

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ВСП «ОДЕСЬКИЙ ТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ ОНТУ»**

Спеціальність: 121 «Інженерія програмного забезпечення»

Освітньо-професійна програма: «Розробка програмного забезпечення»

Група: 4РП-08

Дипломний проект

здобувачки освіти денної форми навчання

РП.08.13.000.ДП

***ЛОМАЖУК
АЛІНИ ГЕННАДІЇВНИ***

**м. Одеса
2025 р.**

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ВСП «ОДЕСЬКИЙ ТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ ОНТУ»

Спеціальність: 121 «Інженерія програмного забезпечення»

Освітньо-професійна програма: «Розробка програмного забезпечення»

Група: 4РП-08

ПОЯСНОВАЛЬНА ЗАПИСКА

до дипломного проекту на тему:

Розробка мобільного застосунку-помічника майстра манікюру

Проектний матеріал складається з пояснювальної записки на 75 сторінках та графічного (презентаційного) матеріалу на 14 аркушах (слайдах)

Дипломник _____ (Ломажук А.Г.)

Керівник _____ (Жадан А.С.)

Консультанти:

з економічного розділу _____ (Канський М.Ю.)

з розділу охорони праці та техніки безпеки _____ (Чорновол Н.І.)

з нормоконтролю _____ (Петрашова В.І.)

старший консультант _____ (Кривченко Ю.В.)

До захисту допущений

Голова циклової комісії _____ (Кривченко Ю.В.)

Завідувач відділення _____ (Краснокутська К.Г.)

Захист «21» _____ 06 2025 р. Протокол ЕК № 1

Оцінка ЕК _____ 5/90

Секретар ЕК _____

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ВСП «ОДЕСЬКИЙ ТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ ОНТУ»

Відділення комп'ютерних систем Комісія КТ та ПІ
Спеціальність 121 «Інженерія програмного забезпечення»
Освітньо-професійна програма «Розробка програмного забезпечення»

ЗАТВЕРДЖУЮ:
Заст. дир. з НВР Беркань І.В.
« 12 » 05 2025 р.

ЗАВДАННЯ

на дипломний проект

Здобувачі освіти Ломажук Аліні Геннадіївні
(прізвище, ім'я, по батькові)

1. Тема проекту Розробка мобільного застосунку-помічника майстра манікюру.

затверджена наказом по коледжу від « 14 » листопада 2024р. № 246

2. Термін здачі закінченого проекту _____

3. Вихідні данні до проекту _____

1. Передбачити UI/UX відповідний до цільової аудиторії та предметної області;

2. Використовувати мобільні кросплатформні технології;

3. Забезпечити застосунок необхідним функціоналом для майстрів манікюру.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які необхідно розробити)



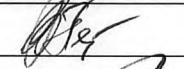
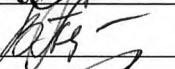


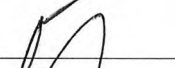
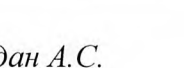

1. Аналіз предметної області; 2. Технології та засоби розробки;

3. Проектування мобільного застосунку; 4. Розробка кросплатформного мобільного застосунку; 5. Тестування мобільного застосунку;

6. Економічний розрахунок; 7. Аспекти охорони праці та техніки безпеки.

5. Перелік графічного (презентаційного) матеріалу (з точним зазначенням обов'язкових креслень, кількості слайдів)
Титул; Технології мобільного застосунку; Навігація у мобільному застосунку; Архітектура мобільного застосунку; Екран «Послуги за категоріями»; . Екран «Клієнтська база»; Екран «Запис»; Екран «Палітра кольорів»; Алгоритм перевірки кольорів на сумісність; Програмний код перевірки на компліментарність; Екран «Генератор малюнків»; Алгоритм генерації кольорів; Екран «Галерея зображень»; Функція «розповсюдити».

6. Консультанти по проекту, із зазначенням розділів проекту, що їх стосується

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв
Основний розділ	Жадан А.С.		
Економічний розділ	Канський М.Ю.		
Розділ охорони праці	Чорновол Н.І.		
Нормоконтроль	Петрашова В.І.		
Старший консультант	Кривченко Ю.В.		

7. Дата видачі завдання 12.05.2025

Керівник Жадан А.С.

Завдання прийняв до виконання Ломажук А.Г.


(підпис)

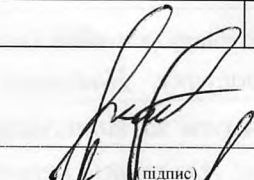
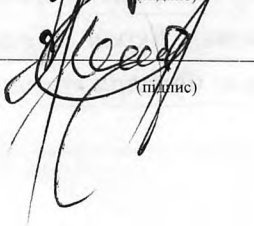
(підпис)

КАЛЕНДАРНИЙ ПЛАН

№ з/р	Назва етапів дипломного проекту	Термін виконання етапів дипломного проекту (роботи)	Відмітка про виконання
1	Формування вступу	15.05.25	виконано
2	Дослідження предметної області	16.05.25	виконано
3	Огляд існуючих рішень	19.05.25	виконано
4	Вибір технічної літератури	20.05.25	виконано
5	Аналіз технологій розробки	22.05.25	виконано
6	Проектування кросплатформного застосунку	23.05.25	виконано
7	Розробка кросплатформного застосунку	26.05.25	виконано
8	Тестування кросплатформного застосунку	27.05.25	виконано
9	Оформлення пояснювальної записки	30.05.25	виконано
10	Підготовка графічних матеріалів	04.06.25	виконано
11	Економічний розрахунок	06.06.25	виконано
12	Опис аспектів охорони праці та техніки безпеки	10.06.25	виконано
13	Підведення висновків	13.06.25	виконано
14	Підготовка доповіді для захисту	16.06.25	виконано

Дипломник

Керівник


(підпис)

(підпис)

ЗМІСТ

Вступ.....	7
1 Основний розділ.....	8
1.1 Аспекти предметної області.....	8
1.1.1 Існуючі виклики.....	8
1.1.2 Існуючі аналоги.....	9
1.2 Проектування мобільного застосунку.....	13
1.2.1 Визначена функціональність.....	13
1.2.2 Обрані технології.....	14
1.2.3 Визначена архітектура.....	16
1.2.4 Проектування основних елементів дизайну.....	18
1.3 Розробка мобільного застосунку.....	20
1.3.1 Налаштування оточення.....	20
1.3.2 Створення екрану навігації.....	22
1.3.3 Створення екрану «Прайс лист».....	24
1.3.4 Створення екрану «Клієнтська база».....	26
1.3.5 Створення екрану «Запис клієнтів».....	27
1.3.6 Створення екрану «Палітра кольорів».....	29
1.3.7 Створення екрану «Генератор зображень».....	32
1.3.8 Створення екрану «Галерея зображень».....	35
1.4 Тестування веб-застосунку.....	36
1.4.1 Обґрунтування виду тестування.....	36
1.4.2 Тестування екрану «Прайс лист».....	38
1.4.3 Тестування екрану «Клієнтська база».....	40
1.4.4 Тестування екрану «Запис» (Календар).....	43
1.4.5 Тестування екрану «Палітра кольорів».....	45
1.4.6 Тестування екрану «Генератор зображень».....	47
1.4.7 Тестування екрану «Галерея зображень».....	49
2 Економічний розділ.....	51
2.1 Резюме.....	51

2.2	Визначення трудомісткості розробки ПЗ.....	51
2.3	Розрахунок ціни програмного продукту	54
3	Розділ охорони праці та техніки безпеки.....	56
3.1	Основні положення.....	56
3.2	Небезпечні та шкідливі фактори в роботі програміста	56
3.3	Вимоги до виробничого середовища	56
3.1	Мікроклімат.....	57
3.2	Шум.....	57
3.3	Освітлення	57
3.4	Електробезпека	58
3.5	Організація робочого місця	58
3.4	Пожежна безпека	60
3.5	Підведення підсумків	60
	Висновки.....	61
	Перелік використаних інформаційних джерел	62
	Додаток А. Фрагмент програмного коду генерації малюнків манікюру	63
	Додаток Б. Слайди мультимедійної презентації	68

ВСТУП

Індустрія краси та сфера послуг майстрів манікюру постійно зростає, що зумовлює потребу у сучасних інструментах для організації робочого процесу. Майстрам необхідно ефективно вести клієнтську базу, планувати записи, контролювати прайс-листи та мати доступ до палітр кольорів і творчих ідей. Відсутність зручного цифрового рішення ускладнює ці завдання та знижує продуктивність роботи [1].

Метою цієї роботи є розробка мобільного застосунку-помічника, який сприятиме автоматизації рутинних процесів майстра манікюру, полегшить керування інформацією про клієнтів і послуги, а також забезпечить інструменти для творчої роботи та планування. Такий застосунок покликаний підвищити ефективність і зручність повсякденної діяльності майстра.

Для досягнення поставленої мети необхідно провести аналіз потреб користувачів, визначити основний функціонал, розробити зручний інтерфейс та організувати логічну структуру застосунку з ключовими екранами – прайс-листом, клієнтською базою, календарем записів, палітрою кольорів, генератором ідей і галереєю робіт. Важливим аспектом є також забезпечення інтуїтивної навігації між розділами.

Розробка застосунку передбачає використання сучасних технологій, зокрема фреймворку React Native, який дозволяє створювати кросплатформенні мобільні застосунки для операційних систем iOS та Android на основі однієї кодової бази. Для організації навігації застосовується бібліотека `@react-navigation/bottom-tabs`, що забезпечує зручний і зрозумілий доступ до основних розділів застосунку. Використання цих технологій сприяє швидкій розробці, гнучкості та масштабованості продукту.

Отже, актуальність роботи обумовлена потребою в ефективних цифрових рішеннях для професіоналів індустрії краси, а реалізація запропонованого мобільного застосунку допоможе покращити організацію роботи і підвищити рівень обслуговування клієнтів.

					<i>РП 08. 13 000. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		7

1 ОСНОВНИЙ РОЗДІЛ

1.1 Аспекти предметної області

1.1.1 Існуючі виклики

У процесі розробки мобільного застосунку-помічника для майстрів манікюру виникає низка актуальних викликів, які зумовлені як специфікою професійної діяльності користувачів, так і особливостями мобільних технологій. Ці виклики стосуються як технічного, так і функціонального аспектів реалізації проекту.

Універсальність і простота інтерфейсу.

Майстри манікюру, як правило, не є технічними фахівцями, тому застосунок має бути максимально інтуїтивно зрозумілим, з адаптивною навігацією та зрозумілою логікою користування. Забезпечення зручного доступу до основних розділів – прайс-листа, бази клієнтів, календаря записів, палітри кольорів, генератора дизайнів та галереї – потребує ретельного UX-дизайну та ефективної організації простору інтерфейсу.

Робота в офлайн-режимі.

Оскільки застосунок передбачено як автономний (офлайн), необхідно забезпечити збереження даних локально на пристрої користувача. Це створює виклик у контексті надійності збереження інформації (особливо при оновленнях), а також організації резервного копіювання та міграції даних між пристроями.

Оптимізація локальної бази даних.

Зберігання інформації про клієнтів, записи в календарі, палітру кольорів, зображення з галереї тощо потребує ефективного управління локальними базами даних (наприклад, SQLite або AsyncStorage). Виклик полягає в тому, щоб реалізувати структуровану та швидкодіючу систему зберігання без втрати продуктивності.

Робота з мультимедіа.

Функціонал галереї та палітри вимагає обробки зображень (наприклад, зразків дизайнів, кольорових рішень, фото з робіт). Це потребує оптимізації

					<i>РП 08. 13 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		8

використання пам'яті пристрою, обробки великих файлів та забезпечення сумісності з різними розмірами і форматами зображень.

Відсутність синхронізації з хмарними сервісами.

Через офлайн-природу застосунку не передбачено автоматичну синхронізацію з хмарними платформами. Це обмежує можливості збереження даних на віддалених серверах, що ускладнює відновлення інформації у разі втрати пристрою або пошкодження даних.

Масштабованість та підтримка.

Застосунок повинен мати потенціал до подальшого розширення функціональності – наприклад, додавання аналітики, системи нагадувань, ведення статистики клієнтів тощо. Проектування архітектури має враховувати можливість масштабування без повної перебудови логіки застосунку.

Безпека персональних даних.

Збереження особистих даних клієнтів (ПІБ, контактів, історії візитів) зобов'язує забезпечити відповідний рівень конфіденційності та захисту інформації, навіть у межах локального сховища. Виникає потреба у реалізації простих механізмів авторизації або блокування доступу до застосунку [1][2][3].

Успішна реалізація мобільного застосунку вимагає комплексного підходу до вирішення як інтерфейсних, так і архітектурних викликів, орієнтованого на потреби цільової аудиторії – майстрів індустрії краси.

1.1.2 Існуючі аналоги

У процесі розробки програмного забезпечення важливим етапом є аналіз існуючих аналогів, що дозволяє виявити сильні та слабкі сторони вже реалізованих рішень, а також визначити напрямки для вдосконалення. Для цілей даної дипломної роботи було проаналізовано низку мобільних застосунків, які орієнтовані на спеціалістів у сфері нігтьового сервісу, зокрема майстрів манікюру. Оскільки розроблюваний застосунок має працювати в офлайн-режимі, особлива увага приділялася продуктам з підтримкою автономної роботи.

Nail Book (iOS, Android).

Nail Book – це мобільний застосунок, орієнтований переважно на клієнтів,

					<i>РП 08. 13 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		9

проте має функціонал для майстрів, зокрема ведення фотогалереї, збереження дизайнів та нотаток. Суттєвим недоліком є обмежені можливості персоналізації сервісів для конкретного майстра, а також необхідність підключення до Інтернету для деяких функцій [4].

На рисунку 1.1 зображено сторінку мобільного застосунку Nail Book в мобільному магазині застосунків PlayMarket.

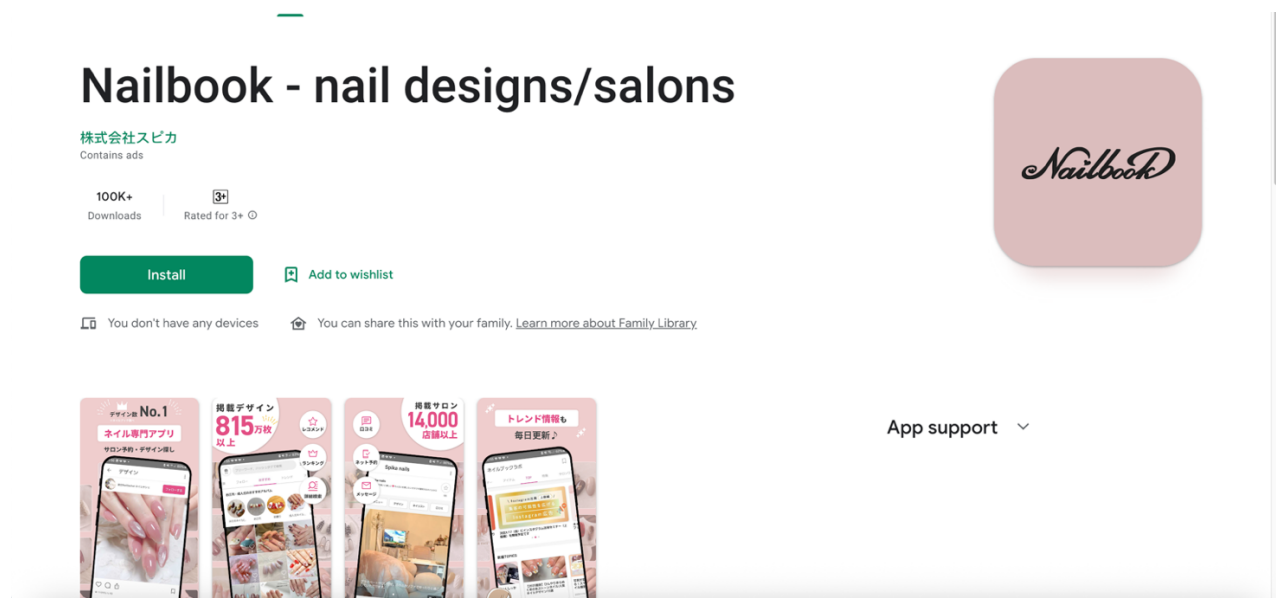


Рисунок 1.1. Nail Book в PlayMarket

SalonAppy.

SalonAppy позиціонується як інструмент для управління салонами краси. Застосунок включає функції запису клієнтів, управління розкладом, нарахування бонусів і ведення клієнтської бази. Однак цей продукт орієнтований радше на колективну роботу в межах салону, а не на індивідуальних майстрів, які працюють самостійно. Додатково, деякі функції є доступними лише в платній версії, що обмежує його використання для початківців [5].

Застосунок дозволяє ефективно керувати роботою салону та покращувати комунікацію з клієнтами. Однак він більше підходить для командної роботи, ніж для індивідуальних майстрів.

На рисунку 1.2 зображено сторінку мобільного застосунку SalonAppy в мобільному магазині застосунків PlayMarket.

					РП 08. 13 001. 00 ДП ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		10

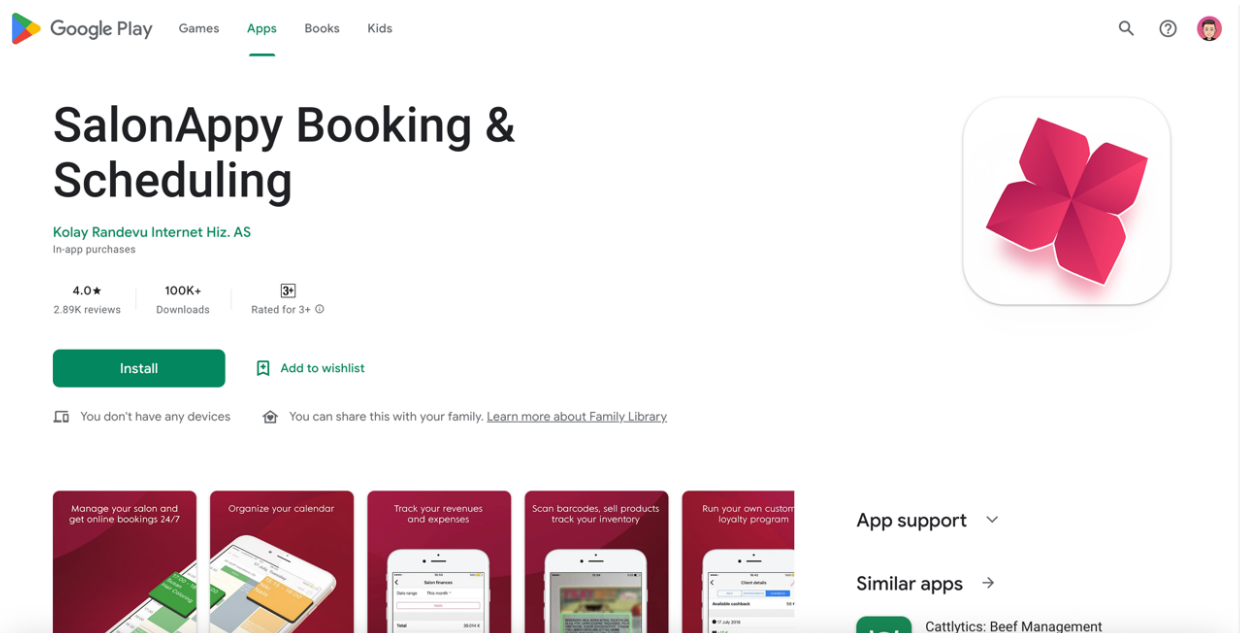


Рисунок 1.2. Salon Appy в PlayMarket

Booksy.

Booksy – один із найпопулярніших застосунків для запису клієнтів у сфері б'юті-послуг. Хоча він має розвинений функціонал календаря, нагадувань і профілів клієнтів, його основна перевага полягає у взаємодії з клієнтами через Інтернет. Через це офлайн-функціональність Booksy є суттєво обмеженою. Крім того, застосунок вимагає реєстрації, що може бути зайвим бар'єром для деяких користувачів [6].

На рисунку 1.3 зображено головну сторінку сервісу Booksy.

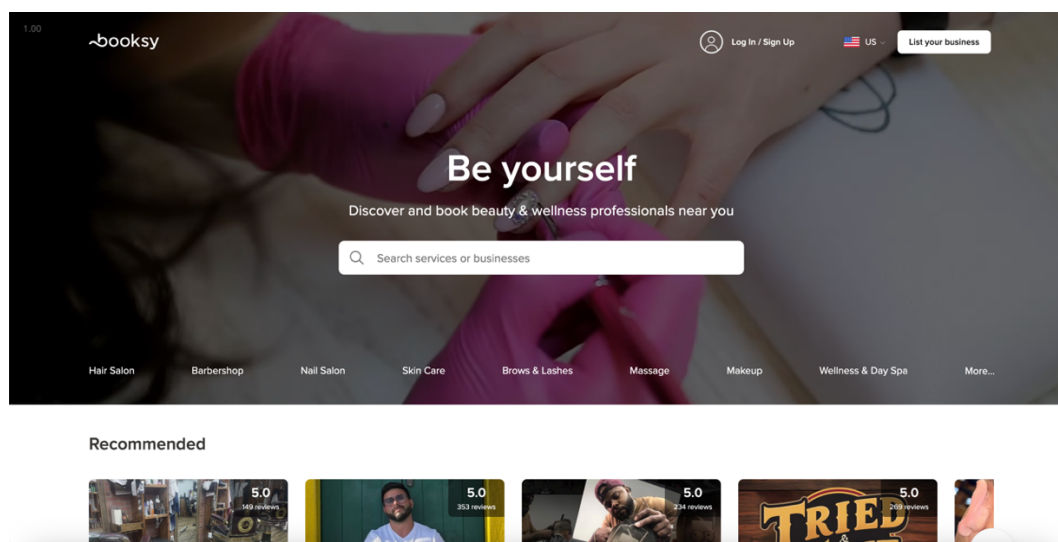


Рисунок 1.3. Головна сторінка сервісу Booksy

Програмне забезпечення загального призначення.

					<i>РП 08. 13 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		11

Багато майстрів манікюру змушені використовувати не спеціалізовані рішення, а універсальні інструменти, як-от Google Календар, Google Таблиці або нотатки в телефоні. Незважаючи на гнучкість таких інструментів, вони не враховують специфіку роботи в сфері нейл-індустрії: облік палітри кольорів, дизайнів, фотогалереї робіт, генераторів ідей тощо [7].

На рисунку 1.4 зображено сервіс Google Calendar.

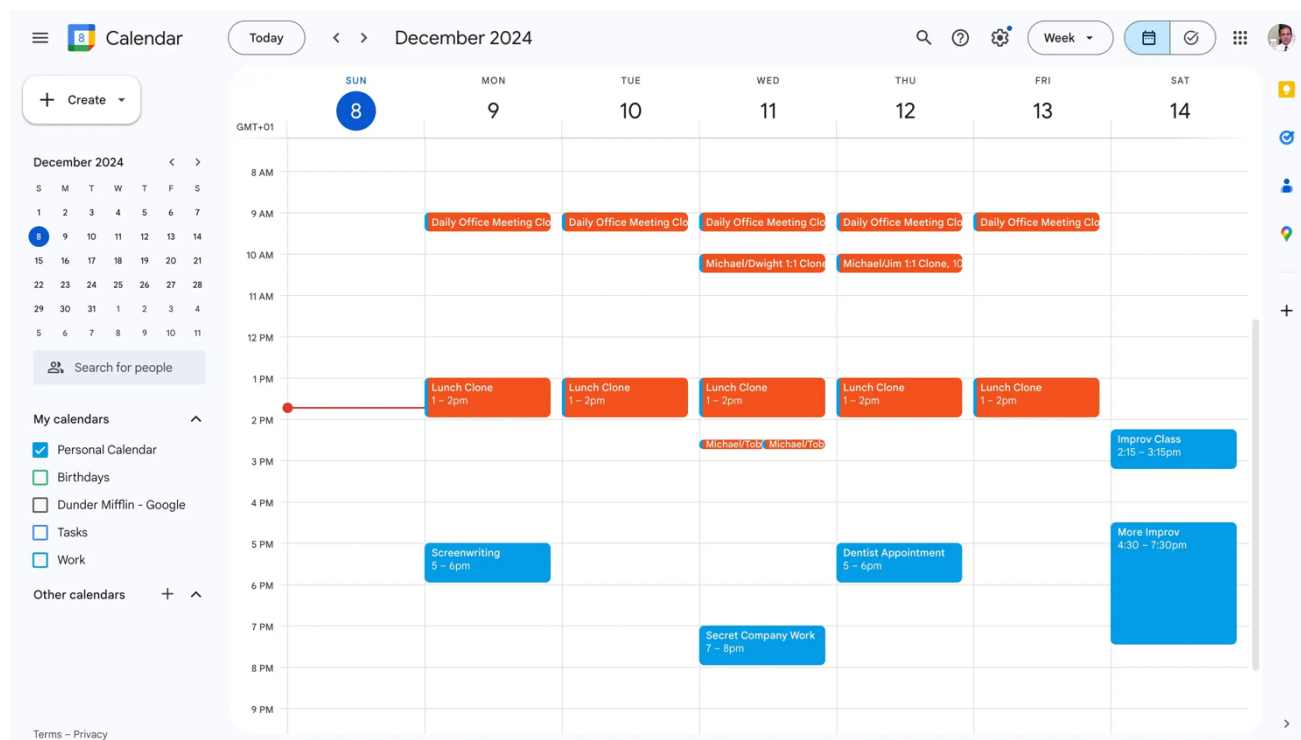


Рисунок 1.4. Сервіс Google Calendar

Аналіз існуючих аналогів показав, що на ринку присутні як універсальні, так і вузькоспеціалізовані застосунки для майстрів манікюру, проте жоден з них не поєднує всі ключові компоненти в офлайн-режимі. Запропонований у дипломному проекті мобільний застосунок вирізняється інтеграцією кількох критично важливих функцій: прайс-лист, база клієнтів, календар записів, палітра кольорів, генератор ідей та галерея – усе в межах однієї автономної платформи. Це забезпечує зручність використання навіть у віддалених умовах або при відсутності стабільного Інтернет-з'єднання.

Завдяки цьому застосунок не лише оптимізує робочі процеси майстра, а й підвищує якість обслуговування клієнтів. Його автономність робить інструмент

надійним помічником у будь-яких умовах, зокрема під час виїзних сесій або роботи в салонах без доступу до мережі.

1.2 Проєктування мобільного застосунку

1.2.1 Визначена функціональність

У межах дипломного проєкту реалізовано мобільний офлайн-застосунок, що виконує роль персонального помічника для майстра манікюру. Основна мета застосунку – забезпечення швидкого доступу до ключової інформації, оптимізація щоденних робочих процесів та підвищення зручності взаємодії з клієнтами.

Розроблена система має зручну навігацію у вигляді нижнього таб-меню, що дозволяє швидко перемикатися між основними модулями. Функціональність застосунку охоплює наступні компоненти:

Прайс-лист.

Модуль, що дозволяє майстру зберігати та редагувати вартість послуг. Це спрощує комунікацію з клієнтами та пришвидшує процес ознайомлення з актуальними цінами.

База клієнтів.

Інтерфейс для ведення обліку клієнтів, включаючи їх контактні дані, історію візитів, вподобання та індивідуальні нотатки. Забезпечує швидкий доступ до інформації та персоналізований підхід до обслуговування.

Календар записів.

Вбудований календар, що дозволяє створювати, переглядати та редагувати записи клієнтів за датами. Дає змогу ефективно планувати графік роботи та уникати накладок у розкладі.

Палітра кольорів.

Цифрова палітра, де зберігаються кольори лаків, доступні для вибору. Модуль дозволяє швидко презентувати клієнту наявні варіанти, що спрощує процес прийняття рішень.

Генератор ідей.

Інструмент для випадкової генерації ідей дизайну нігтів на основі

					<i>РП 08. 13 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		13

попередньо заданих шаблонів, стилів або кольорів. Може слугувати джерелом натхнення для майстра або клієнта.

Галерея робіт.

Розділ для збереження фотографій виконаних робіт. Забезпечує візуальне портфоліо, яке можна демонструвати новим клієнтам для ознайомлення з рівнем майстерності.

Кожен з модулів працює в офлайн-режимі, що є особливо актуальним для користувачів, які можуть не мати постійного доступу до мережі Інтернет. Такий підхід гарантує автономність застосунку, стабільність його роботи та збереження усіх даних без потреби синхронізації з хмарними сервісами.

Загальна структура та набір функцій були визначені на основі аналізу потреб цільової аудиторії – майстрів манікюру, які працюють як в умовах салону, так і приватно. Реалізована функціональність спрямована на забезпечення професійного інструменту для щоденного використання, який об'єднує організаційні, інформаційні та візуальні засоби підтримки діяльності спеціаліста.

1.2.2 Обрані технології

Під час розробки мобільного застосунку-помічника для майстра манікюру було прийнято низку архітектурних рішень, які забезпечують ефективність, зручність користування та масштабованість системи. Застосунок орієнтовано на автономне використання без постійного підключення до мережі, що накладає певні вимоги до зберігання даних та організації логіки інтерфейсу.

Обрана платформа та фреймворк.

Для створення мобільного застосунку було використано React Native – кросплатформену технологію, що дозволяє створювати нативні мобільні інтерфейси для Android та iOS за допомогою JavaScript та React. Це рішення дозволяє скоротити час і витрати на розробку, забезпечуючи при цьому високу продуктивність і нативний вигляд інтерфейсу.

Переваги використання React Native:

- підтримка коду для кількох платформ одночасно;
- велика екосистема бібліотек;

					<i>РП 08. 13 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		14

- зручна інтеграція з нативними модулями;
- активна спільнота та гарна документація [8].

На рисунку 1.5 зображено обрані технології.

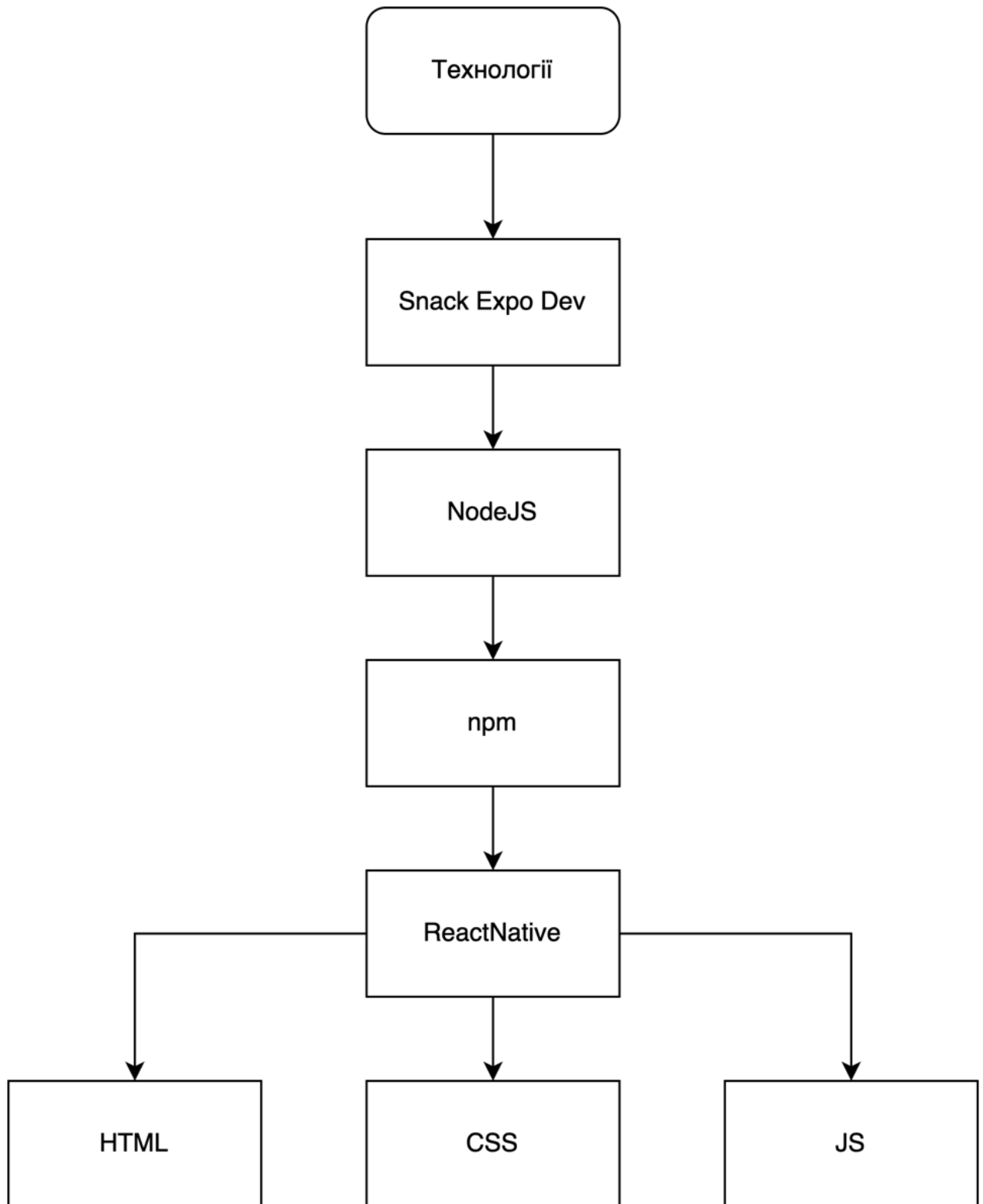


Рисунок 1.5. Обрані технології

1.2.3 Визначена архітектура

Архітектура побудована за принципами модульності та розділення відповідальностей. Основні функціональні блоки реалізовано у вигляді окремих екранів:

1. Прайс: модуль для перегляду та редагування прайсу на послуги;
2. Клієнти: база клієнтів з можливістю збереження контактів та нотаток;
3. Запис: система планування візитів;
4. Палітра: візуальне представлення кольорів та їх сумісність;
5. Генератор: допоміжний інструмент для генерації дизайнів або ідей;
6. Галерея: перегляд і збереження фото виконаних робіт.

На рисунку 1.6 зображено схему екранів.

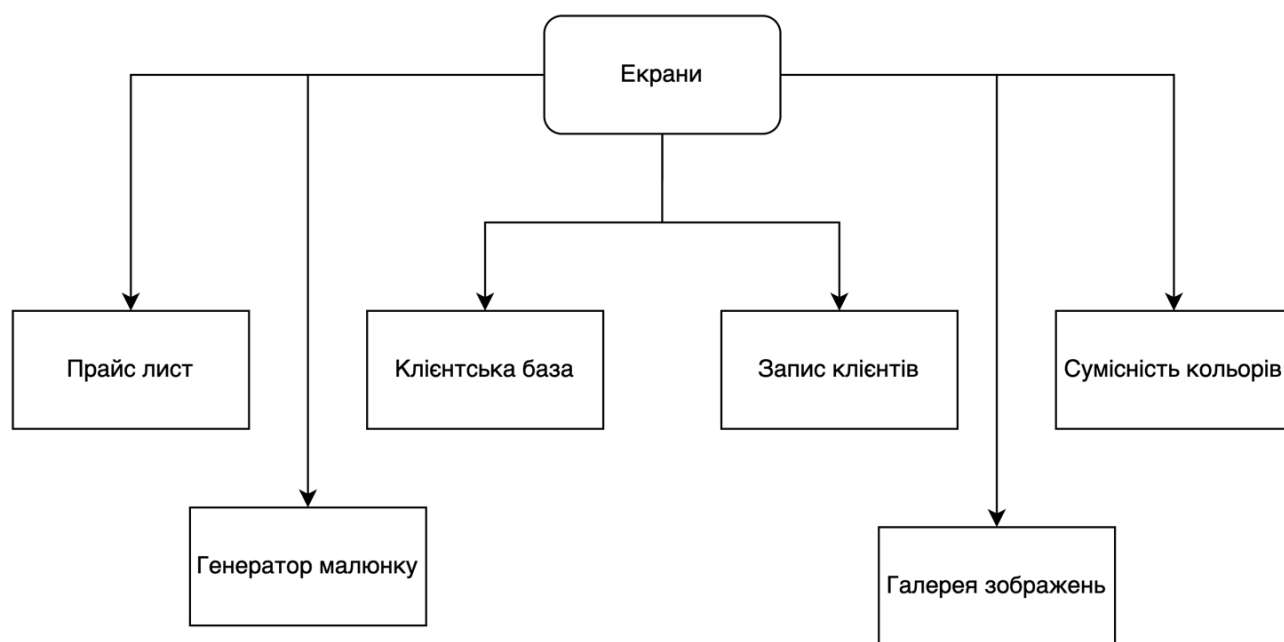


Рисунок 1.6. Схема екранів

Для навігації між екранами використано бібліотеку `@react-navigation/bottom-tabs`, що забезпечує інтуїтивний доступ до функціональних частин програми через нижню панель вкладок. Іконки вкладок реалізовано у вигляді емодзі, що спрощує сприйняття і підвищує візуальну привабливість.

Зберігання даних.

Оскільки застосунок працює в офлайн-режимі, критичною є можливість

зберігати дані локально. Для цього було обрано бібліотеку `@react-native-async-storage/async-storage`, яка дозволяє зберігати інформацію у вигляді ключ-значення на пристрої користувача. Такий підхід забезпечує швидкий доступ до даних без необхідності мережевого з'єднання.

Використані бібліотеки.

Для реалізації функціональності було використано низку спеціалізованих бібліотек:

1. `@react-native-community/datetimepicker`: вибір дати і часу для календаря;
2. `react-native-color-picker`: вибір кольору у палітрі;
3. `react-native-image-picker`: додавання фотографій до галереї;
4. `@react-native-picker/picker`: динамічний вибір параметрів;
5. `@react-native-community/slider`: елементи керування інтенсивністю чи параметрами;
6. `react-native-vector-icons`: використання нативних іконок для кращої ідентифікації функцій.

Організація інтерфейсу користувача.

Інтерфейс побудовано з урахуванням принципів UX-дизайну: простота, мінімалізм, логічна послідовність дій користувача. Головний акцент зроблено на візуальні символи та кольорове оформлення, яке відповідає стилістиці б'юті-сфери.

Обґрунтування офлайн-режиму.

Оскільки застосунок орієнтовано на використання під час прийому клієнтів у приміщенні без гарантії стабільного інтернету, обрано офлайн-архітектуру. Всі дані обробляються та зберігаються локально, що гарантує доступність функцій за будь-яких умов. У майбутньому можливе додавання синхронізації з хмарним сховищем як додаткової опції.

Таким чином, архітектура застосунку забезпечує гнучкість та стабільність роботи навіть без інтернет-з'єднання. Завдяки модульній побудові та використанню сучасних бібліотек, застосунок легко масштабувати та доповнювати новими функціями.

					<i>РП 08. 13 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		17

На рисунку 1.7 зображено загальну архітектуру застосунку.

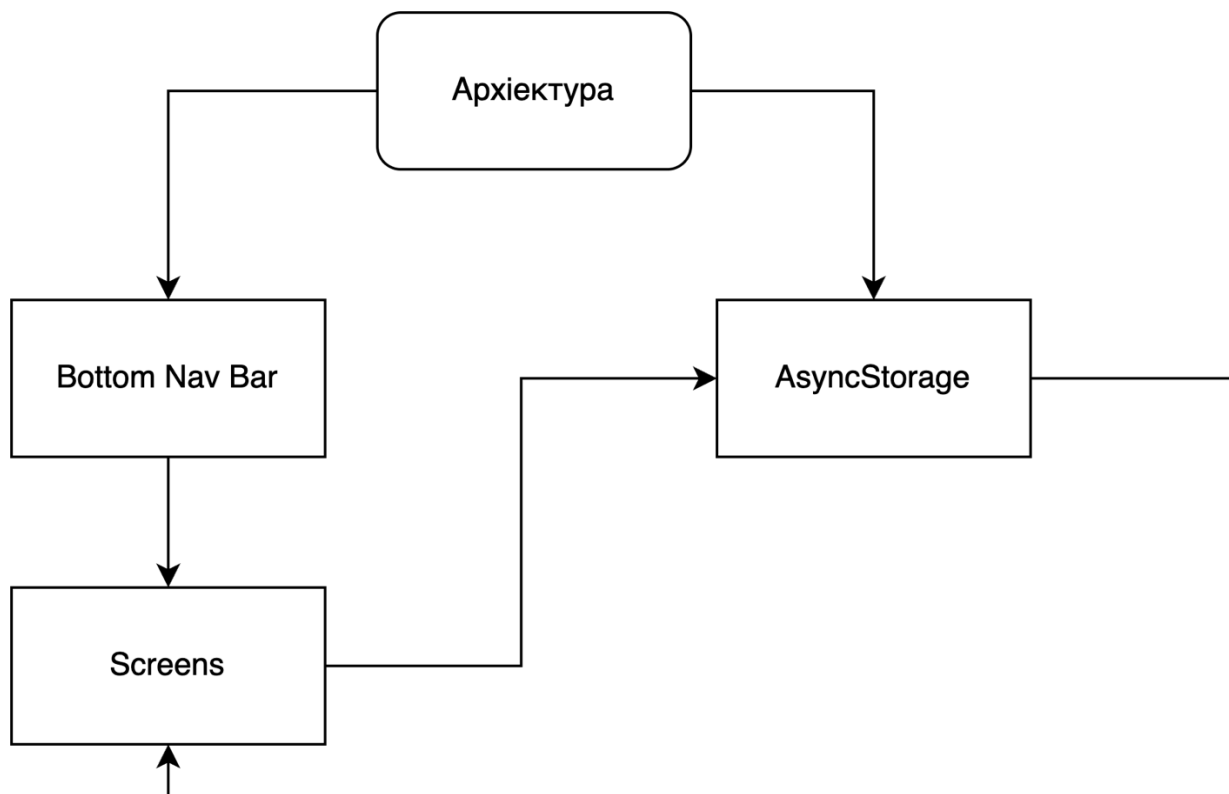


Рисунок 1.7. Загальна архітектура застосунку

1.2.4 Проектування основних елементів дизайну

При розробці мобільного застосунку-помічника майстра манікюру особливу увагу приділено дизайну інтерфейсу користувача, який має бути інтуїтивно зрозумілим, зручним та естетично привабливим. Основні вимоги до дизайну включають:

1. Простота та зрозумілість: інтерфейс має бути мінімалістичним, щоб користувач міг легко орієнтуватись серед функціоналу без зайвого навантаження.
2. Візуальна привабливість: використання приємної кольорової гами з акцентом на відтінки рожевого та ніжних пастельних кольорів, що відповідають індустрії краси.
3. Уніфікованість стилю: всі екрани та компоненти дотримуються єдиного стилю з чіткими межами, закругленими кутами та помірною контрастністю.
4. Адаптивність: дизайн враховує різні розміри екранів мобільних пристроїв,

Зм.	Арк.	№ докум.	Підпис	Дата

РП 08. 13 001. 00 ДП ПЗ

Арк.

18

забезпечуючи комфортне використання на смартфонах з різними роздільними здатностями.

Дизайн навігації.

Навігація у застосунку реалізована за допомогою нижньої панелі вкладок (bottom tab navigator), що є поширеним і зручним рішенням для мобільних додатків. Такий підхід дозволяє швидко переключатися між основними функціональними розділами: прайс-листом, базою клієнтів, календарем записів, палітрою кольорів, генератором і галереєю.

Для покращення сприйняття візуальною іконкою вкладки використано емодзі, що інтуїтивно асоціюються з відповідним розділом (наприклад, 💰 – прайс, 👤 – клієнти, 📅 – календар). Активна вкладка виділяється кольором deerpink, що відповідає загальній кольоровій гамі застосунку.

Навігаційна структура реалізована у середовищі React Native із застосуванням бібліотек @react-navigation/native та @react-navigation/bottom-tabs. Це забезпечує плавний перехід між екранами, високу продуктивність і кросплатформність.

На рисунку 1.8 зображено дизайн навігації.

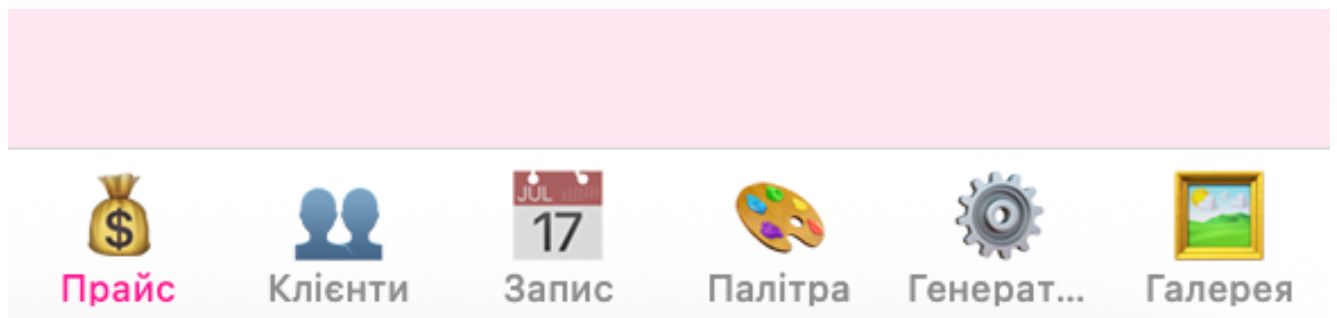


Рисунок 1.8. Дизайн навігації

Дизайн екранів.

Кожен основний екран застосунку відповідає окремій функції і реалізований як самостійний компонент React Native.

Основні елементи інтерфейсу до екранів включають:

- Форми вводу інформації. Використовується компонент TextInput із зрозумілими плейсхолдерами.

- Кнопка додавання або оновлення послуги з яскравим кольоровим оформленням та зрозумілим текстом.
- Список інформації, згрупований за категоріями, реалізований через `SectionList` – це дозволяє впорядкувати інформацію та покращити її сприйняття.
- Кнопки редагування та видалення для кожної послуги з відповідними піктограмами.

Дизайн виконаний у світлій кольоровій гамі з акцентом на різнокольорові відтінки, що створює приємне візуальне середовище. Всі елементи мають закруглені кути та помірні відступи, що сприяє сучасному вигляду та зручності користування.

1.3 Розробка мобільного застосунку

1.3.1 Налаштування оточення

У процесі розробки мобільного застосунку-помічника для майстра манікюру було створено офлайн застосунок з використанням інструментів та бібліотек, орієнтованих на кросплатформенну розробку. Основним фреймворком було обрано `React Native`, який забезпечує можливість створення нативних інтерфейсів для `Android` та `iOS` за допомогою мови `JavaScript`.

Середовище розробки.

Розробка застосунку здійснювалася у середовищі [Snack Expo](#) – онлайн-платформі, яка дозволяє запускати, тестувати та ділитися кодом без потреби встановлення локального інструментарію. `Snack Expo` є частиною екосистеми `Expo`, яка надає попередньо налаштоване середовище з базовим набором бібліотек для `React Native`. Такий підхід є доцільним для швидкого прототипування, а також для підтримки офлайн-функціональності, що була ключовою вимогою проєкту.

Структура проєкту.

Проєкт має модульну структуру, що передбачає розподіл функціональних складових за окремими екранами:

1. `PriceListScreen`: екран для перегляду прайсу послуг;

					<i>РП 08. 13 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		20

2. ClientBaseScreen: база даних клієнтів;
3. CalendarScreen: екран календаря для ведення записів;
4. ColorPaletteScreen: палітра кольорів;
5. GeneratorScreen: генератор дизайнів або порад;
6. GalleryScreen: галерея виконаних робіт.

Навігація між екранами реалізована за допомогою бібліотеки `@react-navigation/bottom-tabs`, що забезпечує нижню панель вкладок з використанням емодзі як іконок.

Основні залежності

Для забезпечення функціональності застосунку були використані такі бібліотеки:

1. `@react-navigation/native`: основна навігаційна бібліотека;
2. `@react-navigation/bottom-tabs`: реалізація нижнього таб-меню;
3. `react-native-vector-icons`: підтримка іконок, зокрема для майбутнього розширення інтерфейсу;
4. `react-native-color-picker`: вибір кольору в палітрі;
5. `react-native-image-picker`: завантаження зображень до галереї;
6. `@react-native-community/datetimepicker`: вибір дати та часу для календаря;
7. `@react-native-community/slider`: підтримка повзунків для інтерфейсу;
8. `@react-native-picker/picker`: реалізація випадаючих списків;
9. `@react-native-async-storage/async-storage`: зберігання даних користувача в офлайн-режимі;
10. `react-native-safe-area-context`: коректне відображення інтерфейсу на різних пристроях;
11. `react-native-screens` та `react-native-svg`: оптимізація продуктивності та підтримка SVG-графіки.

Файл конфігурації `package.json` містив відповідні записи для усіх вищезазначених залежностей, що були автоматично інтегровані Snack Expo при створенні проєкту.

Нижче наведено код файлу `package.json`

					<i>РП 08. 13 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		21

```

{
  "dependencies": {
    "react-native-svg": "15.11.2",
    "react-native-screens": "~4.10.0",
    "@react-navigation/native": "*",
    "react-native-color-picker": "*",
    "react-native-image-picker": "*",
    "react-native-vector-icons": "10.2.0",
    "@react-native-picker/picker": "2.11.0",
    "@react-navigation/bottom-tabs": "*",
    "@react-native-community/slider": "4.5.6",
    "react-native-safe-area-context": "5.4.0",
    "@react-native-community/datetimepicker": "8.3.0",
    "react-native-vector-icons/MaterialIcons": "*",
    "@react-native-async-storage/async-storage": "2.1.2"
  }
}

```

Запуск застосунку.

Після створення проєкту в Snack Expo, запуск застосунку відбувався в браузері або у мобільному застосунку Expo Go, шляхом сканування згенерованого QR-коду. Такий підхід дозволив швидко тестувати застосунок на різних пристроях без необхідності локального складання.

1.3.2 Створення екрану навігації

Головний екран мобільного застосунку-помічника майстра манікюру виконує роль основної точки входу для користувача. Його головним призначенням є забезпечення зручної та інтуїтивної навігації між ключовими функціональними розділами застосунку, що дозволяє майстру швидко переходити до потрібних сервісів. Компонент створений на базі React Native із застосуванням бібліотеки @react-navigation, що гарантує кросплатформену сумісність та стабільність навігаційного процесу.

Компонент головного екрану реалізує панель вкладок (Tab Navigator), яка відображається внизу екрану пристрою. Кожна вкладка відповідає окремому розділу застосунку, таким як: прайс-лист, база клієнтів, календар, палітра кольорів, генератор та галерея.

Основні функції компоненту включають:

1. Візуалізація та інтерактивність навігаційної панелі, що дозволяє швидко перемикатися між розділами.
2. Відображення інтуїтивно зрозумілих іконок у вигляді емодзі для покращення

					<i>РП 08. 13 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		22

сприйняття та швидкого орієнтування.

3. Зміна кольору активної вкладки, що сигналізує про поточний активний екран.
4. Ініціалізація та управління станом навігації в межах застосунку.

Нижче наведено код основної функції App, яка реалізує головний екран із навігацією.

```
import React from 'react';
import { Text } from 'react-native';
import { NavigationContainer } from '@react-navigation/native';
import { createBottomTabNavigator } from '@react-navigation/bottom-tabs';

import PriceListScreen from './screens/PriceListScreen';
import ClientBaseScreen from './screens/ClientBaseScreen';
import CalendarScreen from './screens/CalendarScreen';
import ColorPaletteScreen from './screens/ColorPaletteScreen';
import GeneratorScreen from './screens/GeneratorScreen';
import GalleryScreen from './screens/GalleryScreen';

const Tab = createBottomTabNavigator();

export default function App() {
  return (
    <NavigationContainer>
      <Tab.Navigator
        screenOptions={({ route }) => ({
          tabBarActiveTintColor: 'deeppink',
          tabBarIcon: ({ color, size }) => {
            let emoji;

            switch (route.name) {
              case 'Прайс':
                emoji = '💰';
                break;
              case 'Клієнти':
                emoji = '👥';
                break;
              case 'Календар':
                emoji = '📅';
                break;
              case 'Палітра':
                emoji = '🎨';
                break;
              case 'Генератор':
                emoji = '⚙️';
                break;
              case 'Галерея':
                emoji = '🖼️';
                break;
              default:
                emoji = '?';
            }
          }
        })
      />
      <Text style={{ fontSize: size }}>{emoji}</Text>
    </Tab.Navigator>
  );
}
```

					<i>РП 08. 13 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		23

```

    >
    <Tab.Screen name="Прайс" component={PriceListScreen} />
    <Tab.Screen name="Клієнти" component={ClientBaseScreen} />
    <Tab.Screen name="Календар" component={CalendarScreen} />
    <Tab.Screen name="Палітра" component={ColorPaletteScreen} />
    <Tab.Screen name="Генератор" component={GeneratorScreen} />
    <Tab.Screen name="Галерея" component={GalleryScreen} />
  </Tab.Navigator>
</NavigationContainer>
);
}

```

Пояснення коду

1. Компонент `NavigationContainer` обгортає навігаційну структуру і відповідає за управління станом навігації.
2. `createBottomTabNavigator` створює панель вкладок у нижній частині екрану.
3. У властивості `screenOptions` визначається колір активної вкладки (`tabBarActiveTintColor`) та іконки у вигляді емодзі, які вибираються залежно від імені вкладки (`route.name`).
4. Для кожної вкладки задається відповідний компонент екрану (наприклад, `PriceListScreen` для вкладки «Прайс»).

1.3.3 Створення екрану «Прайс лист»

Основна мета екрану «Прайс лист» забезпечити зручний інтерфейс для створення, редагування, зберігання та групування переліку послуг, які надає майстер. Даний компонент виконує функцію цифрового каталогу, де кожна послуга описується за чотирма параметрами: назвою, ціною, тривалістю та категорією (групою), до якої вона належить (наприклад, «Манікюр», «Педикюр» тощо).

Функціональність компоненту:

Компонент реалізує повноцінну CRUD-функціональність (Create, Read, Update, Delete) для роботи з переліком послуг. Основні можливості:

1. Додавання нової послуги з введенням назви, ціни, тривалості та групи.
2. Редагування наявної послуги з можливістю зміни будь-якого поля.
3. Видалення послуги із локального списку.
4. Збереження списку послуг у локальне сховище за допомогою `AsyncStorage`, що дозволяє зберігати інформацію між сеансами.

					РП 08. 13 001. 00 ДП ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		24

5. Групування послуг за категоріями з виведенням у вигляді розділів у компоненті `SectionList`.

Це дозволяє майстру швидко оновлювати ціни та переглядати структуру наданих послуг без потреби у зовнішньому з'єднанні з мережею або сервером.

Основною функціональною одиницею є метод `addOrUpdateService`, який відповідає за додавання нової або оновлення наявної послуги. Нижче розглянемо логіку реалізації цієї функції:

```
const addOrUpdateService = () => {
  if (!name || !price || !duration || !group) return;

  const newService = { name, price, duration, group };
  let updatedServices = [...services];

  if (editingIndex !== null) {
    updatedServices[editingIndex] = newService;
    setEditingIndex(null);
  } else {
    updatedServices.push(newService);
  }

  setServices(updatedServices);
  saveServices(updatedServices);
  setName('');
  setPrice('');
  setDuration('');
  setGroup('');
};
```

Принцип дії:

1. Валідація даних: перед додаванням перевіряється, чи всі поля (назва, ціна, тривалість, група) заповнені. Якщо хоча б одне порожнє – функція не виконується.
2. Формування об'єкта послуги: на основі введених даних створюється об'єкт `newService`.
3. Визначення режиму роботи:
 - Якщо `editingIndex` не дорівнює `null`, отже відбувається редагування – елемент у масиві `services` за відповідним індексом оновлюється.
 - Якщо ж `editingIndex` є `null`, додається новий об'єкт до списку.
4. Оновлення стану та збереження:
 - Масив `services` оновлюється відповідним чином.
 - Новий список зберігається у локальне сховище через функцію `saveServices`.

					<i>РП 08. 13 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		25

- Очистка полів введення – забезпечує зручність для наступного введення.

1.3.4 Створення екрану «Клієнтська база»

Екран «Клієнтська база» призначений для обліку та збереження інформації про клієнтів майстра манікюру. Його основною метою є надання можливості швидкого додавання, перегляду, редагування та видалення записів про клієнтів, що суттєво полегшує організацію робочого процесу, дозволяючи майстру ефективніше планувати та управляти своїм часом і контактами.

Функціонал екрану охоплює наступні можливості:

1. Додавання нового клієнта: користувач вводить ім'я, прізвище, номер телефону та нотатки, які зберігаються у локальному сховищі пристрою.
2. Редагування наявного клієнта: за потреби дані клієнта можна змінити через інтерфейс.
3. Видалення клієнта: реалізована функція видалення запису з бази.
4. Автоматичне збереження: усі зміни зберігаються у локальному сховищі AsyncStorage після кожної операції.
5. Перегляд списку клієнтів: реалізовано прокручуваний список, де відображаються всі внесені клієнти з відповідними нотатками.

Інтерфейс компонента створено за допомогою бібліотеки React Native, що дозволяє забезпечити кросплатформенність застосунку та зручність взаємодії з користувачем.

Ключовою функцією компоненту є `addOrUpdateClient`, яка реалізує логіку додавання нового клієнта або оновлення вже наявного запису. Нижче подано фрагмент коду та його докладний аналіз:

```
const addOrUpdateClient = () => {
  if (!firstName || !lastName || !phone) return;

  const newClient = { firstName, lastName, phone, notes };
  let updatedClients = [...clients];

  if (editingIndex !== null) {
    updatedClients[editingIndex] = newClient;
    setEditingIndex(null);
  } else {
    updatedClients.push(newClient);
  }

  setClients(updatedClients);
}
```

					<i>РП 08. 13 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		26

```
saveClients(updatedClients);

setFirstName('');
setLastName('');
setPhone('');
setNotes('');
};
```

Ця функція виконує наступні дії:

1. Валідація даних: перевіряється, чи введені обов'язкові поля (firstName, lastName, phone). Якщо хоча б одне з них порожнє – функція припиняє виконання.
2. Формування об'єкта клієнта: створюється об'єкт newClient, який містить дані клієнта.
3. Оновлення або додавання:
 - Якщо індекс editingIndex не є null, отже користувач редагує вже існуючий запис – елемент масиву clients замінюється.
 - Інакше – новий запис додається в кінець масиву.
4. Збереження змін: оновлений масив клієнтів передається у функцію saveClients, яка записує дані в AsyncStorage.
5. Очищення полів вводу: після завершення операції форма скидається до початкового стану.

1.3.5 Створення екрану «Запис клієнтів»

Екран «Запис клієнтів» дозволяє автоматизувати процес реєстрації клієнтів на обрані послуги у визначену дату й час. Цей компонент забезпечує користувачеві (майстру) зручний інтерфейс для керування записами, вибору клієнта та послуги, встановлення дати й часу, а також перегляду та видалення наявних записів. Таким чином, компонент сприяє ефективному плануванню робочого часу та підвищенню якості обслуговування.

Функціональність компоненту:

1. Компонент реалізовано як функціональний React Native компонент CalendarScreen, який включає у себе такі функціональні можливості:
2. Вибір дати та часу: інтеграція з компонентом DateTimePicker дозволяє зручно обирати як дату, так і час для запису.

					РП 08. 13 001. 00 ДП ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		27

3. Завантаження даних: при завантаженні компонента виконуються асинхронні функції для отримання списку клієнтів, послуг та вже наявних записів із локального сховища AsyncStorage.
4. Вибір клієнта та послуги: через випадуючі списки (Picker) майстер може обрати відповідного клієнта та потрібну послугу зі збережених даних.
5. Збереження нового запису: після вибору всіх параметрів, запис зберігається до локального сховища у форматі JSON.
6. Відображення записів: усі записи, збережені у AsyncStorage, відображаються у списку, що оновлюється в режимі реального часу.
7. Видалення записів: передбачена можливість видалення окремих записів за допомогою відповідної кнопки.

Центральною функцією компонента є saveAppointment, що реалізує логіку збереження нового запису клієнта. Вона забезпечує додавання нового об'єкта запису до локального сховища після перевірки наявності вибраного клієнта та послуги.

```
const saveAppointment = async () => {
  if (!selectedClient || !selectedService) return;

  const appointment = {
    date: date.toString(),
    client: selectedClient,
    service: selectedService
  };

  const updated = [...appointments, appointment];
  setAppointments(updated);
  await AsyncStorage.setItem('appointments', JSON.stringify(updated));
};
```

Функція виконує наступні дії:

1. Перевірка валідності введених даних: якщо клієнт або послуга не вибрані, функція завершиться достроково.
2. Формування об'єкта запису: створюється новий об'єкт, що містить дату, ПІБ клієнта та назву послуги.
3. Оновлення стану appointments: новий запис додається до масиву наявних записів.
4. Збереження у AsyncStorage: оновлений список записів перетворюється у JSON-формат та зберігається локально для забезпечення постійного доступу.

					<i>РП 08. 13 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		28

1.3.6 Створення екрану «Палітра кольорів»

Компонент «Палітра кольорів» розроблений для мобільного застосунку-помічника майстра манікюру з метою надання користувачу зручного інструменту для вибору та комбінування кольорів. Даний екран дозволяє майстру візуально експериментувати з палітрою, оцінювати сумісність обраних кольорів та отримувати рекомендації щодо їх гармонійності, що сприяє підвищенню якості послуг і задоволеності клієнтів. Інтерфейс компонента включає інтуїтивно зрозумілі елементи управління, які полегшують процес підбору відтінків навіть для початківців. Крім того, компонент підтримує збереження обраних палітр для подальшого використання, що дозволяє майстру швидко повторювати або адаптувати популярні комбінації кольорів у своїй роботі. Такий підхід допомагає зекономити час і зробити процес створення дизайну нігтів більш творчим і ефективним.

Основні функції компонента включають:

1. Вибір від 2 до 3 кольорів за допомогою інтерактивного кольорового селектора.
2. Можливість додавати або видаляти кольори у межах допустимої кількості.
3. Візуальний перегляд обраних кольорів у вигляді кольорових блоків.
4. Перевірка сумісності кольорів на основі аналізу їх тональних характеристик у кольоровому просторі HSL.
5. Вивід результату перевірки сумісності з поясненням типу гармонії (комплементарність, трикутна гармонія або відсутність гармонії).
6. Обробка некоректних або невалідних значень кольорів із відображенням відповідного повідомлення про помилку.

Основною логічною функцією компоненту є `checkCompatibility`, яка виконує перевірку сумісності обраних кольорів. Вона реалізована наступним чином:

```
const checkCompatibility = () => {
  if (colors.length < 2 || colors.length > 3) {
    setCheckResult('Виберіть 2 або 3 кольори');
    return;
  }

  try {
    const hues = colors.map(c => {
```

					<i>РП 08. 13 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		29

```

    if (!c.match(/^#([0-9a-f]{6})$/i)) throw new Error('Invalid color');
    const rgb = c.match(/\w\w/g).map(x => parseInt(x, 16));
    return rgbToHsl(...rgb)[0];
  });

  const angleDiff = (a, b) => {
    const diff = Math.abs(a - b) % 360;
    return diff > 180 ? 360 - diff : diff;
  };

  if (colors.length === 2) {
    const diff = angleDiff(hues[0], hues[1]);
    setCheckResult(diff > 150 && diff < 210 ? 'Кольори комплементарні' : 'Кольори
не дуже комплементарні');
  } else if (colors.length === 3) {
    const diff01 = angleDiff(hues[0], hues[1]);
    const diff12 = angleDiff(hues[1], hues[2]);
    const diff20 = angleDiff(hues[2], hues[0]);
    if (
      Math.abs(diff01 - 120) < 30 &&
      Math.abs(diff12 - 120) < 30 &&
      Math.abs(diff20 - 120) < 30
    ) setCheckResult('Кольори утворюють трикутну гармонію');
    else setCheckResult('Кольори не гармонійні по колу Ітана');
  }
} catch {
  setCheckResult('Помилка в кольорах. Введіть валідні кольори.');
```

Пояснення роботи функції:

1. Спершу виконується перевірка кількості обраних кольорів. Якщо вибрано менше двох або більше трьох – користувачу виводиться повідомлення з проханням вибрати коректну кількість.
2. Для кожного кольору здійснюється перевірка формату (шестнадцятковий RGB), далі конвертація кольору з RGB у відтінок (hue) у кольоровому просторі HSL за допомогою функції rgbToHsl.
3. Визначається різниця кутів (відтінків) між кольорами, з урахуванням циклічності колірної кола.
4. Якщо вибрано два кольори, перевіряється, чи їх різниця лежить у межах 150–210 градусів (комплементарність). Для трьох кольорів перевіряється близькість різниць між відтінками до 120 градусів із допустимою похибкою, що означає трикутну гармонію.
5. Результат перевірки записується у стан checkResult, який відображається на екрані.

На рисунку 1.9 зображено блок схему алгоритму перевірки кольорів.

					<i>РП 08. 13 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		30

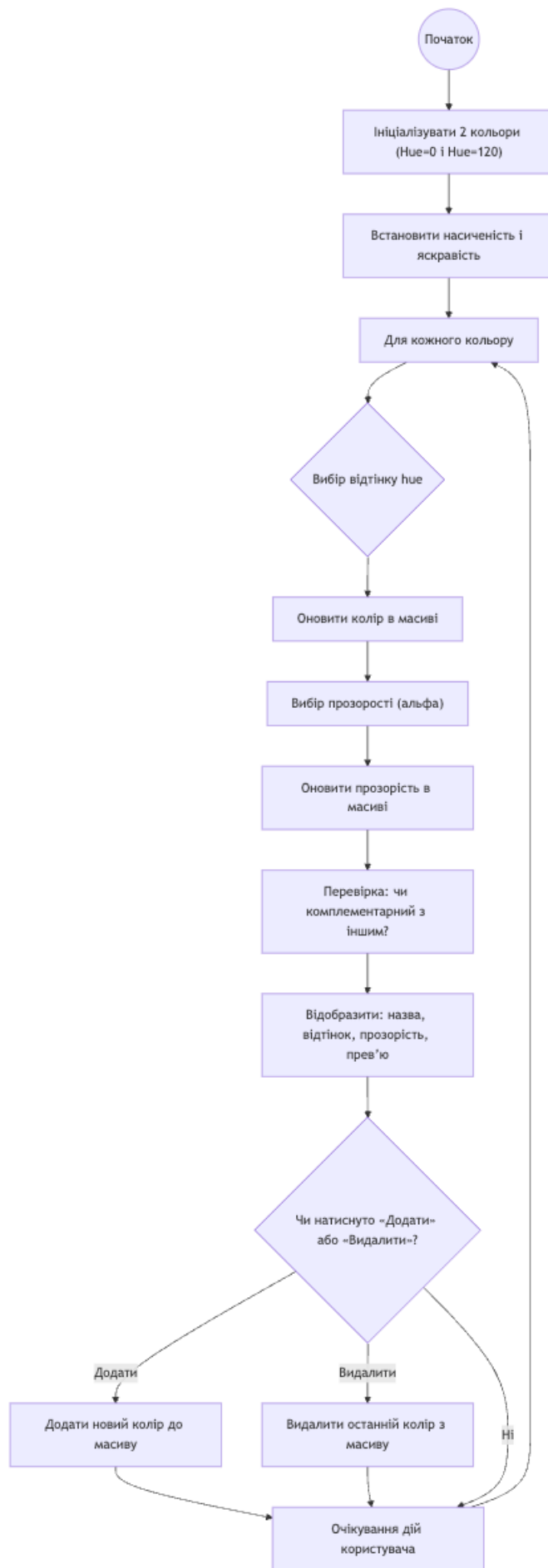


Рисунок 1.9. БСА перевірки кольорів

Зм.	Арк.	№ докум.	Підпис	Дата

РП 08. 13 001. 00 ДП ПЗ

Арк.

31

1.3.7 Створення екрану «Генератор зображень»

Компонент «Генератор зображень» розроблений для створення унікальних абстрактних зображень, які можуть слугувати елементами дизайну нігтьової пластини у мобільному застосунку-помічнику майстра манікюру. Цей екран дозволяє користувачу вибирати кольори та автоматично генерувати комбінації простих геометричних фігур (кола, прямокутники, лінії) у заданій кольоровій палітрі. Завдяки цьому майстер манікюру отримує інструмент для натхнення та швидкого створення оригінальних візерунків. Інтерфейс генератора має прості та зрозумілі налаштування, що дозволяють регулювати кількість, розмір, прозорість і розташування фігур для досягнення бажаного ефекту, що значно розширює можливості творчості користувача. Користувач може обирати різні стилі розміщення елементів, змінювати їхню інтенсивність та комбінувати відтінки, що дає змогу створювати як мінімалістичні, так і яскраві, насичені дизайни. Згенеровані зображення можна зберігати у внутрішній галереї застосунку, що дозволяє майстрам швидко повернутися до улюблених або популярних варіантів дизайну. Окрім того, компонент підтримує функцію експорту створених зображень для подальшого використання у портфоліо чи демонстрації клієнтам, що підвищує рівень професіоналізму і допомагає краще комунікувати з замовниками. Такий інструмент суттєво оптимізує процес підбору дизайну, роблячи його більш інтерактивним, цікавим та доступним навіть для користувачів без спеціальних навичок малювання. Загалом, «Генератор зображень» сприяє підвищенню креативності майстрів та розширенню їхніх можливостей у створенні індивідуальних стильних образів.

Основні функції компоненту включають:

1. Вибір трьох основних кольорів з фіксованого набору.
2. Автоматична генерація випадкової комбінації геометричних фігур із використанням обраних кольорів.
3. Можливість збереження створених малюнків у локальному списку.
4. Перегляд та вибір зі збережених зображень.
5. Повторна генерація нових композицій за допомогою кнопки «Згенерувати».

					<i>РП 08. 13 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		32

Інтерфейс користувача реалізовано за допомогою React Native, використовуючи компонент Svg для векторної графіки, що забезпечує чітке відображення зображень на різних розмірах екранів.

Головним функціональним компонентом є функція GeneratorScreen, яка реалізує логіку генератора. Вона управляє станами вибраних кольорів, поточних фігур, списком збережених малюнків, а також взаємодією з користувачем.

```
export default function GeneratorScreen() {
  const [color1, setColor1] = useState('#FF6B6B');
  const [color2, setColor2] = useState('#FFD93D');
  const [color3, setColor3] = useState('#6BCB77');
  const [shapes, setShapes] = useState(generateShapes([color1, color2, color3]));
  const [savedDrawings, setSavedDrawings] = useState([]);
  const [selectedSavedIndex, setSelectedSavedIndex] = useState(null);

  const regenerate = () => {
    const newShapes = generateShapes([color1, color2, color3]);
    setShapes(newShapes);
    setSelectedSavedIndex(null);
  };

  const save = () => {
    const newDrawing = generateShapes([color1, color2, color3]);
    setSavedDrawings([...savedDrawings, newDrawing]);
    Alert.alert('Збережено!', 'Абстракція нігтьової пластини збережена.');
```

```
    setSelectedSavedIndex(savedDrawings.length);
  };

  const onColorPick = (pickerIndex, color) => {
    if (pickerIndex === 1) setColor1(color);
    if (pickerIndex === 2) setColor2(color);
    if (pickerIndex === 3) setColor3(color);
    setSelectedSavedIndex(null);
  };

  return (
    <View style={styles.container}>
      { /* Відображення згенерованого малюнка, вибір кольорів, кнопки */ }
    </View>
  );
}
```

Пояснення роботи функції:

1. Хук useState зберігає поточні кольори (color1, color2, color3), масив поточних фігур (shapes), список збережених малюнків (savedDrawings) та індекс вибраного збереження (selectedSavedIndex).
2. Функції regenerate та save реалізують відповідно повторну генерацію та збереження поточного зображення. Функція onColorPick дозволяє змінювати вибрані кольори.

					<i>РП 08. 13 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		33

Таким чином, GeneratorScreen є ключовим компонентом, що забезпечує інтерактивність та гнучкість при створенні дизайнерських візерунків для нігтьової пластини в мобільному застосунку.

На рисунку 1.10 зображено блок схему алгоритму генерації кольорів.

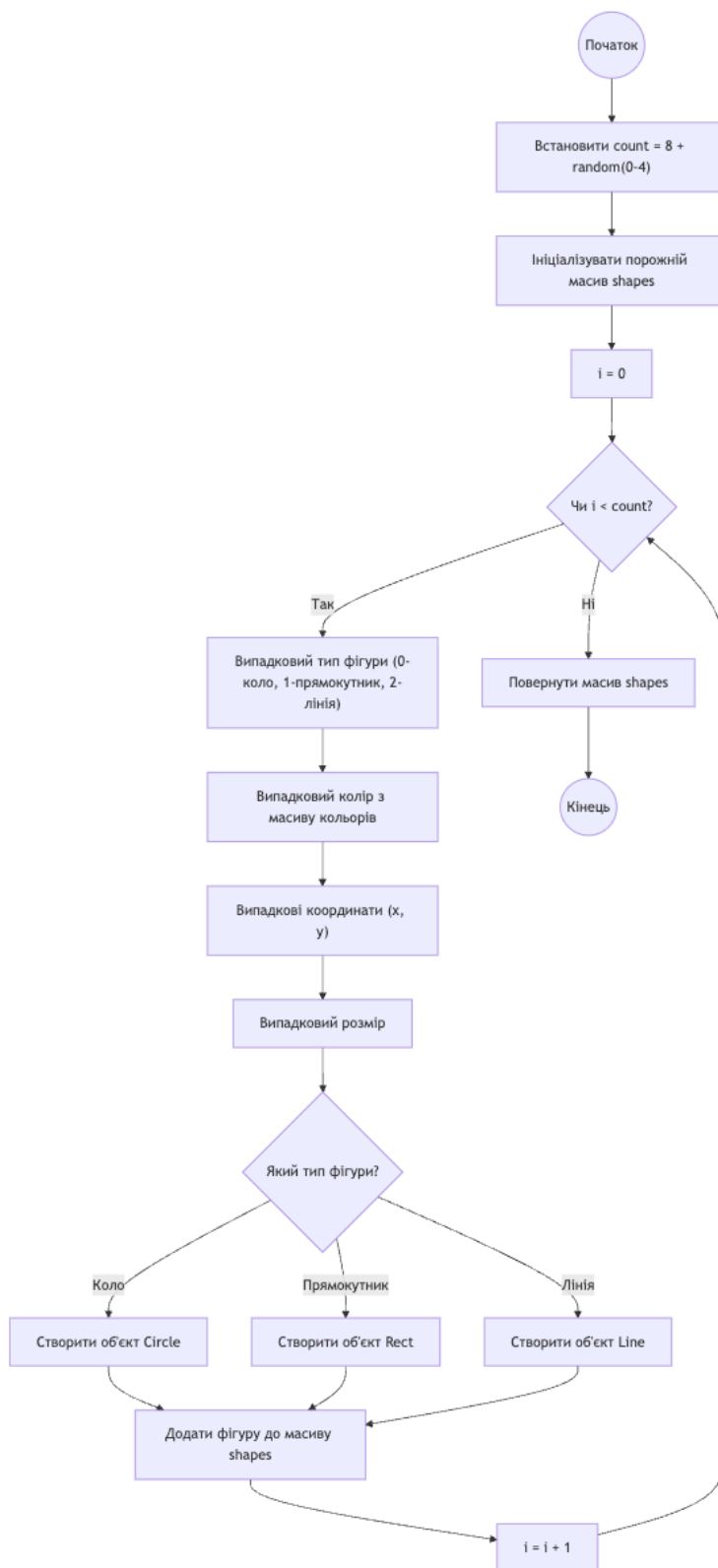


Рисунок 1.10. БСА генерації кольорів

Зм.	Арк.	№ докум.	Підпис	Дата

РП 08. 13 001. 00 ДП ПЗ

Арк.

34

1.3.8 Створення екрану «Галерея зображень»

Компонент «Галерея зображень» призначений для організації та управління фотографіями клієнтів, що дозволяє майстру зручно зберігати, переглядати та видаляти знімки робіт. Цей екран забезпечує можливість вибору клієнта зі списку, створення нових фотографій за допомогою камери пристрою та їх подальшого збереження в локальному сховищі застосунку. Таким чином, компонент сприяє ефективній систематизації фотоархіву майстра, підвищуючи зручність і швидкість доступу до інформації.

Компонент виконує такі основні функції:

1. Завантаження списку клієнтів: Отримання та відображення списку клієнтів із локального сховища (AsyncStorage) для вибору конкретного користувача.
2. Вибір клієнта: Користувач може обрати клієнта зі списку, що визначає, до якого профілю будуть додаватися нові фотографії.
3. Зйомка фотографій: Виклик камери пристрою з перевіркою відповідних дозволів (особливо актуально для Android), з можливістю робити фото для обраного клієнта.
4. Збереження фотографій: Локальне зберігання знімків у форматі URI з асоціацією до конкретного клієнта.
5. Перегляд галереї: Відображення списку збережених фотографій із вказівкою імені клієнта, якому належить фото.
6. Видалення фотографій: Можливість видаляти небажані знімки із підтвердженням дії користувача через діалогове вікно.

Основною функцією екрану «Галерея зображень» є функція `openCamera`, яка відповідає за відкриття камери для зйомки фото клієнта. Код функції наведено нижче:

```
const openCamera = async () => {
  if (!selectedClient) {
    Alert.alert('Виберіть клієнта', 'Будь ласка, виберіть клієнта зі списку перед зйомкою');
    return;
  }

  const hasPermission = await requestCameraPermission();
  if (!hasPermission) {
    Alert.alert('Дозвіл не надано', 'Без доступу до камери не можливо робити фото');
  }
}
```

					РП 08. 13 001. 00 ДП ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		35

```

    return;
  }

  launchCamera(
    {
      mediaType: 'photo',
      cameraType: 'back',
      saveToPhotos: false,
    },
    (response) => {
      if (response.didCancel) {
        console.log('Користувач скасував');
      } else if (response.errorCode) {
        Alert.alert('Помилка камери', response.errorMessage || 'Невідома помилка');
      } else {
        const uri = response.assets && response.assets[0].uri;
        if (uri) {
          const newPhotos = [...photos, { clientId: selectedClient.id, uri }];
          setPhotos(newPhotos);
          savePhotos(newPhotos);
        }
      }
    }
  );
};

```

Пояснення роботи функції:

1. Якщо клієнт не вибраний (!selectedClient), показується повідомлення з проханням вибрати клієнта, і функція завершується.
2. Виконується асинхронний запит дозволу через requestCameraPermission(). Якщо дозвіл не отримано, показується повідомлення про відсутність доступу, і функція завершується.
3. Викликається launchCamera для зйомки фото. Якщо користувач скасовує дію – нічого не відбувається. Якщо є помилка – виводиться повідомлення. Якщо фото зроблено успішно, URI додається до масиву photos, оновлюється стан та локальне сховище.

1.4 Тестування веб-застосунку

1.4.1 Обґрунтування виду тестування

У процесі розробки мобільного застосунку-помічника для майстра манікюру важливою складовою стало забезпечення його коректної та стабільної роботи. З огляду на особливості програмної реалізації застосунку, було прийнято рішення використовувати мануальне (ручне) тестування як основний вид перевірки

					<i>РП 08. 13 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		36

функціональності.

Розроблений застосунок реалізовано з використанням бібліотеки React Native, що дає змогу створювати кросплатформені мобільні інтерфейси з використанням JavaScript та React. Основна навігація реалізована за допомогою бібліотеки @react-navigation, зокрема модуля createBottomTabNavigator, який забезпечує інтуїтивно зрозумілу нижню панель навігації між основними екранами застосунку: «Прайс», «Клієнти», «Календар», «Палітра», «Генератор» та «Галерея».

Кожен із цих екранів представлений окремим функціональним компонентом, імпортованим у головний файл застосунку. З огляду на орієнтацію проекту на індивідуальне користування (один майстер – один застосунок), а також відсутність складної бізнес-логіки або зовнішніх API-зв'язків на ранньому етапі, доцільно було обрати саме мануальне тестування, яке дозволяє безпосередньо перевірити інтерфейс, взаємодію з елементами керування та загальну зручність використання.

Мануальне тестування дало змогу:

1. Перевірити коректність навігації між екранами;
2. Візуально оцінити відображення емодзі-іконок відповідно до назви вкладки;
3. Перевірити відповідність дизайну очікуванням цільової аудиторії (зокрема використання кольору deerpink, що відповідає естетиці б'юті-сфери);
4. Виявити помилки або неочікувану поведінку на етапі рендерингу компонентів на реальних пристроях.

Автоматизоване тестування на цьому етапі не було запроваджено, оскільки застосунок перебуває у стадії первинної розробки з частими змінами інтерфейсу, і витрати на написання та підтримку автоматизованих сценаріїв наразі є недоцільними. У майбутньому, з розширенням функціоналу, планується інтеграція автоматизованих тестів (наприклад, з використанням фреймворку Detox) для перевірки стабільності при оновленнях.

Отже, вибір мануального тестування повністю виправданий як на етапі розробки MVP (Minimum Viable Product), так і в контексті проекту, орієнтованого



					<i>РП 08. 13 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		37

на конкретного індивідуального користувача з чіткими очікуваннями щодо функціональності й зручності.

1.4.2 Тестування екрану «Прайс лист»



Основна мета екрану "Прайс лист" є забезпечити зручний інтерфейс для зберігання, редагування, групування та виведення переліку послуг, які надає майстер. Екран дозволяє швидко орієнтуватися у власному прайсі, що особливо актуально під час консультацій із клієнтами чи в процесі планування робочого часу. Збереження даних відбувається локально, що забезпечує офлайн-доступ до інформації, навіть без підключення до Інтернету. Інтерфейс розроблено з урахуванням простоти та інтуїтивності, що дозволяє легко додавати нові послуги, редагувати їхні ціни та структурувати прайс за категоріями. Крім того, можливість групування послуг сприяє кращій організації інформації і дозволяє швидше знаходити потрібні позиції під час роботи. Такий підхід значно підвищує ефективність управління послугами та сприяє комфортній взаємодії майстра з клієнтами.

Екран реалізує такі функціональні можливості:

1. Додавання послуги: користувач заповнює поля (назва, ціна, тривалість, група) та натискає кнопку для додавання. Послуга зберігається у AsyncStorage.
2. Редагування існуючої послуги: натискання кнопки редагування  переносить дані з відповідного запису у форму, де користувач може внести зміни.
3. Видалення послуги: натискання на кнопку  видаляє відповідний запис із локального сховища.
4. Групування записів: послуги автоматично групуються за категорією (напр., "Манікюр", "Педикюр") за допомогою компонента SectionList.
5. Збереження даних: використання AsyncStorage дозволяє уникнути втрати інформації при перезапуску застосунку.

Тест кейси ручного тестування:

					РП 08. 13 001. 00 ДП ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		38

1. Додавання нової послуги
Умови: усі поля заповнено коректно
Очікуваний результат: послуга додається до списку, форма очищується, дані зберігаються у сховищі
2. Відмова від додавання при незаповнених полях
Умови: одне або декілька полів залишено порожніми
Очікуваний результат: дія не виконується, попередження не показується, але помилка не виникає
3. Редагування існуючої послуги
Умови: натиснуто кнопку  біля певного запису
Очікуваний результат: дані переносяться у форму, після збереження запис оновлюється
4. Видалення послуги
Умови: натиснуто кнопку  біля послуги
Очікуваний результат: запис зникає зі списку, зміни зберігаються в AsyncStorage
5. Групування послуг
Умови: додано послуги з різними значеннями поля "група"
Очікуваний результат: список правильно розділено на секції, заголовки відображаються коректно
6. Перезапуск застосунку
Умови: застосунок перезапущено після збереження декількох записів
Очікуваний результат: список послуг повністю відновлюється з AsyncStorage

Проведене ручне тестування підтвердило стабільну та передбачувану роботу екрану "Прайс лист", що є важливою складовою загального функціоналу мобільного застосунку.

На рисунку 1.11 зображено екран «Прайс лист».

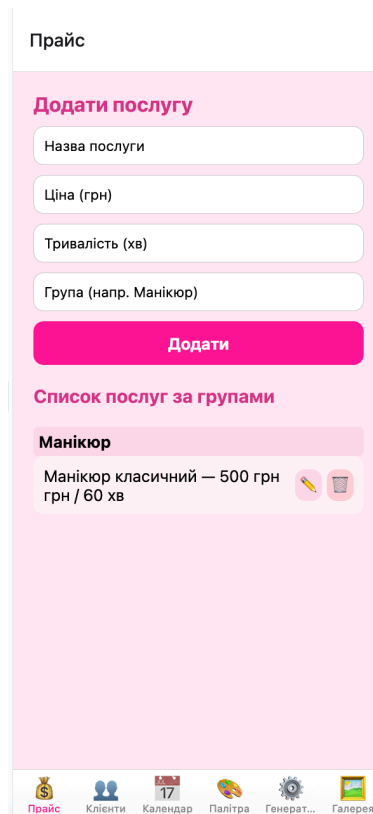


Рисунок 1.11. Екран «Прайс лист»

1.4.3 Тестування екрану «Клієнтська база»

Екран «Клієнтська база» слугує для збереження, перегляду, редагування та видалення інформації про клієнтів майстра манікюру. Це дозволяє зручно організувати контактну базу з можливістю додавання індивідуальних нотаток. Уся інформація зберігається у локальному сховищі пристрою (AsyncStorage), що забезпечує її доступність навіть у режимі офлайн.

Екран реалізує такі функціональні можливості:

1. Додавати нових клієнтів (ім'я, прізвище, номер телефону, нотатки);
2. Редагувати дані існуючих клієнтів;
3. Видаляти клієнтів;
4. Переглядати список усіх збережених клієнтів;
5. Зберігати інформацію у AsyncStorage.

Тест кейси ручного тестування:

1. Додавання нового клієнта
 Умови: Заповнено всі поля (Ім'я, Прізвище, Телефон), натиснуто кнопку «Додати»

					РП 08. 13 001. 00 ДП ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		40


Очікуваний результат: Клієнт з'являється у списку, дані зберігаються в AsyncStorage

2. Додавання з порожніми обов'язковими полями

Умови: Одне або кілька обов'язкових полів (Ім'я, Прізвище, Телефон) залишено порожніми, натиснуто кнопку «Додати»

Очікуваний результат: Клієнт не додається, список не змінюється

3. Оновлення існуючого клієнта

Умови: Натиснуто  біля клієнта, внесено зміни у поля, натиснуто кнопку «Оновити»

Очікуваний результат: Дані клієнта змінюються у списку, оновлені дані зберігаються в AsyncStorage

4. Видалення клієнта

Умови: Натиснуто кнопку  біля клієнта

Очікуваний результат: Запис зникає зі списку, зміни зберігаються в AsyncStorage

5. Завантаження даних після перезапуску застосунку

Умови: У сховищі є записи, застосунок перезапущено

Очікуваний результат: Клієнти автоматично з'являються у списку при запуску

6. Додавання нотатки до клієнта

Умови: У полі «Нотатки» введено текст, натиснуто кнопку «Додати»

Очікуваний результат: Клієнт з нотаткою додається до списку, нотатка відображається нижче контактної інформації

7. Відміна редагування через натискання кнопки «Додати»

Умови: Натиснуто , змінено дані, натомість натиснуто кнопку «Додати»

Очікуваний результат: Додається новий запис із новими даними, попередній клієнт не змінюється

8. Очищення полів після додавання

Умови: Клієнта успішно додано

Очікуваний результат: Усі поля введення автоматично очищуються

					<i>РП 08. 13 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		41

9. Додавання кількох клієнтів

Умови: Послідовно додано 5 і більше клієнтів

Очікуваний результат: Усі клієнти зберігаються у списку в тій черговості, в якій були додані

10. Додавання некоректного номера телефону

Умови: У полі «Телефон» введено текст замість цифр, натиснуто «Додати»

Очікуваний результат: Клієнт додається або не додається залежно від реалізованої валідації (у протестованій версії – додається)

Проведене ручне тестування підтвердило стабільну та передбачувану роботу екрану "Клієнтська база", що є важливою складовою загального функціоналу мобільного застосунку.

На рисунку 1.12 зображено екран «Клієнтська база».

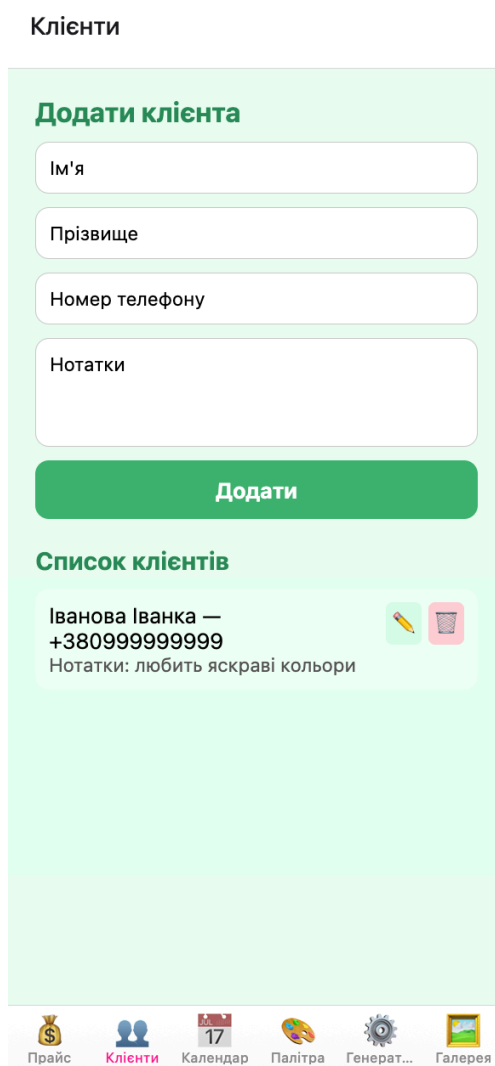


Рисунок 1.12. Екран «Клієнтська база»

					РП 08. 13 001. 00 ДП ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		42


1.4.4 Тестування екрану «Запис» (Календар)

Екран «Запис» є ключовим функціональним елементом мобільного застосунку-помічника майстра манікюру. Його основним призначенням є надання можливості користувачу здійснювати запис клієнтів на прийом, фіксуючи дату, час, обраного клієнта та відповідну послугу. Даний екран також дозволяє переглядати та видаляти раніше збережені записи.

Екран реалізує такі функціональні можливості:

1. Вибір дати та часу запису за допомогою DatePicker;
2. Вибір клієнта із випадаючого списку (Picker) на основі збережених даних у AsyncStorage;
3. Вибір послуги, яку буде надано клієнту;
4. Збереження запису до локального сховища (AsyncStorage);
5. Відображення списку записів у форматі списку з даними про дату, клієнта, номер телефону та послугу;
6. Можливість видалення запису через інтерактивну кнопку.

Тест кейси ручного тестування:

1. Додавання **нового** запису
Умови: Обрано клієнта зі списку, обрано послугу, встановлено дату та час, натиснуто кнопку "Зберегти запис".
Очікуваний результат: Новий запис відображається у списку візитів, дані зберігаються в AsyncStorage.
2. Видалення **існуючого** запису
Умови: У списку записів присутній хоча б один елемент, натиснуто кнопку з іконкою .
Очікуваний результат: Запис зникає зі списку, відповідна інформація видаляється з AsyncStorage.
3. Перевірка **вибору** клієнта
Умови: Відкрито екран запису, зі списку клієнтів обрано одного зі збережених клієнтів.
Очікуваний результат: Значення змінюється, зберігається у стані

					РП 08. 13 001. 00 ДП ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		43

компонента, використовується під час створення запису.

4. Вибір дати та часу
Умови: Натиснуто кнопку "Вибрати дату" або "Вибрати час", у діалоговому вікні обрано нове значення.

Очікуваний результат: Вибране значення з'являється на екрані, зберігається у змінній date.

5. Збереження даних між сесіями

Умови: Створено один або кілька записів, застосунок перезапущено.

Очікуваний результат: Після повторного запуску всі записи завантажуються з AsyncStorage та відображаються у списку.

Проведене ручне тестування підтвердило стабільну та передбачувану роботу екрану "Запис (Календар)", що є важливою складовою загального функціоналу мобільного застосунку.

На рисунку 1.13 зображено екран «Запис (календар)».

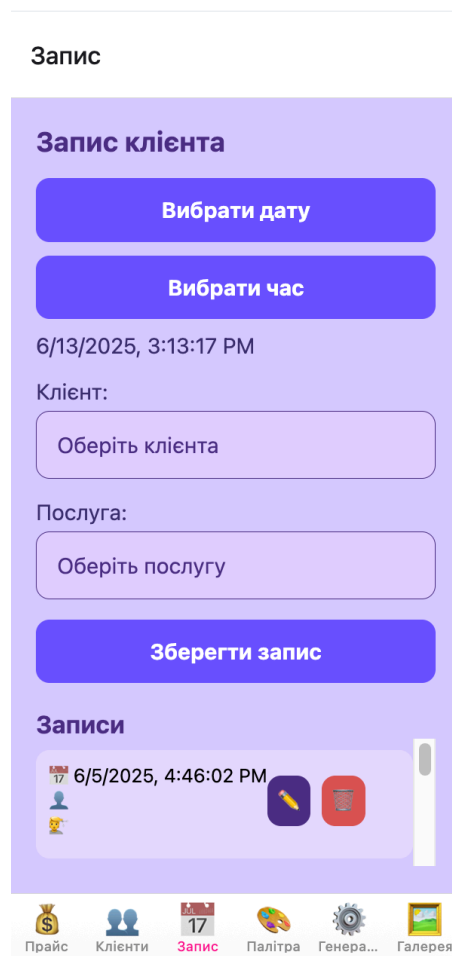


Рисунок 1.13. Екран «Запис (Календар)»

					<i>РП 08. 13 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		44

1.4.5 Тестування екрану «Палітра кольорів»

Екран «Палітра кольорів» призначений для взаємодії користувача з кольоровою палітрою з метою вибору та налаштування 2 або 3 кольорів із можливістю зміни відтінку (Hue) та прозорості (Alpha) кожного з них. Він дозволяє візуально визначати комплементарні кольори та переглядати обрані кольори у модальному вікні. Даний екран може використовуватися у дизайнерських інтерфейсах, ігрових додатках або системах налаштування тем.

Екран реалізує такі функціональні можливості:

1. Вибір відтінку для кожного кольору зі шкали (кожні 10° HSL);
2. Зміна прозорості (Alpha) кожного кольору з кроком 0.05;
3. Визначення та візуальне позначення комплементарних кольорів;
4. Відображення обраного кольору у модальному вікні у форматі RGBA;
5. Додавання нового кольору (до максимум 3 кольорів);
6. Видалення останнього кольору (мінімум 2 кольори).

Тест кейси ручного тестування:

1. Вибір відтінку для кольору

Умови: Користувач натискає на коло-відтінок (Hue) під відповідним кольором.

Очікуваний результат: Відтінок оновлюється, фон кольору змінюється відповідно.

2. Зміна прозорості кольору

Умови: Натиснуто кнопку + або - для збільшення чи зменшення прозорості кольору.

Очікуваний результат: Значення прозорості оновлюється на ± 0.05 , не виходячи за межі [0; 1].

3. Додавання нового кольору

Умови: Кількість кольорів менше 3, натиснуто кнопку "Додати колір".

Очікуваний результат: Додається новий колір із початковими параметрами HSL(0,100%,50%) і alpha=1.

4. Видалення існуючого запису

					РП 08. 13 001. 00 ДП ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		45

Умови: У списку кольорів щонайменше 3 елементи, натиснуто кнопку "Видалити колір".

Очікуваний результат: Останній колір зникає зі списку.

5. Відкриття модального вікна кольору

Умови: Натиснуто на колірний блок.

Очікуваний результат: Відкривається вікно з кольором у форматі RGBA.

6. Визначення комплементарності

Умови: Обрано два кольори з різницею відтінку близько 180°.

Очікуваний результат: Надписи відповідних кольорів стають червоними.

Екран «Палітра кольорів» успішно реалізує інтерфейс для інтерактивного налаштування кольорів із підтримкою базових функцій зміни відтінку, прозорості, визначення комплементарності, а також додавання й видалення кольорових елементів. Проведене тестування засвідчило стабільну роботу всіх функціональних елементів, зокрема контроль меж для кількості кольорів та прозорості, а також коректну обробку подій користувача.

На рисунку 1.14 зображено екран «Палітра кольорів».



Рисунок 1.14. Екран «Палітра кольорів»

					<i>РП 08. 13 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		46

1.4.6 Тестування екрану «Генератор зображень»

Екран «Генератор зображень» реалізує функціональність створення абстрактних SVG-зображень, які імітують структуру та вигляд нігтьової пластини. Він надає користувачу інструменти для вибору кольорів, генерації зображення, збереження створених варіантів та перегляду історії збережених шаблонів. Застосування сховища AsyncStorage забезпечує збереження даних між сеансами роботи з додатком.

Екран реалізує такі функціональні можливості:

1. Вибір трьох кольорів для генерації зображення.
2. Генерація нового зображення на основі обраних кольорів.
3. Збереження згенерованого зображення у список історії.
4. Відображення списку збережених зображень.
5. Завантаження збережених зображень з AsyncStorage під час відкриття екрану.
6. Видалення конкретного зображення з історії та зі сховища.
7. Вибір збереженого зображення для перегляду.

Тест кейси ручного тестування:

1. Генерація нового зображення

Умови: Обрано кольори; натиснуто кнопку «Згенерувати».

Очікуваний результат: В області зображення з'являється нова композиція SVG-фігур.

2. Збереження зображення

Умови: Згенеровано нове зображення; натиснуто кнопку «Зберегти».

Очікуваний результат: Зображення додається до списку збережених; виводиться сповіщення «Збережено!».

3. Перегляд збереженого зображення

Умови: У списку збережень є хоча б один запис; натиснуто на збережене зображення.

Очікуваний результат: Обране зображення виводиться в основній області перегляду.

					<i>РП 08. 13 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		47

4. Видалення існуючого запису

Умови: У списку збережень є хоча б один запис; натиснуто кнопку з іконкою



Очікуваний результат: Вибране зображення зникає зі списку; відповідна інформація видаляється з AsyncStorage.

5. Завантаження збережень при відкритті екрану

Умови: В AsyncStorage присутні збережені записи.

Очікуваний результат: При завантаженні екрану всі збережені зображення відображаються у відповідному розділі.

Проведене тестування підтвердило коректність роботи основних функцій екрану «Генератор зображень». Усі ключові сценарії взаємодії користувача – від створення зображень до їхнього збереження та видалення – виконуються відповідно до очікуваної логіки. Застосування AsyncStorage забезпечує збереження стану між сесіями, що відповідає сучасним вимогам до мобільних застосунків з персоналізованим інтерфейсом.

На рисунку 1.15 зображено екран «Генератор зображень».

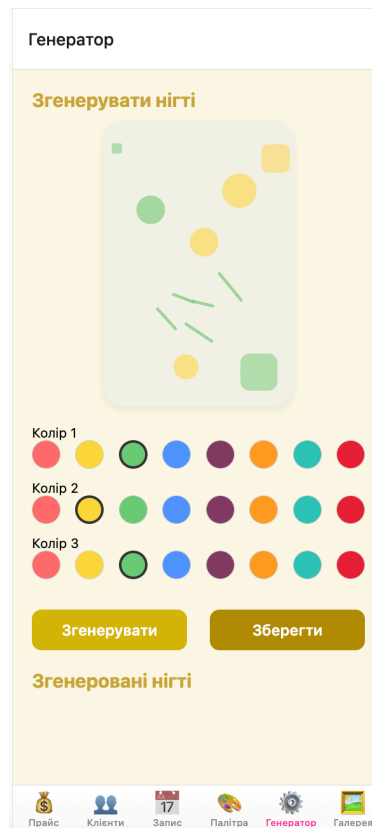


Рисунок 1.15. Екран «Генератор зображень»

					<i>РП 08. 13 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		48

1.4.7 Тестування екрану «Галерея зображень»

Екран «Галерея зображень» призначений для перегляду та управління фотографіями клієнтів у мобільному застосунку. Його головна мета – забезпечити користувачу можливість вибору клієнта, створення фотографій, асоційованих із цим клієнтом, та перегляду й видалення збережених зображень. Дані фотографій та інформація про клієнтів зберігаються локально за допомогою технології AsyncStorage.

Екран реалізує такі функціональні можливості:

1. Вибір клієнта зі списку;
2. Відображення імені обраного клієнта;
3. Створення фото за допомогою камери пристрою;
4. Збереження фото, прив'язаного до конкретного клієнта;
5. Перегляд списку фото з зазначенням клієнта;
6. Видалення фото зі списку;
7. Збереження та завантаження даних з AsyncStorage;
8. Обробка ситуацій, коли клієнти або фото відсутні.

Тест кейси ручного тестування:

1. Вибір клієнта зі списку

Умови: У списку клієнтів є хоча б один клієнт.

Дії: Натиснуто на кнопку «Оберіть клієнта», зі списку вибрано елемент.

Очікуваний результат: Ім'я обраного клієнта відображається на кнопці вибору.

2. Створення фото без вибору клієнта

Умови: Жоден клієнт не обраний.

Дії: Натиснуто кнопку «Зробити фото».

Очікуваний результат: Виводиться сповіщення з текстом «Будь ласка, виберіть клієнта перед зйомкою».

3. Створення фото для обраного клієнта


Умови: Клієнт обраний, дозволено доступ до камери, користувач зробив фото.

					<i>РП 08. 13 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		49

Дії: Натиснуто кнопку «Зробити фото».

Очікуваний результат: Фото додається до списку, зберігається в AsyncStorage.

4. Видалення існуючого запису

Умови: У списку фото присутній хоча б один запис, натиснуто кнопку з іконкою .

Очікуваний результат: Фото зникає зі списку, відповідна інформація видаляється з AsyncStorage.

5. Перевірка порожнього стану

Умови: У сховищі немає фото.

Дії: Екран відкрито.

Очікуваний результат: Виводиться повідомлення «Фото ще не зроблено».

Екран «Галерея зображень» протестовано відповідно до основних сценаріїв взаємодії. Основні функції, включно з вибором клієнта, створенням та видаленням фото, працюють стабільно та відповідно до очікувань. Обробка помилок, таких як відсутність обраного клієнта чи порожні списки, реалізована коректно. Таким чином, інтерфейс забезпечує надійне та інтуїтивно зрозуміле користування фотогалереєю.

На рисунку 1.15 зображено екран «Галерея зображень».

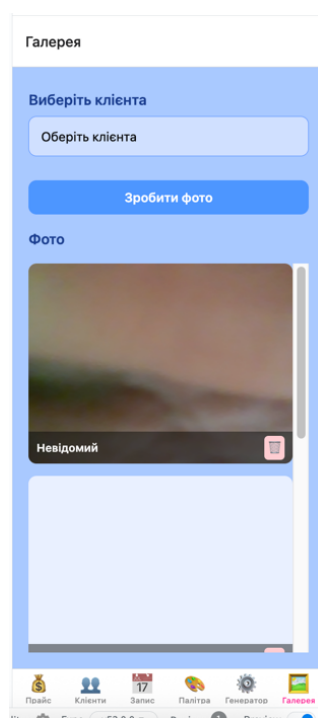


Рисунок 1.16. Екран «Генератор зображень»

					<i>РП 08. 13 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		50

2 ЕКОНОМІЧНИЙ РОЗДІЛ

2.1 Резюме

У дипломному проєкті розроблено мобільний застосунок-помічник для майстра манікюру, що сприяє ефективній організації робочого процесу, дозволяє керувати прайсом, клієнтами, записами та творчими матеріалами. Якість застосунку оцінюється за критеріями зручності використання, функціональності та практичної цінності в професійній діяльності.

2.2 Визначення трудомісткості розробки ПЗ

Тривалість розробки програмного продукту залежить від його обсягу, складності, кваліфікації розробників і встановлених ринком термінів. Метод структурної аналогії дозволяє оцінити обсяг у тисячах умовних машинних команд на основі подібного програмного забезпечення.

Табл. 2.1 містить аналоги ПЗ з подібними функціями; обраний варіант виділено сірим.

Таблиця 2.1. Каталог аналогів

Найменування ПП	Обсяг функції ПП – V_0 , умовних. машинних командах
1. ПП автоматизації засобів по каталогу	680 – 7000
2. ПП автоматизованих розрахунків	1300 – 8600
3. ПП введення інформації	1060 – 5750

Після вибору аналога з обсягом V_0 (умовні машинні команди), трудомісткість визначається за табл. 2.2.

Таблиця 2.2. Обсяг ПП

Обсяг ПП, тис.умов.машинних команд	Норма часу, люд/год
1.00	229
2.00	244
3.00	262
4.00	283
5.00	306
6.00	330
7.00	357

Обсяг ПП, тис.умов.машинних команд	Норма часу, люд/год
8.00	385
9.00	414
10.00	445

На основі отриманого значення за довідником визначають укрупнену норму часу, скориговану коефіцієнтом $K_k = 0,7-0,8$ для умов розробки на комп'ютері:

$$T_{ap} = 306 \times 0,8 = 244,8 \text{ (люд/годин)} \quad (2.1)$$

Трудомісткість визначається для кожного етапу окремо, з урахуванням складності, новизни та використання стандартних модулів, за відповідними формулами:

$$T_{T3} = T^a p \times L_1 \times K_H \quad (2.2)$$

$$T_{TP} = T^a p \times L_2 \times K_H \quad (2.3)$$

$$T_{RP} = T^a p \times L_3 \times K_H \times K_T \quad (2.4)$$

Для розрахунку використовуються такі коефіцієнти:

- L_i – частка i -го етапу (табл. 2.3);
- K_H – коефіцієнт новизни (табл. 2.4);
- K_T – коефіцієнт використання типових програм (табл. 2.5).

Наш варіант виділено сірим.

Таблиця 2.3. Питомі коефіцієнти трудомісткості стадії у загальній трудомісткості розробки ПП

Код стадії	Ступінь новизни		
	А	Б	В
ТЗ (L_1)	0,15	0,12	0,12
ТП (L_2)	0,16	0,15	0,11
РП (L_3)	0,55	0,58	0,61

Таблиця 2.4. Значення коефіцієнта новизни

Код ступеня новизни	Ступінь новизни	Значення K_H
А	Принципово новий ПП	1,75 – 1,2
Б	ПП – розвиток визначеного параметричного ряду	1,0 – 0,8

Код ступеня новизни	Ступінь новизни	Значення K_n
В	ПП, що має аналог	0,7

Таблиця 2.5. Значення коефіцієнта використання типових програм

Ступінь охоплення реалізованих функцій розробленого ПП типовими програмами, %	Значення K_T
60 і вище	0,6
40-60	0,7
20-40	0,8
До 20	0,9

Тепер розраховуємо трудомісткість для всіх етапів і зводимо у табл. 2.6:

Трудомісткість технічного завдання:

$$T_{tz} = Ta * L_1 * K_n = 244,8 * 0,12 * 0,8 = 23,50 \text{ (люд/годин)} \quad (2.2)$$

Трудомісткість розробки технічного проєкту:

$$T_{tp} = Ta * L_2 * K_n = 244,8 * 0,15 * 0,8 = 29,37 \text{ (люд/годин)} \quad (2.3)$$

Трудомісткість розробки робочого проєкту:

$$T_{rp} = Ta * L_3 * K_n * K_m = 244,8 * 0,58 * 0,8 * 0,7 = 79,51 \text{ (люд/годин)} \quad (2.4)$$

Для розрахунків визначили обсяг документації по етапах:

- технічне завдання $N_{tz}=2$ (стор);
- розробка ТП $N_{tp}=45$ (стор);
- розробка робочого проєкту $N_{rp}=4$ (стор);
- пояснювальна записка відповідно $N_{pz}=9$ (стор).

Таблиця 2.6. Розрахунок трудомісткості ПП

Найменування етапів	Розрахунок, годин		
1.ТЗ	$T_{РТЗ}=23,50$	$T_{кк}=0,7*N_{ТЗ}=0,7*2=1,4$	$T_{нк}=0,15*N_{ТЗ}=0,15*2=0,30$
2.Розробка ТП	$T_{РТП}=29,37$	$T_{кк}=0,7*N_{ТП}=0,7*45=31,5$	$T_{нк}=0,15*N_{ТП}=0,15*45=6,75$
3.Розробка РП	$T_{Ррп}=79,51$	$T_{кк}=0,7*N_{рп}=0,7*9=6,3$	$T_{нк}=0,15*N_{рп}=0,15*9=1,35$
4.Розробка ПЗ	$T_{Пз}=1,5*N_{Пз}=1,5*9=13,5$	$T_{кк}=0,7*N_{Тз}=0,7*9=6,3$	$T_{нк}=0,15*N_{Пз}=0,15*9=1,35$

Усього, в т.ч.:	201,07		
- на розробку	Тр=145,82		
- контроль керівника		Ткк=45,5	
- нормоконтроль			Тнк=9,75

2.3 Розрахунок ціни програмного продукту

Розраховуємо основну зарплату виконавців, матеріальні та загальні витрати на розробку ПП. Зарплата наведена в табл. 2.7. З 1 січня 2025 року мінімальна місячна зарплата – 8000 грн, погодинна ставка – 48 грн (згідно зі ст. 8 Закону про Держбюджет України).

Таблиця 2.7. Розрахунок основної заробітної плати виконавців

Найменування робіт	Трудомісткість робіт, години	Погодинна тарифна ставка, грн.	Розрахунок, грн.
1.Розробка ПП	145,82	60,00	8749,20
2.Контроль керівника	45,5	120,00	5460,00
3.Нормоконтроль	9,75	120,00	1170,00
Усього	-	-	Зо= 15379,20

Розраховуємо матеріальні витрати на розробку ПП та наведемо їх у табл. 2.8.

Таблиця 2.8. Розрахунок матеріальних витрат на розробку

Найменування матеріальних витрат	Тип, модель	Кількість	Ціна одиниці, грн.	Вартість, грн.
Папір	Лист А4	70	5.0	350,0
Разом	-	-	-	$V_{Mi}=350,0$
Транспортно – заготівельні Витрати (10%)				40,00
Усього				$V_M=V_{Mi}+V_{тр_з}= 390,00$

За отриманими даними складена калькуляція планової собівартості ПП, наведена в табл. 2.9.

					РП 08. 13 002. 00 ДП ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		54

Таблиця 2.9. Розрахунок статей витрат планової собівартості

Стаття витрат	Значення, грн.	Формула розрахунку
1. Матеріали	390,00	V_M (див. табл. 2.8)
2. Основна заробітна плата	15379,20	Z_o (див. табл. 2.7)
3. Додаткова заробітна плата	1537,92	$Z_d = 0.1 * Z_o = 15379,20 * 0,1$
4. Відрахування до єдиного фонду соціального внеску	3721,77	$V_{с.с.в.} = 0.22 * (Z_o + Z_d) = 0,22 * (15379,20 + 1537,92)$
5. Накладні витрати	6251,68	$V_{нак.} = 0.4 * Z_o = 0.4 * 15379,20$
6. Повна собівартість	27280,57	$C_{пов} = V_M + Z_o + Z_d + V_{с.с.в.} + V_{нак.} = 390,00 + 15379,20 + 1537,92 + 3721,77 + 6251,68$

Розмір прибутку розраховується за формулою:

$$P = (C_n * P) / 100 = (27280,57 * 10) / 100 = 2728,06 \text{ грн.} \quad (2.5)$$

Де p – плановий рівень рентабельності (10-15%).

Оптова ціна розраховується за формулою:

$$C_o = C_n + P = 27280,57 + 2728,06 = 30008,63 \text{ грн.} \quad (2.6)$$

За отриманими даними, ціна реалізації ПП за формулою становить:

$$C_p = C_o + ПДВ = 30008,63 + 30008,63 * 0.2 = 36010,35 \text{ грн.} \quad (2.7)$$

3 РОЗДІЛ ОХОРОНИ ПРАЦІ ТА ТЕХНІКИ БЕЗПЕКИ

3.1 Основні положення

В Україні значна увага приділяється забезпеченню охорони життя і здоров'я громадян, а також створенню безпечних умов праці роботодавцями та керівниками установ і організацій. Водночас кількість нещасних випадків на виробництві залишається надто високою. Покращення умов праці та охорона здоров'я є важливим чинником підвищення матеріального й культурного рівня життя населення, що сприяє зростанню якості й продуктивності праці, покращенню соціально-економічних показників виробництва та зниженню витрат, пов'язаних із травматизмом, професійними захворюваннями і аваріями.

У цьому розділі дипломного проекту розглядаються питання охорони праці програміста.

3.2 Небезпечні та шкідливі фактори в роботі програміста

Забезпечення безпечних і здорових умов праці значною мірою залежить від коректної оцінки небезпечних та шкідливих факторів виробничого середовища. Аналогічні зміни в стані організму людини можуть спричиняти різні фактори. До них належать умови виробничого середовища (підвищений рівень шуму, висока температура, недостатнє або відсутнє освітлення робочої зони, електричний струм, статична електрика тощо), а також надмірні фізичні та розумові навантаження, нервово-емоційне напруження і різноманітні їх комбінації.

3.3 Вимоги до виробничого середовища

У сучасних умовах кожен працівник повинен постійно підтримувати високий рівень фізичної, психологічної та професійної підготовки, а також запобігати травмам і професійним захворюванням. Для оцінки можливого впливу виробничих факторів на здоров'я користувачів відеодисплейних терміналів важливі певні якісні характеристики робочого середовища. Це середовище у приміщеннях (офісах) здебільшого визначається фізичними параметрами, такими як температура, вологість і електричний опір підлоги. Фізико-хімічні показники

					<i>РП 08. 13 003. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		56

містять дані про концентрацію іонів, різноманітних забруднювачів у повітрі, а також інші якісні характеристики середовища.

3.1 Мікроклімат

Висока температура повітря негативно впливає на функціональний стан людини. Хоча теплова генерація дисплеєм стає критичною лише у найспекотніші періоди року, створювати комфортні теплові умови слід постійно. Оптимальні та допустимі параметри мікроклімату в приміщеннях мають враховувати особливості технологічного процесу при роботі з комп'ютерами. Згідно з чинними нормативами України (ДСН 3.3.6.042-99 «Державні санітарні норми мікроклімату виробничих приміщень») у холодну пору року температура повітря, швидкість руху повітря та відносна вологість повинні становити: 22-24°C; 0,1 м/с; 40-60 %. Температура може варіюватися від 21 до 25°C за збереження інших параметрів мікроклімату. В теплу пору року ці показники мають бути: 23-25°C; 0,1-0,2 м/с; 40-60 %. Оптимальним рівнем аероіонізації в зоні дихання користувача вважається вміст легких аерофонів обох знаків у кількості від 150 до 5000 на 1 см³ повітря. Нормалізацію складу повітря робочої зони забезпечують примусова вентиляція, захисні екрани з заземленням та використання іонізаторів.

3.2 Шум

Шум негативно впливає на стан користувача, особливо при тривалому впливі. У людей, чия діяльність пов'язана з обробкою інформації, це проявляється зниженням розумової працездатності, збільшенням кількості помилок, розвитком зорового стомлення, зміною сприйняття кольорів, появою головного болю та зниженням концентрації уваги. Нормований рівень шуму на робочих місцях не повинен перевищувати 50 дБ. Основні методи боротьби з шумом включають усунення або зменшення джерел шуму на етапі проектування, застосування звукопоглинаючих матеріалів та раціональне планування виробничих приміщень.

3.3 Освітлення

Освітлення в приміщеннях із ПК має бути змішаним — природним і штучним. Природне світло повинно забезпечуватись бічним освітленням

					<i>РП 08. 13 003. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		57

відповідно до норм ДБН В.2.5-28-2006 «Природне і штучне освітлення». Для уникнення різких перепадів яскравості між природним світлом і світлом екрана ВДТ необхідно передбачити сонцезахисні засоби, наприклад, металізовані плівки або вертикальні жалюзі з регульованими ламелями. Штучне освітлення в приміщеннях з ВДТ повинно бути комбінованим, із застосуванням люмінесцентних ламп у світильниках загального освітлення. На робочих місцях має бути рівномірна освітленість, здебільшого за рахунок відбитого або розсіяного світла. Світлові відблиски з клавіатури, екрана та інших частин ВДТ у напрямку очей користувача мають бути відсутні. Нормативна освітленість робочих місць становить 300–500 лк.

3.4 Електробезпека

Причинами ураження працівника електричним струмом можуть бути:

- випадковий контакт зі струмоведучими елементами під час виконання робіт поблизу або безпосередньо на цих частинах;
- несправність захисних засобів, якими потерпілий торкався до струмоведучих деталей.

Сила струму, що проходить через тіло людини, залежить від напруги, під якою знаходиться людина, та від опору ділянки тіла, на яку впливає ця напруга. Джерелом напруги живлення є мережа змінного струму з напругою 229 В, що регламентується ГОСТ 25861-83.

Для запобігання ураженню електричним струмом необхідно суворо виконувати правила проведення робіт і технічної експлуатації електроустановок. Важливо виключити доступ працівника до частин обладнання, які знаходяться під небезпечною напругою, до неізольованих елементів, що працюють на низькій напрузі і не підключені до захисного заземлення, а також забезпечувати підключення ПЕОМ до електроживлення через спеціальні вилки з заземлюючим контактом.

3.5 Організація робочого місця

Обладнання та організація робочого місця з ВДТ повинні відповідати

					<i>РП 08. 13 003. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		58

ергономічним вимогам щодо конструкції всіх його елементів і їх розташування, з урахуванням специфіки трудової діяльності (ДСанПіН 3.3.2.-007-98). Конструкція робочого місця та розташування його складових (сидіння, органи керування, засоби відображення інформації) мають відповідати антропометричним, фізіологічним і психологічним потребам, а також характеру виконуваної роботи. Робочі меблі повинні забезпечувати можливість індивідуального регулювання для підтримки зручного положення працівника. Стіл має бути пофарбований матовою фарбою. Дисплей розміщується так, щоб верхній край екрана знаходився на рівні очей на відстані приблизно 70 см, що відповідає нормативним межах від 60 до 90 см. Частота мерехтіння екрана становить 100 Гц, що перевищує мінімально допустимі 70 Гц. Для зниження нервово-емоційного навантаження, зменшення втоми, покращення мозкового кровообігу та подолання негативних наслідків гіподинамії рекомендується виконувати комплекс вправ, наведений у Державних санітарних правилах і нормах роботи з візуальними терміналами ЕОМ (ДСанПіН 3.3.2.007-98).

На рисунку 3.1 зображена організація робочого місця з ПК.

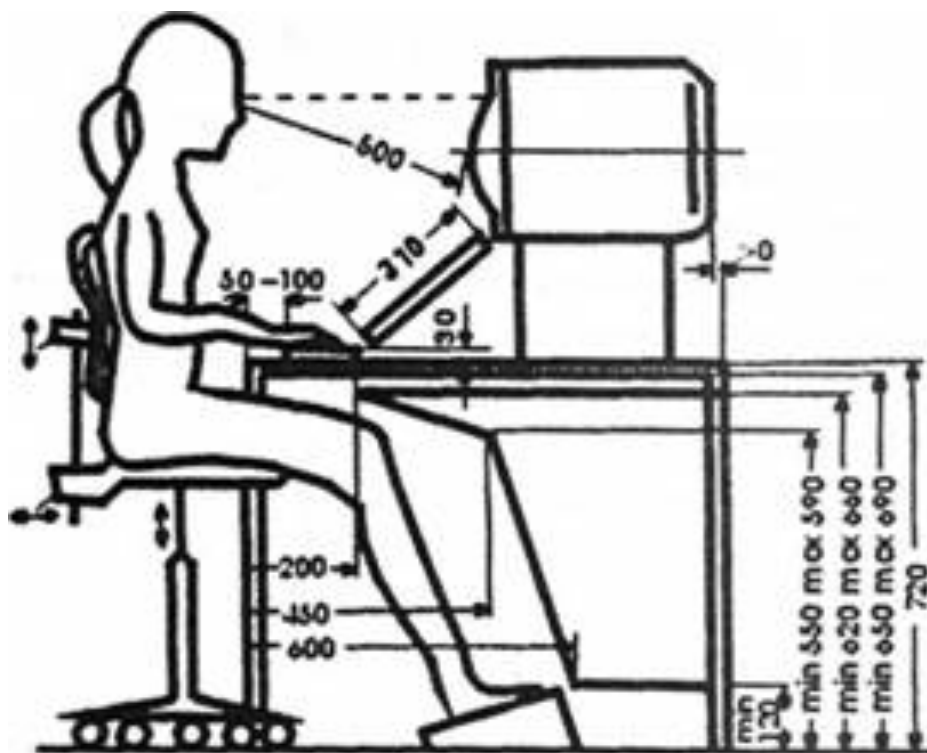


Рисунок 3.1. Організація робочого місця за ПК

3.4 Пожежна безпека

Протипожежний захист приміщень забезпечується автоматичною системою пожежної сигналізації, наявністю засобів для гасіння пожежі, використанням будівельних конструкцій з регламентованими показниками вогнестійкості, а також організацією своєчасної евакуації людей. Усі приміщення мають бути оснащені первинними засобами пожежогасіння: пожежним водопостачанням (пожежними кранами ПК), пожежними щитами з комплектом пожежного інструменту, а також вуглекислотними або порошковими вогнегасниками. У разі виникнення пожежі слід негайно вимкнути електроживлення, зателефонувати за номером 101 до пожежної служби, організувати евакуацію людей згідно з планом і розпочати ліквідацію пожежі.

3.5 Підведення підсумків

В Україні охорона праці є важливим пріоритетом для забезпечення безпеки і здоров'я працівників, незважаючи на те, що рівень нещасних випадків на виробництві залишається високим. Покращення умов праці сприяє підвищенню якості і продуктивності праці, а також зниженню економічних витрат, пов'язаних із травматизмом і професійними захворюваннями. Тому впровадження ефективних заходів з охорони праці програмістів є необхідною складовою сталого розвитку підприємств і організацій.

					<i>РП 08. 13 003. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		60

ВИСНОВКИ

Під час виконання проєкту було проаналізовано потреби майстрів манікюру щодо цифрових інструментів для організації роботи. Виявлено запит на єдиний інструмент для планування, ведення клієнтської бази, керування послугами та презентації робіт.

На основі цього спроектовано структуру мобільного застосунку-помічника з основними функціональними модулями:

1. Прайс-лист: для зберігання, редагування та перегляду переліку послуг із зазначенням цін;
2. База клієнтів: для ведення історії відвідувань, контактної інформації та нотаток;
3. Календар записів: з можливістю планування прийомів, нагадувань і перегляду завантаженості;
4. Палітра кольорів: для збереження улюблених відтінків гель-лаків та зручного пошуку;
5. Генератор ідей: для натхнення, на основі стилістичних шаблонів і випадкових комбінацій;
6. Галерея робіт: для презентації завершених дизайнів клієнтам та в соціальних мережах.

Розробку здійснено з використанням фреймворку React Native, який забезпечує кросплатформність — здатність застосунку працювати як на Android, так і на iOS без необхідності створення окремих версій. Для реалізації навігації між розділами використано бібліотеку `@react-navigation/bottom-tabs`. Інтерфейс інтуїтивно зрозумілий, адаптивний і зручний для щоденного використання.

Застосунок автоматизує рутинні процеси, полегшує управління інформацією та підвищує професійну ефективність майстра. Він відповідає потребам цільової аудиторії, сприяє покращенню обслуговування клієнтів і підвищує конкурентоспроможність фахівця. Усі поставлені завдання реалізовано, а проєкт має потенціал для подальшого розвитку.

					<i>РП 08. 13 000. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		61

ПЕРЕЛІК ВИКОРИСТАНИХ ІНФОРМАЦІЙНИХ ДЖЕРЕЛ

1. Фалькович Г. А. Манікюр : навч. посіб. / Г. А. Фалькович. – Київ : УкрЛіб, 20-- . – Електронне видання. – Посібник охоплює будову нігтя, методики апаратного і комбінованого манікюру, санітарні вимоги та організацію робочого місця. Доступно на dreambook.com.ua, instagram.com, dan-it.com.ua, ukrlib.com.ua.
2. Радченко К. О. Розроблення мобільних застосунків : навч. посіб. / К. О. Радченко. – Київ : КПІ ім. Ігоря Сікорського, 2023. – 16,3 авт.арк. – Конспект лекцій з Android-розробки: інтерфейси, життєвий цикл, робота з БД, тестування. Доступно на ela.kpi.ua, dan-it.com.ua.
3. Стефанов С. React.js. Швидкий старт : навч. посіб. / С. Стефанов. – 2025. – Огляд React: компоненти, JSX, Flux, ESLint, Jest. Доступно на dreambook.com.ua, yakaboo.ua.
4. Nail Book [Електронний ресурс]. – Режим доступу: <https://play.google.com/store/apps/details?id=com.nailbook.app> – Застосунок для обліку клієнтів і записів майстрів манікюру. Дата звернення: 22.05.2025.
5. SalonAppy [Електронний ресурс]. – Режим доступу: <https://play.google.com/store/apps/details?id=com.salonappy.app> – Планування записів, CRM і SMS-нагадування для салонів краси. Дата звернення: 24.05.2025.
6. Booksy [Електронний ресурс]. – Режим доступу: <https://play.google.com/store/apps/details?id=net.booksy.customer> – Онлайн-бронювання послуг краси та керування графіком. Дата звернення: 25.05.2025.
7. Google Forms [Електронний ресурс]. – Режим доступу: <https://forms.google.com> – Інструмент для створення опитувань, форм і тестів. Дата звернення: 27.05.2025.
8. Expo Snack Documentation [Електронний ресурс]. – Режим доступу: <https://docs.expo.dev> – Офіційна документація фреймворку Expo для розробки мобільних застосунків. Дата звернення: 01.06.2025.

					<i>РП 08. 13 000. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		62

ДОДАТОК А. Фрагмент програмного коду генерації

малюнків манікюру

```
import React, { useState, useEffect } from 'react';
import {
  View,
  Text,
  TouchableOpacity,
  ScrollView,
  Alert,
  StyleSheet,
} from 'react-native';
import Svg, { Circle, Rect, Line } from 'react-native-svg';
import AsyncStorage from '@react-native-async-storage/async-storage';

const AVAILABLE_COLORS = [
  '#FF6B6B',
  '#FFD93D',
  '#6BCB77',
  '#4D96FF',
  '#843B62',
  '#FF9F1C',
  '#2EC4B6',
  '#E71D36',
];

const getRandomInt = (max) => Math.floor(Math.random() * max);

const generateShapes = (colors) => {
  const shapes = [];
  const count = 8 + getRandomInt(5);

  for (let i = 0; i < count; i++) {
    const shapeType = getRandomInt(3);
    const color = colors[getRandomInt(colors.length)];
    const x = getRandomInt(180);
    const y = getRandomInt(280);
    const size = 10 + getRandomInt(30);

    switch (shapeType) {
      case 0:
        shapes.push(
          <Circle
            key={`c${i}`}
            cx={x}
            cy={y}
            r={size / 2}
            fill={color}
            opacity={0.6}
          />
        );
        break;
      case 1:
        shapes.push(
          <Rect
            key={`r${i}`}
            x={x}
            y={y}
            width={size}
            height={size}
            fill={color}
            opacity={0.5}
          />
        );
    }
  }
};
```

```

        rx={size / 4}
        ry={size / 4}
      />
    );
    break;
  case 2:
    const x2 = x + getRandomInt(30);
    const y2 = y + getRandomInt(30);
    shapes.push(
      <Line
        key={`l${i}`}
        x1={x}
        y1={y}
        x2={x2}
        y2={y2}
        stroke={color}
        strokeWidth={3}
        opacity={0.7}
        strokeLinecap="round"
      />
    );
    break;
  }
}

return shapes;
};

const STORAGE_KEY = '@savedDrawings';

export default function PriceListScreen() {
  const [color1, setColor1] = useState('#FF6B6B');
  const [color2, setColor2] = useState('#FFD93D');
  const [color3, setColor3] = useState('#6BCB77');
  const [shapes, setShapes] = useState(generateShapes([color1, color2, color3]));
  const [savedDrawings, setSavedDrawings] = useState([]);
  const [savedColorsList, setSavedColorsList] = useState([]);
  const [selectedSavedIndex, setSelectedSavedIndex] = useState(null);

  // Загрузка сохраненных рисунков из AsyncStorage при монтировании
  useEffect(() => {
    (async () => {
      try {
        const savedData = await AsyncStorage.getItem(STORAGE_KEY);
        if (savedData) {
          const savedParams = JSON.parse(savedData);
          const loadedDrawings = savedParams.map(colors => generateShapes(colors));
          setSavedColorsList(savedParams);
          setSavedDrawings(loadedDrawings);
        }
      } catch (e) {
        console.error('Ошибка загрузки сохраненных рисунков', e);
      }
    })();
  }, []);

  // Сохраняем массив выбранных цветов (вместо JSX) в AsyncStorage при обновлении savedColorsList
  useEffect(() => {
    (async () => {
      try {
        await AsyncStorage.setItem(STORAGE_KEY, JSON.stringify(savedColorsList));
      } catch (e) {
        console.error('Ошибка сохранения рисунков', e);
      }
    })();
  });
}

```

```

    }
  })();
}, [savedColorsList]);

const regenerate = () => {
  setShapes(generateShapes([color1, color2, color3]));
  setSelectedSavedIndex(null);
};

const save = () => {
  const currentColors = [color1, color2, color3];
  const newDrawing = generateShapes(currentColors);
  setSavedDrawings([...savedDrawings, newDrawing]);
  setSavedColorsList([...savedColorsList, currentColors]);
  Alert.alert('Збережено!', 'Абстракція нігтьової пластини збережена.');
```

setSelectedSavedIndex(savedDrawings.length);

```

};

const onColorPick = (pickerIndex, color) => {
  if (pickerIndex === 1) setColor1(color);
  if (pickerIndex === 2) setColor2(color);
  if (pickerIndex === 3) setColor3(color);
  setSelectedSavedIndex(null);
};

const deleteDrawing = (index) => {
  Alert.alert(
    'Видалити?',
    'Ви впевнені, що хочете видалити цей малюнок?',
    [
      { text: 'Скасувати', style: 'cancel' },
      {
        text: 'Видалити',
        style: 'destructive',
        onPress: () => {
          const newDrawings = savedDrawings.filter((_, i) => i !== index);
          const newColorsList = savedColorsList.filter((_, i) => i !== index);
          setSavedDrawings(newDrawings);
          setSavedColorsList(newColorsList);
          if (selectedSavedIndex === index) {
            setSelectedSavedIndex(null);
          } else if (selectedSavedIndex > index) {
            setSelectedSavedIndex(selectedSavedIndex - 1);
          }
        }
      }
    ]
  );
};

return (
  <ScrollView style={styles.container}>
    <Text style={styles.title}>Згенерувати нігті</Text>

    <View style={styles.canvasWrapper}>
      <View style={styles.canvas}>
        <Svg height="300" width="200">
          {selectedSavedIndex !== null
            ? savedDrawings[selectedSavedIndex]
            : shapes}
        </Svg>
      </View>
    </View>
  </View>
);

```

```

<View style={styles.colorsSection}>
  {[1, 2, 3].map((index) => {
    const selectedColor =
      index === 1 ? color1 : index === 2 ? color2 : color3;
    return (
      <View key={index} style={styles.colorBlock}>
        <Text style={styles.label}>Колір {index}</Text>
        <View style={styles.colorRow}>
          {AVAILABLE_COLORS.map((color) => (
            <TouchableOpacity
              key={` ${color}-${index}`}
              style={[
                styles.colorDot,
                { backgroundColor: color },
                selectedColor === color && styles.activeDot,
              ]}
              onPress={() => onColorPick(index, color)}
            />
          ))}
        </View>
      </View>
    );
  })}
</View>

<View style={styles.buttonRow}>
  <TouchableOpacity style={styles.buttonPrimary} onPress={regenerate}>
    <Text style={styles.buttonText}>Згенерувати</Text>
  </TouchableOpacity>
  <TouchableOpacity style={styles.buttonSecondary} onPress={save}>
    <Text style={styles.buttonText}>Зберегти</Text>
  </TouchableOpacity>
</View>

<Text style={styles.title}>Згенеровані нігті</Text>
<ScrollView
  horizontal
  style={styles.savedRow}
  showsHorizontalScrollIndicator={false}>
  {savedDrawings.map((drawing, index) => (
    <View key={index} style={styles.savedItemWrapper}>
      <TouchableOpacity
        onPress={() => setSelectedSavedIndex(index)}
        style={[
          styles.savedItem,
          selectedSavedIndex === index && styles.savedItemActive,
        ]}>
        <Svg height="60" width="40" viewBox="0 0 200 300">
          {drawing}
        </Svg>
      </TouchableOpacity>
      <TouchableOpacity
        style={styles.deleteButton}
        onPress={() => deleteDrawing(index)}
      >
        <Text style={{ fontSize: 18, textAlign: 'center' }}>✖</Text>
      </TouchableOpacity>
    </View>
  ))}
</ScrollView>
</ScrollView>
);
}

```

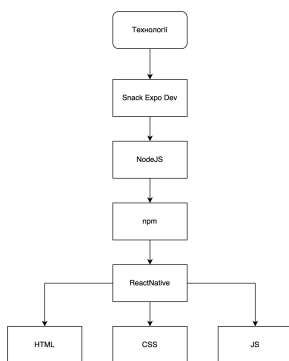
```
const styles = StyleSheet.create({
  container: {
    padding: 20,
    backgroundColor: '#FFF9E6',
    flex: 1,
  },
  title: {
    fontSize: 20,
    fontWeight: 'bold',
    marginBottom: 10,
    color: '#C9A637',
    textAlign: 'left',
  },
  canvasWrapper: {
    alignItems: 'center',
    marginBottom: 20,
  },
  canvas: {
    backgroundColor: '#F5F0E6',
    borderRadius: 20,
    width: 200,
    height: 300,
    overflow: 'hidden',
    shadowColor: '#000',
    shadowOpacity: 0.08,
    shadowRadius: 10,
    shadowOffset: { width: 0, height: 4 },
  },
  colorsSection: {
    marginBottom: 20,
  },
  colorBlock: {
    marginBottom: 12,
  },
  colorRow: {
    flexDirection: 'row',
    flexWrap: 'wrap',
    justifyContent: 'space-between',
    gap: 10,
  },
  colorDot: {
    width: 30,
    height: 30,
    borderRadius: 15,
    borderWidth: 1,
    borderColor: '#ccc',
  },
  activeDot: {
    borderWidth: 3,
    borderColor: '#333',
  },
  buttonRow: {
    flexDirection: 'row',
    justifyContent: 'space-between',
    marginBottom: 20,
    gap: 12,
  },
  buttonPrimary: {
    flex: 1,
    backgroundColor: '#D4B30A',
    paddingVertical: 12,
    borderRadius: 10,
    marginRight: 6,
  },
});
```

Розробка мобільного застосунку-помічника майстра манікюру

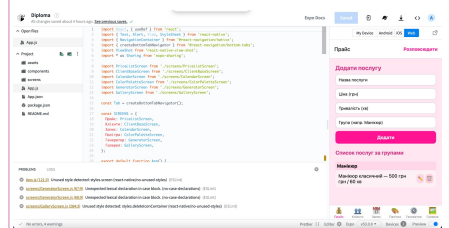
ЗДОБУВАЧКА ОСВІТИ ГРУПИ 4РП-08: ЛОМАЖУК А.Г.

КЕРІВНИК ДИПЛОМНОГО ПРОЄКТУ: ЖАДАН А.С.

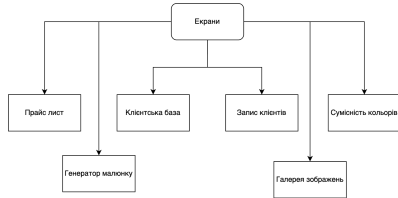
Технології мобільного застосунку



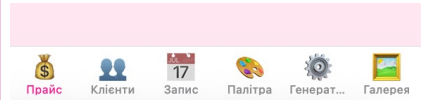
1. SNACK EXPO DEV
2. NODE JS
3. NPM
4. REACTNATIVE
5. HTML
6. CSS
7. JAVASCRIPT



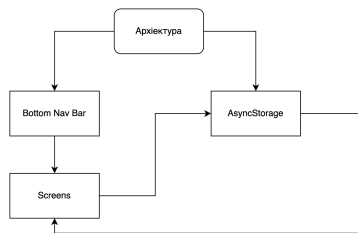
Навігація у мобільному застосунку



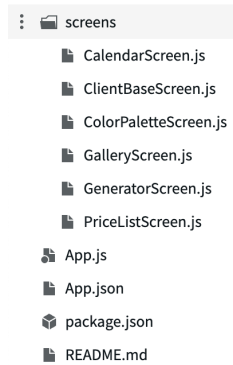
- Прайс Лист
- Клієнтська база
- Запис клієнтів
- Генератор малюнку
- Сумісність кольорів
- Галерея зображень



Архітектура мобільного застосунку



- Bottom Nav Bar
- Screens
- AsyncStorage



Екран «Послуги за категоріями»

Прайс

Додати послугу

Додати

Список послуг за групами

Манікюр

Манікюр класичний — 500 грн
грн / 60 хв



Екран «Клієнтська база»

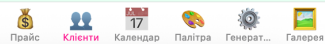
Клієнти

Додати клієнта

Додати

Список клієнтів

Іванова Іванка —
+380999999999
Нотатки: любить яскраві кольори



Екран «Запис»

Запис

Запис клієнта

Вибрати дату

Вибрати час

6/13/2025, 3:13:17 PM

Клієнт:

Оберіть клієнта

Послуга:

Оберіть послугу

Зберегти запис

Записи

6/5/2025, 4:46:02 PM



Екран «Палітра кольорів»

Палітра

Обери 2 або 3 кольори

Колір 1

Відтінок (тапніть щоб вибрати):



Прозорість: 1.00

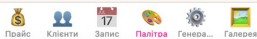


Колір 2

Відтінок (тапніть щоб вибрати):

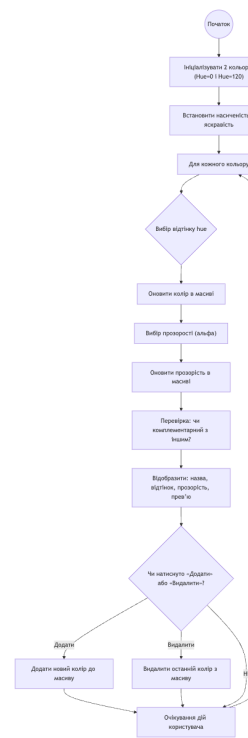


Прозорість: 0.95



Торкніться, щоб закрити

Алгоритм перевірки кольорів на сумісність



Програмний код перевірки на компліментарність

```
100     return [r,g,b];
101   }
102
103   function areComplementary(h1, h2) {
104     const diff = Math.abs(h1 - h2);
105     return (diff > 165 && diff < 195) || (diff > 345 || diff < 15);
106   }
107
108   const complementaryPairs = [];
109   for (let i = 0; i < colors.length; i++) {
110     for (let j = i+1; j < colors.length; j++) {
111       if (areComplementary(colors[i].h, colors[j].h)) {
112         complementaryPairs.push([i, j]);
113       }
114     }
115   }
116 }
```

Екран «Генератор МАЛЮНКІВ»

Генератор

Згенерувати нігті



Колір 1
Колір 2
Колір 3

Three rows of color selection circles, each containing 8 different colors: red, yellow, green, blue, purple, orange, teal, and red.

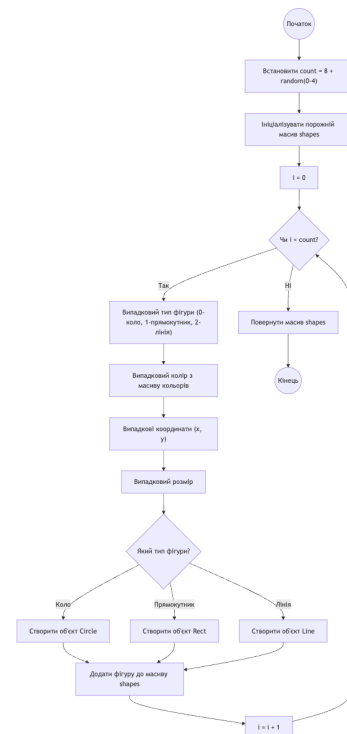
Згенерувати

Зберегти

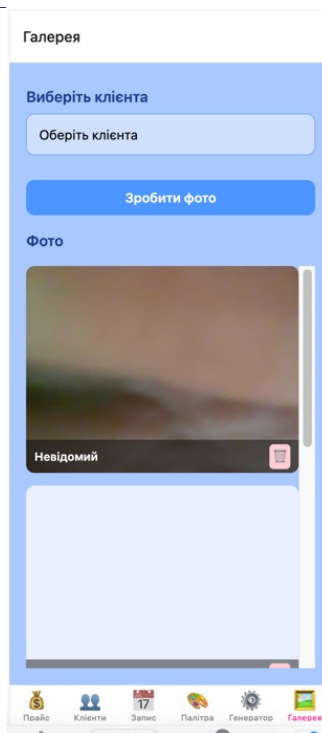
Згенеровані нігті



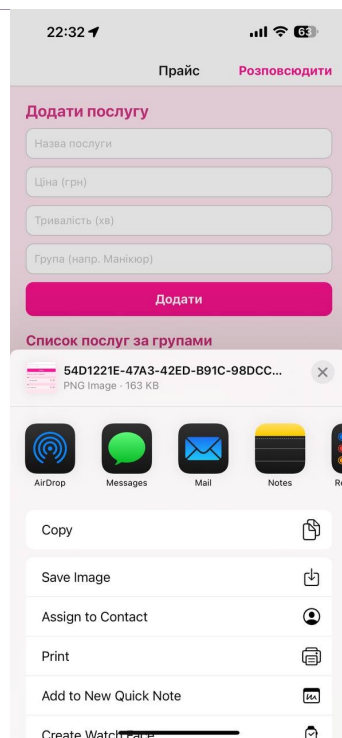
Алгоритм генерації кольорів



Екран «Галерея зображень»



Функція «розповсюдити»



РЕЦЕНЗІЯ

на дипломний проект здобувача (здобувачки) освіти
відділення комп'ютерних систем

Ломажук Аліни Геннадіївни

(прізвище, ім'я та по батькові)

Спеціальність 121 «Інженерія програмного забезпечення»

Освітня програма «Розробка програмного забезпечення»

Керівник дипломного проекту (роботи) Жадан Артур Сергійович

(прізвище, ім'я та по батькові)

Тема дипломного проекту (роботи) Розробка мобільного застосунку-помічника майстра манікюру

Обсяг розрахунково-пояснювальної записки 75 сторінок

Обсяг графічної (презентаційної) частини 14 аркушів (слайдів)

ХАРАКТЕРИСТИКА ДИПЛОМНОГО ПРОЕКТУ (РОБОТИ)

а) заключення про ступінь відповідності виконаного дипломного проекту завданню

Представлений на рецензію дипломний проект відповідає затвердженій темі та виконаний відповідно технічному завданню. Дипломний проект присвячений розробці мобільного застосунку та складається з пояснювальної записки, додатку з програмним кодом та мультимедійної презентації, що містить приклади роботи програми.

б) характеристика виконання кожного розділу дипломного проекту

Пояснювальна записка складається з основного розділу (аналізу предметної області, проектування застосунку, реалізації застосунку, тестування застосунку), економічного розділу, розділу охорони праці та додатків. Перелічені розділи поетапно охоплюють розробку, виконані докладно та обґрунтовано. Розділ охорони праці містить загальну інформацію та вимоги до техніки безпеки оператора КТ. Економічний розділ проекту містить розрахунок витрат на НДР та реалізацію проекту.

в) оцінка якості виконання пояснювальної записки та графічної частини дипломного проекту

Графічна частина складається з 14 слайдів мультимедійної презентації, виконаної у програмному продукті MS PowerPoint, які містять ілюстративні схеми, скріншоти роботи програмного застосунку, передбачені технічним завданням. Пояснювальна записка виконана акуратно та у відповідності до норм. Якість виконання графічної частини проекту та пояснювальної записки добра, розробку виконано у повному обсязі.

г) перелік позитивних якостей дипломного проекту Продумана реалізація офлайн-функціональності із застосуванням AsyncStorage. Архітектура застосунку має чітку модульну структуру з розділенням за функціональними екранами. Обрано React Native як фреймворк, що дозволяє швидко створювати портативний та сучасний UI, а також забезпечує можливість подальшої масштабованості додатка.

д) основні недоліки дипломного проекту Застосування Expo/ SnackExpo має обмеження для реальних умов експлуатації, проте це не було належно проаналізовано. Інтеграція можливості синхронізації з хмарними сервісами могла б підвищити надійність збереження інформації. Деякі недоліки оформлення пояснювальної записки та зображення алгоритмів у графічній частині.

Оцінка розрахункової частини	<u>Відмінно</u>
Оцінка графічної частини	<u>Добре</u>
Загальна оцінка	<u>Відмінно</u>

Прізвище, ім'я, по батькові рецензента к.т.н. Шибасва Наталя Олегівна

Місце роботи і посада рецензента Національний університет «Одеська політехніка», доцент кафедри інформаційних технологій



Підпис: _____

« 20 » червня 2025 р.

ВІДГУК

керівника на дипломний проект здобувача (здобувачки) освіти
відділення комп'ютерних систем

Ломажук Аліні Геннадіївні

(прізвище, ім'я та по батькові)

Спеціальність: 121 «Інженерія програмного забезпечення»

Освітня програма: «Розробка програмного забезпечення»

Тема дипломного проекту: Розробка мобільного застосунку-помічника
майстра манікюру.

ХАРАКТЕРИСТИКА ДИПЛОМНОГО ПРОЕКТУ

а) обсяг і якість виконання проекту (графічного матеріалу і розрахунково-пояснювальної записки) Дипломний проект виконано відповідно технічному завданню. Пояснювальна записка до дипломного проекту містить 75 сторінки. У пояснювальній записці описано етапи розробки мобільного застосунку-помічника майстра манікюру засобами HTML, CSS, JavaScript та ReactNative. Графічна частина складається з окремих 14 слайдів, оформлених у вигляді презентації, передбачених технічним завданням. Якість виконання пояснювальної записки та слайдів добра.

б) самостійність роботи над проектом: Протягом виконання дипломного проекту здобувачка освіти поступово та послідовно виконувала всі етапи, проявляла ініціативу в створенні загальної концепції та реалізації роботи. Всі роботи здобувачка освіти виконувала самостійно, з оглядом на рекомендації керівника.

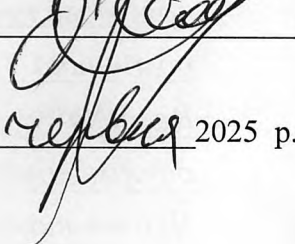
в) теоретична підготовка випускника (випускниці): Здобувачка освіти під час роботи над дипломним проектом вивчила достатньо багато літературних та інтернет-джерел за даною тематикою. Вважаю, що теоретична підготовка дипломника достатня і вона готова до захисту проекту.

г) вміння розв'язувати виробничі та конструкторські питання Під час виконання дипломного проекту здобувачка освіти показала вміння організовано працювати над поставленим завданням, застосовувати знання у галузі програмування та математики, розробляти, встановлювати та налаштовувати спеціалізоване програмне забезпечення, оформлювати слайди та складати презентації, користуючись сучасними комп'ютерними програмними засобами, такими як *Snack Expo Dev, HTML, CSS, JavaScript та ReactNative.*

Оцінка розрахункової частини Добре
Оцінка графічної частини Відмінно
Загальна оцінка Відмінно

Прізвище, ім'я, по батькові керівника дипломного проекту Жадан Артур Сергійович

Місце роботи і посада керівника дипломного проекту ВСП «Одеський технічний фаховий коледж ОНТУ», викладач спецдисциплін циклової комісії комп'ютерної техніки та програмної інженерії.

Підпис 
«16» червня 2025 р.

**ДОЗВІЛ
НА РОЗМІЩЕННЯ
ВИПУСКНОЇ КВАЛІФІКАЦІЙНОЇ РОБОТИ
(ДИПЛОМНОГО ПРОЕКТУ)
В ЕЛЕКТРОННОМУ РЕПОЗИТАРІЇ ВСП «ОТФК ОНТУ»**

Ми, що нижче підписалися,

Ломажук Аліна Геннадіївна,
здобувачка освіти гр. 4РП-08, та

Жадан Артур Сергійович,
керівник дипломного проекту,

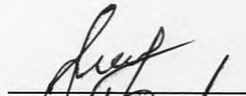
не заперечуємо щодо розміщення електронного варіанту пояснювальної записки до дипломного проекту фахового молодшого бакалавра на тему:

«Розробка мобільного застосунку-помічника майстра манікюру» (авторка – Ломажук А.Г., керівник роботи – Жадан А.С.)

виконаного у ВСП «Одеський технічний фаховий коледж Одеського національного технологічного університету» в 2025 році, у повному обсязі в електронному репозитарії ВСП «ОТФК ОНТУ» для вільного доступу через мережу Інтернет.

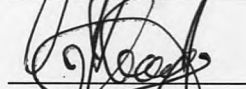
Несемо відповідальність за ідентичність електронного та друкованого варіантів випускної кваліфікаційної роботи і даємо згоду на обробку персональних даних.

Виконавець



/ Ломажук А.Г. /

Керівник



/ Жадан А.С. /

«16» червня 2025 р.

ДОВІДКА

циклової комісії КТ та ПІ
про допуск до захисту дипломного проєкту
здобувача (здобувачки) освіти ІV курсу
відділення комп'ютерних систем групи 4РП-08

Ломажук Аліни Геннадіївни

на тему *Розробка мобільного застосунку-помічника майстра манікюру*

Висновок відповідальної особи за проведення нормоконтролю:

пояснювальна записка до дипломного проєкту виконана з некритичними

порушеннями ДСТУ та оформлена відповідно до вимог Положення про

дипломне проєктування



(підпис)

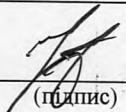
20.06.2025

(дата)

Петрашова В.І.

(П.І.Б.)

Висновок відповідальної особи за перевірку роботи на наявність академічного плагіату *згідно звіту про перевірку від 18.06.2025 р. значення коефіцієнту подібності в роботі становить 16,30%, коефіцієнт цитування – 2,43%.*



(підпис)

20.06.2025

(дата)

Краснокутська К.Г.

(П.І.Б.)

Попередня експертиза (малий захист) дипломного проєкту

здобувача (здобувачки) освіти

Ломажук А.Г.

(П.І.Б.)

проведена « 20 » червня 2025 р.

Висновки *Пояснювальна записка до дипломного проєкту виконана у повному
обсязі. Випускна кваліфікаційна робота (дипломний проєкт) відповідає
вимогам Положення про дипломне проєктування та рекомендована до
захисту.*

Голова ЦК КТ та ПІ


(підпис)

Кривченко Ю.В.

(П.І.Б.)

Звіт подібності

метадані

Назва організації

Odesa Technical Professional College of Odesa National University of Technology

Заголовок

Розробка мобільного застосунку-помічника майстра манікюру

Автор

Науковий керівник / Експерт

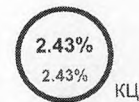
Ломажук Аліна Геннадівна Жадан Артур Сергійович

підрозділ

Відокремлений структурний підрозділ "Одеський технічний фаховий коледж Одеського національного технологічного університету"

Обсяг знайдених подібностей

Коефіцієнт подібності визначає, який відсоток тексту по відношенню до загального обсягу тексту було знайдено в різних джерелах. Зверніть увагу, що високі значення коефіцієнта не автоматично означають плагіат. Звіт має аналізувати компетентна / уповноважена особа.



25

Довжина фрази для коефіцієнта подібності 2

12138

Кількість слів

107145

Кількість символів

Тривога

У цьому розділі ви знайдете інформацію щодо текстових спотворень. Ці спотворення в тексті можуть говорити про МОЖЛИВІ маніпуляції в тексті. Спотворення в тексті можуть мати навмисний характер, але частіше характер технічних помилок при конвертації документа та його збереженні, тому ми рекомендуємо вам підходити до аналізу цього модуля відповідально. У разі виникнення запитань, просимо звертатися до нашої служби підтримки.

Заміна букв		16
Інтервали		0
Мікропробіли		1
Білі знаки		427
Парафрази (SmartMarks)		101

Подібності за списком джерел

Нижче наведений список джерел. В цьому списку є джерела із різних баз даних. Колір тексту означає в якому джерелі він був знайдений. Ці джерела і значення Коефіцієнту Подібності не відображають прямого плагіату. Необхідно відкрити кожне джерело і проаналізувати зміст і правильність оформлення джерела.

10 найдовших фраз

ПОРЯДКОВИЙ НОМЕР	НАЗВА ТА АДРЕСА ДЖЕРЕЛА URL (НАЗВА БАЗИ)	Копір тексту КІЛЬКІСТЬ ІДЕНТИЧНИХ СЛІВ (ФРАГМЕНТІВ)
1	https://card-file.ontu.edu.ua/bitstreams/34a6756b-592f-4b77-a805-183aa03a6a26/download	74 0.61 %
2	https://card-file.ontu.edu.ua/server/api/core/bitstreams/995bdcec-4e4d-4321-8070-4d6badcb8e49/content	36 0.30 %
3	Розробка 3D-гри у жанрі survival-horror з налаштуваннями рівнів складності 6/12/2025 Odesa Technical Professional College of Odesa National University of Technology (Відокремлений структурний підрозділ "Одеський технічний фаховий коледж Одеського національного технологічного університету")	35 0.29 %

4	https://card-file.ontu.edu.ua/server/api/core/bitstreams/a141b658-5fa7-4f90-b0bd-7f0ccaed21e5/content	32 0.26 %
5	https://card-file.ontu.edu.ua/bitstreams/aed610a6-43ef-47e0-9066-e85c89456f3e/download	32 0.26 %
6	https://card-file.ontu.edu.ua/bitstreams/29489599-0581-4ce6-8890-c3b13d9f2e0e/download	31 0.26 %
7	https://card-file.ontu.edu.ua/server/api/core/bitstreams/a141b658-5fa7-4f90-b0bd-7f0ccaed21e5/content	30 0.25 %
8	https://card-file.ontu.edu.ua/bitstreams/1dff552d-7200-49b8-ae1d-ba76a1335685/download	29 0.24 %
9	https://card-file.ontu.edu.ua/bitstreams/1dff552d-7200-49b8-ae1d-ba76a1335685/download	27 0.22 %
10	https://card-file.ontu.edu.ua/bitstreams/bbed74c8-2ea7-44c5-8d00-0fe3fd9790ee/download	25 0.21 %

з домашньої бази даних (0.66 %)

ПОРЯДКОВИЙ НОМЕР	ЗАГОЛОВОК	КІЛЬКІСТЬ ІДЕНТИЧНИХ СЛІВ (ФРАГМЕНТІВ)
1	Розробка 3D-гри у жанрі survival-horror з налаштуваннями рівнів складності 6/12/2025 Odesa Technical Professional College of Odesa National University of Technology (Відокремлений структурний підрозділ "Одеський технічний фаховий коледж Одеського національного технологічного університету")	65 (3) 0.54 %
2	Розробка веб-застосунку для генерації повідомлень із використанням технологій штучного інтелекту 6/14/2025 Odesa Technical Professional College of Odesa National University of Technology (Відокремлений структурний підрозділ "Одеський технічний фаховий коледж Одеського національного технологічного університету")	9 (1) 0.07 %
3	Створення веб-застосунку цифрового помічника з використанням Open AI 5/28/2025 Odesa Technical Professional College of Odesa National University of Technology (Відокремлений структурний підрозділ "Одеський технічний фаховий коледж Одеського національного технологічного університету")	6 (1) 0.05 %

з програми обміну базами даних (0.49 %)

ПОРЯДКОВИЙ НОМЕР	ЗАГОЛОВОК	КІЛЬКІСТЬ ІДЕНТИЧНИХ СЛІВ (ФРАГМЕНТІВ)
1	ДИПЛОМ.ФИНАЛ.docx 6/2/2021 Sumy State University (Кафедра прикладної математики та моделювання складних систем)	22 (1) 0.18 %
2	ПЗДП_ІСТ_КПІ_2025_Рябчун Вероніка Сергіївна 6/9/2025 National Technical University of Ukraine Igor Sikorskyi Kyiv Politech Institute (ФІОТ, К-ра інформаційних систем та технологій)	14 (2) 0.12 %
3	125_Горюк_ВС 11/3/2024 State University of Trade and Economics (Кафедра інженерії програмного забезпечення та кібербезпеки)	13 (1) 0.11 %
4	Hyriavenko_bac_rob.docx 6/11/2021 Sumy State University (Кафедра інформаційних технологій)	11 (1) 0.09 %

з Інтернету (15.14 %)

ПОРЯДКОВИЙ НОМЕР	ДЖЕРЕЛО URL	КІЛЬКІСТЬ ІДЕНТИЧНИХ СЛІВ (ФРАГМЕНТІВ)
---------------------	-------------	---

1	https://card-file.ontu.edu.ua/server/api/core/bitstreams/44c16132-5f53-48e2-b6c0-61e9a2f0fd75/content	516 (43) 4.25 %
2	https://card-file.ontu.edu.ua/bitstreams/1dff552d-7200-49b8-ae1d-ba76a1335685/download	229 (15) 1.89 %
3	https://card-file.ontu.edu.ua/bitstreams/bbed74c8-2ea7-44c5-8d00-0fe3fd9790ee/download	128 (11) 1.05 %
4	https://card-file.ontu.edu.ua/server/api/core/bitstreams/a141b658-5fa7-4f90-b0bd-7f0ccaed21e5/content	127 (9) 1.05 %
5	https://card-file.ontu.edu.ua/bitstreams/34a6756b-592f-4b77-a805-183aa03a6a26/download	116 (4) 0.96 %
6	https://card-file.ontu.edu.ua/bitstreams/aed610a6-43ef-47e0-9066-e85c89456f3e/download	87 (6) 0.72 %
7	https://card-file.ontu.edu.ua/bitstreams/29489599-0581-4ce6-8890-c3b13d9f2e0e/download	71 (5) 0.58 %
8	https://card-file.ontu.edu.ua/bitstreams/549ee9fe-7574-4ae5-b500-9fe2711f33e6/download	66 (6) 0.54 %
9	https://card-file.ontu.edu.ua/bitstreams/62baa43e-b968-4993-bb54-8cf8761a89b2/download	58 (6) 0.48 %
10	https://card-file.ontu.edu.ua/bitstreams/035f6436-20b4-4ee6-8e99-bede670e308b/download	56 (6) 0.46 %
11	https://card-file.ontu.edu.ua/bitstreams/0e72a3b9-bdd7-4711-a3c6-dedc1d4287cc/download	56 (5) 0.46 %
12	https://card-file.ontu.edu.ua/bitstreams/53ed22ad-8700-4162-b97a-082a1ad472d6/download	56 (4) 0.46 %
13	https://card-file.ontu.edu.ua/server/api/core/bitstreams/995bdcec-4e4d-4321-8070-4d6badcb8e49/content	54 (2) 0.44 %
14	https://www.jscamp.app/docs/react-navigation/reactnative07/	31 (2) 0.26 %
15	https://card-file.ontu.edu.ua/bitstreams/5240e379-7721-49f0-8ee8-27140b0b473a/download	30 (3) 0.25 %
16	https://card-file.ontu.edu.ua/bitstreams/6cf43324-8f08-4031-ba42-f80b18efbbc8/download	24 (3) 0.20 %
17	https://card-file.ontu.edu.ua/bitstreams/bbaf3f38-16a8-4070-bead-5562769b7c71/download	23 (1) 0.19 %
18	http://www.dstu.dp.ua/Portal/Data/5/10/5-10-kl25.pdf	22 (1) 0.18 %
19	https://card-file.ontu.edu.ua/bitstreams/2b75599e-e1ac-412d-bf09-10d2eb49022f/download	21 (2) 0.17 %
20	https://aboutreact.com/react-native-login-and-signup/	21 (3) 0.17 %
21	https://github.com/transistorsoft/react-native-background-geolocation/issues/2303	20 (3) 0.16 %
22	http://ukrefs.com.ua/print.page.1.171894-Proshivka-chipa-kartridzha-Samsung-SCX-4200.html	11 (1) 0.09 %
23	https://card-file.ontu.edu.ua/bitstreams/7b1e10b9-0ac2-4b07-afc4-8cdfff7db780/download	10 (1) 0.08 %
24	https://www.stud24.ru/management/kadrovj-menedzhmenti/495966-1914465-page9.html	5 (1) 0.04 %

Список прийнятих фрагментів (немає прийнятих фрагментів)

ПОРЯДКОВИЙ НОМЕР	ЗМІСТ	КІЛЬКІСТЬ ОДНАКОВИХ СЛІВ (ФРАГМЕНТІВ)
------------------	-------	---------------------------------------

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ВСП «ОДЕСЬКИЙ ТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ ОНТУ»

Спеціальність: 121 «Інженерія програмного забезпечення»
Освітньо-професійна програма: «Розробка програмного забезпечення» Група: 4РП- 08

Дипломний проект
здобувачки освіти денної форми навчання РП. 08.13.000.ДП