

Міністерство освіти і науки України  
Національний технічний університет  
«Дніпровська політехніка»

Інститут електроенергетики  
(інститут)

Факультет інформаційних технологій  
(факультет)

Кафедра Програмного забезпечення комп'ютерних систем  
(повна назва)

**ПОЯСНЮВАЛЬНА ЗАПИСКА**  
**кваліфікаційної роботи ступеня**  
*бакалавра*

(назва освітньо-кваліфікаційного рівня)

студента *Богдана Даниїла Сергійовича*  
(ПІБ)

академічної групи *121-19ск-1*  
(шифр)

спеціальності *121 Інженерія програмного забезпечення*  
(код і назва спеціальності)

освітньої програми *Інженерія програмного забезпечення*  
(назва освітньої програми)

на тему: *Розробка мобільного додатку персональний електронний*  
записник справ

Керівники	Прізвище, ініціали	Оцінка за шкалою		Підпис
		рейтинговою	інституційною	
кваліфікаційної роботи	<i>доц. Кабак Л.В.</i>			
<b>розділів:</b>				
спеціальний	<i>доц. Кабак Л.В.</i>			
економічний	<i>доц. Касьяненко Л.В.</i>			
<b>Рецензент</b>				
<b>Нормоконтролер</b>	<i>доц. Гуліна І.Г.</i>			

Дніпро  
2022

Міністерство освіти і науки України  
НТУ «Дніпровська політехніка»

**ЗАТВЕРДЖЕНО:**

завідувач кафедри  
програмного забезпечення комп'ютерних систем  
(повна назва)

І.М. Удовик

(підпис)

(прізвище, ініціали)

«    »                      2022 року

**ЗАВДАННЯ**

на кваліфікаційну роботу

бакалавра

(назва освітньо-кваліфікаційного рівня)

студента 121-19ск-1  
(група)

Богдан Д.С.  
(прізвище та ініціали)

тема кваліфікаційної роботи  
електронний записник справ

Розробка мобільного додатку персональний

затверджена наказом ректора НТУ «ДП» від «18» травня 2022 р. № 268-с

Розділ	Зміст виконання	Термін виконання
Спеціальний	На основі матеріалів проектно-технологічної та інших науково-технічних джерел провести аналіз стану рішення проблеми та постановку задачі. Обґрунтувати вибір та здійснити реалізацію методів вирішення проблеми	13.05.2022 р.
Економічний	Провести розрахунок трудомісткості розробки програмного забезпечення, витрат на створення ПЗ й тривалості його розробки	27.05.2022 р.

Завдання видав

(підпис)

доц. Кабак Л.В.

(посада, прізвище, ініціали)

Завдання прийняв до виконання

(підпис)

Богдан Д.С.

(прізвище, ініціали)

Дата видачі завдання: 14.01.2022 р.

Термін подання кваліфікаційної роботи до ЕК: 13.06.2022 р.

## РЕФЕРАТ

Пояснювальна записка: \_\_\_с., \_\_\_ рис., \_\_\_ табл., \_\_\_ дод., \_\_\_ джерел.

Об'єкт розробки: мобільний додаток персональний електронний записник справ.

Мета кваліфікаційної роботи: Розробка мобільного додатку персональний електронний записник справ.

У вступі виконується аналіз сучасного стану проблеми, уточнюється постановка завдання, мета кваліфікаційної роботи та галузь її застосування, обґрунтовується актуальність теми. У першому розділі проводиться дослідження предметної області та існуючих рішень, визначається актуальність завдання та призначення розробки, розроблюється постановка завдання.

У другому розділі обирається платформа для розробки, виконується проектування програми і її розробка, наводиться опис алгоритму і структури функціонування системи, визначаються вхідні і вихідні дані, наводяться характеристики складу параметрів технічних засобів, описується робота програми.

В економічному розділі визначено трудомісткість розробленої інформаційної системи, проведений підрахунок вартості роботи по створенню програми та розраховано час на його створення.

Практичне значення роботи полягає в тому, що розроблений додаток може бути використаний користувачами, зацікавленими у плануванні справ та подій на кожен день.

Актуальність програмного продукту визначається наявністю планування часу на кожен день.

Список ключових слів: ПРОГРАМА, ВИТРАТИ, МОБІЛЬНИЙ ДОДАТОК, ПРОГРАМУВАННЯ, ІНФОРМАЦІЙНА СИСТЕМА.

## **ABSTRACT**

Explanatory note: \_\_\_pages, \_\_\_figures, \_\_\_tables, \_\_appendices, \_\_\_sources.

Object of development: mobile application personal electronic recorder.

The purpose of the qualification work: development of a mobile application personal electronic recorder.

The introduction analyzes the current state of the problem, specifies the task, the purpose of the qualification work and the field of its application, substantiates the relevance of the topic.

In the first section the research of the subject area and existing decisions is carried out, the urgency of the task and the purpose of development is determined, the statement of the task is developed.

The second section selects the platform for development, performs program design and development, describes the algorithm and structure of the system, determines the input and output data, provides characteristics of the parameters of technical means, describes the program.

The economic section determines the complexity of the developed information system, calculates the cost of work to create a program and calculates the time for its creation.

Relevance of the software product by the availability of time planning for each day.

List of key words: PROGRAM, COSTS, CLIENT, MOBILE APP, PROGRAMMING, INFORMATION SYSTEM.

## ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ.....	5
ВСТУП.....	6
РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ.....	7
1.1. 7.....	7
1.2. 8.....	8
1.3. 8.....	8
1.4. 9.....	9
1.5. <b>Ошибка! Закладка не определена.</b> .....	9
1.5.1. <b>Ошибка! Закладка не определена.</b> .....	9
1.5.2.41.....	10
1.5.3.410	
1.5.4.410	
РОЗДІЛ 2. ПРОЄКТУВАННЯ ТА РОЗРОБКА ПРОГРАМНОГО ПРОДУКТУ.	11
2.1. 111	
2.2. 11.....	11
2.3. 11.....	11
2.4. <b>Ошибка! Закладка не определена.</b> .....	15
2.4.1. Проєктування системи.....	15
2.4.2. Створення сторінок.....	18
2.5. <b>Ошибка! Закладка не определена.</b> .....	26
2.6. 26.....	26
2.6.1. 266	
2.6.2. 27.....	27

2.6.3.	276
2.6.4.	288
РОЗДІЛ 3. ЕКОНОМІЧНИЙ РОЗДІЛ.....	36
3.1.	36.....36
3.2.	38.....39
ВИСНОВКИ.....	41
ДОДАТОК А.....	42
ДОДАТОК Б.....	52
ДОДАТОК В.....	90
ДОДАТОК Г.....	91

## **ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ**

БД – база даних;

ПД – програмний додаток;

ПЗ – програмне забезпечення;

ІС – інформаційна система;

ОС – операційна система;

ОП – оперативна пам'ять;

СУБД – система управління базами даних;

SQL – Structured Query Language;

XML – eXtensible Markup Language.

## ВСТУП

Планування — одна з найважливіших умінь людини яка допомагає продуктивніше організувати час на кожен день та виконувати повсякденні справи.

Для планування своїх справ можна використовувати блокнот та ручку, але у час, коли людина більшу частину свого дня проводить в русі, є необхідність впровадити цей процес у її життя. І в цьому випадку незамінним інструментом стає телефон або планшет. Сучасний світ дуже складно уявити без техніки, яка завжди поруч з нами, в цьому випадку, вона може послужити не лише, як засіб зв'язку або дозвілля, а як особистий фінансовий помічник. Перевагами в такому методі обліку стануть: зручність, компактність, надійність. Адже ваш помічник завжди разом з вами: він може нагадувати про витрати, ним можна користуватися у транспорті, на перерві й навіть під час роботи, адже це швидко і легко.

Метою кваліфікаційної роботи є проектування та розробка мобільного додатку операційної системи Android, назвою обрано «ToDoCount», додаток створений для кожного хто вже планує свій час або хоче почати планувати його.

Поданий проект допомагає організувати свої події і справи. Також однією з особливостей додатку є наявність слідкування за денними витратами за день та реалізацією нагадувань справ або повідомлень.

Ведення щоденника — це приємний бонус та заміна рукописного денного щоденника, куди можна написати нотатки на вибраний день.

У наш час існує багато додатків для планування, одні з найкращих мобільних планерів являється «Any.do» та «Мои дела».

В даному дипломному проекті за допомогою мови програмування Java та SQLite розроблений мобільний додаток, що призначений для допомоги в вирішенні повсякденних справ.



## РОЗДІЛ 1

### АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ

#### 1.1. Загальні відомості з предметної області

Об'єктом дослідження та аналізу предметної галузі є мобільні додатки для планування справ з метою організації свого часу. Метою проведення аналізу предметної галузі є виявлення, класифікація, формалізація інформації про усі аспекти предметної галузі, що впливають на характеристики кінцевого програмного продукту.

Були впроваджені наступні роботи:

- збір даних про об'єкт автоматизації та здійснюваних ним видів діяльності;
- вивчення та оцінка якості функціонування об'єкта та здійснюваних ним видів діяльності, виявлення проблем, рішення яких можливе завдяки засобам автоматизації;
- оцінка технічно-економічної, соціальної, практичної доцільності створення програмного продукту;
- виявлення та аналіз вимог користувача до програмного продукту;
- аналіз ринку та потенційних конкурентоспроможних вже існуючих рішень;
- оцінка ефективності прийнятих проектних рішень для створення програмного продукту.

Для збору та обробки інформації про аналогічні проектні рішення були застосовані наступні методи:

- обсервація;
- аналіз документів;
- порівняльний аналіз;
- прогнозування;

- експеримент.

Планування — одна з найважливіших умінь людини яка допомагає продуктивніше організувати час на кожен день та виконувати повсякденні справи.

В ході аналізу мною було виявлено, що у наш час існують багато мобільних додатків для планування.

Одними з найкращих мобільних планерів являється «Any.do».

Any.DO — це функціональний планувальник завдань, з біло-блакитним інтерфейсом. Програма містить величезну кількість функцій, починаючи від прогнозування завдань, віджетів і повідомлень, закінчуючи кооперативною роботою, синхронізацією даних і зручними фішками. Знову ж таки, вона зручна у використанні.

Недоліки теж існують:

- дуже велика кількість контенту, при цьому є перезавантаження ОП;
- в порівнянні з іншими дуже дорогий платний режим \$2.99/місяць.

## **1.2. Призначення розробки та область застосування**

Мобільний персональний електронний записник справ «ToDoCount» має багато властивостей для ведення якісного планування: ведення справ, подій та витрат. Програма нагадує про справи та події у постановлений користувачем час. Дуже простий та сучасний інтерфейс допомагає легко орієнтуватися у програмі.

## **1.3. Підстава для розробки**

Підставою для розробки кваліфікаційної роботи бакалавра на тему «Розробка мобільного додатку персональний електронний записник справ» є наказ по Національному технічному університету «Дніпровська політехніка» від \_\_.\_\_. 2021р. № \_\_\_\_ -\_\_.

## **1.4. Постановка завдання**

Завдання даної кваліфікаційної роботи є спроектувати та розробити мобільний додаток для ведення справ, подій та витрат під ОС Android.

Програма повинна виконувати функції для будь-якого користувача:

1. Ведення списку справ, подій і витрат;
  - 1.1. Створення, редагування, видалення справ. Також повинна бути можливість додавати нагадування;
  - 1.2. Створення, редагування, видалення подій. Також повинна бути можливість додавати нагадування;
  - 1.3. Створення, редагування, видалення витрат.
2. Ведення щоденника та його редагування;
3. Систему оповіщення, при здійсненні подій та справ;
4. Підрахунок загальних витрат за день;
5. Реалізація зручного та швидкого інтерфейсу.

Користувацький інтерфейс мобільного додатку має містити в собі такі особливості:

- бути зручним та інтуїтивно-зрозумілим;
- мати приємну кольорову гаму;
- бути сумісним з потребами і можливостями користувача;
- забезпечувати простоту переходу від виконання однієї функції до іншої.

## **1.5. Вимоги до програми або програмного виробу**

### **1.5.1. Вимоги до функціональних характеристик**

Мобільний додаток призначений для планування ведення своїх подій і справ. Також однією з особливостей додатку є наявність слідкування за денними витратами за день та реалізацією нагадувань справ або повідомлень.

Ведення щоденника — це приємний бонус та заміна рукописного денного

щоденника, куди можна написати нотатки на вибраний день.

Для того, щоб почати роботу з програмою користувачу необхідно просто встановити додаток.

### **1.5.2. Вимоги до інформаційної безпеки**

До розроблювального додатка сформовані наступні вимоги забезпечення її безпеки та цілісності:

- при відмові системи, дані, які були введені в систему, не повинні бути пошкоджені;
- як система, так і оброблені дані, повинні легко переноситися; - час відновлення після збою 2 хвилини.

Надійність роботи розробленого програмного забезпечення залежить від надійності операційної системи, під управлінням якої вона буде функціонувати, і розроблюваного ПЗ.

### **1.5.3. Вимоги до складу та параметрів технічних засобів**

Для коректної роботи на мобільних пристроях необхідно:

- версія Android 4.0 і вище;
- оперативна пам'ять – 1 Гб і вище.

### **1.5.4. Вимоги до інформаційної та програмної сумісності**

Для розробленої програми додаткові програмні засоби не потрібні, бо вона має власний інтерфейс і працює на мобільному додатку.

Для розробки програмного продукту необхідна наявність наступних програм та систем:

- Android Studio;
- MS SQL Server;
- GitHub.

## РОЗДІЛ 2

### ПРОЄКТУВАННЯ ТА РОЗРОБКА ПРОГРАМНОГО ПРОДУКТУ

#### 2.1. Функціональне призначення програми

Результатом кваліфікаційної роботи має бути додаток для кожного хто планує свій час або хоче почати планувати його. Інтерфейс повинен бути простим та інтуїтивно-зрозумілим для користувача.

Експлуатаційне призначення – автоматизація та спрощення ведень персональних справ. Розроблена система дозволяє звести використання паперу, значно зменшує витрати часу, так як розроблена, як мобільний додаток.

Функціональне призначення – надати можливість користувачеві у ролі будь-якої людини, різного віку планувати свій час. Також записувати свої думки у щоденник з автозбереженням.

#### 2.2. Опис застосованих математичних методів

Під час проєктування та розробки даного мобільного додатку використовувалися прості арифметичні операції над даними системи.

#### 2.3. Опис використаних технологій та мов програмування

Для написання програмного продукту було обрано мову програмування Java та середовище AndroidStudio.

Java — об'єктно-орієнтована мова програмування, що розробляється компанією Sun Microsystems з 1991 року і офіційно випущена 23 травня 1995 року. Спочатку нову мову програмування називали Oak (James Gosling) і розроблялась вона для побутової електроніки, але згодом була перейменована в Java і стала використовуватися для написання аплетів, додатків і серверного програмного забезпечення. Програми на Java можуть бути трансльовані в байт-код, що виконується на віртуальній java-машині (JVM) — програмі, яка

оброблює байт-код і передає інструкції обладнанню, як інтерпретатор, але з тією відмінністю, що байт-код, на відміну від тексту, обробляється значно швидше. Мова Java зародилась як частина проекту створення передового програмного забезпечення для різних побутових приладів. Реалізація проекту була почата на мові C ++, але незабаром виник ряд проблем, найкращим засобом боротьби з якими була зміна самого інструмента — мови програмування. Стало очевидним, що необхідна платформи-незалежна мова програмування, яка дозволить створювати програми, які не доводилося б компілювати окремо для кожної архітектури і можна було б використовувати на різних процесорах під різними операційними системами. Мова Java потрібна для створення інтерактивних продуктів для мережі Internet. Фактично, більшість архітектурних рішень, прийнятих при створенні Java, було продиктовано бажанням надати синтаксис, схожий з C і C ++. В Java використовуються практично ідентичні угоди для оголошення змінних, передачі параметрів, операторів і для управління потоком виконання коду. В Java додані всі хороші риси C ++[4].

Три ключові елементи об'єдналися в технології мови Java:

- Java надає для широкого використання свої аплети (applets) — невеликі, надійні, динамічні, які не залежать від платформи активні мережеві додатки, що вбудовуються в сторінки Web;

- Java вивільняє міць об'єктно-орієнтованої розробки додатків, поєднуючи простий і знайомий синтаксис з надійним і зручним в роботі середовищем розробки. Це дозволяє програмісту швидко створювати нові програми і нові аплети;

- Java надає програмісту багатий набір класів об'єктів для ясного абстрагування багатьох системних функцій, використовуваних при роботі з вікнами, мережею і для введення-виведення. Ключова риса цих класів полягає в тому, що вони забезпечують створення незалежних від використовуваної платформи абстракцій для широкого спектра системних інтерфейсів. Важливою

особливістю технології Java є гнучка система безпеки, в рамках якої виконання програми повністю контролюється віртуальною машиною. Будь-які операції, які перевищують встановлені повноваження програми (наприклад, спроба несанкціонованого доступу до даних або з'єднання з іншим комп'ютером), викликають негайне переривання[7].

AndroidStudio — інтегроване середовище розробки (IDE) для платформи Android, представлене 16 травня 2013 року на конференції Google I/O менеджером по продукції корпорації Google — Еллі Паверс (англ. Ellie Powers). 8 грудня 2014 року компанія Google випустила перший стабільний реліз Android Studio 1.0.

Android Studio прийшло на зміну плагіну ADT для платформи Eclipse. Середовище побудоване на базі вихідного коду продукту IntelliJ IDEA Community Edition, що розвивається компанією JetBrains. Android Studio розвивається в рамках відкритої моделі розробки та поширюється під ліцензією Apache 2.0.

Особливості AndroidStudio:

- живі макети (layout): редагувальник WYSIWYG — живе кодування — подання (rendering) програми в реальному часі;
- консоль розробника: підказки по оптимізації, допомога по перекладу;
- Android-орієнтований рефакторинг та швидкі виправлення;
- шаблони для створення поширених Android дизайнів та компонентів;
- багатий редактор макетів (layouts) що дозволяє користувачам перетягнути і покласти (drag-and-drop) компоненти користувацького інтерфейсу, як варіант, переглянути одночасно макети (layouts) на різних конфігураціях екранів [9].

SQLite — полегшена реляційна система керування базами даних. Втілена у вигляді бібліотеки, де реалізовано багато зі стандарту SQL-92. Сирцевий код SQLite поширюється як суспільне надбання (англ. public domain), тобто може використовуватися без обмежень та безоплатно з будь-якою метою.

Фінансову підтримку розробників SQLite здійснює спеціально створений консорціум, до якого входять такі компанії, як Adobe, Oracle, Mozilla, Nokia, Bentley[en] і Bloomberg.

Особливістю SQLite є те, що вона не використовує парадигму клієнт-сервер, тобто рушій SQLite не є окремим процесом, з яким взаємодіє додаток, а надає бібліотеку, з якою програма компілюється і рушій стає складовою частиною програми. Таким чином, як протокол обміну використовуються виклики функцій (API) бібліотеки SQLite. Такий підхід зменшує накладні витрати, час відгуку і спрощує програму. SQLite зберігає всю базу даних (включають визначення, таблиці, індекси і дані) в єдиному стандартному файлі на тому комп'ютері, на якому виконується додаток. Простота реалізації досягається за рахунок того, що перед початком виконання транзакції весь файл, що зберігає базу даних, блокується; ACID-функції досягаються зокрема за рахунок створення файлу-журналу [8].



## 2.4. Опис структури програми та алгоритмів її функціонування

### 2.4.1. Проєктування системи

Use-case діаграма (діаграма прецедентів) необхідна для відображення різних сценаріїв взаємодії між акторами і прецедентами (рис. 2.1).

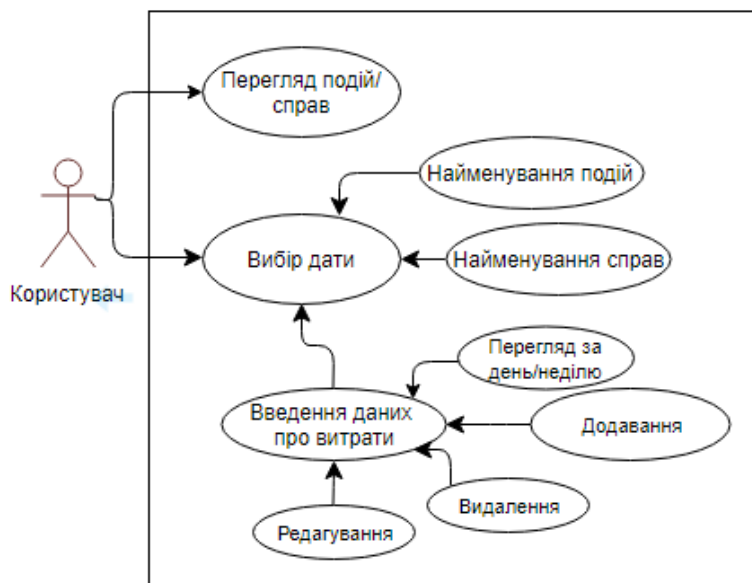


Рис. 2.1. Use-case діаграма додатку

База даних мобільного додатку складається з 7 пов'язаних між собою таблиць: Date — «Дата», Event — «Подія», Events — «Події», Task — «Справа», Tasks — «Справи», Spending — «Витрата», Spendings — «Витрати». Перелік таблиць БД, їх поля та призначення полів наведено у таблицях 2.1 – 2.7.

Таблиця 2.1 – Таблиця Date — «Дата»

Поле	Призначення
Date	Дата
Diary	Щоденик

Таблиця 2.2 – Таблиця Event — «Подія»

Поле	Призначення
IP_event	Ідентифікатор події

Name_event	Найменування події
Reminder_event	Нагадування події
Time_event	Час події
Comm_event	Коментарій події

Таблиця 2.3 – Таблиця Events — «Події»

Поле	Призначення
IP_event	Ідентифікатор події
Date	Дата

Таблиця 2.4 – Таблиця Task — «Справа»

Поле	Призначення
IP_task	Ідентифікатор справи
Name_task	Найменування справи
Reminder_task	Нагадування справи
Comm_task	Коментарій справи

Таблиця 2.5 – Таблиця Tasks — «Справи»

Поле	Призначення
IP_task	Ідентифікатор справи
Date	Дата

Таблиця 2.6 – Таблиця Spending — «Витрата»

Поле	Призначення
IP_spending	Ідентифікатор витрати
Purpose _ spending	Найменування витрати
Suma _ spending	Сума витрати

Таблиця 2.7 – Таблиця Spendings — «Витрати»

Поле	Призначення
IP_spending	Ідентифікатор витрати
Date	Дата

ER-діаграма зв'язку перелічених таблиць приведена на рисунку 2.2.

Для реалізації програми дипломного проекту була обрана СУБД SQLite, в якій були створені таблиці.

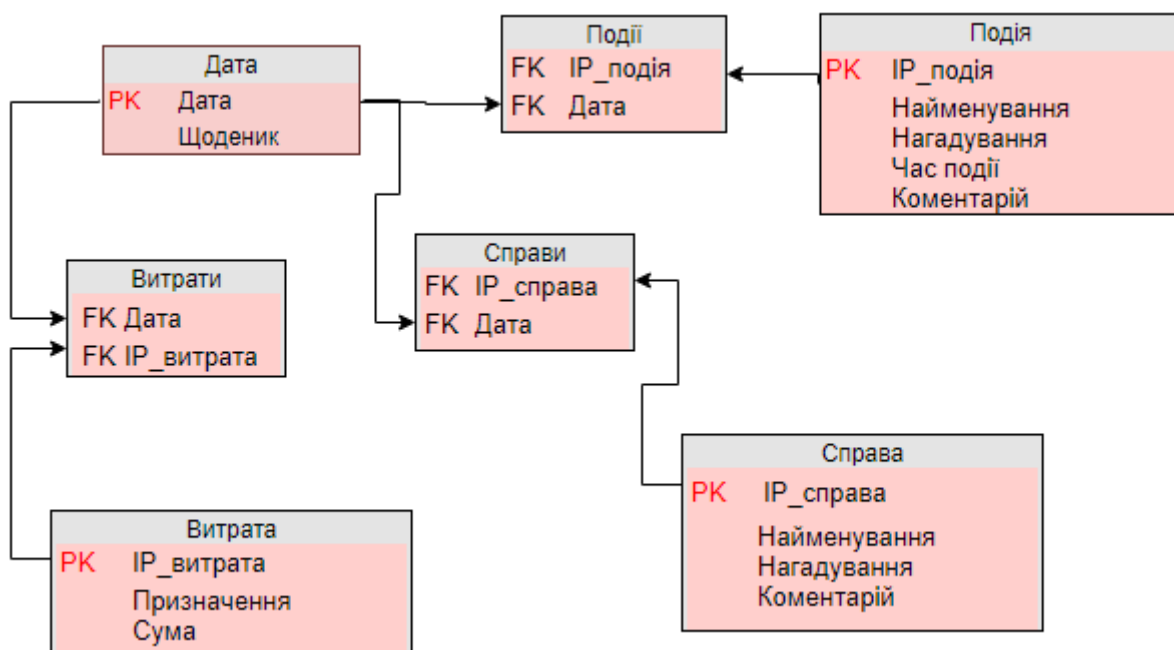


Рисунок 2.2 – ER-діаграма ToDoCount

Також було спроектовано діаграму діяльності програми, яку можна побачити на рисунку 2.3. Вона допомагає розробнику виявити потрібні та не потрібні функції та атрибути у додатку. Допомагає приблизитися до кінцевої форми інтерфейсу і структурувати увесь хід додатку.

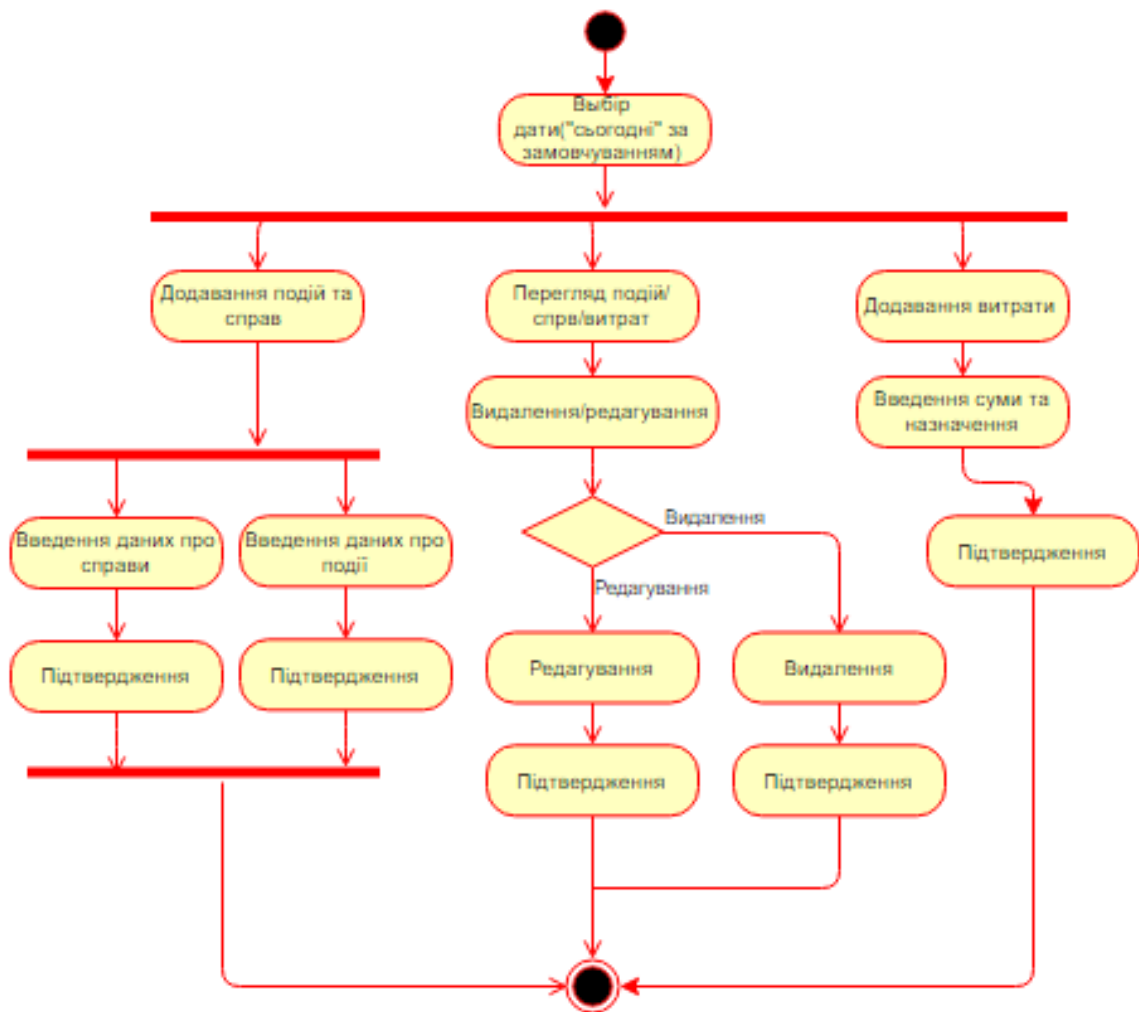


Рисунок 2.3 – Діаграма діяльності

#### 2.4.2. Створення сторінок

З самого початку було спроектовано загальний макет мобільного додатку, на якому відображається розташування основних блоків інформації. Макет дизайну мобільного додатку наведено на рисунку 2.4.

Код інтерфейсу програми представлено у Додатку А.



Рисунок 2.4 – Макет дизайну мобільного додатку

Далі обираються кольори, які в подальшому були використані: темно сірий — для фону; світло синій — для основних елементів; темно жовтий — для тексту. Розроблено логотип додатку, який представлено на рисунку 2.5.



Рисунок 2.5 – Логотип мобільного додатку

Графічний інтерфейс забезпечує взаємодію користувача з програмним додатком. Користувацький інтерфейс програми кваліфікаційної роботи складається з діяльностей (форм), фрагментів та компонентів, які на них розташовані. Стартова форма створюється шляхом зміни властивостей в маніфесті додатка.

Створено схему діяльностей яка подана на рисунку 2.6.

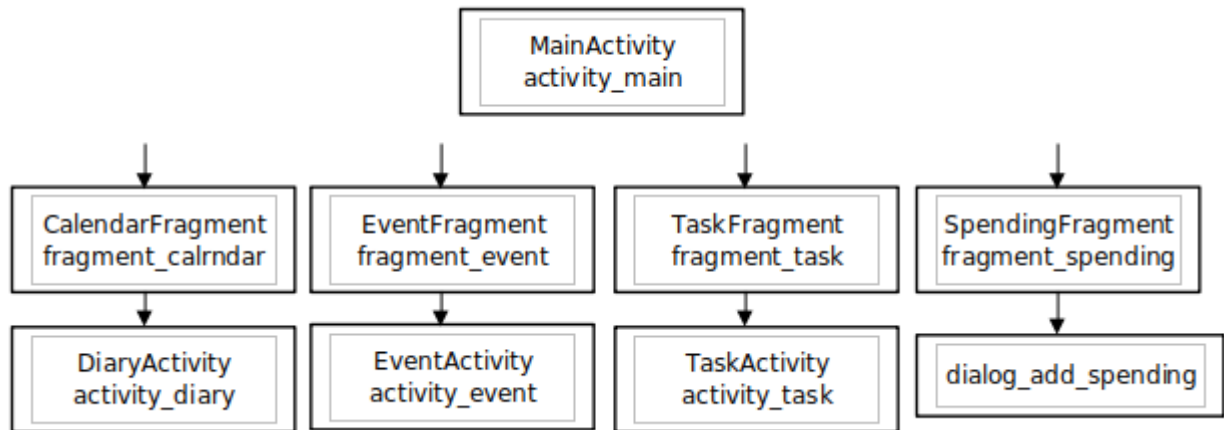


Рисунок 2.6 – Схема діяльностей мобільного додатку

Код кожної діяльності і фрагмента представлено у Додатку А-Б.

CalendarFragment призначено для ведення календаря. Зображення форми наведено на рисунку 2.7. На формі розташовано:

- календар;
- кнопка переходу у активність DiaryActivity;
- статичний текст для виведення загальної суми;
- текстовий заголовок, для виведення дати.

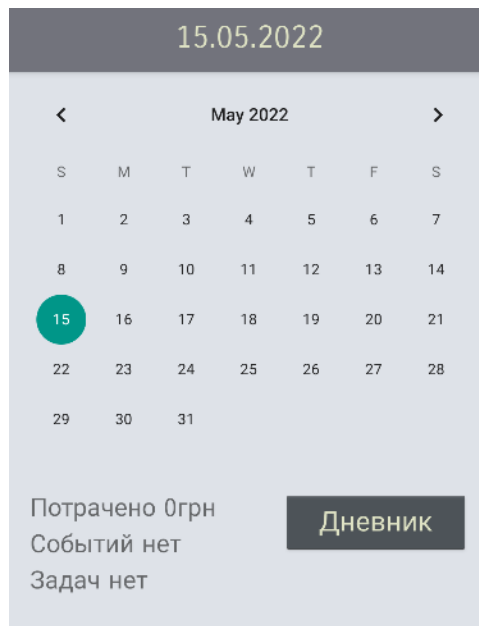


Рисунок 2.7 – Створення форми календаря

DiaryActivity призначений для ведення щоденника. Зображення форми представлено на рисунку 2.8.

На формі розташовано:

- текстовий редактор;
- кнопка повернення;
- текстовий заголовок.



Рисунок 2.8 – Форма щоденника

EventFragment призначений для обліку подій. Приклад переліку подій на певну дату наведено на рисунку 2.9.

На формі можуть бути розташовані:

- елементи списку (в якому зображена подія);
- заголовок з відображенням дати;
- кнопка додавання події (для переходу на діяльність EventActivity).

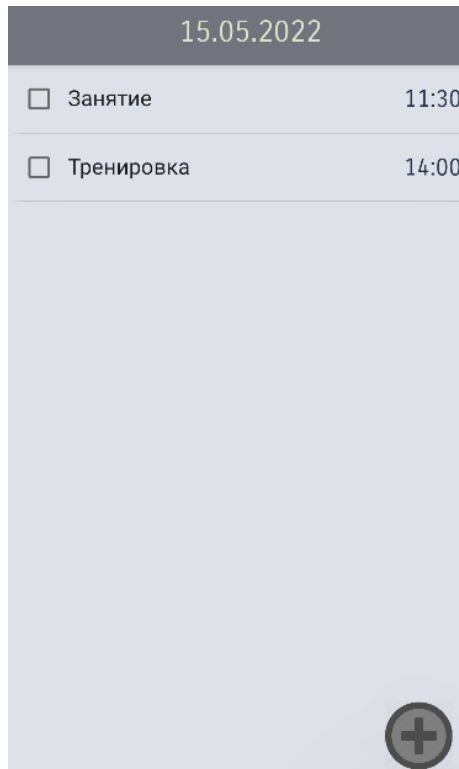


Рисунок 2.9 – Форма введення подій

EventActivity призначена для додавання подій. Зображення форми наведено на рисунку 2.10.

На формі розташовано:

- поле введення назви події;
- перемикач для виводу діалогового вікна часу;
- поле для введення тексту (коментаря);
- кнопка повернення;
- кнопка збереження.



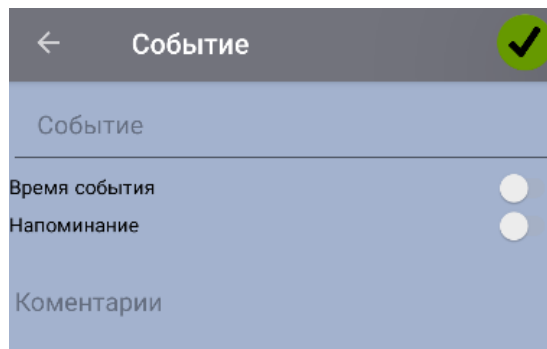


Рисунок 2.10 – Форма додавання подій

TaskFragment для обліку справ. Зображення форми представлено на рисунку 2.11. На формі можуть бути розташовані:

- елементи списку (перелік справ);
- заголовок з відображенням дати;
- кнопка додавання події (для переходу на діяльність TaskActivity).

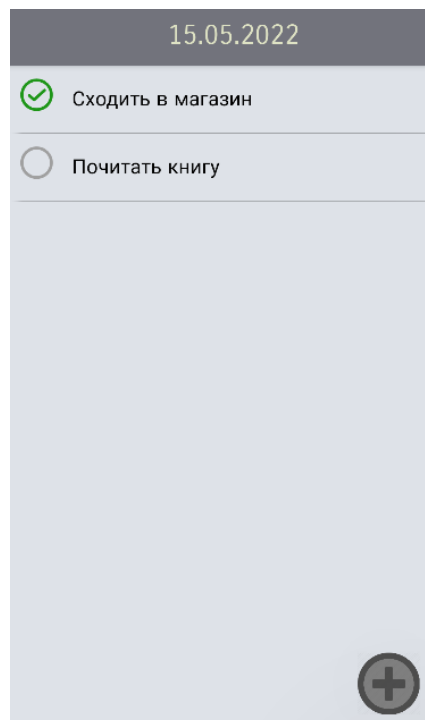


Рисунок 2.11 – Форма обліку справ

TaskActivity призначена для додавання справ. Зображення форми наведено на рисунку 2.12.

На формі розташовані:

- поле введення назви справи;

- перемикач для виводу діалогового вікна часу;
- поле для введення тексту (коментаря);
- кнопка повернення;
- кнопка збереження.

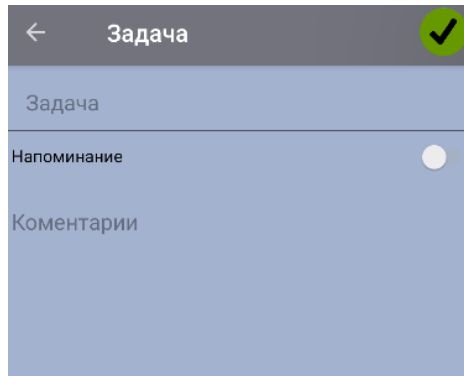


Рисунок 2.12 – Форма додавання справ

SpendingFragment призначена для обліку витрат. Зображення форми наведено на рисунку 2.13.

На формі можуть бути розташовані:

- елементи списку (перелік витрат);
- заголовок для відображення дати;
- кнопка додавання витрат (для переходу на діалогове вікно `dialog_add_spending`).

15.05.2022	
АТБ	240
Джузеппе	480
Маршрутка	30

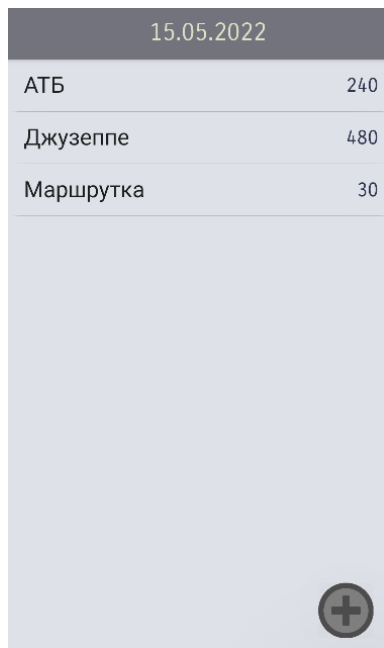


Рисунок 2.13 – Форма обліку витрат

Dialog\_add\_spending призначено для додавання витрат. Зображення форми наведено на рисунку 2.14.

На формі розташовані:

- поле введення призначення витрат;
- поле введення суми витрати;
- кнопка збереження;
- кнопка відміни.

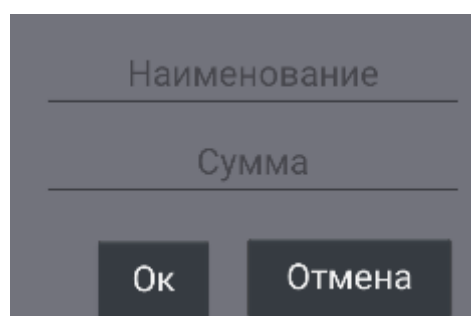


Рисунок 2.14 – Форма додавання витрат

Dialog\_setting призначено для відображення опцій пунктів меню, Зображення форми наведено на рисунку 2.15.

На формі розташовані:

- кнопка редагування обраного пункту;
- кнопка видалення обраного пункту.

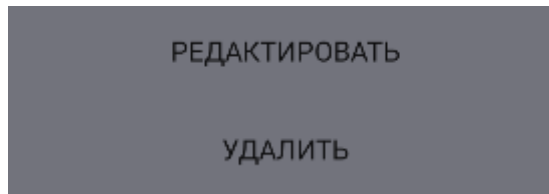


Рисунок 2.15 – Форма для відображення опцій

## **2.5. Обґрунтування та організація вхідних та вихідних даних програми**

Вхідними даними програми є: найменування справ та подій, ведення щоденника та запис витрат.

Вихідними даними програми є: створені справи та події, повідомлення за встановленим часом, аналіз витрат за день.

Всі дані зберігаються в базі даних проєкту.

## **2.6. Опис розробленого програмного продукту**

### **2.6.1. Використані технічні засоби**

Для роботи даного додатку необхідно мобільний пристрій (наприклад, смартфон), що відповідає таким вимогам:

- операційна система Android версії 4.0 і вище;
- оптимальна роздільна здатність дисплею – 1280\*720.

Для тестування та демонстрації роботи даного додатку використовувався мобільний пристрій.

## 2.6.2. Використані програмні засоби

Програмний додаток розроблений у середовищі програмування Android Studio, на мові програмування Java і мовою розмітки сторінки XML. База даних конструювалася в СУБД Workbench.

## 2.6.3. Виклик та завантаження програми

Для того щоб встановити додаток на мобільний пристрій потрібно обрати файл інсталятор в файловій системі мобільного, файл має назву ToDoCount.apk та натиснути кнопку встановити, як це зображено на рисунку 2.16. Після вдалого встановлення додатку буде виведено повідомлення, приклад якого наведено на рисунку 2.17.

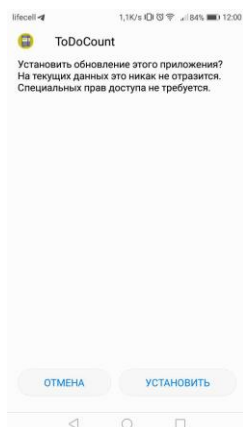


Рисунок 2.16 – Встановлення додатку на мобільний пристрій

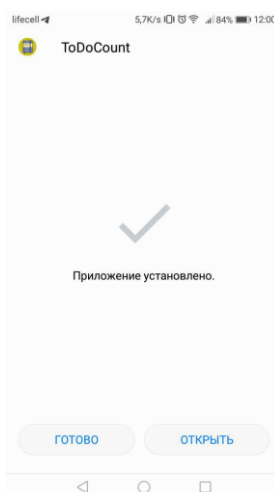


Рисунок 2.17 – Повідомлення про успішне встановлення додатку

## 2.6.4. Опис інтерфейсу користувача

При запуску додатку на екрані буде відображений календар з пунктами головного меню, як це зображено на рисунку 2.18.

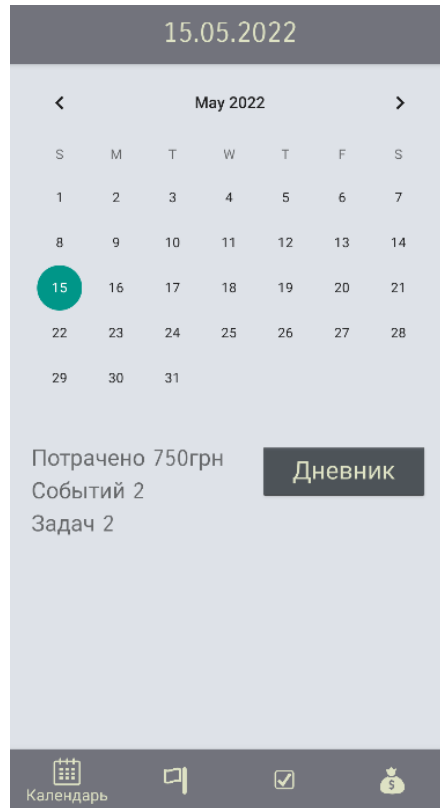


Рисунок 2.18 – Форма календаря

Після входу у додаток буде можливо виконання наступних дій:

- додавання, редагування та видалення події;
- додавання, редагування та видалення справи;
- додавання, редагування та видалення витрат за добу;
- записувати свої враження у щоденнику.

Для того щоб ввести нову подію треба натиснути на кнопку з прапорцем, як це зображено на рисунку 2.19, після чого здійсниться перехід до форми подій, яку зображено на рисунку 2.20.



Рисунок 2.19 – Навігаційна кнопка «Події»

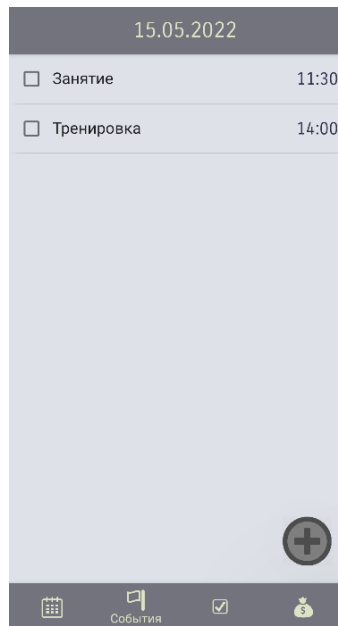


Рисунок 2.20 – Форма обліку подій

Для того щоб додати подію треба натиснути кнопку додавання, яку представлено на рисунок 2.21.



Рисунок 2.21 – Кнопка додавання

Після натискання на кнопку додавання відкриється форма додавання події, яку зображено на рисунках 2.22 – 2.23. Для збереження введеної події потрібно натиснути зелену галочку.

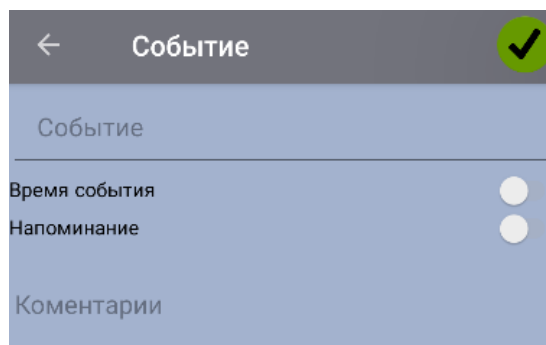


Рисунок 2.22 – Форма додавання події

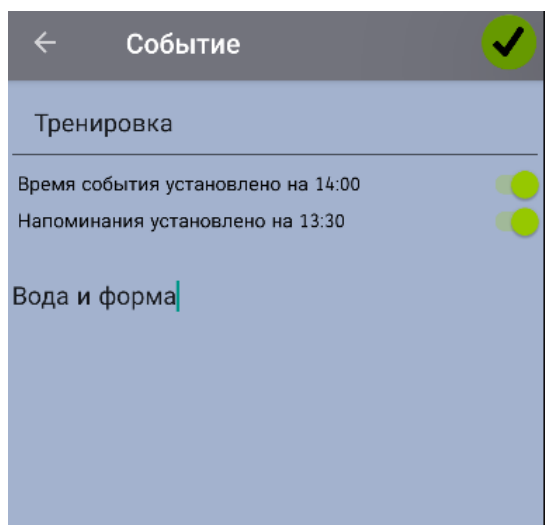


Рисунок 2.23 – Заповнена форма додавання

Для вибору часу створена форма для часу яку видно на рисунок 2.24.

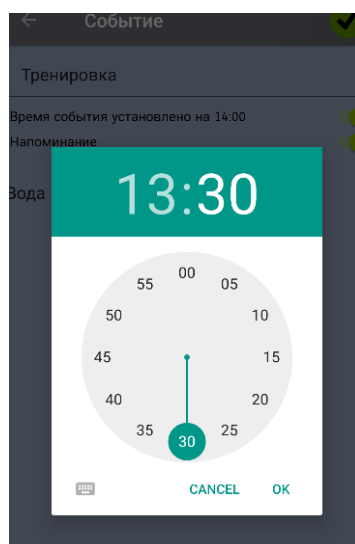


Рисунок 2.24 – Заповнена форма додавання

Для того щоб почати роботу зі справами треба натиснути на кнопку з галочкою яку можна побачити на рисунку 2.25 і ви перейдете до форми подій яку видно на рисунку 2.26.



Рисунок 2.25 – Навігаційна кнопка «Справи»



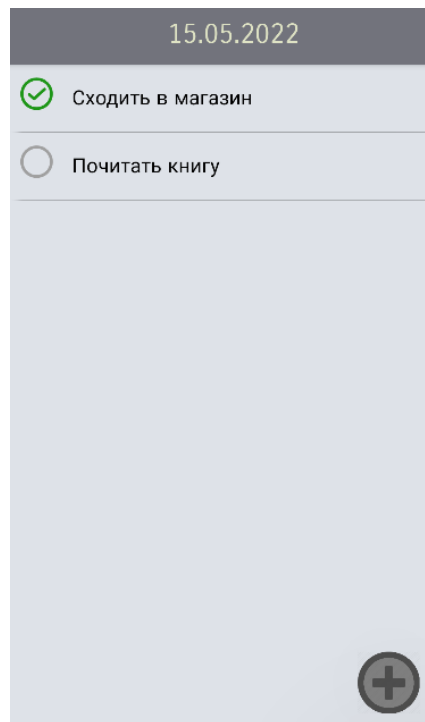


Рисунок 2.26 – Форма обліку справ

Для того щоб додати справу треба натиснути кнопку додавання (дивись на рисунок 2.21).

Після натискання на кнопку додавання відкриється форма додавання справи яку видно на рисунок 2.27 – 2.28. Для збереження потрібно натиснути зелену галочку.

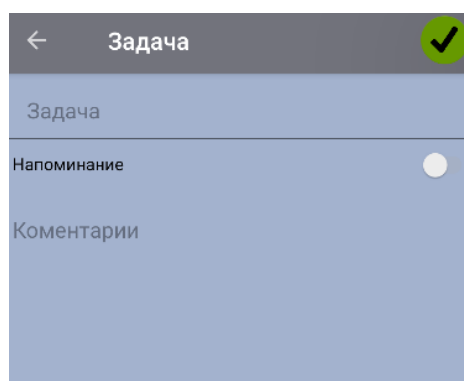


Рисунок 2.27 – Форма додавання справи

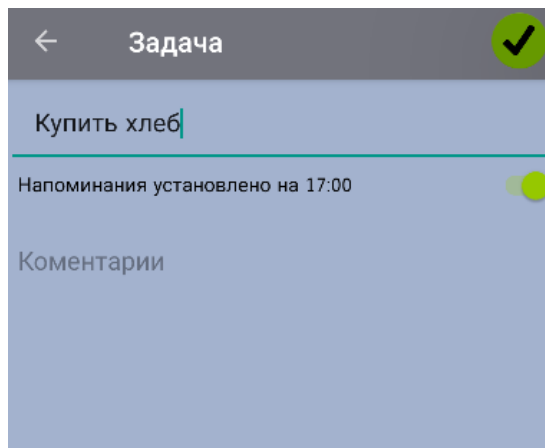


Рисунок 2.28 – Заповнена форма додавання справи

Для того щоб почати роботу з витратами треба натиснути на кнопку з грошима це можна побачити на рисунку 2.29 і ви перейдете до форми витрати яку видно на рисунку 2.30.



Рисунок 2.29 – Навігаційна кнопка «Витрати»

15.05.2022	
АТБ	240
Джузеппе	480
Маршрутка	30

Рисунок 2.30 – Форма обліку витрат

Для того щоб додати витрату треба натиснути кнопку додавання (див. рис. 2.21).

Після натискання на кнопку додавання відкриється форма додавання витрат, яка представлена на рисунках 2.31 – 2.32. Для збереження введеної інформації потрібно натиснути кнопку «ОК».

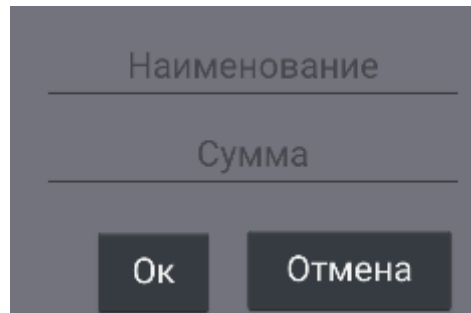


Рисунок 2.31 – Форма додавання справи

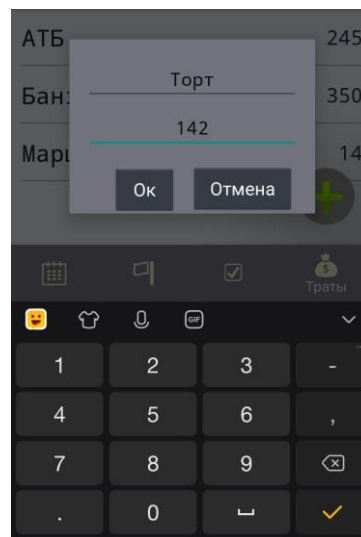


Рисунок 2.32 – Заповнена форма додавання справи

При натисканні на пункт будь-якого списку відкриється форма опцій яка наведена на рисунок 2.33. Для того щоб відредагувати пункт списку натисніть кнопку «Відредагувати».

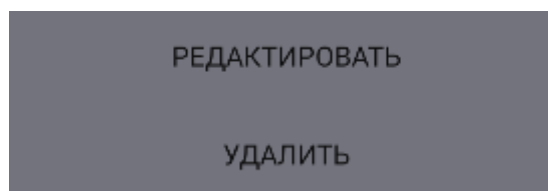


Рисунок 2.33 – Форма опцій

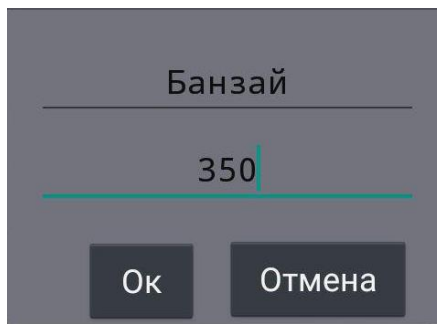


Рисунок 2.34 – Редагування

Для того відкриття вікна ведення щоденника треба перейти на форму календаря і натиснути кнопку «Щоденник». З’явиться форма, яка представлена на рисунку 2.35.



Рисунок 2.35 – Форма «Щоденник»

При додаванні нагадування будуть створені повідомлення, які працюють у встановлений користувачем час. Робота повідомлення представлена на рисунку 2.36.

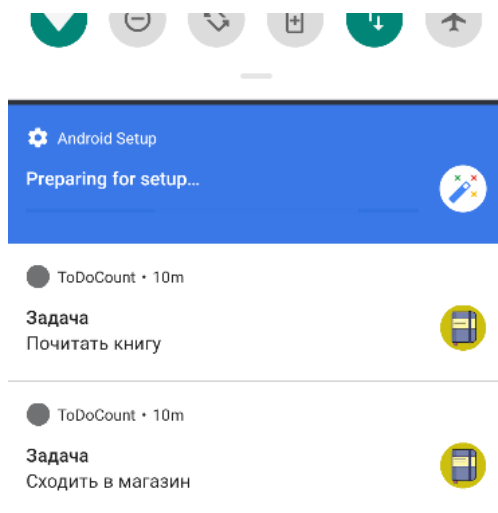


Рисунок 2.36 – Реалізація повідомлень

## РОЗДІЛ 3

### ЕКОНОМІЧНИЙ РОЗДІЛ

#### 3.1. Обчислення трудомісткості розробки програмного забезпечення

Вхідні дані:

1.  $q$  – передбачуване число операторів – 1500.

2.  $C$  – коефіцієнт складності програми – 1,4.

3.  $p$  – коефіцієнт корекції програми в ході її розробки – 0,093.

4.  $B$  – коефіцієнт збільшення витрат – 1,25.

5.  $k$  – коефіцієнт кваліфікації програміста – 0,8.

6.  $C_{\text{ПР}}$  – середня годинна заробітна плата програміста, грн/год. Згідно зі статистичними даними з сайту DOU.ua, для програміста кваліфікації Junior Software Engineer заробітна плата при розробці на Java складає в середньому 750\$ на місяць. З урахуванням 40 годинного робочого тижня та при чинному курсі Національного банку України 1 долар = 29,46 гривень, отримаємо значення 117 грн/год.

7.  $B_k$  – число виконавців – 1.

8.  $F_p$  – місячний фонд робочого часу (при 40 годинному робочому тижні  $F_p = 176$  годин).

9.  $C_{\text{МЧ}}$  – вартість машино-години ЕОМ, грн/год – 4,8.

Нормування праці в процесі створення ПЗ істотно ускладнено в силу творчого характеру праці програміста. Тому трудомісткість розробки ПЗ може бути розрахована на основі системи моделей з різною точністю оцінки.

Трудомісткість розробки ПЗ можна розрахувати за формулою (3.1):

$$t = t_o + t_u + t_a + t_n + t_{отл} + t_d, \text{ людино-годин,} \quad (3.1)$$

де  $t_o$  – витрати праці на підготовку й опис поставленої задачі (приймається 50),

$t_u$  – витрати праці на дослідження алгоритму рішення задачі,

$t_a$  – витрати праці на розробку блок-схеми алгоритму,

$t_n$  – витрати праці на програмування по готовій блок-схемі,

$t_{отл}$  – витрати праці на налагодження програми на ЕОМ,

$t_d$  – витрати праці на підготовку документації.

Складові витрати праці визначаються через умовне число операторів у ПЗ, яке розробляється. Умовне число операторів (підпрограм) визначається за формулою (3.2):

$$Q = q * C * (1 + p), \quad (3.2)$$

де  $q$  – передбачуване число операторів,

$C$  – коефіцієнт складності програми,

$p$  – коефіцієнт кореляції програми в ході її розробки.

$$Q = 1500 * 1,4 * (1 + 0,08) = 2\,295,30$$

Витрати праці на вивчення опису задачі  $t_u$  визначається з урахуванням уточнення опису і кваліфікації програміста представлені в формулі (3.3):

$$t_u = \frac{Q * B}{(75.85) * k}, \text{ людино – годин,} \quad (3.3)$$

де  $B$  – коефіцієнт збільшення витрат праці внаслідок недостатнього опису задачі,

$k$  – коефіцієнт кваліфікації програміста, обумовлений від стажу роботи з даної спеціальності.

$$t_u = \frac{2\,295,30 * 1,25}{75.85 * 0.8} = 44.83, \text{ людино – годин.}$$

Витрати праці на розробку алгоритму рішення задачі розраховуються за формулою (3.4):

$$t_a = \frac{Q}{(20..25) * k}, \text{ людино – годин.}$$

$$t_a = \frac{2\,295,30}{20 * 0,8} = 143,46, \text{ людино – годин.} \quad (3.4)$$

Витрати на складання програми по готовій блок-схем розраховуються за формулою (3.5):

$$t_n = \frac{Q}{(20..25) * k}, \text{ людино – годин.}$$

$$t_n = \frac{2\,295,30}{24 * 1} = 119,55, \text{ людино – годин.} \quad (3.5)$$

Витрати праці на налагодження програми на ЕОМ розраховуються за формулою (3.6):

$$t_{отл} = \frac{Q}{(4..5) * k}, \text{ людино – годин.}$$

$$t_{отл} = \frac{2\,295,30}{5 * 1} = 573,83, \text{ людино – годин.} \quad (3.6)$$

Витрати праці на підготовку документації представлені у формулі (3.7):

$$t_d = t_{др} + t_{до}, \quad (3.7)$$

де  $t_{др}$  – трудомісткість підготовки матеріалів і рукопису, що розраховується за формулою 3.8.

$$t_{др} = \frac{Q}{(15..20) * k}, \text{ людино – годин,}$$

$$t_{др} = \frac{2\,295,30}{16 * 1} = 179,32 \text{ людино – годин.} \quad (3.8)$$

$t_{до}$  – трудомісткість редагування, печатки й оформлення документації, що розраховується за формулою 3.9.

$$t_{до} = 0,75 * t_{др}, \text{ людино – годин.} \quad (3.9)$$

$$t_{до} = 0,75 * 179,32 = 134,49, \text{ людино – годин.}$$

$$t_d = 179,32 + 134,49 = 313,81, \text{ людино – годин.}$$

$$t = 44,83 + 143,46 + 119,55 + 573,83 + 313,81 = 1245,47, \text{ людино – годин}$$

### 3.2. Обчислення витрат на створення програмного забезпечення

Витрати на створення ПЗ  $K_{ПО}$  включають витрати на заробітну плату виконавця програми  $Z_{ЗП}$  і витрат машинного часу, необхідного на налагодження програми на ЕОМ. Розраховується  $K_{ПО}$  за формулою (3.10)



$$K_{\text{ПО}} = Z_{\text{ЗП}} + Z_{\text{МВ}}, \text{ грн.} \quad (3.10)$$

Заробітна плата виконавців визначається за формулою (3.11):

$$Z_{\text{ЗП}} = t * C_{\text{ПР}}, \text{ грн,} \quad (3.11)$$

де:  $t$  – загальна трудомісткість, людино-годин,

$C_{\text{ПР}}$  – середня годинна заробітна плата програміста, грн/година.

$$Z_{\text{ЗП}} = 1245,47 * 117 = 145\,719,84, \text{ грн.}$$

Вартість машинного часу, необхідного для налагодження програми на ЕОМ визначається за формулою (3.12):

$$Z_{\text{МВ}} = t_{\text{отл}} * C_{\text{мч}}, \text{ грн,} \quad (3.12)$$

де  $t_{\text{отл}}$  – трудомісткість налагодження програми на ЕОМ, год,

$C_{\text{мч}}$  – вартість машино-години ЕОМ, грн/год.

$$Z_{\text{МВ}} = 573,83 * 4,5 = 2\,582,21 \text{ грн.}$$

$$K_{\text{ПО}} = 145\,719,84 + 2\,582,21 = 148\,302,06, \text{ грн.}$$

Визначені в такий спосіб витрати на створення програмного забезпечення є частиною одноразових капітальних витрат на створення АСУП.

Очікуваний період створення ПЗ визначається за формулою (3.13):

$$T = \frac{t}{B_k * F_p}, \text{ міс,} \quad (3.13)$$

де  $B_k$  - число виконавців,

$F_p$  - місячний фонд робочого часу.

$$T = \frac{1245,47}{1 * 176} = 7,08, \text{ міс.}$$

Висновки: обчислено трудомісткість розробленого додатку (1245,47 людино годин), проведено обчислення вартості роботи по створенню програми (148 302,06 грн.) та визначено час на її створення (7,08 місяців).

## ВИСНОВКИ

Кваліфікаційна робота «Розробка мобільного додатку персональний електронний записник справ» був реалізований засобами середовища Android Studio 3.0. Він надає можливість планувати свій час та організовувати свої плани.

Мобільний додаток може зберігати дані користувача про події, справи та витрати на певні дати та забезпечує можливість перегляду, додавання, редагування та видалення цих даних.

Розроблений мобільний додаток має сучасний і дуже простий графічний інтерфейс, а користувач потребує лише базових знань роботи зі смартфоном.

Розроблена пояснювальна записка акцентує увагу на інтерфейсі мобільного додатку та інструкції користувача, що спрощує його використання.

Мобільний додаток може бути корисним та актуальним для кожного користувача будь якого віку, тому що допомагає організувати свої денні та недільні плани.

Постановка задачі містить інформацію про вимоги до використовуваних технічних засобів, опис інструментальних засобів, що використовувались при розробці програми. Опис етапів реалізації містить докладну інформацію про проектування, розробку бази даних та інтерфейсу мобільного додатку. Опис програмного продукту являє собою інструкцію роботи з мобільним додатком.

В даному дипломному проекті за допомогою мови програмування Java та SQLite розроблений мобільний додаток, що призначений для допомоги в вирішенні повсякденних справ.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. ДСТУ 3008-95 Документація. Звіти у сфері науки і техніки. Структура і правила оформлення.
2. Мартин, Г. Практическое руководство [Електронний ресурс] / Грабер Мартин // Лори. – 2014. – Режим доступу до ресурсу: [http://www.htbook.ru/kompjutery\\_i\\_seti/programmirovanie/sql-dlya-nachinayuschih..](http://www.htbook.ru/kompjutery_i_seti/programmirovanie/sql-dlya-nachinayuschih..)
3. Николай П. HTML, JavaScript, PHP и MySQL. Джентльменский набор Web-мастера [Електронний ресурс] / Прохоренок Николай // БХВ-Петербург. – 2010. – Режим доступу до ресурсу: <http://forcoder.ru/html-css/html-javascript-php-i-mysql-dzhentlmskij-nabor-web-mastera-1120>.
4. Корнелл Г. Java. Библиотека профессионала. Том 1. Основы / Г. Корнелл, К. Хорстманн. – Калифорния, 2016. – 866 с. – (Вильямс). – (978-5-8459-2084-3; 10).
5. Изучаем Java на примерах и задачах [Електронний ресурс] // Наука и техника. – 2016. – Режим доступу до ресурсу: <http://forcoder.ru/java/izuchaem-java-na-primerah-i-zadachah-1528>.
6. Head First Java [Електронний ресурс] // O`Reilly. – 2009. – Режим доступу до ресурсу: <http://forcoder.ru/java/head-first-java-1127>.
7. Полный справочник по Java [Електронний ресурс] // Вильямс. – 2007. – Режим доступу до ресурсу: <http://forcoder.ru/java/polnyj-spravochnik-po-java-377>.
8. Особенности SQLite [Електронний ресурс] // DB Browser for SQLite. – 2016. – Режим доступу до ресурсу: <https://sqlitebrowser.org/>.
9. Розробка на AndroidStudio [Електронний ресурс] // AndroidDev. – 2021. – Режим доступу до ресурсу: <https://developer.android.com/studio>.

10. Документація. Звіти у сфері науки і техніки. Структура і правила оформлення : ДСТУ 3008-95. – Чинний від 1996–01–01. – К. : Держстандарт України, 1996. – 39 с.

11. ДСТУ 2394-94 Інформація та документація. Комплектування фонду, бібліографічний опис, аналіз документів. Терміни та визначення. – Чинний від 01.01.1995. - Київ : Держстандарт України, 1994. – 88 с.

12. Методичні вказівки з виконання економічного розділу в дипломних проектах студентів спеціальності “Комп’ютерні системи ” / Укладачі О.Г. Вагонова, Нікітіна О.Б. Н.Н. Романюк – Дніпропетровськ: Національний гірничий університет. – 2013. – 23с.

13. Цехнер, М. Программирование игр под Android / Марио Цехнер. – СПб. : Питер, 2013. – 688 с.

14. Benjamin Pierce. Types and Programming Languages. — MIT Press, 2002. – 221с.

15. Raghav Sood, Pro Android Augmented Reality. – Apress, 2012. – 343.

16. Android Studio Developers, Create Notifications [Електронний ресурс] // AndroidDev. – 2019. – Режим доступу до ресурсу: <https://developer.android.com/training/notify-user/build-notification>.

17. Методичні рекомендації до виконання кваліфікаційних робіт бакалаврів напряму підготовки 122 «Комп’ютерні науки» галузі знань 12 Інформаційні технології/, Л.М. Коротенко , О.С. Шевцова; Нац. гірн. ун-т. – Д : ДВНЗ НГУ, 2019. – 65 с.

18. Методичні рекомендації щодо написання, оформлення та представлення учнівських науково-дослідницьких робіт учнів – членів Малої академії наук України / Г.Г. Півняк, Л.М. Коротенко, І.М. Удовик, Є.М. Головня – Д.: ДВНЗ «Національний гірничий університет», 2017. – 24 с.

19. Скорочення слів в українській мові у бібліографічному описі. Загальні правила та вимоги : ДСТУ 3582-97. – Чинний від 1998–07–01. – К. : Держстандарт України, 1998. – 24 с. – (Державний стандарт України).

20. Харди, Б. Программирование под Android. Для профессионалов : [пер. с англ.] / Брайан Харди, Билл Филлипс. – СПб. : Питер, 2014. – 592 с. – (Для профессионалов).