

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА
Факультет інформаційних технологій
Кафедра прикладних інформаційних систем**

**ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА
БАКАЛАВРА
НА ТЕМУ**

Мобільний застосунок «Електронний щоденник»

Галузь знань **12 «Інформаційні технології»**

Спеціальність **122 «Комп'ютерні науки»**

Освітня програма **«Прикладне програмування»**

Освітній рівень: бакалавр

Виконала: студентка 4 курсу, групи ПП-41

_____ Терещук А.А. _____
(прізвище та ініціали)

Керівник _____ Шолохов О.В. _____
(прізвище та ініціали)

_____ к.ф.-м.н., доц. _____
(науковий ступінь, звання)

Унікальність тексту 92%

Випускна кваліфікаційна робота бакалавра допущена до захисту
рішенням кафедри *прикладних інформаційних систем*

Протокол № _____ від _____ р.
зав. кафедри _____ Плескач В.Л.

КАЛЕНДАРНИЙ ПЛАН ВИКОНАННЯ КВАЛІФІКАЦІЙНОЇ РОБОТИ БАКАЛАВРА

Номер	Назва етапів кваліфікаційної роботи бакалавра	Термін виконання етапів кваліфікаційної роботи бакалавра	Відмітка про виконання
1.	Вибір теми та наукового керівника кваліфікаційної роботи бакалавра	14.10.2022	виконано
2.	Видача завдання кваліфікаційної роботи бакалавра	24.10.2022	виконано
3.	Настановча групова співбесіда з питань кваліфікаційної роботи бакалавра	31.10.2022	виконано
4.	Затвердження плану кваліфікаційної роботи бакалавра	01.11.2022	виконано
5.	Підбір та вивчення літературних та інших джерел з теми дослідження	08.11.2022	виконано
6.	Підготовка і подання науковому керівнику першого варіанту I розділу роботи	21.12.2022	виконано
7.	Підготовка і подання науковому керівнику першого варіанту II розділу роботи	31.01.2023	виконано
8.	Підготовка і подання науковому керівнику першого варіанту III розділу роботи	30.03.2023	виконано
9.	Подання роботи у першому варіанті	28.04.2023	виконано
10.	Оформлення пояснювальної записки кваліфікаційної роботи бакалавра	03.05.2023	виконано
11.	Подання кваліфікаційної роботи бакалавра на попередній захист	22.05.2023	виконано
12.	Врахування зауважень керівника і подання роботи в остаточному варіанті (з відповідним висновком про допуск) на кафедрі	26.05.2023	виконано
13.	Затвердження роботи в цілому (підготовка письмового відгуку керівника, письмова рецензія на бакалаврської роботу)	12.06.2023	виконано
14.	Захист кваліфікаційної роботи бакалавра	27.06.2023	

Здобувач вищої освіти _____

(підпис)

Керівник _____

(підпис)

ВІДОМІСТЬ ДИПЛОМНОЇ РОБОТИ

Складові частини дипломної роботи	Обсяг, арк.
Титульний аркуш	1
Календарний план дипломної роботи	1
Відомість дипломної роботи	1
Анотація	1
Анотація (іноземною мовою-англійською)	1
Зміст	1
Вступ	3
1	9
2	21
3	12
Висновки	1
Перелік використаних джерел	5
Додатки	6

				ДП ХХХХ 00.000.00		
	ПІБ	Підп.	Дата		Лист	Листів
Розробн.	Герещук А.А.		26.05.2023	Відомість дипломної роботи		
Керівн.	Шолохов О.В.		26.05.2023			
Н/контр.	Макаренко С.А.		26.05.2023			
Плескач	Плескач В.Л.					

АНОТАЦІЯ

Дипломна робота: 51 с., 25 рис., 2 табл., 45 джерел, 1 додаток.

Ця дипломна робота присвячена проектуванню та розробленню мобільного застосунку для електронного щоденнику.

Метою дипломної роботи є створення застосунку для ефективної організації діяльності і життя людей за допомогою методів планування часу.

Для досягнення поставленої мети необхідно вирішити такі **задачі**:

- дослідити та проаналізувати ринок мобільних застосунків на предмет потреб користувачів і очікувань щодо особистого щоденника;
- розглянути базові підходи до проектування інтерфейсів електронних щоденників;
- розробити методики проектування інтерфейсу електронного щоденнику із подальшою реалізацією системи.

Об'єктом дослідження є процеси полегшення планування особистого життя та дозвілля.

Предметом дослідження є засоби розроблення дизайну інтерфейсу мобільного застосунку та функціональних можливостей, що впливають на зручність використання, ефективність в управлінні й організації завдань з огляду продуктивності користувача та задоволення його потреб.

Методи дослідження.

Аналіз сучасних методів розробки інтерфейсу органайзера, вибір критеріїв та показників якості інтерфейсу мобільного застосунку, визначення найважливіших параметрів застосунку, розроблення методу створення організації на основі отриманих результатів. Розроблення інтерфейсу користувача та проведення експериментальної оцінки його ефективності.

Ключові слова: застосунок, мобільний застосунок, органайзер, інтерфейс, електронний щоденник.

ABSTRACT

Thesis: 51 pages, 25 figures, 2 tables, 45 sources, 1 appendix.

This thesis is dedicated to the design and development of a mobile application for an electronic diary.

The purpose of the thesis is to effectively organize the activities and lives of people through time planning methods.

To achieve this goal, it is necessary to solve the following **problems**:

- research and analyze the mobile application market for user needs and expectations of a personal diary;
- consider basic approaches to the design of electronic diary interfaces;
- develop the methods for designing an electronic diary interface with subsequent implementation of the system.

The object of the study is the processes of facilitating the planning of personal life.

The subject of research is the means of developing the design of the mobile application interface and its functionality.

Research methods.

Analysis of modern methods of developing the organizer interface, selection of criteria and quality indicators of the mobile application interface, determination of the most important parameters of the application, development of a method for creating an organization based on the results obtained. Development of the user interface and experimental evaluation of its effectiveness.

Keywords: application, mobile application, organizer, interface, electronic diary.

ЗМІСТ

ВСТУП	7
РОЗДІЛ 1 ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ВИЗНАЧЕННЯ ЦІЛЕЙ	10
1.1 Тенденції та характеристики ринку мобільних застосунків	10
1.2 Огляд платформ для мобільних застосунків та їхній аналіз	11
1.3 Аналізаційний огляд аналогів органайзерів	13
1.3.1 Evernote: універсальний електронний щоденник із потужними можливостями	13
1.3.2 Notes: помічник для нотаток та організацій ідей	15
1.3.3 Any.do – найкращий в питаннях організації та досягнення мети	16
1.3.4 Google Календар: зручне та ефективне керування часом	17
РОЗДІЛ 2 ОГЛЯД ОСНОВНИХ ПІДХОДІВ ДО ДИЗАЙНУ ІНТЕРФЕЙСІВ ЕЛЕКТРОННИХ ЩОДЕННИКІВ	19
2.1 Фактори, які впливають на дизайн інтерфейсів електронних щоденників	19
2.2 Огляд можливих систем керування базами даних	30
2.3 Аналіз інструментів розробки програмного забезпечення	33
2.4 Технічні вимоги	38
РОЗДІЛ 3 ПРОЕКТУВАННЯ МЕТОДИКИ ДЛЯ РОЗРОБКИ ІНТЕРФЕЙСУ ЕЛЕКТРОННОГО ЩОДЕННИКА	40
3.1 Аналіз потреб користувачів	40
3.2 Проектування зручного та привабливого інтерфейсу	44
3.3 Створення дизайну	46
3.4 Функціональність мобільного застосунку	48
3.5 Тестування застосунку	51
ВИСНОВОК	53
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	54
ДОДАТКИ	58

ВСТУП

Сьогодні більшість людей не знають, як планувати свій день. Через це багато часу витрачається на непотрібні речі, які виявляються не особливо важливими. Також із плином часу саме мобільні пристрої стали невід'ємною частиною нашого повсякденного життя. Розробка мобільного застосунку електронного щоденника дозволить користувачам мати постійний доступ до свого особистого щоденника в будь-який час і в будь-якому місці.

Сьогодні ефективне планування часу – запорука успіху сучасної людини. Тож розгляд даного питання та з'ясування того, як організувати свій день так, щоб мати вільний час після роботи, стало неабияк на часі. Ось чому в 21 столітті, столітті технологій, люди почали створювати календарі, планувальники, органайзери всіх видів, за допомогою яких можна зберігати всю цю інформацію.

Якщо раніше для планування часу використовувалися фізичні носії, де зберігався облік майбутніх транзакцій, то зараз на зміну їм приходять сучасні технології, де з'являється все більше нових можливостей, таких як: легке сортування, класифікація, призначення міток електронним спискам справ і встановлення нагадувань. Важко не погодитись із тим, що ефективний список справ – це зручний інструмент для спрощення повсякденного життя.

Дослідження мобільного застосунку "електронний щоденник" є актуальним з кількох причин. По-перше, зростання використання смартфонів та мобільних пристроїв створює потребу в ефективних інструментах для організації та управління особистими записами і завданнями. Електронні щоденники можуть забезпечити зручний спосіб зберігання, оновлення та доступу до цих даних з будь-якого місця та в будь-який час.

По-друге, електронні щоденники можуть відігравати важливу роль у покращенні продуктивності та організації робочого часу. Вони дозволяють користувачам створювати список завдань, встановлювати нагадування та

стежити за своїм прогресом. Зручність та доступність мобільних застосунків роблять їх ефективними інструментами для керування часом та досягнення поставлених цілей.

Крім того, використання мобільних застосунків електронного щоденника може сприяти полегшенню роботи та покращенню організації в різних професійних сферах, включаючи менеджмент, освіту, медицину та багато інших. Вони дозволяють зберігати важливі дані, записи, нотатки та документи в одному місці, забезпечуючи швидкий доступ до них і зручне керування інформацією.

Електронний щоденник може бути надзвичайно корисним для сучасних людей з різних причин:

- **Організація та планування:** Електронний щоденник дозволяє легко організувати та планувати щоденні обов'язки, завдання, зустрічі та події. Ви можете створювати розклади, встановлювати нагадування та пріоритети, що допомагає вам краще управляти своїм часом та досягати своїх цілей.
- **Зручний доступ до інформації:** Електронний щоденник забезпечує зручний доступ до вашої інформації з будь-якого пристрою з підтримкою мобільних технологій. Ви можете переглядати, редагувати та оновлювати свої записи навіть під час перебування в дорозі або поза офісом.
- **Збереження та організація інформації:** Електронний щоденник дозволяє зберігати різну інформацію, включаючи тексти, фотографії, голосові нотатки тощо. Ви можете організовувати свої записи за категоріями, тегами або датами, що допомагає знайти потрібну інформацію швидко та ефективно.

Отже, **актуальність дослідження** полягає у потенціалі застосунку для поліпшення організації, продуктивності та керування часом, а також у

можливості застосування в різних професійних сферах та побутовому використанні.

Метою дипломної роботи є створення застосунку для ефективної організації діяльності і життя людей за допомогою методів планування часу..

Завдання дослідження:

- дослідити та проаналізувати ринок мобільних застосунків на предмет потреб користувачів і очікувань щодо особистого щоденника;
- розглянути базові підходи до проектування інтерфейсів електронних щоденників;
- розробити методики проектування інтерфейсу електронного щоденнику із подальшою реалізацією системи.

Об'єктом дослідження є процеси полегшення планування особистого життя та дозвілля.

Предметом дослідження є засоби розроблення дизайну інтерфейсу мобільного застосунку та функціональних можливостей, що впливають на зручність використання, ефективність в управлінні й організації завдань з огляду продуктивності користувача та задоволення його потреб.

Методи дослідження: аналіз сучасних методів розробки інтерфейсу органайзера, вибір критеріїв та показників якості інтерфейсу мобільного застосунку, визначення найважливіших параметрів застосунку, розроблення методу створення організації на основі отриманих результатів. Розроблення інтерфейсу користувача та проведення експериментальної оцінки його ефективності.

Практичне значення одержаних результатів полягає в наданні користувачам потужного інструменту для організації їхнього особистого життя та підтримки їхнього розвитку. Застосунок сприяє поліпшенню продуктивності, плануванню та організації робочих та особистих завдань, а також забезпечує збереження важливих моментів, думок і емоцій.

Структура роботи:

Кваліфікаційна робота бакалавра складається зі вступу, трьох розділів, розподілених на підрозділи та висновку.

РОЗДІЛ 1 ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ВИЗНАЧЕННЯ ЦІЛЕЙ

1.1 Тенденції та характеристики ринку мобільних застосунків

Останні дані свідчать, що середньостатистична людина щодня проводить за власним телефоном 3 години 15 хвилин. І один з п'яти користувачів смартфонів витрачає більше 4,5 годин на свій телефон щодня [1]. Цікавим фактом є те, що середня кількість використання смартфонів в будні дні перевищує вихідні. Люди перевіряють свої телефони в середньому 58 разів на день, а 52% цих перевірок (тобто близько 30 разів на добу) припадає на робочий час. Ця поведінка може стати звичкою, оскільки половина усіх сеансів за телефоном розпочинається менш ніж через 3 хвилини після попереднього [2].

Ринок мобільних застосунків знаходиться в швидкому розвитку, і прогнози провідних компаній свідчать про подальше прискорення цього зростання. З'являється все більше розробників, компаній і самостійних програм. Зараз спостерігається збільшена конкуренція серед застосунків з подібною функціональністю, деякі ставлять акцент на дизайн, інші пропонують нові функції, а ще інші активно рекламують свої продукти. Згідно з аналізом ринку мобільних застосунків, прогнозується, що до кінця 2023 року щорічна кількість завантажень застосунків по всьому світу досягне 299 мільярдів. Це значне зростання порівняно з 2020 роком, коли глобальна кількість завантажень становила близько 247 мільярдів [3]. За даними Grand View Research, Inc., очікується, що глобальний ринок програмного забезпечення для управління ефективністю досягне \$102,98 млрд до 2027 року з річним приростом у 13,4% [4].

Європейці, які залишаються гедоністами, не поступаються собі у розвагах, фізичній активності та саморозвитку (рис. 1.1). Популярність

розважальних програм, таких як Netflix, Disney+ і Twitch, зросла на 28% в березні 2023 року порівняно з лютим, зі загальною кількістю завантажень досягаючи 132 мільйони. Завантаження категорії "Книги" зросли на 21%. У категорії "Музика" зростання склало лише 13%. Завантаження новин збільшилися на 66%, а здоров'я та фітнесу - на 46%, а продуктивність - на 38% [5].

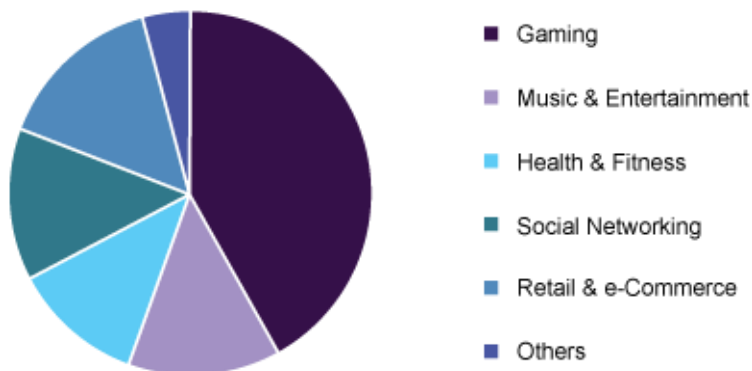


Рисунок 1.1 – Аналіз мобільних застосунків за категоріями

Підприємства, які активно вкладаються у мобільний сектор, досягають значного успіху і отримують значний прибуток. Наприклад, три найбільші IPO (Initial Public Offering) у 2020 році були пов'язані з компаніями, які мають ключову присутність у мобільному бізнесі. На кінець року загальна ринкова капіталізація мобільних компаній становила 544 мільярди доларів [6].

1.2 Огляд платформ для мобільних застосунків та їхній аналіз

Android має більшу частку на світовому ринку, складаючи 68.61% від загальної кількості користувачів. Прогнозується, що ця тенденція продовжиться і в майбутньому [7]. З іншого боку, програми для iOS спрямовані на аудиторію з вищою купівельною спроможністю, що робить їх привабливими для розробників (рис. 1.2).



Рисунок 1.2 – Ринок мобільних операційних систем

Хоча Android має більше користувачів загалом, Apple продовжує домінувати на ринках країн з вищим рівнем життя, таких як Північна Америка, Південна Америка, Європа та Азія [8]. З іншого боку, Android показує свою перевагу в країнах, що розвиваються (рис. 1.3).



Рисунок 1.3 – Карта користувачів за регіонами

Розробка для iOS є швидшою та менш затратною через меншу фрагментацію, у той час як Android має багато версій та різноманітні пристрої, що ускладнює процес розробки та тестування програм, затрачаючи багато часу та ресурсів. Крім того, лише Apple використовує операційну систему iOS, а кількість моделей смартфонів і планшетів обмежена. Apple отримує до 45% більше доходу від одного користувача, ніж Android, і користувачі iOS частіше здійснюють покупки в програмі на 10% [9].

Програми для iPhone мають вищу якість та кращий дизайн. Успішні люди віддають перевагу iOS через його простий у використанні інтерфейс, який

допомагає економити час. Крім того, iOS відомий своєю більш високою безпекою порівняно з Android. Це пояснюється тим, що App Store здійснює сувору перевірку програм на віруси після їх завантаження, що надає користувачам впевненість. З іншого боку, Google Play Market має багато різних застосунків, включаючи низькі якості, тоді як App Store попередньо перевіряє всі пропозиції, демонструючи високу якість і контроль продуктів [10].

1.3 Аналізаційний огляд аналогів органайзерів

Для забезпечення створення унікальних застосунків без негативного впливу на користувачів під час розробки, був здійснений аналіз інтерфейсу, що схожий на програми менеджера, та обрані такі популярні програми для платформи iOS, як Google Calendar, Evernote, Notes та Any.do [12].

Аналіз аналогів є важливою частиною дослідження, оскільки він дозволяє оцінити наявні рішення та продукти на ринку, які задовольняють потреби та вимоги користувачів. В контексті розробки мобільного застосунку "Електронний щоденник", аналіз аналогів допоможе отримати уявлення про функціонал, особливості та переваги наявних електронних щоденників на ринку.

Цей аналіз дозволить також ідентифікувати проблеми та недоліки існуючих рішень, а також виявити можливості для покращення та розробки унікальних функцій у нашому власному застосунку. Аналізаційний огляд сприятиме створенню конкурентоспроможного продукту, який відповідає потребам та вимогам користувачів і надає значну перевагу на ринку електронних щоденників.

1.3.1 Evernote: універсальний електронний щоденник із потужними можливостями

Evernote – популярний застосунок для нотаток та підвищення продуктивності, який дозволяє користувачам створювати, організовувати та

синхронізувати свої нотатки на різних пристроях (рис. 1.4). Він пропонує широкий спектр функцій, включаючи форматування тексту, контрольні списки, вкладення, аудіозаписи та сканування документів. Користувачі можуть створювати ідеї, складати списки справ, зберігати веб-статті, фотографії та співпрацювати з іншими користувачами [11].

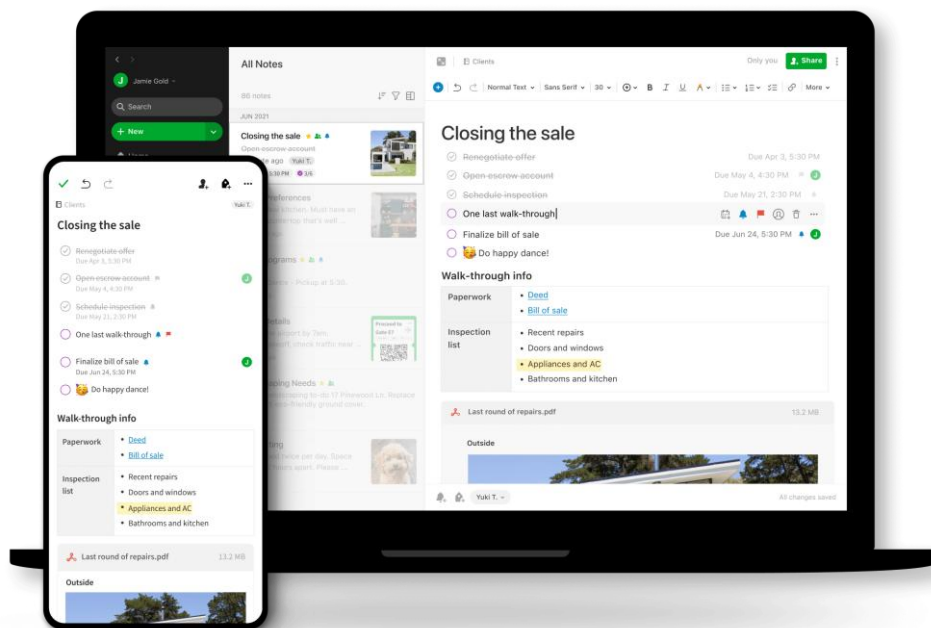


Рисунок 1.4 – Інтерфейс Evernote

Однією з ключових переваг Evernote є його кросплатформна сумісність, що дозволяє користувачам отримувати доступ до своїх нотаток зі смартфонів, планшетів і комп'ютерів. Застосунок також має потужну функцію пошуку, що дозволяє легко знаходити конкретні нотатки або вміст у нотатках. Evernote підтримує синхронізацію через свій хмарний сервіс, що забезпечує користувачам доступ до їх нотаток на всіх пристроях.

Крім того, Evernote надає організаційні можливості, такі як теги та фільтри пошуку, які допомагають користувачам класифікувати та ефективно знаходити свої нотатки. Застосунок також інтегрується з іншими інструментами та сервісами, що дозволяє користувачам підключати Evernote до своїх програм електронної пошти, календарів та інших інструментів продуктивності.

В цілому, Evernote – це універсальний і функціональний застосунок, який допомагає людям залишатися організованими, фіксувати ідеї та ефективно співпрацювати. Він став популярним вибором для особистого та професійного використання завдяки своїй гнучкості, доступності та надійному набору функцій [12].

1.3.2 Notes: універсальний помічник для нотаток та організацій ідей

Notes – це програма для створення та керування цифровими нотатками, доступна на настільних та мобільних платформах iOS та macOS [14]. Вона надає зручний та доступний спосіб для записування швидких думок, створення списків справ, зберігання важливої інформації та організації нотаток. Ці програми часто мають функції форматування тексту, можливість додавання зображень або вкладень, синхронізацію на різних пристроях та потужний пошук для швидкого знаходження потрібних нотаток. Застосунок Notes дозволяє користувачам записувати думки, створювати ескізи, списки та миттєво додавати фотографії до них (рис. 1.5).

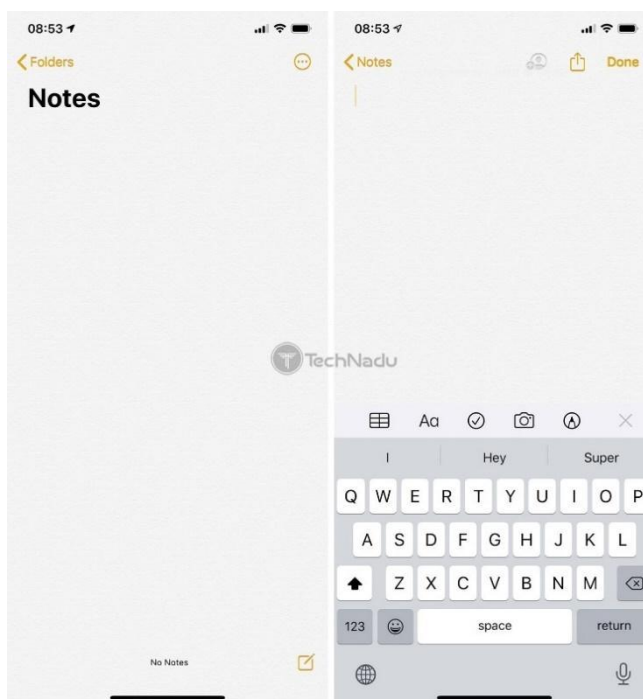


Рисунок 1.5 – Інтерфейс Notes

Програма Notes широко використовується для особистої організації, навчання, професійної роботи та творчих проектів. Вона допомагає розробляти й зберігати нотатки про зустрічі, створювати списки справ, зберігати веб-статті, дослідницькі матеріали, ідеї мозкового штурму та важливу інформацію.

Оскільки Notes є стандартною програмою для iOS, нотатки синхронізуються на всіх пристроях користувача через iCloud. Вона є важливим компонентом операційної системи macOS, а її продуктивність, інтеграція зі стандартними програмами та простий дизайн приваблюють користувачів. Однак, деяким недоліком є обмежена функціональність текстового редактора та можлива незручність в його використанні [15].

1.3.3 Any.do – найкращий в питаннях організації та досягнення мети

Any.do – це програмний продукт, який допомагає вам керувати нотатками та нагадуваннями. У Any.do немає чіткого розрізнення між нагадуваннями та замітками. Єдине відмінність полягає у тому, що нагадування мають час повідомлення, а решта властивостей є однаковими. Any.do організовує ваші завдання відповідно до дати нагадування, таких як "сьогодні", "завтра", "майбутнє" і "минуле". Синхронізація відбувається автоматично. Користувач може перетягувати завдання між різними списками та нотатками, і при цьому прикріплювати файли, записи звуку та нагадування до будь-якого елемента [16].

Any.do також надає можливість переглядати та керувати календарем пристрою разом з завданнями в програмі. Застосунок має багато корисних доповнень, включаючи "панель швидкого додавання", яка дозволяє швидко розміщувати щоденні завдання на панелі сповіщень, а також набір красиво оформлених віджетів для головного екрана. Система спливаючих сповіщень розумно нагадує про пропущені дзвінки.

Інтерфейс програми має зрозумілий і простий у навігації дизайн з безліччю корисних функцій для створення, організації та просування практично будь-якого типу списків. Мінімалістичний дизайн робить інтерфейс Any.do ще більш простим і зручним для користувачів (рис. 1.6).

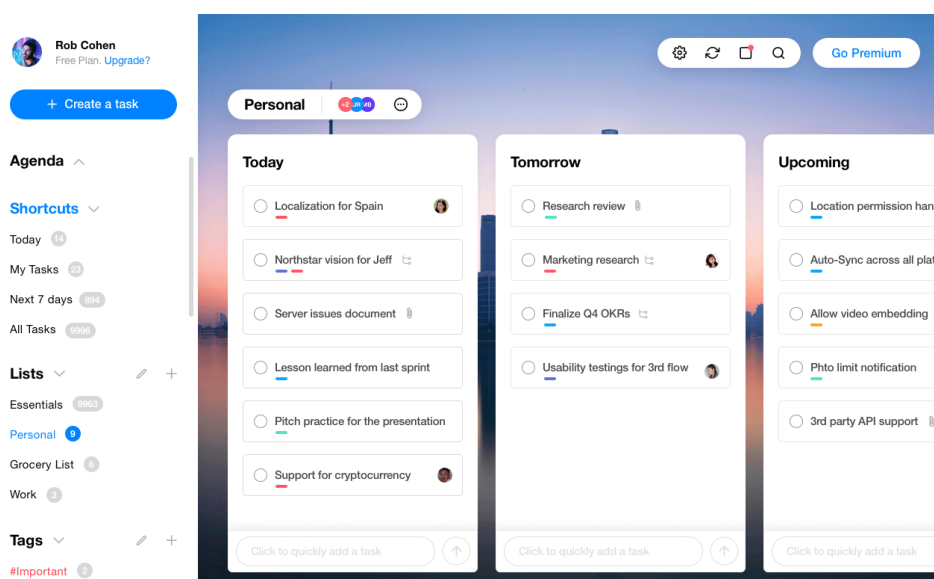


Рисунок 1.6 – Інтерфейс Any.do

В нижній панелі навігації є три вкладки: "Завдання", "Календар" і "Моменти". На сторінці "Завдання" всі замітки відсортовані за датою, і користувач може побачити свої плани не лише на сьогодні, а й на найближчі дні. Також є можливість відзначати завдання як виконані, натиснувши на порожнє коло біля них. При створенні нотаток і нагадувань користувач може використовувати кольорові мітки та позначки для користувачів.

Основні функції Any.do легкі у використанні, але є також платна підписка, яка надає додаткові можливості, такі як повторювані завдання, кольорові мітки та нотатки, геолокаційні нагадування та інші [17].

1.3.4 Google Календар: зручне та ефективне керування часом

Google Календар є першим і найпопулярнішим сервісом для організації зустрічей та подій. Випущений у бета-версії в 2006 році і офіційно представлений в 2009 році, цей застосунок доступний для платформ iOS і macOS. За допомогою Google Календара можна встановлювати час зустрічей, створювати повторювані події, встановлювати нагадування та надсилати запрошення іншим учасникам за допомогою електронної пошти [18].

Інтерфейс календаря Google відповідає стилю Material Design (рис. 1.7), зробленому зрозумілим для користувача завдяки чіткій розстановці проміжків і використанню кольорових акцентів. Дати і дні тижня виділяються збільшеним розміром, щоб забезпечити більшу читабельність. Панель інструментів, яка дозволяє змінювати режим перегляду та керувати календарем, розташована у верхній частині, а рядок пошуку доступний через піктограму.

Для відображення та керування кількома календарями на одній сторінці у вигляді стовпців, користувач може вибрати відповідний тип відображення дат. Залежно від відповіді користувача на запрошення на подію, стан відображається різними способами: "Так" позначається одним кольором, "Можливо" – похилою лінією, а "Ні" – пунктирною лінією. Користувач також може отримувати нагадування про події та дії за допомогою push-повідомлень або електронної пошти. Календар підключається до облікового запису Gmail для забезпечення зручності використання.



Рисунок 1.7 – Інтерфейс календаря Google

Переваги Google Calendar [19]:

- 1) Google Calendar є популярним інструментом для організації часу та календарних подій. Ось кілька переваг, які надає Google Calendar:
- 2) Зручний доступ і синхронізація: Google Calendar доступний на різних платформах, включаючи веб-версію, мобільні пристрої з операційною системою Android та iOS. Календар можна синхронізувати з іншими пристроями і застосунками, що дозволяє з легкістю організувати свій розклад незалежно від пристрою.
- 3) Множина календарів і подій: Google Calendar дозволяє створювати необмежену кількість календарів, що дозволяє розділяти різні типи подій і організувати їх у відповідних групах. Наприклад, можна створити окремий календар для особистих подій, робочих зустрічей, важливих дат тощо.
- 4) Доступність інтеграцій: Google Calendar інтегрується з іншими популярними сервісами Google, такими як Gmail, Google Meet, Google Tasks та іншими. Це дозволяє зручно планувати зустрічі, надсилати запрошення, створювати завдання і зберігати всю релевантну інформацію в одному місці.

- 5) Нагадування і повідомлення: Google Calendar надає можливість встановлювати нагадування для подій, що допомагає не забувати про заплановані зустрічі та завдання. Також можна налаштувати повідомлення електронною поштою або сповіщення на мобільний пристрій, щоб завжди бути в курсі найближчих подій.
- 6) Загальний доступ до календаря: Google Calendar дозволяє ділитися своїм календарем з іншими людьми. Це зручно для спільного планування подій, співпраці в робочих групах або просто для поділу розкладу з родиною та друзями.

РОЗДІЛ 2 ОГЛЯД ОСНОВНИХ ПІДХОДІВ ДО ДИЗАЙНУ ІНТЕРФЕЙСІВ ЕЛЕКТРОННИХ ЩОДЕННИКІВ

а. 2.1 Фактори, які впливають на дизайн інтерфейсів електронних щоденників

Ефективне управління часом – актуальне завдання для будь-якої особи, незалежно від її професії. Ця сфера, також відома як тайм-менеджмент, отримала значну увагу у формі книг та курсів, які надають різні методики планування часу. Використання програмного забезпечення в цій галузі не обмежується лише встановленням розкладу, але й забезпечує зберігання інформації. Людям часто важко запам'ятати або впорядкувати всю необхідну інформацію, яка повинна бути доступна у структурованому вигляді, тому ефективне управління часом має велике значення [20].

Розробники вкладають значні зусилля і детально вивчають кожен аспект, щоб відповісти потребам користувачів. Вони займаються дослідженням користувацького досвіду, враховуючи емоції, переконання, вподобання, фізіологічні та психологічні реакції, поведінку та досягнення, які виникають перед, під час і після використання системи.

Дизайн інтерфейсу користувача є процесом, за допомогою якого дизайнери створюють інтерфейси у програмному забезпеченні та комп'ютеризованих пристроях, зосереджуючись на зовнішньому вигляді та стилі [21]. Дизайнери прагнуть створити інтерфейси, які користувачі вважають легкими у використанні та приємними. Крім того, інтерфейс повинен додавати цінності бренду та підвищувати довіру користувачів. Гарний дизайн є емоційним дизайном. Користувачі асоціюють позитивні враження з брендами, які їм подобаються на всіх рівнях і зберігають магію приємного та безпроблемного досвіду. Вимоги, настрої та життєві цілі цільової аудиторії визначають особливості дизайну інтерфейсу та його ергономіку. Аналіз ринку та конкурентних продуктів дозволяє виявити основні недоліки та помилки у розробці, що сприяє створенню нових якісних продуктів.

Інтерфейс забезпечує комфортну взаємодію між користувачем і пристроєм. Під час розробки інтерфейсу програми, розробники повинні враховувати ергономіку та пристрої, з якими програма взаємодіє. Користувачі повинні бути основним критерієм побудови інтерфейсу та створення дизайну застосунків.

Щоб створити чудовий інтерфейс, слід дотримуватися таких рекомендацій [22]:

- Забезпечення легкості навігації. Найкращим варіантом є не ускладнювати навігацію програми великим меню, але варто уникати повного його відсутності. Блоки навігації не повинні перекривати основні області програми. Наприклад, можна використовувати "гамбургер-меню" або зробити меню прихованим в лівій шторці програми. Дизайн інтерфейсу мобільних застосунків має бути максимально продуманим і простим, щоб користувач зміг без зусиль досягти своїх цілей.
- Використання інтуїтивно зрозумілих піктограм. Найкраще використовувати піктограми, які користувачі можуть легко розпізнати, наприклад, ті, що використовуються в більшості програм. Це дозволить користувачам швидко і ефективно орієнтуватися в програмі. Піктограми повинні бути чіткими, з чітко виділеними елементами, наприклад, з тінями кнопок (рис. 2.1).



Рисунок 2.1 – Стандартні іконки для платформи IOS

Наведені рекомендації допоможуть створити зручний та привабливий інтерфейс, який задовольняє потреби користувачів:

1. Простий фон. Дизайн застосунків для iOS створює основу для тенденції мінімалізму та спрощення. У той час як веб-дизайнери додають красиві теми та фони під час створення веб-сайтів, мобільні дизайнери роблять фонові зображення простими або розмитими, щоб відобразити навігацію та функціональність програми більш виразно. Це також дозволяє скоротити час завантаження та зробити взаємодію з користувачами більш комфортною [23];
2. Інтерфейс має бути простим, містити лише ті елементи, які служать меті користувача.
3. Не менш важливою є система ієрархії та розбірливості. Правильне вирівнювання – запорука успіху і зазвичай вибирають вирівнювання по краю. Важливо також звернути увагу на ключові характеристики, що використовуються: колір, яскравість і контраст. Варто уникати надмірного використання кольорів або кнопок; використовувати для тексту розмір шрифту, напівжирний шрифт/щільність, курсив, великі літери та міжлітерні інтервали.
4. Необхідно звести до мінімуму кількість дій для виконання завдання, але зосередитись на одній основній функції на сторінці, а також направляти користувача, вказуючи необхідну дію, використовуючи прогресивне розкриття інформації для спрощення складних завдань.
5. Хорошою практикою є розміщення елементів керування поруч із об'єктом, яким користувач хоче керувати. Наприклад, кнопка відправлення форми повинна бути поруч з формою.
6. Використання зворотного зв'язок для інформування користувачів про відповіді/дії системи.
7. Використання відповідних шаблонів дизайну інтерфейсу користувача, для підтримки сталості бренду;

8. Забезпеченні кроків, які користувачі можуть зробити природно, незалежно від їхнього контексту.

Ескізи та креслення дозволяють визначити, чи є UX-дизайн розробленого інтерфейсу ергономічним. В даному випадком розробникам надважливо зрозуміти, як поводить ся людина, коли вперше відкриває програму, які колірні схеми їй більше до вподоби, яке розташування елементів дизайну інтерфейсу буде для нього найбільш оптимальним.

Основні вимоги до інтерфейсу допомагають забезпечити зручність використання, ефективність та задоволення користувачів [24]. Ось кілька причин, чому важливо розуміти ці вимоги:

- Задоволення потреб користувачів: Розуміння вимог до інтерфейсу дозволяє розробникам зосередитись на тому, як зробити застосунок зручним та привабливим для цільової аудиторії. Врахування потреб користувачів щодо навігації, взаємодії з елементами інтерфейсу, доступності та зручності використання допомагає побудувати приємний та задовільний досвід користувача.
- Ефективність та продуктивність: Добре спроектований інтерфейс допомагає користувачам швидко знаходити потрібну інформацію, виконувати завдання та отримувати результати. Ефективна організація елементів інтерфейсу, зрозумілість інтеракцій та простота використання сприяють підвищенню продуктивності користувачів.
- Збільшення залученості та ретенції користувачів: Перші враження від інтерфейсу мобільного застосунку мають велике значення для залучення та утримання користувачів. Зрозумілий та привабливий дизайн, легка навігація та приємний досвід користування можуть збільшити зацікавленість і відданість користувачів вашому застосунку.
- Сумісність з платформою: Основні вимоги до інтерфейсу також включають дотримання стандартів та рекомендацій конкретної мобільної платформи (наприклад, Android чи iOS). Відповідність до принципів

дизайну платформи сприяє інтеграції з екосистемою та поліпшує взаємодію з іншими застосунками та сервісами.

- Зменшення помилок та підтримка: Якщо інтерфейс застосунку є логічним, зрозумілим та інтуїтивно зрозумілим, це допомагає зменшити кількість помилок, збитків та підтримки, необхідної для користувачів. Чим більше користувачі розуміють та легко використовують інтерфейс, тим менше ймовірність неправильного введення даних або незадоволення від використання застосунку.

Для успішного дизайну та використання мобільного застосунку важливо розуміти основні вимоги до його інтерфейсу, що відповідають потребам та очікуванням користувачів. Розробка інтерфейсу мобільного застосунку має свої особливості, які потрібно враховувати для створення ефективного та зручного дизайну. Однією з головних особливостей є обмежений екран мобільного пристрою, тому важливо максимально ефективно використовувати доступний простір. Мінімалістичний дизайн, компактні елементи та легка навігація є ключовими факторами для комфортного користувацького досвіду.

Мобільні пристрої забезпечують взаємодію за допомогою дотику, тому розробка інтерфейсу повинна враховувати цей аспект, забезпечуючи легку натискання, прокручування та переміщення елементів за допомогою пальців. Розміщення кнопок та інших інтерактивних елементів повинно бути зручним для використання однією рукою. Оскільки мобільні пристрої мають різні розміри екранів, інтерфейс мобільного застосунку повинен бути адаптивним, щоб забезпечити оптимальне відображення та навігацію на різних пристроях.

Орієнтація екрану (вертикальна або горизонтальна) також має вплив на дизайн інтерфейсу мобільного застосунку. Інтерфейс повинен добре виглядати та бути функціональним незалежно від обраної орієнтації. Крім того, розробка інтерфейсу мобільного застосунку повинна враховувати контекст використання, такий як рух або використання однією рукою, та забезпечувати відповідний контраст та зручність використання в різних умовах.

Для створення добре пристосованого інтерфейсу мобільного застосунку необхідно ретельно розглянути ці особливості, проводити тестування та отримувати зворотний зв'язок від користувачів. Інтерфейс повинен бути інтуїтивно зрозумілим, зручним та ефективним, щоб забезпечити приємний користувацький досвід на мобільних пристроях.

Окрім цього, при розробці мобільних застосунків важливо враховувати вимоги та рекомендації магазинів застосунків, таких як Google Play та App Store. Дотримання цих вимог полегшує процес розміщення застосунків у магазинах та забезпечує впізнаваність для користувачів [25].

Розробка інтерфейсу мобільного застосунку вимагає ретельного розгляду цих особливостей, а також тестування та отримання зворотного зв'язку від користувачів. Інтерфейс повинен бути інтуїтивно зрозумілим, зручним та ефективним, щоб забезпечити приємний користувацький досвід на мобільних пристроях.

У App Store користувачі iOS найбільше цінують оригінальність рішень UX та UI [26]. Якщо мобільні застосунки розроблені відповідно до рекомендацій, App Store полегшує роботу, і користувачі раді бачити щось знайоме та давно улюблене. Тому необхідно орієнтуватися в загальних вимогах і не забувати стежити за оновленнями.

Скелет екрана програми – це загальна структура та компоненти, які визначають основний макет та розміщення елементів на екрані програми. Скелет екрана програми складається з рядка стану, панелі навігації, панелі вкладок, модальних/спливаючих вікон/сповіщень, панелі пошуку, вибору.

Рядок стану містить основну інформацію про систему, таку як поточний оператор, час, стан батареї тощо. Він візуально пов'язаний із панеллю навігації та використовує ту саму фонову заливку. Для відповідності стилю програми, що розробляється, є два варіанти кольору вмісту рядка стану – білий і чорний. Незалежно від пристрою, що використовується, висота рядка стану завжди

становить 20 pt. Єдиним винятком є iPhone X, де рядок стану має 44 pt через «поле» [27].

На панелі навігації зберігаються основні елементи навігації програми або блок керування для певного екрана. Він завжди розташовується у верхній частині нижче лінії стану і інтегрується з лінією стану (рис. 2.2).

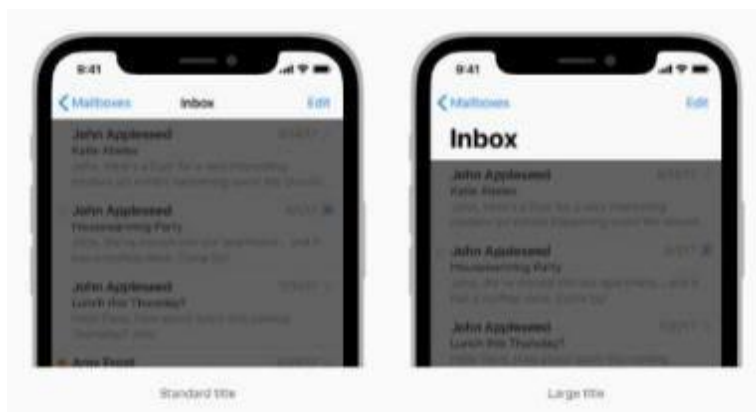


Рисунок 2.2 – Рядок стану та панель навігації

Висота панелі без рядка стану на будь-якому пристрої (крім iPhone X) становить 44 pt. В iOS 10+ за бажанням панель навігації можна розгорнути, тобто створити додаткову область висотою 52 точки з великим заголовком рядка. Не варто забувати про відступи – в iOS вони повинні бути кратними восьми. В основній частині панелі навігації зазвичай встановлюють відступ 8 pt, а в додатковій частині з заголовком – 16 pt.

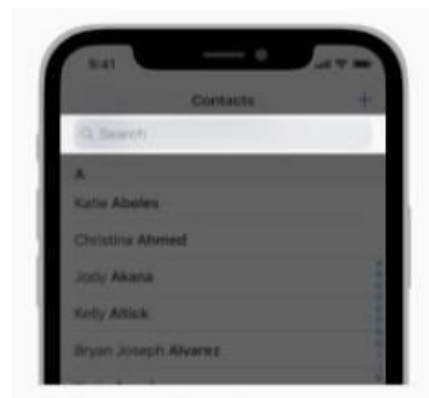
Панель вкладок використовується лише для швидкого доступу до головного екрана програми. Файли карток з'являються в нижній частині екрана програми, що дозволяє швидко перемикатися між різними частинами програми. Панелі вкладок є напівпрозорими, можуть мати фонову тінь, підтримувати однакову висоту в усіх орієнтаціях екрана та приховуватись, коли відображається клавіатура. Файл може містити будь-яку кількість вкладок, але кількість видимих залежить від розміру та орієнтації пристрою. Якщо деякі вкладки неможливо відобразити через обмежений горизонтальний простір, остання видима стає вкладкою «Додатково», яка відображає інші вкладки у списку на окремому екрані.

Висота вкладки становить 49 пунктів на всіх моделях iPhone. Піктограми можуть мати заголовок або без нього, але вміст розташовується по центру в будь-якому випадку. Розмір значка становить 24 пт. Слід враховувати, що



область натискання становить 44 pt – інші іконки в зоні натискання розміщувати не рекомендується. Щоб знати, як розмістити значки на панелі вкладок, потрібно визначити їхню кількість та розділити ширину екрана на кількість значків/вкладок. Потім кожен значок центрується відповідно до ширини та висоти області мітки, незалежно від того, чи є текст під знаком (рис. 2.3).

За замовчуванням рядок пошуку відображається у двох різних стилях: видимому та мінімальному. Обидві версії мають однаковий функціонал. Видиму версію важко не помітити, вона гарно виділяється суцільною заливкою, зазвичай білою. Текст-заповнювач відображається всередині панелі, коли користувач не вводить текст. Після введення критерію пошуку він зникає, а праворуч з'являється кнопка видалення введеного (рис. 2.4).



Селектор використовується для вибору значення зі списку. Цікаво, що в одному інструменті вибору можна використовувати кілька типів списків. Хорошим прикладом його застосування є вибір дати народження, де розробники використовують вибір дня, місяця та року в одному вікні (рис. 2.5).



Рисунок 2.5 – Селектор

Розробники платформи IOS приділили особливу увагу вибору шрифтів, написавши в рекомендаціях, що необхідно використовувати власний шрифт «San Francisco» (табл. 2.1). Він представлений двома зображеннями: «SF UI Display» і «SF UI Text. Display». Перший здебільшого використовується для стандартних елементів інтерфейсу, «текст» – для заголовків і основного тексту. Текст прийнято робити з невеликим зміщенням [28].

Таблиця 2.1 – Розміри шрифтів основних елементів інтерфейсу

Елемент	Розмір	Накреслення	Тип шрифту
Large Nav Bar Title	34	bold	Текст
Nav bar title	17	Semibold	Текст
Nav bar button	17	Regular	Текст
Search bar	17	Regular	Текст
Tab bar button	10	Regular	Текст
Action sheets	20	Regular/Semibold	Дисплей

Система координат, яку використовує iOS для розміщення вмісту на екрані, базується на точкових вимірюваннях, виражених у пікселях на дисплеї. Дисплей стандартної чіткості має щільність пікселів 1:1 (або @1x), де один піксель дорівнює одній точці. Дисплеї з високою роздільною здатністю мають вищу щільність пікселів і коефіцієнти масштабування 2,0 або 3,0 (позначаються @2x і @3x). Тому для дисплея з високою роздільною здатністю необхідне зображення з великою кількістю пікселів (рис. 2.6).

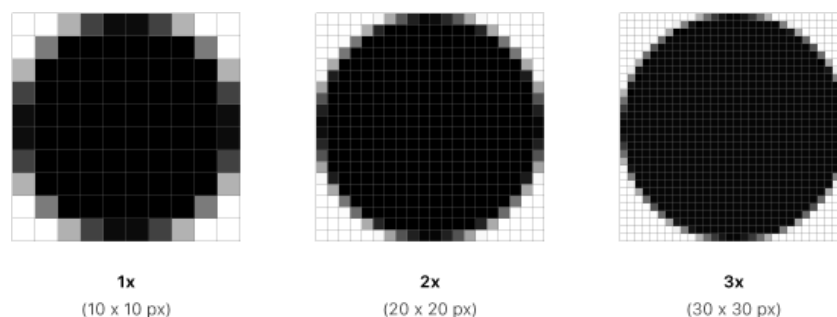


Рисунок 2.6 – Три кругові графіки з різною роздільною здатністю та кількістю пікселів

Коло ліворуч – це стандартна роздільна здатність із масштабним коефіцієнтом 1, тобто 10x10 пікселів. Середнє коло має високу роздільну здатність із масштабним коефіцієнтом 2 і розміром 20x20 пікселів. Коло праворуч має високу роздільну здатність із масштабним коефіцієнтом 3 і розміром 30x30 пікселів.

Звичайно, в сучасних реаліях недостатньо вимагати від таких мобільних застосунків традиційних функцій: наявності календарів, адресних книг, блокнотів. Не менш важливим аспектом є мобільність, можливість синхронізації з портативними пристроями.

Типова програма-органайзер є інструментом, призначеним для допомоги в організації та керуванні різними аспектами життя та роботи користувача. Основна мета органайзера полягає в управлінні завданнями, календарем, контактами, нагадуваннями та іншими важливими елементами. Вона надає зручний і централізований спосіб організації та доступу до цих інформаційних ресурсів [29]. Ось деякі основні функції та можливості типової програми-органайзера:

- Календар: В програмі-органайзері користувач може створювати події, записувати зустрічі, встановлювати нагадування та організувати свій графік роботи. Календар дозволяє планувати та відстежувати події на різні дати та часу.
- Завдання і списки: Користувач може створювати завдання, призначати їм пріоритети, встановлювати терміни виконання та стежити за їх

виконанням. Органайзер допомагає керувати списками завдань, організувати їх за категоріями та статусом.

- **Контакти:** В органайзері можна зберігати контакти людей, включаючи їх імена, номери телефонів, адреси електронної пошти та іншу інформацію. Це дозволяє зручно керувати та швидко знаходити необхідні контакти.
- **Нагадування:** Програма-органайзер може надсилати нагадування користувачеві про важливі події, завдання, зустрічі або інші дедлайни. Це допомагає не пропустити важливі моменти та пам'ятати про них.
- **Нотатки:** Користувач може створювати нотатки або записи для збереження важливої інформації, ідей, списків та інших деталей. Органайзер дозволяє створювати структуровані нотатки та зручно організувати їх.
- **Синхронізація та доступ з різних пристроїв:** Багато програм-органайзерів підтримують синхронізацію даних між різними пристроями, такими як комп'ютери, смартфони, планшети тощо. Це дозволяє користувачу мати доступ до своїх даних з будь-якого пристрою та в будь-який час.
- **Різноманітність інструментів:** Органайзери можуть мати інші корисні функції, такі як калькулятор, годинник, галерея, файлообмінник та інші інструменти, що полегшують організацію та роботу користувача.

в.

с. 2.2 Огляд можливих систем керування базами даних

Розробка електронного щоденника вимагає вибору відповідної системи керування базами даних (СКБД), яка буде забезпечувати надійне та ефективне зберігання даних. Існує кілька можливих варіантів СКБД, які можуть бути використані для електронного щоденника залежно від конкретних потреб та вимог проекту.

Одним з найпопулярніших варіантів є MySQL, яка є широко використовуваною та має велику спільноту користувачів. Вона пропонує швидкодію та підтримку мови SQL, що сприяє зручному управлінню та

запитам до бази даних. MySQL є надійним рішенням для зберігання даних щоденника [30].

Інший варіант – PostgreSQL, відкрита система керування базами даних з розширеними можливостями та підтримкою для різних типів даних [31]. PostgreSQL володіє потужним функціоналом та активно розвивається спільнотою розробників. Вона може бути варіантом для проектів, які потребують більш гнучкого та розширеного функціоналу бази даних.

Для простих застосунків з невеликим обсягом даних може бути варто розглянути використання SQLite [32]. Вона є легкою та вбудовуваною СКБД, яка не вимагає окремого сервера та зберігає базу даних у вигляді файлу. SQLite є ефективним вибором для електронних щоденників з обмеженими потребами в розширених функціях та масштабуванні.

Крім того, Microsoft SQL Server може бути зручним варіантом для користувачів, які працюють у середовищі Microsoft та мають доступ до SQL Server. Вона пропонує широкий функціонал, високу продуктивність та інтеграцію з іншими продуктами Microsoft.

Нарешті, MongoDB може бути цікавим вибором, якщо планується зберігання даних щоденника у вигляді документів. Вона є документоорієнтованою СКБД, яка надає гнучкість та здатність працювати з різними типами даних [33].

Бібліотека Volley є інструментом розробки для мережевого взаємодії в Android-додатках. Вона надає зручні інструменти для виконання HTTP-запитів, обробки відповідей сервера і кешування даних. Програма не використовує локальну базу даних, оскільки немає необхідності зберігати дані на самому пристрої. Замість цього програма отримує всі дані з бази даних сервера через реалізацію протоколу передачі даних HTTP та інтерфейсу API на веб-сервері. Ця взаємодія відома як архітектура клієнт-сервер (рис. 2.7). Бібліотека Volley була обрана для роботи через її безліч переваг, головне, що бібліотека проста в розумінні та використанні [34].

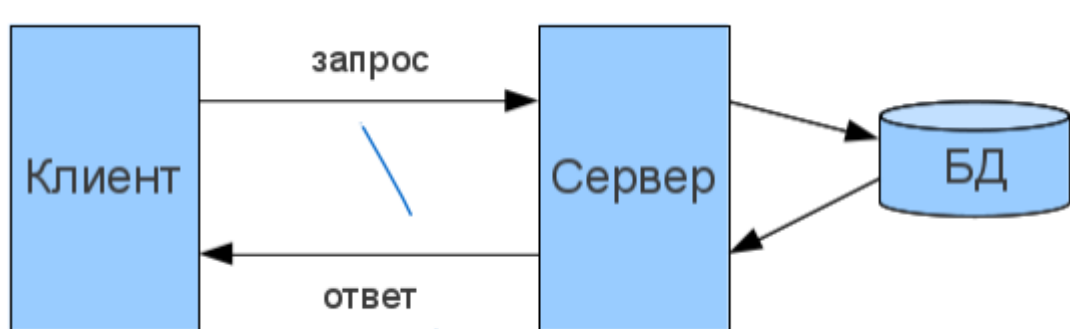


Рисунок 2.7 – Архітектура мережі «Клієнт-Сервер»

Сервер постійно очікує запитів від клієнтів і завжди готовий відповісти дописом або отримати запит. Для відправки запитів на сервери в Android використовуються різні бібліотеки, такі як Retrofit, OkHttp, Volley.

Зазвичай в Android код проблеми з мережею розміщує AsyncTask, тому розробнику варто подбати про відображення ProgressBar у функціях doInBackground() і onPostExecute(). Але завдяки Volley їм більше не потрібно думати про такі процедури. За документацією Volley бібліотека має безліч функцій:

1. Автоматичне планування мережевих запитів.
2. Кілька одночасних підключень до мережі.
3. Volley надає засоби налагодження.
4. Пріоритет запиту на підтримку.
5. Просте налаштування, наприклад повторна спроба та відкат.

Щоб надсилати запити за допомогою бібліотеки Volley, потрібно використовувати 2 основні класи: Request і Request queue. Клас Request містить деталі, необхідні для створення запиту, такі як тип запиту, URL-адреса, вибір даних запиту.

Вибір СКБД для електронного щоденника залежить від різних факторів, таких як потреби проекту, обсяг даних, доступність ресурсів та знання команди розробників. Незалежно від обраної СКБД, важливо ретельно спланувати та

налагодити структуру бази даних щоденника для забезпечення ефективної та надійної роботи застосунку.

d. 2.3 Аналіз інструментів розробки програмного забезпечення

У процесі планування розробки мобільного застосунку було вирішено, що проект буде орієнтований на одну з найпоширеніших і популярних операційних систем для мобільних пристроїв – Android [35]. Операційна система має багато незаперечних переваг перед конкурентами, такими як IOS, Symbian і BlackBerry, найбільш очевидними перевагами є [36]:

1. Операційна система Android є найпопулярнішою операційною системою у світі.
2. Великий вибір пристроїв: Android працює на широкому спектрі пристроїв, включаючи смартфони, планшети, смарт-годинники, телевізори, автомобільні системи і багато іншого. Це дає користувачам можливість вибирати пристрій, який найкраще відповідає їхнім потребам і бюджету.
3. Відкрита платформа: Android є відкритою платформою, що означає, що розробники можуть створювати та впроваджувати свої власні застосунки, модифікувати операційну систему та використовувати різноманітні налаштування. Це сприяє інноваціям, розширенню функціоналу та великому вибору застосунків у Google Play Store.
4. Налаштування та персоналізація: Android надає користувачам багато налаштувань та можливостей персоналізації. Ви можете змінювати теми, шрифти, розташування елементів на екрані, додавати віджети та налаштовувати різні параметри за своїм смаком.
5. Широкий вибір застосунків: Google Play Store, офіційний магазин застосунків для Android, має одну з найбільших кількостей застосунків серед усіх мобільних платформ. Ви зможете знайти тисячі

безкоштовних та платних застосунків для різних потреб, включаючи соціальні мережі, ігри, продуктивність, фотографію, навігацію та багато іншого.

6. Інтеграція з Google-сервісами: Android має глибоку інтеграцію з сервісами Google, такими як Gmail, Google Drive, Google Календар та інші. Це дозволяє легко синхронізувати дані із вашими обліковими записами Google та отримувати доступ до них з різних пристроїв.
7. Багатофункціональність: Android пропонує багато функцій, що полегшують використання мобільних пристроїв. Серед них можна виділити сповіщення на екрані блокування, розширені можливості мультимедіа, налаштування швидкого доступу та багато інших.

Одним із найважливіших елементів розробки будь-якої програми є вибір інтегрованого середовища розробки (IDE). Для створення застосунків на платформі Android підійдуть такі середовища розробки як Eclipse, NetBeans, Android studio. Для роботи було обрано останній варіант, який з 2013 року доступний безкоштовно (рис. 2.10). Android Studio – це універсальна IDE Google, яка дозволяє створювати та тестувати програми, які працюють на смартфонах, планшетах, ноутбуках і годинниках під керуванням операційної системи Android [37].

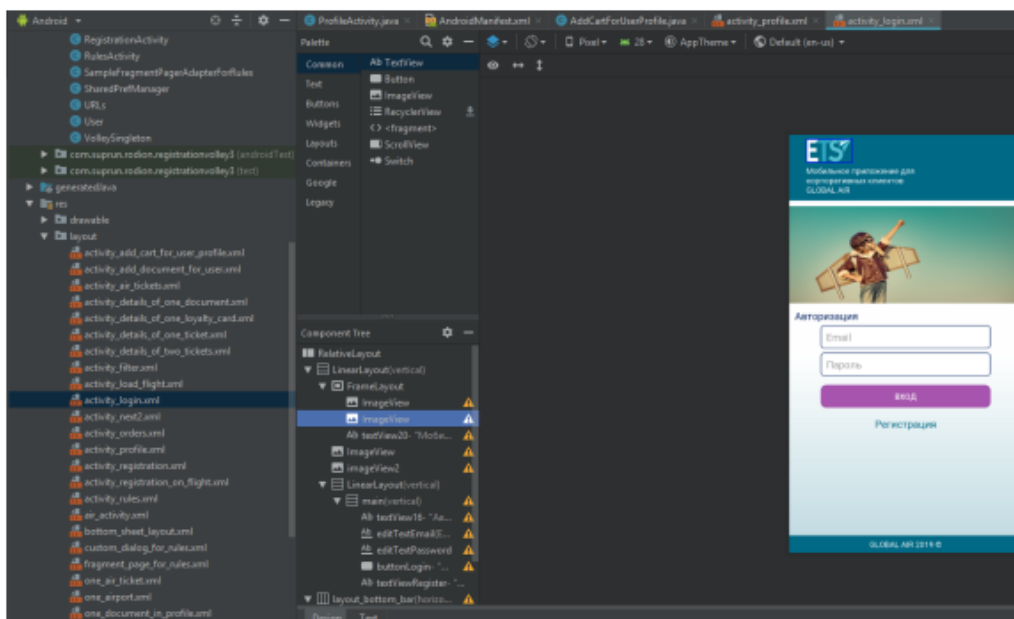


Рисунок 2.10 – Android Studio

Основні особливості Android Studio [38]:

- Розширені інструменти для розробки: Android Studio надає багатий набір інструментів для розробки Android-додатків, включаючи редактор коду, візуальний редактор макетів, дизайнер інтерфейсу користувача, налагоджувач, аналізатор коду та інші. Ці інструменти допомагають розробникам швидше і ефективніше створювати застосунки.
- Інтеграція з Android SDK та Google сервісами: Android Studio повністю підтримує Android SDK та надає зручний інтерфейс для управління компонентами SDK, такими як SDK-платформи, інструменти збірки, бібліотеки та інше. Вона також має вбудовану підтримку для Google Play сервісів та багатьох інших сервісів Google, що спрощує інтеграцію з ними в застосунки.
- Підтримка мов програмування: Android Studio підтримує різні мови програмування для розробки Android-додатків, включаючи Java та Kotlin. Ви можете вибрати мову, яка найбільше вам підходить або навіть використовувати обидві мови в одному проекті.
- Плагіни та розширення: Android Studio має велику кількість плагінів та розширень, які дозволяють розширити його функціональність. Ви можете використовувати плагіни для автоматизації завдань, покращення продуктивності, інтеграції з іншими інструментами розробки та багатьох інших потреб.
- Емулятори та реальні пристрої: Android Studio надає можливість запускати та тестувати застосунки на емуляторах Android, що дозволяє перевіряти роботу застосунків на різних версіях Android та різних типах пристроїв (рис. 2.11). Крім того, ви можете підключати реальні пристрої для тестування та налагодження застосунків.



Рисунок 2.11 – Емулятор у Android Studio

Емулятор Android використовується як цільова платформа для запуску та тестування мобільних застосунків. Емулятор має зручну унікальну функцію – перегляд показників продуктивності під час запуску програми (рис. 2.12).

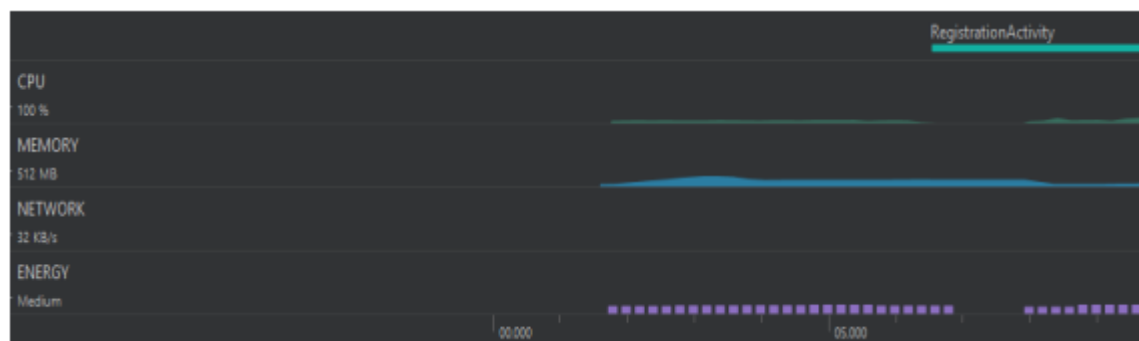


Рисунок 2.12 – Показники продуктивності в Android Studio

Java є однією з основних мов програмування, яку використовують розробники Android-додатків у Android Studio [39]. Ось деякі переваги використання мови програмування Java для розробки Android-додатків:

Підтримка Android SDK: Java має повну підтримку Android SDK та API-інтерфейсів, що дозволяє розробникам легко використовувати всі можливості платформи Android для створення функціональних застосунків.

Велике співтовариство розробників: Java є однією з найпоширеніших мов програмування, і має велике співтовариство розробників, які активно допомагають одне одному. Це означає, що ви можете знайти велику кількість ресурсів, документацію, уроків та підтримку спільноти для розв'язання будь-яких питань, що виникають під час розробки.

Об'єктно-орієнтована природа: Java є мовою програмування з об'єктно-орієнтованою природою, що дозволяє розробникам створювати чистий, модульний і легко зрозумілий код. Це полегшує розробку, тестування та підтримку застосунків.

Багатоплатформовість: Одна з переваг використання мови програмування Java полягає в тому, що вона є багатоплатформовою. Це означає, що ви можете використовувати Java для розробки не тільки Android-додатків, але й застосунків для інших платформ, таких як настільні комп'ютери, веб-програми та серверні застосунки.

Розширюваність: Java має велику кількість бібліотек та фреймворків, які дозволяють розробникам швидко розширювати функціональність своїх застосунків. З використанням цих бібліотек і фреймворків ви можете ефективно використовувати готові рішення для таких завдань, як робота з мережею, базами даних, графікою та багатьма іншими.

е. 2.4 Технічні вимоги

- Вимоги до структурно-функціонального розділу

Програми Android використовують у своїй роботі вікна (схожі на Windows), однак у цій системі зазначені вікна мають іншу назву – Activity (Активність). При розробці програми необхідно максимально оптимізувати код

і кількість дій, тому що велика кількість дій може призвести до збоїв у роботі програми та доставляти незручності користувачеві.

Функціональна структура мобільного застосунку повинна включати:

- зручний інтерфейс;
 - завантажувати інформацію з бази даних по протоколу HTTP;
 - прості та зрозумілі механізми взаємодії з застосунком;
 - авторизація та автентифікація користувача;
 - можливість пошуку за допомогою фільтрів характеристик
- Вимоги до інтерфейсу програми

Інтерфейс користувача повинен враховувати такі принципи:

Зручність. Необхідно використовувати усталені та знайомі стандарти для використання програм в операційній системі Android. Кнопки, посилання та поля мають бути помітними та мати такі розміри, щоб користувачі могли їх натискати.

Інформаційна архітектура. Зважаючи на обмеження розмірів, структура елементів відображення повинна бути максимально простою та зручною. Екран повинен мати оптимальну кількість кнопок і позначення шляхів користувача. Дизайн мобільного застосунку має бути зрозумілим. Спеціальні дії. Хорошою практикою є обмеження кількості обов'язкових полів, щоб звільнити користувачів від довгих текстових записів.

Варто надати дані "за замовчуванням", коли це можливо. Колірне рішення повинно бути в м'яких тонах, які приємні оку. Усі сторінки (екрани мобільного застосунку) мають відповідати одному стилю, який базується на власному інтерфейсі Material Design. Під час роботи на пристрої бажаною орієнтацією екрана є книжкова. Мова інтерфейсу – українська.

- Вимоги до технічного забезпечення

Вся інформація в застосунку повинна зберігатися в базі даних і бути доступною багаторазово. Варто враховувати наступні пункти:

- операційна система Android 5.0 і вище;
- процесор – Qualcomm Snapdragon 625 1 ГГц або краще;
- оперативна пам'ять – 1 Гб і вище;
- екран дисплея – починаючи з 3,5 дюймів по діагоналі, починаючи з роздільної здатності 960*640;
- дисковий простір – близько 70 МБ при використанні;
- постійне підключення до Інтернету.

Отже, на сьогоднішній день вибір мобільних операційних систем досить широкий. Однак характер їх розвитку у кожного з них дуже різний. iOS виключається через високий ризик труднощів у розробці та розповсюдженні застосунків, а також високу ціну на публікацію застосунків та отримання інструментів розробки. Windows Mobile було виключено через надзвичайно низький рівень проникнення операційної системи та відсутність підтримки розробників.

Тому як інструмент розробки мобільного застосунку «Електронний щоденник» було обрано Android Studio, яка є новішою платформою на основі Java порівняно з іншими інструментами розробки Android. Технічні вимоги для мобільного застосунку електронного щоденника можуть варіюватися в залежності від конкретних потреб і функціональності, яку ви бажаєте реалізувати в застосунку. Однак, основні технічні вимоги можуть включати наступні аспекти:

- Платформа: Розгляньте, для якої мобільної платформи ви плануєте розробити застосунок – Android, iOS або обидві. Це важливо, оскільки розробка для кожної платформи може мати відмінні вимоги та особливості.
- Операційна система: Специфіку підтримки операційних систем можна визначити, наприклад, для Android - підтримка версій Android 7.0 (Nougat) і вище, а для iOS - підтримка версій iOS 11 і вище.

- **Мінімальні вимоги апаратного забезпечення:** Визначте мінімальні характеристики апаратного забезпечення (наприклад, процесор, оперативна пам'ять, внутрішня пам'ять) для оптимальної продуктивності та функціональності застосунку.
- **Аутентифікація та безпека:** Розгляньте можливі методи аутентифікації користувачів, такі як вхід за допомогою пароля, PIN-коду, відбитків пальців або розпізнавання обличчя. Також важливо врахувати заходи безпеки, такі як шифрування даних, захист від несанкціонованого доступу та захист від атак.
- **База даних:** Виберіть підходящу систему керування базами даних для зберігання та управління даними щоденника, як описано раніше. Врахуйте функції такі як резервне копіювання, синхронізація, оптимізація запитів та швидкодія.
- **Користувацький інтерфейс:** Розробіть зручний та привабливий користувацький інтерфейс, який відповідає потребам користувачів. Використовуйте сучасні елементи дизайну, інтуїтивно зрозумілі інтеракції та підтримку різних розмірів екрану.
- **Синхронізація даних:** Якщо планується синхронізація даних між пристроями або зберігання даних в хмарному сервісі, врахуйте відповідні механізми синхронізації та забезпечення конфіденційності даних.
- **Локалізація:** Якщо ви плануєте випустити застосунок на різних мовах або для різних регіонів, врахуйте можливості локалізації, щоб забезпечити, що інтерфейс і контент будуть доступними та зрозумілими для користувачів.
- **Тестування:** Забезпечте належне тестування застосунку, включаючи функціональне тестування, тестування продуктивності та сумісності з різними пристроями і версіями операційних систем.

РОЗДІЛ 3 ПРОЕКТУВАННЯ МЕТОДИКИ ДЛЯ РОЗРОБКИ ІНТЕРФЕЙСУ ЕЛЕКТРОННОГО ЩОДЕННИКА

f. 3.1 Аналіз потреб користувачів

Після аналізу сучасних підходів до проектування інтерфейсу було сформовано етапи розробки дизайну інтерфейсу електронного щоденника (рис. 3.1):

1. Аналіз користувача. Щоб розробити хороший користувальницький інтерфейс для мобільної програми, необхідно спочатку знати своїх користувачів та зрозуміти, що спонукає їх використовувати застосунок, а потім почати розробку.
2. Створення макету. Warframe – модель інтерфейсу мобільних застосунків майбутнього, яка починається з ескізу на папері.
3. Дизайн інтерфейсу. Зовнішній вигляд є одним з найважливіших етапів створення мобільного застосунку. Зрештою, це перше, на що звертають увагу користувачі, заходячи в програму. Але щоб створити дизайн, який сподобається користувачам, необхідно враховувати наступні параметри: особливості керівництва по платформі, унікальність, зручність інтерфейсу, основні елементи повинні бути масштабними, менше тексту та більше візуалізації, оброблені іконки.
4. Тестування. Після того як дизайн мобільного застосунку готовий, його залишилося протестувати. Це допомагає виявити недоліки та представити досконалість клієнтам

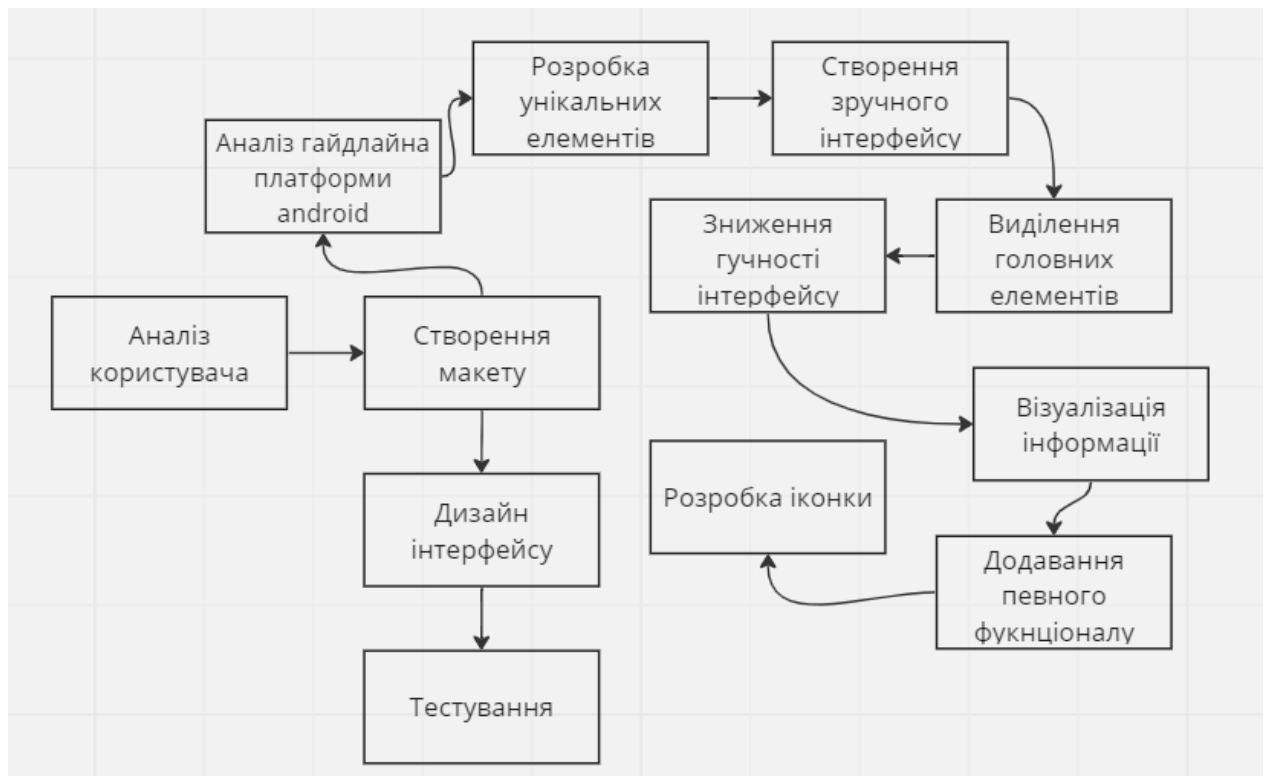


Рисунок 3.1 – Схема дизайну інтерфейсу електронного щоденника

Перш ніж почати створювати прототип програми, потрібно створити чіткі особи користувача [40]. Для цього необхідно:

- 1) Дослідити цільову аудиторію: Ретельно вивчити характеристики та потреби людей, які можуть бути зацікавлені у використанні електронного щоденника. Розглянути такі фактори, як вікова група, професійна діяльність, освіта, інтереси та потреби.
- 2) Провести аналіз конкурентів: Вивчити, які мобільні застосунки для щоденників вже існують на ринку і хто їх цільова аудиторія. Проаналізувати їхні переваги та недоліки, а також спробувати визначити, які групи користувачів вони адресують.
- 3) Виконати інтерв'ю та опитування: Провести співбесіди та опитування з потенційними користувачами електронного щоденника. Задати їм питання про їхні потреби, вимоги та очікування від такого застосунку. Це дозволить отримати важливі відгуки та інсайти, що допоможуть зрозуміти, хто саме є цільовою аудиторією.

- 4) Проаналізувати дані та зробити висновки: Після збору даних з різних джерел варто піддати їх аналізу, виділити спільні риси та особливості у потребах та вимогах користувачів. Визначити ключові групи користувачів, які мають схожі характеристики та вимоги.
- 5) Створити персонажі користувачів: На основі отриманих даних створити персонажів користувачів, які представляють різні групи аудиторії. Кожен персонаж повинен мати ім'я, вікові характеристики, професійні дані, потреби, цілі та інші відповідні деталі, що допоможуть відображати їх як реальних користувачів.

Аналіз показав, що потенційними користувачами мобільного застосунку "електронний щоденник" можуть бути різними категоріями людей, які зацікавлені в організації та веденні щоденникових записів за допомогою мобільного пристрою (рис. 3.2). Основні групи потенційних користувачів можуть

включати:

	<p>Поведінка</p> <ul style="list-style-type: none"> • працює в офісі • має мало вільного часу
<p>Демографія</p> <ul style="list-style-type: none"> • працююча людина • вік 25+ • вища освіта 	<p>Потреби</p> <ul style="list-style-type: none"> • контролювати свій час • мати чіткий розпорядок дня • не забувати про призначені зустрічі

- 1) Студенти: Студенти можуть використовувати мобільний застосунок "електронний щоденник" для ведення записів про свої навчальні завдання, графік занять, терміни здачі робіт, плани на день або тиждень.
- 2) Професіонали: Різні професіонали, такі як менеджери, підприємці, фрілансери та інші, можуть використовувати мобільний застосунок

- "електронний щоденник" для планування своїх робочих завдань, зустрічей, дедлайнів та інших пов'язаних з роботою аспектів.
- 3) Організатори подій: Організатори подій, такі як конференції, семінари, виставки тощо, можуть використовувати мобільний застосунок "електронний щоденник" для планування розкладу подій, спілкування з учасниками, відстеження деталей і оновлень.
 - 4) Люди, які прагнуть особистого розвитку: Люди, які цікавляться особистим розвитком, можуть використовувати мобільний застосунок "електронний щоденник" для ведення записів про свої цілі, досягнення, рефлексії, плани та ідеї.
 - 5) Користувачі, що прагнуть покращити свою продуктивність: Люди, які бажають управляти своїм часом, підвищити ефективність та покращити продуктивність, можуть використовувати мобільний застосунок "електронний щоденник" для планування, організації завдань, відстеження досягнень та управління пріоритетами.

г. 3.2 Створення зручного та привабливого інтерфейсу

Незалежно від докладності технічного завдання, завжди існує ризик пропустити деталі. Для отримання першого уявлення про взаємодію мобільного застосунку з користувачем необхідний прототип. Прототип є динамічною моделлю продукту, що імітує взаємодію користувача з інтерфейсом. Це дозволяє швидко отримати зворотній зв'язок від клієнтів або представників цільової аудиторії. Інтерактивний прототип передбачає намалювання всіх екранів майбутнього мобільного застосунку у спеціальній програмі та встановлення зв'язків між ними.

Прототипи дозволяють протестувати програму на ранньому етапі. Клієнти бачать всі екрани та можуть оцінити кнопки та інші елементи

інтерфейсу. На цій стадії розробник може зрозуміти, чи потребує його сценарій взаємодії з користувачем доопрацювання.

Прототипи дають змогу побачити застосунок у дії та переконатись, наскільки розробники правильно розуміють ідеї та вимоги до проекту. Вони дозволяють оцінити програму з точки зору користувача, визначити, чи потрібно змінювати схему взаємодії програми з користувачем. При виявленні невідповідностей можна уникнути додаткових фінансових і часових затрат.

Figma – це програма для веб-дизайнерів, за допомогою якої можна створювати не тільки прототипи, але й кінцеві інтерфейси для веб-сайтів і застосунків. Процес роботи в програмі інтуїтивно зрозумілий, а можливість спільної одночасної роботи над проектами залучає все більше команд до реалізації прототипів у Figma [41]. Переваги програми Figma:

1. Figma працює повністю в хмарі, що дозволяє безперешкодно співпрацювати між членами команди. Кілька дизайнерів можуть працювати над одним і тим же проектом одночасно, що дозволяє легко обмінюватися дизайнами та надавати зворотний зв'язок. Це виключає необхідність обміну файлами та керування версіями головного болю.
2. Кросплатформенна сумісність: Figma працює на різних операційних системах, включаючи Windows, macOS та Linux. Це дозволяє дизайнерам отримати доступ до своїх проектів з будь-якого пристрою з підключенням до Інтернету, забезпечуючи гнучкість і зручність.
3. Редагування в режимі реального часу: за допомогою Figma кілька користувачів можуть редагувати файл дизайну одночасно, дозволяючи співпрацю в режимі реального часу. Дизайнери можуть бачити курсори один одного, що полегшує спілкування та ефективну роботу разом. Ця функція особливо корисна для віддалених команд або дизайнерів, які працюють в різних місцях.
4. Компоненти дизайну та бібліотеки: Figma дозволяє створювати багаторазові компоненти дизайну та бібліотеки. Компоненти дизайну

можуть бути легко розділені і повторно використані в декількох проектах, забезпечуючи послідовність і прискорення процесу проектування. Зміни, внесені до компонента, поширюються автоматично, заощаджуючи час і зусилля.

5. Прототипування та дизайн взаємодії: Figma пропонує надійні можливості прототипування, що дозволяє дизайнерам створювати інтерактивні та анімовані прототипи. Ви можете визначити взаємодії, переходи і жести, щоб імітувати користувальницький досвід і перевірити функціональність ваших конструкцій.
6. Figma спрощує процес передачі дизайну розробникам. Він надає інструменти для створення специфікацій дизайну, експорту активів та перевірки деталей дизайну, що полегшує розробникам точне розуміння та реалізацію конструкцій.
7. Плагіни та інтеграції: Figma підтримує широкий спектр плагінів та інтеграцій з іншими інструментами та сервісами дизайну. Ці плагіни розширюють функціональність Figma, пропонуючи додаткові функції та інтеграції з популярними системами дизайну, інструментами управління проектами та платформами співпраці.
8. Керування версіями та історія: Figma зберігає детальну історію змін дизайну, що дозволяє переглядати попередні версії та при необхідності повертатися до старих ітерацій. Ця функція забезпечує душевний спокій і допомагає підтримувати послідовність дизайну.
9. Спільнота та ресурси: Figma має процвітаючу спільноту дизайнерів, які діляться ресурсами, шаблонами та підручниками. Цей аспект спільноти надає можливості для навчання, натхнення та співпраці з іншими дизайнерами.

Мінусом цієї програми є те, що Figma працює лише при підключенні до Інтернету.

h. 3.3 Створення дизайну

Дизайн інтерфейсу визначає якість користувацького досвіду та зручність використання. Інтерфейс користувача передає враження про продукт і його функціональність. Простота використання та адаптація до різних платформ є ключовими аспектами у дизайні мобільних застосунків. Застосування нативних компонентів та дотримання рекомендацій щодо людського інтерфейсу Android є важливими елементами успішного проектування.

Взаємодія з користувачем або інтерфейс користувача є важливою частиною створення дизайну. Інтерфейс користувача розповідає нам про досвід роботи користувача з продуктом під час і після роботи. Простота використання або юзабіліті програми забезпечує простоту використання та інтуїтивне розуміння інтерфейсу [42].

Адаптація дизайну до іншої платформи є невід'ємною частиною роботи дизайнера мобільних застосунків. Мета цієї роботи – органічно інтегрувати дизайн у шаблони взаємодії з користувачем. Крім того, адаптація розробки спрощує розробку за рахунок використання нативних компонентів платформи. Щоб належним чином налаштувати свій дизайн, необхідно дотримуватися рекомендацій щодо людського інтерфейсу в Android [43].

Android відрізняється від інших платформ трьома основними ознаками:

2. Чіткість. У всій системі текст змінюється за розміром, піктограми чіткі, декорування вишукане, але доречне, а підвищена увага до функцій керує дизайном. Доречний колір, типографіка, графіка та елементи інтерфейсу тонко підкреслюють важливий вміст і передають інтерактивність.
3. Різниця. Плавний рух і зрозумілий красивий інтерфейс допомагають людям розуміти вміст і взаємодіяти з ним, не конкуруючи з ним. Вміст зазвичай заповнює весь екран, тоді як напівпрозорість і розмиття зазвичай пропонують більше вмісту. Мінімальне використання панелей,

градієнтів і тіней робить інтерфейс легким і просторим, забезпечуючи максимальну цінність.

4. Глибина. Чітка візуальна ієрархія та реалістичний рух передають відчуття ієрархії, яскравості та розуміння. Дотик і видимість покращують занурення та забезпечують доступ до функцій та іншого вмісту без втрати контексту. Переходи створюють відчуття глибини під час навігації вмістом.

Щоб досягти максимального ефекту, слід пам'ятати про наведені нижче принципи, представляючи особу своєї програми. До них відноситься естетична цілісність, що стосується ступеня, до якого зовнішній вигляд і поведінка програми поєднуються з її функціональністю. Послідовна програма реалізує знайомі стандарти та парадигми, використовуючи надані системою елементи інтерфейсу, знайомі значки, стандартні стилі тексту та узгоджену термінологію. Застосунок містить функції та поведінку, які очікують люди. Пряме маніпулювання вмістом екрана також є привабливим і простим для розуміння. Зворотній зв'язок підтверджує дії та показує результати, тримаючи людей в курсі. Люди швидше навчаються, коли віртуальні об'єкти та дії програми є метафорами знайомого досвіду, реального чи цифрового. Метафори добре працюють на Android, оскільки люди фізично взаємодіють з екраном.

В Android люди контролюють усе, а не програми. Застосунки можуть пропонувати дії або попереджати про небезпечні наслідки, але часто застосунок помилково приймає рішення [44].

а. 3.4 Функціональність мобільного застосунку

Встановивши застосунок на смартфон, серед списку з'явиться відповідна іконка із написом Dear Diary (рис.3.3).

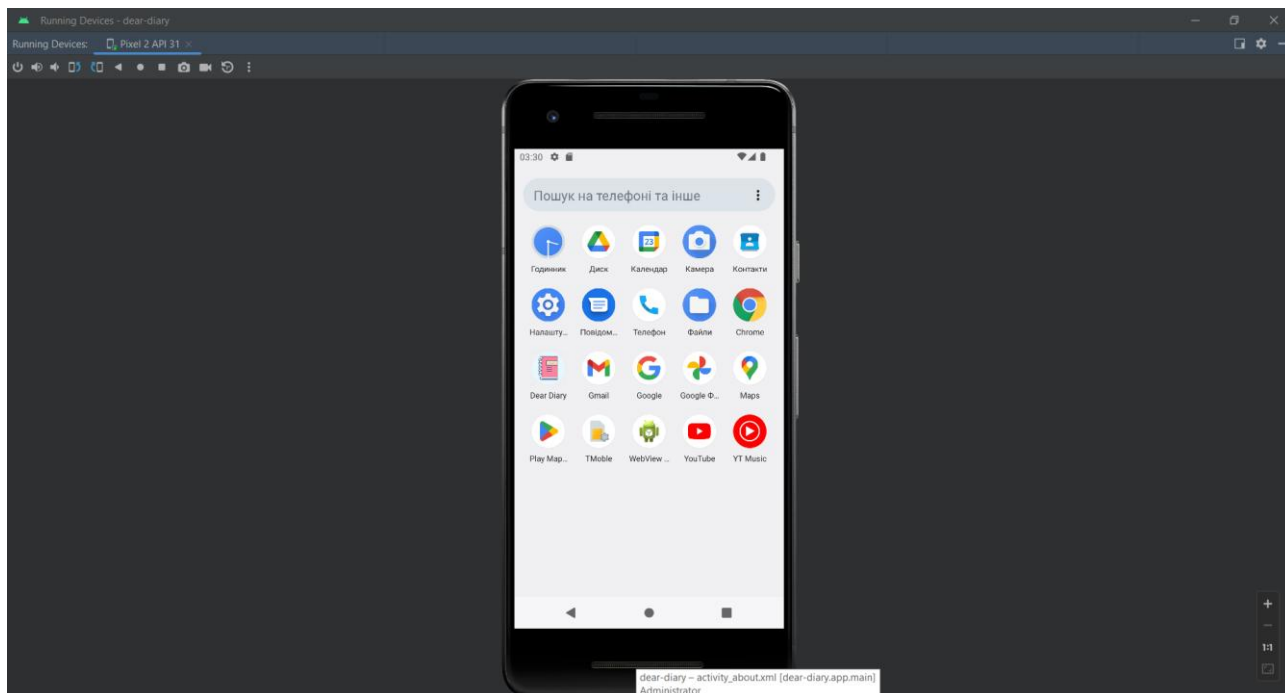


Рисунок 3.3 – Стартове вікно застосунку

Під час натискання на застосунок відображається його назва із плавною анімацією (рис.3.4).

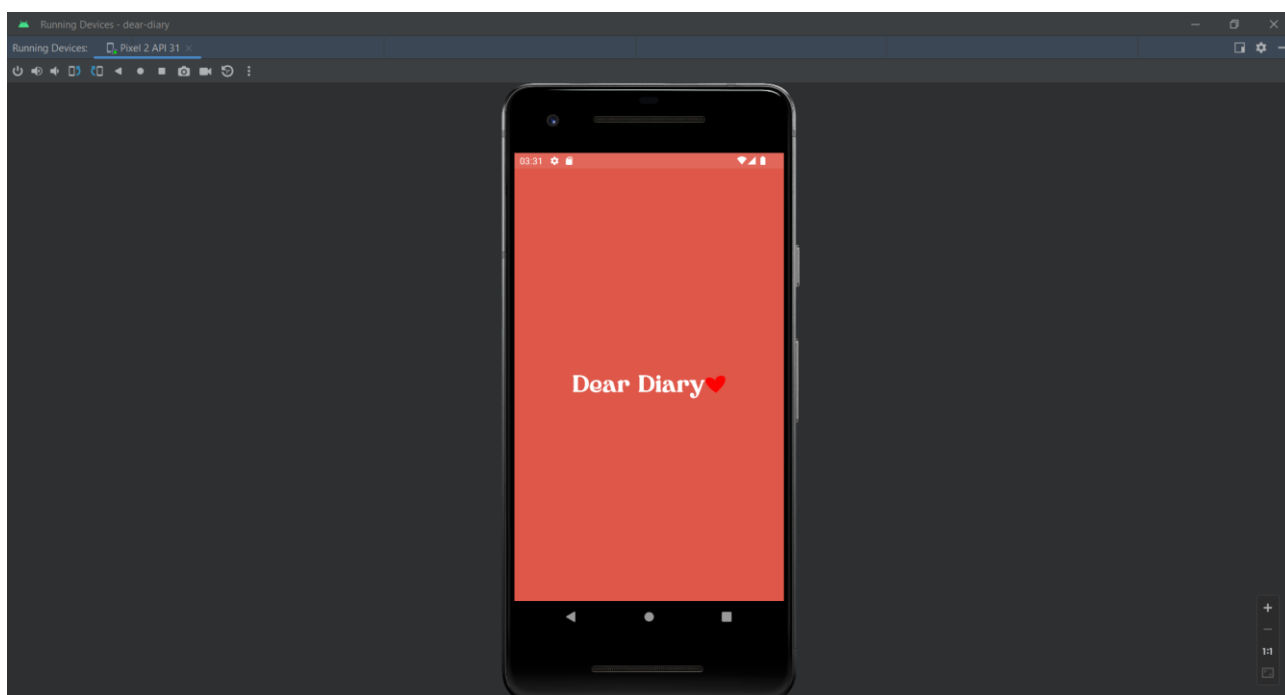


Рисунок 3.4 – Завантаження застосунку

Після успішного завантаження користувача очікує стандартний вигляд застосунку (рис.3.5).

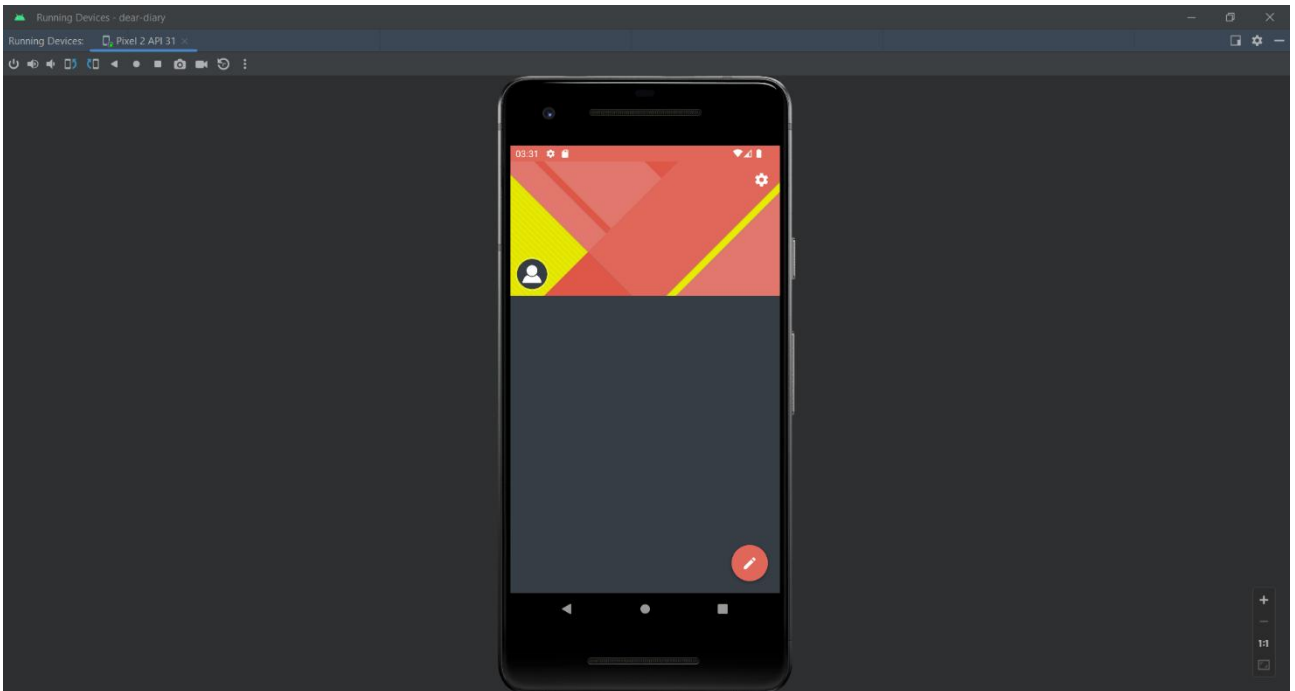


Рисунок 3.5 – Стандартний вигляд застосунку

Ключовим завданням користувача є ведення застосунку як свого особистого щоденнику із додаванням заголовку для власних думок/цілей/планів на майбутнє (рис.3.6).

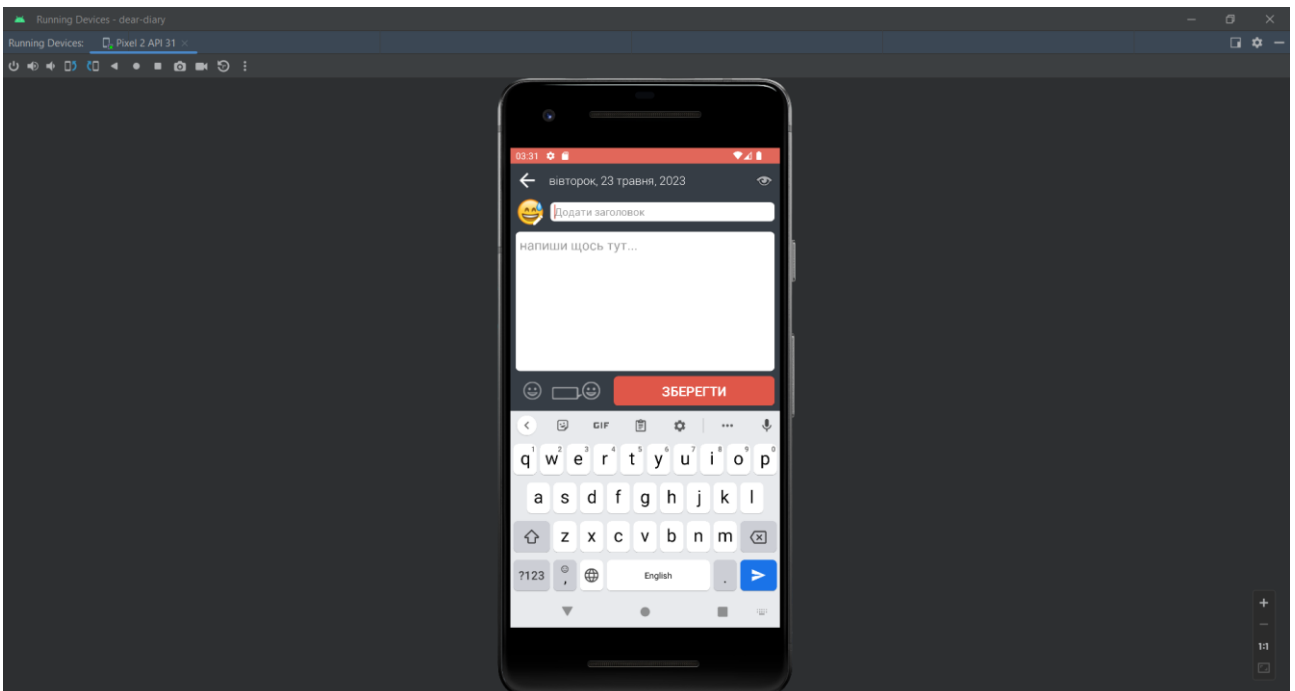
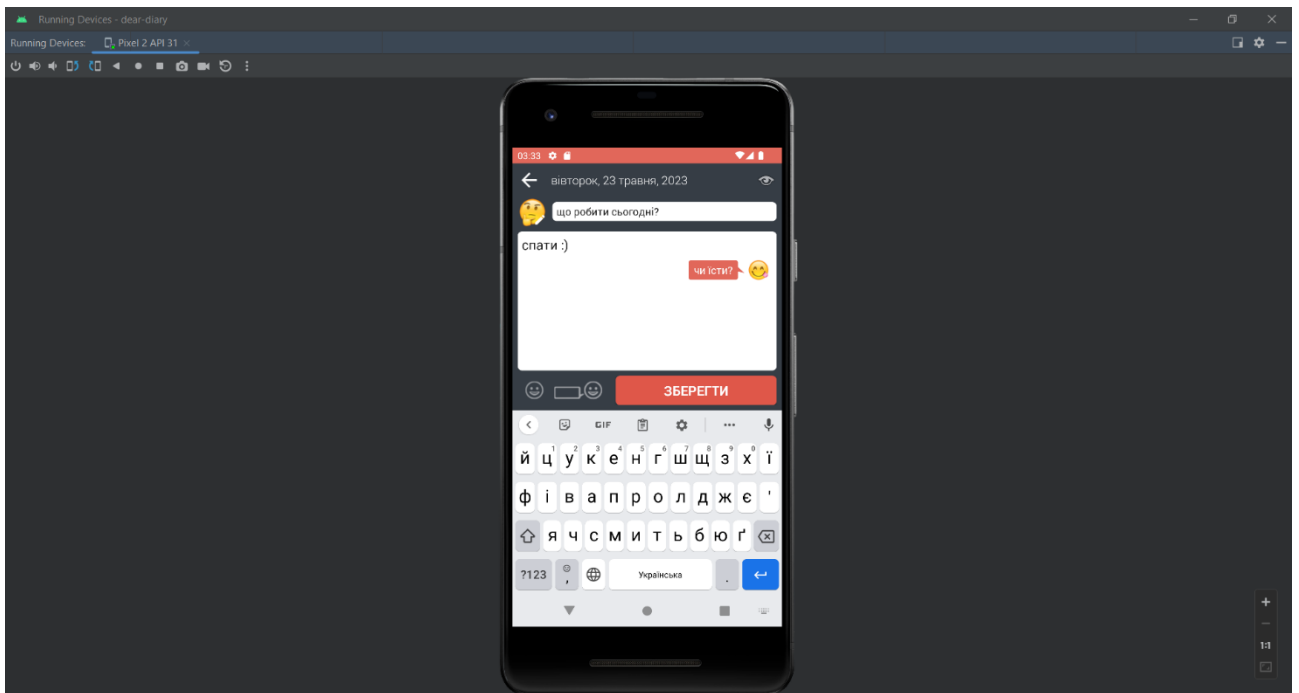


Рисунок 3.6 – Створення особистої події

Примітки можна додавати за допомогою цікавого інтерфейсу, що нагадує



стандартну переписку у соціальних мережах (рис.3.7).

Рисунок 3.7 – Додавання приміток до події

Після успішного додавання власної події, користувач може бачити її на головній вкладці (рис.3.8).

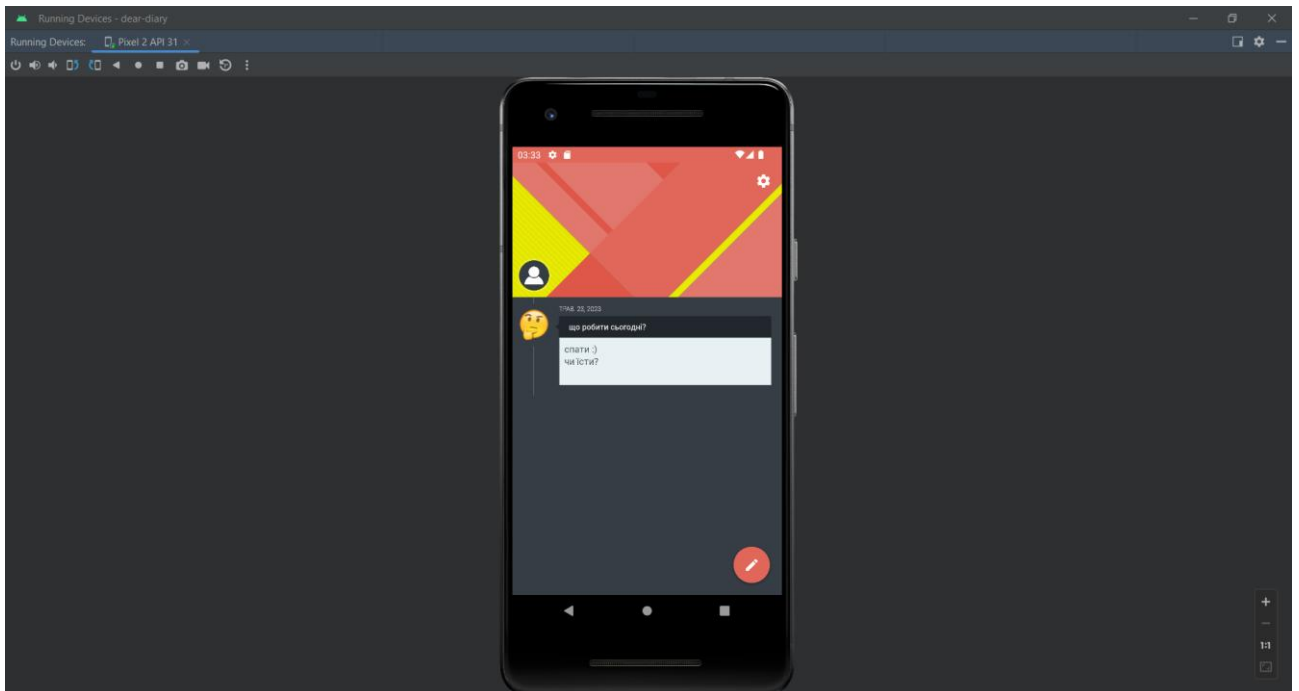


Рисунок 3.8 – Вигляд події на головному екрані

б. 3.5 Тестування застосунку

Під час тестування [45], усі помилки, що виникали у процесі роботи, одразу ж виправлялися.

Таблиця 3.1 – Тестування застосунку

Номер тесту	Опис тесту	Очікуваний результат	Статус
1	Запуск застосунку та перегляд створених записів	Відображення записів за наявності та початкового екрану	Пройдено
2	Створення запису	Додавання заголовку, іконку емоджі, редагування відображення написів (колір, розміщення, емоджі)	Пройдено
3	Додавання запису до головного екрану	Відображення запису на гілці записів	Пройдено
4	Редагування створеного запису	Зміна емоджі, записів та заголовку	Пройдено
5	Зміна фотографій	Заміна на будь-яку фотографію із галереї фонового та особистого зображень	Пройдено
6	Додавання паролю блокування та перевірка роботи	Встановлення паролю для облікового запису, вихід із програми та поява екрану для введення паролю під час запуску застосунку	Пройдено
7	Створення резервної копії	Додавання файлу резервної копії до файлів на смартфоні	Пройдено

ВИСНОВОК

В результаті дипломної роботи було вивчено процес розробки програми електронний щоденник та був розроблений спеціальний мобільний застосунок, який містить необхідні інструменти для покращення планування та організації діяльності людини. А також проаналізовано симуляцію розробленої програми, розглянуто фундаментальний підхід до проектування інтерфейсу електронного

щоденника, досліджено існуючі стандарти та показники якості інтерфейсів мобільних застосунків.

В ході даної роботи були досягнуті наступні результати:

- досліджено та проаналізовано ринок мобільних застосунків на предмет потреб користувачів і очікувань щодо особистого щоденника;
- розглянуті базові підходи до проектування інтерфейсів електронних щоденників;
- розроблені методики проектування інтерфейсу електронного щоденнику із подальшою реалізацією системи.

Результати також демонструють, що були вибрані інструменти розробки сервісу: середовище розробки Android Studio з використанням технології MVC для розробки програмного забезпечення. Розроблено архітектурне рішення побудови клієнтської та серверної частин програми. Застосунок було протестовано, результати чого підтверджують, що програма працює та відповідає вимогам. Розроблений сервіс також дозволяє робити нотатки та анотації в календарі, редагувати наявні, позначати завдання як виконані або видаляти їх з бази.

Працівники або студенти університету можуть використовувати результати виконаної роботи у власних цілях. Застосунок відкритий для оновлень і покращень роботи та функціональності. У подальших дослідженнях існує необхідність у розширенні роботи мобільного застосунку та додаванні функціоналу для зручності користувачів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Time Spent Using Smartphones (2023 Statistics). *Exploding Topics*: веб-сайт. URL: <https://explodingtopics.com/blog/smartphone-usage-stats> (дата звернення: 28.04.2023).
2. Screen Time Statistics 2023 (Your Smartphone Is Hurting You...). *Elite Content Maker*: веб-сайт. URL: <https://elitecontentmarketer.com/screen-time-statistics/> (дата звернення: 28.04.2023).
3. How much money do mobile apps make? (2023 update). *Creative*: веб-сайт. URL: <https://www.creative.onl/how-much-money-do-mobile-apps-make/> (дата звернення: 28.04.2023).
4. Mobile Application Market Size, Share & Trends Analysis Report ByStore Type (Google Store, Apple Store), By Application (Gaming, Music & Entertainment, Health & Fitness), By Region, And Segment Forecasts, 2020 – 2027. *Grand View Research*: веб-сайт. URL: https://www.grandviewresearch.com/industry-analysis/mobile-application-market?utm_source=prnewswire&utm_medium=referral&utm_campaign=ict_21-sep-20&utm_term=mobile-application-market&utm_content=rd (дата звернення: 01.05.2023).
5. Mobile Application Market Size, Share & Trends Report. *Grand View Research*: веб-сайт. URL: <https://www.grandviewresearch.com/industry-analysis/mobile-application-market> (дата звернення: 01.05.2023).
6. Smartphone Market Size, Share, Growth | Analysis Report. *Fortune Business Insights*: веб-сайт. URL: <https://www.fortunebusinessinsights.com/industry-reports/smartphone-market-100308> (дата звернення: 01.05.2023).
7. Mobile Operating System Market Share Worldwide. *Statcounter*: веб-сайт. URL: <https://gs.statcounter.com/os-market-share/mobile/worldwide> (дата звернення: 15.05.2023).

8. iOS Still First Priority For Mobile Developers. *Techcrunch*: веб-сайт. URL: <https://techcrunch.com/2013/08/27/forrester-ios-first/> (дата звернення: 15.05.2023).
9. Android vs Apple Market Share. *BankMyCell*: веб-сайт. URL: <https://www.bankmycell.com/blog/android-vs-apple-market-share/> (дата звернення: 16.05.2023).
10. Android vs iOS users: A detailed behavioral comparison. *Appmysite*: веб-сайт. URL: <https://www.appmysite.com/blog/android-vs-ios-users-a-detailed-behavioural-comparison/> (дата звернення: 16.05.2023).
11. 15 Best Organization Apps & Software in 2023. *Friday*: веб-сайт. URL: <https://friday.app/p/organization-apps> (дата звернення: 16.05.2023).
12. Evernote Documentation. *Evernote*: веб-сайт. URL: <https://evernote.com/intl/en> (дата звернення: 23.05.2023).
13. Evernote Review 2023: App Features, Pricing, Pros & Cons. *CRM*: веб-сайт. URL: <https://crm.org/news/evernote-review> (дата звернення: 23.05.2023).
14. Notes. *App Store*: веб-сайт. URL: <https://apps.apple.com/us/app/notes/id1110145109> (дата звернення: 24.05.2023).
15. 13 Things You Didn't Know Apple's Notes App Could Do. *Pcmag*: веб-сайт. URL: <https://www.pcmag.com/how-to/things-you-didnt-know-apples-notes-app-could-do> (дата звернення: 24.05.2023).
16. Any.do Documentation. *Any.do*: веб-сайт. URL: <https://www.any.do/to-do-list/#> (дата звернення: 26.05.2023).
17. Any.do Review. *Techradar*: веб-сайт. URL: <https://www.techradar.com/reviews/anydo> (дата звернення: 26.05.2023).
18. Google Календар. *Google*: веб-сайт. URL: <https://developers.google.com/calendar> (дата звернення: 27.05.2023).

19. Google Calendar Software Reviews & Ratings. *Software Advice*: веб-сайт. URL: <https://www.softwareadvice.com/scheduling/google-calendar-profile/reviews/> (дата звернення: 27.05.2023).
20. Тайм-менеджмент: методи ефективного управління часом. *Indigo*: веб-сайт. URL: <https://indigo.co.ua/ua/blog/tajm-menedzhment-metody-effektivnogo-upravleniya-vremenem> (дата звернення: 27.05.2023).
21. Най Л. Researching UX: Analytics: книга. SitePoint Pty. Ltd, 2017. 106 с.
22. A Comprehensive Guide To Mobile App Design. *Smashing Magazine*: веб-сайт. URL: <https://www.smashingmagazine.com/2018/02/comprehensive-guide-to-mobile-app-design/> (дата звернення: 30.05.2023).
23. Wigdor D. Brave D.W. NUI World: Designing Natural User Interfaces для Touch and Gesture: книга. Morgan Kaufmann, 2011. 264 с.
24. Як створюється інтерфейс мобільний застосунків. *Wezom*: веб-сайт. URL: <https://wezom.com.ua/ua/blog/dizajn-interfejsov-mobilnyh-prilozhenij> (дата звернення: 30.05.2023).
25. The Fundamental Guide to Mobile Usability. *Toptal*: веб-сайт. URL: <https://www.toptal.com/designers/mobile-ui/la-guia-fundamental-para-la-usabilidad-mobile> (дата звернення: 30.05.2023).
26. Busting the myth that Apple doesn't do User Research. Experience Dynamics: веб-сайт. URL: <https://experiencedynamics.com/blog/2014/09/busting-myth-apple-doesnt-do-user-research> (дата звернення: 30.05.2023).
27. UI Design Dos and Don'ts. *Apple Developer*: веб-сайт. URL: <https://developer.apple.com/design/tips/> (дата звернення: 01.06.2023).
28. Human Interface Guidelines. *Apple Developer*: веб-сайт. URL: <https://developer.apple.com/design/human-interface-guidelines> (дата звернення: 01.06.2023).
29. The power of a planner app. *Monday*: веб-сайт. URL: <https://monday.com/blog/productivity/planner-apps/> (дата звернення: 01.06.2023).

30. MySQL Documentation. *MySQL*: веб-сайт. URL: <https://dev.mysql.com/doc/> (дата звернення: 02.06.2023).
31. PostgreSQL Documentation. *PostgreSQL*: веб-сайт. URL: <https://www.postgresql.org/docs/> (дата звернення: 02.06.2023).
32. SQLite Documentation. *SQLite*: веб-сайт. URL: <https://www.sqlite.org/docs.html> (дата звернення: 02.06.2023).
33. MongoDB Documentation. *MongoDB*: веб-сайт. URL: <https://www.mongodb.com/docs/> (дата звернення: 02.06.2023).
34. Volley Overview. *GitHub*: веб-сайт. URL: <https://google.github.io/volley/> (дата звернення: 02.06.2023).
35. Android. *Wikipedia*: веб-сайт. URL: [https://en.wikipedia.org/wiki/Android_\(operating_system\)](https://en.wikipedia.org/wiki/Android_(operating_system)) (дата звернення: 03.06.2023).
- 36.8 Advantages of Android OS. *Iihtsrt*: веб-сайт. URL: <https://iihtsrt.com/advantages-of-android-os/> (дата звернення: 03.06.2023).
37. Android Studio Documentation. *Android Developers*: веб-сайт. URL: <https://developer.android.com/docs> (дата звернення: 03.06.2023).
38. The Major Advantages of Android Studio App Development. *Indian App Developers*: веб-сайт. URL: <https://www.indianappdevelopers.com/blog/advantages-of-android-studio-app-development/> (дата звернення: 03.06.2023).
39. Learn Java For Android App Development. *GeekForGeeks*: веб-сайт. URL: <https://www.geeksforgeeks.org/learn-java-for-android-app-development-a-complete-guide/> (дата звернення: 03.06.2023).
40. Як визначити цільову аудиторію для вашої ніші? *Web-promo*: веб-сайт. URL: <https://web-promo.ua/ua/blog/produktivnyj-marketing-kak-opredelit-czelevuyu-auditoriyu-dlya-vashej-nishi/> (дата звернення: 03.06.2023).

41. How does Figma help make mobile app development easier? Figma: веб-сайт.
URL: <https://forum.figma.com/t/how-does-figma-help-make-mobile-app-development-easier/35901> (дата звернення: 04.06.2023).
42. User interaction and user interface design. *Edutechwiki*: веб-сайт. URL:
https://edutechwiki.unige.ch/en/User_interaction_and_user_interface_design
(дата звернення: 04.06.2023).
43. UI layer. *Android Developers*: веб-сайт. URL:
<https://developer.android.com/topic/architecture/ui-layer> (дата звернення:
04.06.2023).
44. Why People Love Android. *Pocket Now*: веб-сайт. URL:
<https://pocketnow.com/why-people-love-android/> (дата звернення:
04.06.2023).
45. Кулішова Н.Є., Глуходід М.І. Метод оптимізації процесу тестування веб-ресурсів з урахуванням багатокритеріального підходу: книга. *Eastern-European Journal of Enterprise Technologies*, No. 6/2 (60), 2012. 60-63 с.

ДОДАТКИ

```

public class AboutActivity extends AppCompatActivity implements
View.OnClickListener {
    private AdView mAdView;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        Shared.initialize(getBaseContext());

        setContentView(R.layout.activity_about);

        mAdView = (AdView) findViewById(R.id.ad_view);
        AdRequest adRequest = new AdRequest.Builder()
            .build();
        mAdView.loadAd(adRequest);

        findViewById(R.id.btnBack).setOnClickListener(this);

        TextView t5 = (TextView) findViewById(R.id.textView5);

        TextView t1 = (TextView) findViewById(R.id.textView1);
        TextView t2 = (TextView) findViewById(R.id.textView2);
        TextView t3 = (TextView) findViewById(R.id.textView3);
        TextView t4 = (TextView) findViewById(R.id.textView4);

        t5.setTypeface(Shared.appfontBold);
        t1.setTypeface(Shared.appfontBold);
        t2.setTypeface(Shared.appfontLight);
        t3.setTypeface(Shared.appfontBold);
        t4.setTypeface(Shared.appfontLight);

        try {
            t2.setText("v"+getPackageManager().getPackageInfo(getPackageName(),
0).versionName);
        } catch (PackageManager.NameNotFoundException e) {
            e.printStackTrace();
        }
    }

    @Override
    public void onBackPressed() {
        super.onBackPressed();
        overridePendingTransition(0, android.R.anim.fade_out);
    }

    @Override
    public void onClick(View v) {
        switch (v.getId())
        {
            case R.id.btnBack:
                finish();
                overridePendingTransition(0, android.R.anim.fade_out);
                break;
        }
    }

    @Override

```

```

protected void onDestroy() {
    super.onDestroy();
    if (mAdView != null) {
        mAdView.destroy();
    }
}

@Override
protected void onPause() {
    super.onPause();
    if (mAdView != null) {
        mAdView.pause();
    }
}

@Override
protected void onResume() {
    super.onResume();
    if (mAdView != null) {
        mAdView.resume();
    }
}
}

```

Рисунок А1. Основний клас визначення роботи застосунку

```

<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
xmlns:emojicon="http://schemas.android.com/apk/res-auto"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:background="@color/bg_default"
tools:context="com.vondev.diary.MainActivity">

<hani.momanii.supernova.emoji_library.Helper.EmojiIconTextView
    android:id="@+id/emojiTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text=""
    android:textSize="45dp"
    emojicon:emojiconSize="45sp"/>

</android.support.constraint.ConstraintLayout>

```

Рисунок А2. Розмітка головної сторінки

```

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:emojicon="http://schemas.android.com/apk/res-auto"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:id="@+id/mainLayout"
android:background="@color/bg_default" >

<RelativeLayout

```

```

android:layout_width="match_parent"
android:layout_height="wrap_content"
android:id="@+id/actionBar"
>

<ImageButton
    android:id="@+id/btnBack"
    android:layout_width="35dp"
    android:layout_height="35dp"
    android:layout_alignParentLeft="true"
    android:layout_centerVertical="true"
    android:layout_alignParentStart="true"
    android:layout_margin="8dp"
    android:layout_marginLeft="0dp"
    android:adjustViewBounds="true"
    android:background="@drawable/icon_state"
    android:scaleType="fitXY"
    app:srcCompat="@mipmap/ic_arrow_back_white_24dp" />

<RelativeLayout
    android:id="@+id/dateWrapper"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_toRightOf="@+id/btnBack"
    android:clickable="true"
    android:background="@drawable/icon_state"
    android:layout_centerVertical="true">

    <TextView
        android:id="@+id/txtDate"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="left|center_vertical"
        android:layout_centerVertical="true"
        android:textColor="@color/putih"
        android:layout_margin="8dp"
        android:textSize="18dp"
        android:text="Вівторок, травень 23, 2023" />

</RelativeLayout>

<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_centerVertical="true"
    android:layout_alignParentEnd="true"
    android:layout_alignParentRight="true"
    android:orientation="horizontal">

    <ImageButton
        android:id="@+id/btnPreview"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="8dp"
        android:layout_marginRight="8dp"
        android:layout_marginTop="8dp"
        android:background="@drawable/icon_state"
        app:srcCompat="@android:drawable/ic_menu_view" />

    <ImageButton
        android:id="@+id/btnDelete"
        android:layout_width="wrap_content"

```

```

        android:layout_height="wrap_content"
        android:layout_marginBottom="8dp"
        android:layout_marginRight="8dp"
        android:layout_marginTop="8dp"
        android:background="@drawable/icon_state"
        app:srcCompat="@android:drawable/ic_menu_delete" />

</LinearLayout>

</RelativeLayout>

<RelativeLayout
    android:id="@+id/titleWrapper"
    android:layout_width="match_parent"
    android:layout_height="45dp"
    android:layout_below="@+id/actionBar">

    <hani.momanii.supernova_emoji_library.Helper.EmojiconTextView
        android:id="@+id/emojiTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerVertical="true"
        android:layout_marginLeft="8dp"
        android:gravity="center|center_vertical"
        android:paddingBottom="0dp"
        android:text=""
        android:textSize="45dp"
        emojiicon:emojiiconSize="45sp"/>

    <ImageView
        android:id="@+id/imageView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="30dp"
        android:layout_marginTop="25dp"
        app:srcCompat="@mipmap/ic_edit_white_24dp" />

    <RelativeLayout
        android:id="@+id/titleTextWrapper"
        android:layout_width="match_parent"
        android:layout_height="45dp"
        android:layout_marginLeft="8dp"
        android:layout_marginRight="8dp"
        android:layout_marginTop="8dp"
        android:layout_marginBottom="8dp"
        android:paddingLeft="6dp"
        android:paddingRight="6dp"
        android:layout_toRightOf="@+id/emojiTextView"
        android:background="@drawable/rounded_layout">

        <hani.momanii.supernova_emoji_library.Helper.EmojiconEditText
            android:id="@+id/txtTitle"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:ems="10"
            android:textSize="16sp"
            android:background="@color/putih"
            android:inputType="textPersonName|textNoSuggestions"
            android:hint="@string/add_title"
            android:imeOptions="actionSend"

```

```

        android:focusable="true"
        android:focusableInTouchMode="true"
        emojiicon:emojiiconSize="26sp" />

    </RelativeLayout>
</RelativeLayout>

<ScrollView
    android:id="@+id/scrollDiaryWrapper"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_above="@+id/bottomWrapper"
    android:layout_below="@+id/titleWrapper"
    android:layout_margin="8dp"
    android:background="@drawable/rounded_layout"
    android:clickable="true"
    android:fillViewport="true"
    android:focusable="true"
    android:focusableInTouchMode="true">
    <LinearLayout
        android:id="@+id/diaryWrapper"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        android:padding="6dp"
        android:weightSum="1">
        <hani.momanii.supernova.emoji.library.Helper.EmojiiconEditText
            android:id="@+id/txtContent"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:background="@null"
            android:ems="10"
            android:focusable="true"
            android:focusableInTouchMode="true"
            android:gravity="top|left"
            android:hint="@string/start_typing"
            android:inputType="textMultiLine|textNoSuggestions"
            android:tag="content"
            android:layout_weight="1"
            android:textSize="20sp"
            emojiicon:emojiiconSize="26sp" />
    </LinearLayout>
</ScrollView>

```

Рисунок А3. Розмітка додавання особистої події до щоденника

Table: diary_detail

Column	Data Type
code	INTEGER
code_diary	INTEGER
type	TEXT
color	INTEGER
content	TEXT
flip	INTEGER
emoji	TEXT

Table: diary

Column	Data Type
code	INTEGER
title	TEXT
post_date	DATETIME
feel	TEXT

Table: settings

Column	Data Type
code	TEXT
name	TEXT
value	TEXT
value_blob	BLOB

Рисунок А4. База даних застосунку у табличному вигляді