

SensorManager містить в собі різні сенсори, що наявні у пристрою [8]. Нам буде потрібен саме акселерометр, для цього достатньо у менеджера сенсорів викликати метод getDefaultSensor, який приймає параметром, який саме сенсор потрібно повернути. Отримати акселерометр можна вказавши параметр Sensor.TYPE_ACCELEROMETER.

Щоб реєструвати зміни, нам потрібно зареєструвати об'єкт, який буде «прослуховувати» дії сенсора та виконувати вказану нами операцію, якщо дія відбулась.

Щоб створити «прослуховувач», нам потрібно створити об'єкт SensorEventListener та реалізувати в ньому метод onSensorChanged. Саме в цьому методі ми і будемо виконувати обрахування відволікань користувача.

Якщо «прослуховувач» зареєстрував, що відбувалась зміна позиції телефону, тоді ми маємо перевірити, чи запущена зараз якась сесія, і якщо так, то яка її частина, оскільки ми маємо ігнорувати, якщо користувач користується телефоном під час перерви – це у нашій системі ніяк не впливає на продуктивність робочої частини.

У методі onSensorChanged ми отримуємо на вхід об'єкт SensorEvent, що містить значення, на скільки змінилась позиція телефону по кожній осі координат. Нам потрібно враховувати, що якщо телефон змістився мінімально і тільки по одній осі, то це потенційно випадковий рух і зараховувати це як відволікання не варто.

Також потрібно додати перевірку, що визначає, як довго користувач використовує телефон під час такого відволікання, та не зараховувати кожен зміну як окреме відволікання, якщо користувач вирішив, наприклад, декілька хвилин посидіти в соціальних мережах замість виконання задач.

Інструментарій для збереження цих даних у нас уже є, оскільки для кожного налаштування – таймера, відео та музики – вже є окреме поле в базі даних, яке описано вище як «оцінка відволікань».

Вираховувати цю оцінку для однієї сесії будемо як відсоток часу, що користувач провів відволікаючись, до того ж мінімальний час кожного відволікання буде 30 секунд, що будемо вважати за час, який потрібен для повторного входження в контекст роботи.

Для налаштування сесії, коли вона була запущена більше одного разу «оцінку відволікань» будемо обраховувати як середнє арифметичне від вже записаного значення та нового значення з останнього запуску. Це дає нам змогу надати останнім успіхам більший пріоритет, ніж старим.

Представлені можливості повністю задовольняють основні потреби нашого додатку.

РОЗДІЛ 3

РОЗШИРЕННЯ ДО СТАНДАРТНОГО ВИКОРИСТАННЯ ТЕХНІКИ POMODORO

3.1 Опис потенційно корисних розширень

Хоч і сама техніка Pomodoro показує гарні результати, її можна покращити, об'єднавши з іншими частими практиками, які використовуються для продуктивнішого виконання завдань.

Одним з найочевидніших покращень буде використання її з музичним супроводом. Численні дослідження показують, що певні види музики можуть бути корисними для нас під час роботи. Деякі види музики допомагають саме у випадках, коли ми вивчаємо щось нове, оскільки покращують нашу здатність обробляти інформацію. Інші ж види музики можуть просто допомагати блокувати відволікаючі фонові шуми.

Використання музики як спосіб концентрації - це доволі розповсюджена практика. Це легко можна підтвердити, зайшовши в один з найпопулярніших музичних сервісів – Spotify. Вони містять окремий розділ музики, який називається «Focus», та містить 20 офіційних музичних збірок з різними видами музики, що корисні для концентрації.

Другим покращенням, яке варто реалізувати в додатку, є можливість додавати відео, що буде відображатись на фоні під час роботи. Спочатку додавання відео у інструмент для збільшення концентрації уваги на задачі може здатись абсурдним, але насправді за допомогою цієї інтеграції я хочу реалізувати один з корисних психологічних трюків.

Коли ми знаходимося біля людей, які зосереджено працюють чи навчаються, ми підвищуємо нашу продуктивність на 15% [8]. Саме з такою метою на відеосервісі YouTube є сотні відео «Працюй зі мною», де люди просто декілька годин знімають, як вони навчаються або працюють, які набирають по декілька мільйонів переглядів.

Тому я вважаю корисною можливість додавати такі відео в своєму додатку. Звісно ж це може бути корисним не для всіх, тому вбудований відео-програвач можна буде приховати, щоб він не відволікав і не загромождував місце на екрані телефону.

3.2 Вибір відеострімінгового сервісу для інтеграції

Проблема вибору сервісу для інтеграції відео в додаток не стояла, оскільки YouTube є гігантом в цій сфері і не має аналогів, які в даному контексті підходили б краще.

Компанія надає офіційний прикладний програмний інтерфейс, для доступу до всієї необхідної в додатку інформації. А також, окремий розділ – YouTube Android Player API, що дає змогу включати функції відтворення відео та добірки у власних додатках. Він визначає методи завантаження та відтворення відео та добірок відео з YouTube, а також дозволяє налаштовувати й керувати відтворенням відео.

Використовуючи цю частину прикладного програмного інтерфейсу, можна завантажувати або вставляти відео в режим перегляду програвача, вбудованого в інтерфейс додатку. Після цього можна керувати відтворенням програмно. Наприклад, можна відтворити, призупинити або перейти до певної точки у поточному завантаженому відео [10].

Але оскільки частина з програвачем має доволі багато проблем та не оновлювалась уже декілька років, щоб ці проблеми виправити, було вирішено

використовувати обгортку на цей програвач, а саме `android-youtube-player` від PierfrancescoSoffritti.

Він забезпечує простий перегляд та інтерфейс, який можна легко інтегрувати в кожен Android додаток. Оскільки програвач реалізований так, що він працює всередині `WebView`, тому додатково завантажувати додаток YouTube на пристрій користувача не потрібно, що, варто зауважити, не суперечить правилам використання офіційного програвача.

Обгортка дозволяє змінювати інтерфейс програвача відповідно до побажань та навіть повністю замінити користувацьким інтерфейсом розробника.

Щоб почати використовувати додаток достатньо просто додати залежність на бібліотеку та додати елемент інтерфейсу, що буде містити програвач, а саме елемент `YouTubePlayerView`, що лежить за шляхом `com.pierfrancescosoffritti.androidyoutubeplayer.core.player.views`. Це відразу додасть програвач до додатку, достатньо просто передати ідентифікатор відео, яке потрібно показати.

Щоб програмно керувати програвачем, потрібно знайти вже доданий до інтерфейсу елемент і напряму до нього звертатись.

Для роботи додатку нам потрібні наступні операції програвача:

1. `getYouTubePlayerWhenReady` – параметром приймає функцію зворотного виклику. Як тільки програвач буде готовий до відтворення – функція виконається. Всі дії з запуском та контролем відео будуть відбуватись саме в цій функції зворотного виклику;
2. `loadVideo` – параметрами приймає ідентифікатор відео та час в секундах з якого потрібно ввімкнути відео. Завантажує та відразу запускає відео;

3. `cueVideo` – параметри має такі ж як і метод `loadVideo`. Єдина різниця, що цей метод не запускає відразу відео, а тільки завантажує його та відображає його обкладинку;
4. `play` – запускає завантажене відео;
5. `pause` – зупиняє відео;
6. `mute` – вимикає звук у відео. Цей метод потрібен в додатку для того, щоб одночасно мати відео без звуку та музику з музичного сервісу;
7. `unmute` – вмикає звук у відео. Цей метод потрібен, якщо користувач хоче мати лише YouTube як джерело звуку.

Всі інші методи програмно нам не потрібні, але користувач зможе сам використовувати їх прямо через програвач. Наприклад, перемотувати відео на певну кількість часу.

Для зручності користувача також додано власний метод, який використовує регулярний вираз для визначення ідентифікатора відео з посилання YouTube різного вигляду.

3.3 Вибір музичного стрімінгового сервісу для інтеграції

Музичних стрімінгових сервісів зараз існує величезна кількість, і більшість з них надають публічний прикладний програмний інтерфейс, за допомогою якого можна спілкуватись з сервісом та реалізувати весь потрібний в додатку функціонал.

Розглянемо найпопулярніші з них, оскільки в них міститься найбільша кількість доступних музичних композицій.

3.3.1 Музичний сервіс SoundCloud

SoundCloud – це популярна стрімінгова платформа для публікування музичних композицій. Основний функціонал є безкоштовним, але багато композицій популярних виконавців заблоковані в безкоштовній версії. Для доступу до таких композицій потрібно придбати одну з підписок – «SoundCloud Go» або «SoundCloud Pro».

В основному цю платформу більше використовують тільки починаючі виконавці, які публікують і намагаються поширити свої композиції або ж певні музичні обробки вже існуючих.

Прикладний програмний інтерфейс, який надає сервіс, охоплює всі потрібні функції – дає доступ до збережених композицій користувача, його музичних збірок та загалом усіх композицій, які присутні в сервісі та доступні цьому користувачу за його підпискою (або за її відсутності).

До того ж, значною перевагою є наявність офіційного контент-модуля, що вбудовується у веб-сторінку або в браузер. Завдяки цьому, не потрібно писати свій інтерфейс для управління музикою. Але, на жаль, його можна використовувати тільки, якщо програмна частина написана на JavaScript, тому для цього додатку такий метод не підходить.

Одним же з недоліків, який присутній у цьому сервісі, є відсутність офіційних музичних збірок, які користувач міг би додавати до таймера. Тобто для того, щоб знайти фонову музику для своїх Pomodoro сесій, користувач має або створити свою власну добірку, або ж проаналізувати добірки інших користувачів. Це додає певну незручність для використання цієї функціональності в додатку.

Більше того, у платформи відсутня нормальна система рекомендацій, тобто навіть знайшовши потрібну музичну добірку, користувач не матиме

можливість переглянути аналогічні до неї, що змушує знаходити новий різноманітний контент власноруч.

3.3.2 Музичний сервіс YouTube Music

YouTube Music – це відносно нова платформа для прослуховування музики, що розроблена відеохостингом YouTube.

Цей сервіс відрізняється від інших, оскільки він має в собі основу не композиції безпосередньо, а звук з відео, що розміщене на YouTube. Тобто це просто інструмент, який надає користувачам можливість зручно шукати та прослуховувати всі музичні відеозаписи на YouTube.

До списку доступних до прослуховування композицій та добірок входить велика кількість відомих виконавців та будь-які відео з їх відео-сервісу, що знаходиться в категорії «музика». Для тих користувачів, які мають підписку «YouTube Premium», доступна функція, що надає можливість слухати музичний супровід будь-якого відео доступного на YouTube [11].

Використання музичного сервісу саме від YouTube було б значною перевагою, оскільки для інтеграції у додаток відео, використовується саме ця продукція. Тому управління, авторизація та інтеграція загалом мали б бути значно простіше. Але, на жаль, YouTube не надає зараз прикладний програмний інтерфейс для роботи саме з музичним їх сервісом.

Є неофіційна бібліотека для доступу до YouTube Music, але вона теж не надає доступу для відтворення музики в додатку.

3.3.3 Музичний сервіс Spotify

Spotify - стрімінговий сервіс, що дозволяє прослуховувати музичні композиції та подкасти. Надає послуги легального онлайнного стрімінгу музичних композицій більшості найпопулярніших медіакорпорацій, серед яких BBC, Sony, EMI, Warner Music Group та Universal [12].

Цей сервіс є одним з найбільших та найпопулярніших сервісів для прослуховування музики і налічує понад 422 мільйони активних користувачів [13].

Після дослідження аналогів до цього сервісу, я прийшла до висновку, що саме він надає найкращу систему рекомендацій відносно твоїх прослуховувань, а також містить добре розроблений каталог з офіційних категорій з музичними добірками.

До того ж, сервіс містить окремий розділ, який називається «Focus», та містить 20 офіційних музичних збірок з різними видами музики, що корисні для концентрації.

Це надає зручність навіть для тих користувачів додатку, що раніше не були знайомі зі Spotify та не мають певних заготовлених добірок, що можна використовувати під час техніки Pomodoro.

Spotify надає зручний прикладний програмний інтерфейс, що дозволяє дістати всю потрібну інформацію для додатку.

Більше того, сервіс надає свій офіційний SDK (від англ. software Development Kit) — набір із засобів розробки, утиліт і документації [14]. Він містить в собі програвач, який надає змогу під'єднуватися до встановленого на телефон додатку Spotify та зручно і просто керувати музикою через нього, без додаткової авторизації та без необхідності самостійно робити запити до

сервісу кожного разу, коли потрібно зробити маніпуляції з поточною композицією та її станом.

Для використання цього функціоналу свій додаток потрібно зареєструвати на офіційній платформі Spotify для розробників, отримати звідти ідентифікатор клієнта, що буде використовувати безпосередньо в додатку.

Також потрібно згенерувати Development Fingerprint – це унікальний ключ, що генерується з пристрою, на якому відбувається розробка та додається до системи, якій треба ідентифікувати дії з цього пристрою. Цей ключ потрібно теж зберегти на платформі для розробників у своєму зареєстрованому додатку.

Далі, для того щоб під'єднатися до додатку Spotify зі свого додатку, потрібно просто викликати статичну функцію з функціоналу, що надає сервіс, а саме `SpotifyAppRemote.connect` з параметрами для підключення, які будуть містити в собі ідентифікатор клієнту та веб-адресу для переадресації при підключенні (вона теж задається на платформі для розробників). Також для успішного підключення має бути правильно введений Development Fingerprint, про який було описано вище.

Після успішного підключення ми отримаємо об'єкт типу `SpotifyAppRemote`, який дозволяє мати доступ до будь-якої заданої користувачем композиції чи добірки, а також дозволяє керувати нею, її гучністю, тощо.

Завдяки цьому функціоналу, ми можемо керувати програвачем викликаючи відповідний метод об'єкту. Але важливо розуміти, що під кожним таким викликом відбувається певне звертання до прикладного програмного інтерфейса. Розглянемо детальніше базові виклики, які будуть необхідні в нашому додатку:

1. Метод GET, звертання за `/v1/me/player/` – повертає поточний стан програвача, тобто дозволяє дізнатись чи він ініціалізований, чи зараз грає якась композиція, чи вона на паузі та інші деталі.
2. Метод GET, звертання за `/v1/me/player/currently-playing` – повертає інформацію про поточну композицію, що відтворюється.
3. Метод POST, звертання за `/v1/me/player/next` – перемикає стан програвача на наступну композицію та вмикає її, якщо програвач був на паузі.
4. Метод POST, звертання за `/v1/me/player/previous` – перемикає стан програвача на початок композиції, якщо пройшло більше 5 секунд від її початку або перемикає на попередню пісню інакше. В будь-якому випадку вмикає результуючу композицію, якщо програвач був на паузі.
5. Метод PUT, звертання за `/v1/me/player/pause` – ставить програвач на паузу.
6. Метод PUT, звертання за `/v1/me/player/play` – починає відтворення для програвача або знімає його з паузи.

Далі робота з програвачем від Spotify зводиться до правильної синхронізації її з таймером в додатку.

ВИСНОВКИ

У роботі представлений опис проблеми, що спонукала на створення нового додатку, розглянуто різні методи, їх корисність та можливість використання у межах цієї роботи. Проаналізовано роботу техніку Pomodoro, її варіації, способи використання та існуючі платформи, що реалізують саме цю техніку. За результатами було виявлено недоліки, яких потрібно уникнути при створенні свого додатку, а також переваги, які варто реалізувати.

Обґрунтовано актуальність створення нового додатку для мобільного пристрою на базі Android, що реалізовує зручну роботу з технікою Pomodoro та дозволяє налаштовувати усі необхідні параметри відповідно до роботи, яку потрібно виконати.

Додаток має змогу відслідковувати відволікання користувача під час робочої частини сесії та на основі цього рекомендувати налаштування за яких продуктивність найбільша.

Більше того, додаток надає можливість користувачеві використати допоміжні техніки для концентрації уваги та збільшення продуктивності, такі як прослуховування музичних композицій та відтворення на фоні відео, що за правильного використання також може покращити ефективність. Рациональність такого підходу детально розглянута та обґрунтована у даній роботі.

Було проведено аналіз та порівняння стрімінгових відео- та аудіоплатформ, що потенційно можна було б використовувати для інтеграції в додаток. Обрані платформи було детально розглянуто та в подробицях описана взаємодія з ними в рамках створення додатку та в рамках взаємодії користувача з наданим інтерфейсом.

В результаті було створено додаток, описано процес його створення, які технології та елементи були використані та як саме була реалізована робота з цими аспектами.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Too Much Information, Too Little Time: How the Brain Separates Important from Unimportant Things in Our Fast-Paced Media World [Електронний ресурс]. – Режим доступу: URL: - <https://kids.frontiersin.org/articles/10.3389/frym.2017.00023> (переглянуто 24.05.2022) - Назва з екрана.
2. Burnout: Prevention, Treatment, and Advice for Employees & Employers [Електронний ресурс]. – Режим доступу: URL: - https://zety.com/blog/burnout?utm_source=google&utm_medium=sem&utm_campaign=6540106315&utm_term=%2Bburnout%20%2Bprevention&network=g&device=c&adposition=&adgroupid=85084943440&placement=&gclid=Cj0KCQjwvqeUBhCBARIsAOdt45ZIday8GQQgeiqQnGnKKf8WyOrk_QRXUTUAucN4xe_aoy4Oq7oyY4EaAunLEALw_wcB (переглянуто 24.05.2022) - Назва з екрана.
3. Бейлі К. Рік продуктивності. Експерименти з часом, увагою та енергією. / К. Бейлі. – К.: «Наш Формат», 2019. – 95 с.
4. 10 top strategies on improving work efficiency | TonyRobbins [Електронний ресурс]. – Режим доступу: URL: - <https://www.tonyrobbins.com/business/improve-work-efficiency> (переглянуто 24.05.2022) - Назва з екрана.
5. Psychological Detachment From Work During Leisure Time: The Benefits of Mentally Disengaging From Work [Електронний ресурс]. – Режим доступу: URL: - https://www.researchgate.net/profile/Sabine-Sonnentag/publication/228079939_Psychological_Detachment_From_Work_During_Leisure_Time_The_Benefits_of_Mentally_Disengaging_From_Work/links/553d073d0cf245bdd7699a41/Psychological-Detachment-From-Work-During-Leisure-Time-The-Benefits-of-Mentally-Disengaging-From-Work.pdf (переглянуто 24.05.2022) - Назва з екрана.

6. Cirillo F. The Pomodoro Technique / F. Cirillo. – Італія: «FC Garage GmbH», 2009. – 5 с.
7. Save data using SQLite | Android Developers [Електронний ресурс]. – Режим доступу: URL: - <https://developer.android.com/training/data-storage/sqlite> (переглянуто 24.05.2022) - Назва з екрана.
8. SensorManager | Android Developers [Електронний ресурс]. – Режим доступу: URL: - <https://developer.android.com/reference/android/hardware/SensorManager> (переглянуто 24.05.2022) - Назва з екрана.
9. Sitting Near a High-Performer Can Make You Better at Your Job [Електронний ресурс]. – Режим доступу: URL: - <https://insight.kellogg.northwestern.edu/article/sitting-near-a-high-performer-can-make-you-better-at-your-job> (переглянуто 24.05.2022) - Назва з екрана.
10. YouTube Android Player API | Google Developers [Електронний ресурс]. – Режим доступу: URL: - <https://developers.google.com/youtube/android/player/> (переглянуто 24.05.2022) - Назва з екрана.
11. Google announces YouTube Music and YouTube Premium [Електронний ресурс]. – Режим доступу: URL: - <https://www.theverge.com/2018/5/17/17364056/youtube-music-premium-google-launch> (переглянуто 24.05.2022) - Назва з екрана.
12. What artists need to know about Spotify and the major labels [Електронний ресурс]. – Режим доступу: URL: - <https://unlockyoursound.com/major-labels-spotify/> (переглянуто 24.05.2022) - Назва з екрана.
13. Spotify – About Spotify [Електронний ресурс]. – Режим доступу: URL: - <https://newsroom.spotify.com/company-info/> (переглянуто 24.05.2022) - Назва з екрана.
14. Android SDK Quick Start | Spotify for Developers [Електронний ресурс]. – Режим доступу: URL: - <https://developer.spotify.com/documentation/android/quick-start/> (переглянуто 24.05.2022) - Назва з екрана.