

**Міністерство освіти і науки України
Луцький національний технічний університет
Факультет комп'ютерних та інформаційних технологій
Кафедра інженерії програмного забезпечення**

**КВАЛІФІКАЦІЙНА РОБОТА
ЗА СТУПЕНЕМ ВИЩОЇ ОСВІТИ «БАКАЛАВР»**

**РОЗРОБКА ВЕБ-СИСТЕМИ КЕРУВАННЯ ГРУМІНГ-ПОСЛУГАМИ
ТВАРИН З ВИКОРИСТАННЯМ KOTLIN ТА FIREBASE**

**DEVELOPMENT OF A WEB-BASED SYSTEM FOR MANAGING
ANIMAL GROOMING SERVICES USING KOTLIN AND FIREBASE**

спеціальність 121 «Інженерія програмного забезпечення»
освітня програма «Інженерія програмного забезпечення»

Виконав: здобувач вищої освіти
Групи ІПЗ-43
Бандура Вадим Ростиславович

Керівник:
Повстяна Юлія Славомирівна

Кваліфікаційну роботу
допущено до захисту
«___» _____ 2026 р.
к.т.н., доцент
Гарант освітньої програми:
К.т.н., доцент
Ліщина Наталія Миколаївна

(підпис)

ЛУЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

Факультет комп'ютерних та інформаційних технологій

Кафедра інженерії програмного забезпечення

Ступінь вищої освіти: бакалавр

Галузь знань: 12 Інформаційні технології

Спеціальність: 121 «Інженерія програмного забезпечення»

Освітня програма: «Інженерія програмного забезпечення»

ЗАТВЕРДЖУЮ

Завідувач кафедри

«__» _____ 20__ р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧУ ВИЩОЇ ОСВІТИ

Бандурі Вадиму Ростиславовичу

1. Тема кваліфікаційної роботи: «Розробка веб-системи керування грумінг-послугами тварин з використанням Kotlin та Firebase»

Керівник роботи: к.т.н., доцент, Повстяна Юлія Славомирівна

затверджені наказом закладу вищої освіти від «31» грудня 2025р. № 541/01-02

2. Строк подання здобувачем вищої освіти кваліфікаційної роботи «29» травня 2026 р.

3. Вихідні дані до роботи: предметна область організації грумінг-послуг, процеси взаємодії клієнтів із сервісами онлайн-запису, методи автоматизації бронювання послуг; технології розробки програмного забезпечення (Kotlin), засоби створення інтерактивних користувацьких інтерфейсів; хмарна платформа Firebase, зокрема Firebase Authentication для реалізації автентифікації та Cloud Firestore для зберігання і обробки даних; вимоги до інформаційних систем щодо масштабованості, надійності, безпеки та зручності використання.

4. Зміст розрахунково-пояснювальної записки (перелік питань, що потрібно розробити): аналіз сучасного стану проблеми, існуючих методів і засобів її розв'язання, аналіз, визначення вимог до розроблюваного програмного забезпечення та проектування програмного забезпечення, опис функціонального наповнення об'єкта проектування, практична реалізація об'єкта проектування.

5. Перелік графічного матеріалу: 18 рисунків, 3 лістинги, 1 таблиця

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис	
		завдання видав	завдання прийняв
Аналіз предметної області	<i>Повстяна Ю. С.</i>		
Специфікації вимог до розробленої системи	<i>Повстяна Ю. С.</i>		
Розробка об'єкта проектування	<i>Повстяна Ю. С.</i>		
Нормоконтроль	<i>Суринович О. М.</i>		
Гарант ОП	<i>Ліщина Н. М.</i>		
Показник запозичень тексту	_____ %		
Академічна доброчесність	<i>Повстяна Ю. С.</i>		

7. Дата видачі завдання «03» лютого 2026р.

№	Назва етапів кваліфікаційної роботи бакалавра	Строк виконання етапів роботи	Примітка
1	Огляд літературних джерел по темі кваліфікаційної роботи бакалавра	до 14.02.2026 р.	
2	Аналіз проблеми розробки та впровадження об'єкту проектування	до 03.03.2026 р.	
3	Обґрунтування вибору шляхів, технологій і засобів вирішення поставленого завдання	до 14.03.2026 р.	
4	Розробка функціонально-структурної схеми роботи об'єкта проектування та проектування бази даних	до 14.04.2026 р.	
5	Практична реалізація об'єкта проектування та розробка бази даних	до 20.04.2026 р.	
6	Тестування та налагодження об'єкта проектування	до 04.05.2026 р.	
7	Здача чистового варіанту кваліфікаційної роботи бакалавра на кафедрі	до 29.05.2026 р.	

Здобувач вищої освіти _____

Вадим БАНДУРА

Керівник кваліфікаційної роботи _____

Юлія ПОВСТЯНА

АНОТАЦІЯ

Бандура В. Р. Розробка веб-системи керування грумінг-послугами тварин з використанням Kotlin та Firebase . Рукопис. Кваліфікаційна робота бакалавра ОП «Інженерія програмного забезпечення» спеціальності «Інженерія програмного забезпечення». Луцький національний технічний університет. Луцьк, 2026.

Кваліфікаційна робота бакалавра складається зі вступу, трьох розділів, висновків, списку використаних джерел.

У першому розділі проведено аналіз предметної області, пов'язаної з організацією грумінг-послуг, розглянуто сучасні підходи до автоматизації процесів запису клієнтів, проаналізовано існуючі програмні рішення та визначено їх переваги й недоліки. На основі проведеного аналізу обґрунтовано доцільність розробки власної інформаційної системи.

У другому розділі визначено функціональні та нефункціональні вимоги до веб-системи, виконано її проєктування з використанням UML-нотації, а також обґрунтовано вибір інструментальних засобів, методів і технологій розробки, зокрема мови програмування Kotlin та хмарної платформи Firebase.

У третьому розділі описано процес практичної реалізації веб-системи, включаючи розробку користувацького інтерфейсу, реалізацію механізмів авторизації, пошуку салонів та створення записів. Окрему увагу приділено розробці структури бази даних у Cloud Firestore, забезпеченню захисту інформації та проведенню тестування і налагодження програмного забезпечення.

У висновках підсумовано отримані результати, підтверджено досягнення поставленої мети та виконання всіх завдань кваліфікаційної роботи бакалавра.

Ключові слова: веб-система, грумінг-послуги, онлайн-запис, інформаційна система, Kotlin, Firebase, Cloud Firestore, автентифікація, база даних.

ABSTRACT

Bandura V. R. Development of a Web-Based System for Managing Animal Grooming Services Using Kotlin and Firebase. Manuscript. Bachelor's qualification work in Software Engineering, specialty «Software Engineering». Lutsk National Technical University. Lutsk, 2026.

The bachelor's qualification work consists of an introduction, three chapters, conclusions, a list of references.

In the first chapter, the problem domain related to grooming service organization is analyzed. Modern approaches to client booking automation are reviewed, existing software solutions are examined, and their advantages and disadvantages are identified. The necessity of developing a dedicated information system is substantiated.

In the second chapter, functional and non-functional requirements for the web system are defined, the system is designed using UML notation, and the choice of tools, methods, and development technologies, including Kotlin and Firebase, is justified.

The third chapter describes the practical implementation of the web system, including the development of the user interface, implementation of authentication mechanisms, salon search, and booking creation. Special attention is paid to the database structure design using Cloud Firestore, as well as information security and software testing.

The conclusions summarize the obtained results, confirm the achievement of the research objective, and verify the fulfillment of all defined tasks.

Keywords: web system, grooming services, online booking, information system, Kotlin, Firebase, Cloud Firestore, authentication, database.

ЗМІСТ

ВСТУП	7
РОЗДІЛ 1 АНАЛІЗ ІСНУЮЧИХ РІШЕНЬ У СФЕРІ ГРУМІНГ-ПОСЛУГ І ПОСТАНОВКА ЗАВДАНЬ НА КВАЛІФІКАЦІЙНУ РОБОТУ	10
1.1 Аналіз сучасного стану проблеми.....	10
1.2 Постановка завдання на кваліфікаційну роботу бакалавра.....	13
РОЗДІЛ 2 СПЕЦИФІКАЦІЯ ВИМОГ ДО РОЗРОБЛЮВАНОЇ СИСТЕМИ ..	15
2.1 Аналіз, визначення вимог до розроблюваного програмного забезпечення та проектування програмного забезпечення	15
2.2 Вибір засобів, методів і алгоритмів вирішення поставленого завдання ...	21
РОЗДІЛ 3 РОЗРОБКА ВЕБ-СИСТЕМИ КЕРУВАННЯ ГРУМІНГ-ПОСЛУГАМИ ТВАРИН З ВИКОРИСТАННЯМ KOTLIN ТА FIREBASE ...	24
3.1 Практична реалізація об'єкта проектування	24
3.2 Тестування та налагодження інформаційно-комп'ютерної системи.....	33
3.3 Реалізація бази даних	34
3.4 Оцінка ефективності та результатів роботи веб-системи	37
ВИСНОВКИ.....	40
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	42

ВСТУП

У сучасних умовах розвитку інформаційного суспільства спостерігається стрімке впровадження цифрових технологій у всі сфери діяльності людини. Особливої актуальності набуває автоматизація процесів у сфері послуг, де швидкість обслуговування, зручність взаємодії з клієнтами та ефективне управління ресурсами є ключовими факторами конкурентоспроможності. Однією з таких сфер є індустрія догляду за домашніми тваринами, яка активно розвивається у зв'язку зі зростанням кількості власників домашніх улюбленців та підвищенням рівня відповідальності за їхній догляд.

Грумінг-послуги, що включають гігієнічний та естетичний догляд за тваринами (миття, стрижка, чистка, догляд за кігтями тощо), стали невід'ємною частиною сучасного ринку послуг. З кожним роком зростає попит на якісні та професійні послуги грумінгу, що, у свою чергу, зумовлює необхідність удосконалення процесів їх надання. Водночас більшість малих і середніх закладів, що надають такі послуги, використовують застарілі або неефективні методи ведення обліку, зокрема паперові журнали, електронні таблиці чи неінтегровані програмні рішення.

Зазначені підходи мають низку суттєвих недоліків: обмеженість у масштабуванні, високий ризик втрати або пошкодження даних, відсутність централізованого доступу до інформації, складність у плануванні графіків роботи майстрів та записів клієнтів, а також недостатній рівень взаємодії з користувачами. Це призводить до зниження ефективності роботи закладу, збільшення часу обслуговування клієнтів та ймовірності виникнення помилок.

У зв'язку з цим актуальним є розроблення сучасної веб-системи керування грумінг-послугами, яка забезпечить автоматизацію основних бізнес-процесів, зокрема ведення бази клієнтів і їхніх тварин, управління записами на послуги, формування розкладу роботи, облік наданих послуг, а

також взаємодію з користувачами через зручний інтерфейс. Така система дозволить підвищити якість обслуговування, оптимізувати роботу персоналу та зменшити витрати часу на виконання рутинних операцій.

Особливу роль у створенні подібних систем відіграє вибір сучасних інструментів і технологій розробки. Використання мови програмування Kotlin є доцільним завдяки її сучасному синтаксису, підтримці об'єктно-орієнтованого та функціонального програмування, високому рівню безпеки та сумісності з існуючими платформами. Kotlin дозволяє зменшити кількість помилок у коді, підвищити продуктивність розробки та забезпечити підтримуваність програмного продукту.

У свою чергу, платформа Firebase надає широкий спектр інструментів для розробки сучасних веб-застосунків, включаючи хмарне зберігання даних, автентифікацію користувачів, хостинг, а також можливості для масштабування та синхронізації даних у реальному часі. Використання Firebase дозволяє значно скоротити час розробки, спростити реалізацію серверної частини та забезпечити надійність зберігання даних.

Поєднання зазначених технологій створює передумови для розробки ефективної, масштабованої та зручної у використанні веб-системи, яка відповідатиме сучасним вимогам до інформаційних систем у сфері послуг. Крім того, така система може бути адаптована до різних типів закладів, що надають послуги догляду за тваринами, та інтегрована з іншими цифровими сервісами.

Актуальність теми дослідження також зумовлена необхідністю підвищення рівня цифровізації малого бізнесу в Україні, що є важливим фактором його розвитку та конкурентоспроможності. Впровадження інформаційних систем у діяльність підприємств дозволяє оптимізувати бізнес-процеси, покращити якість обслуговування клієнтів та забезпечити ефективне управління ресурсами.

Метою кваліфікаційної роботи бакалавра є розробка веб-системи керування грумінг-послугами тварин із використанням мови програмування

Kotlin та платформи Firebase для автоматизації процесів обслуговування клієнтів і підвищення ефективності функціонування закладів.

Для досягнення поставленої мети необхідно вирішити такі завдання:

- провести аналіз предметної області та існуючих рішень у сфері грумінг-послуг;
- визначити функціональні та нефункціональні вимоги до веб-системи;
- обґрунтувати вибір технологій і засобів розробки;
- спроектувати архітектуру системи та побудувати UML-моделі;
- розробити структуру бази даних і механізми взаємодії з нею;
- реалізувати основні функціональні модулі веб-системи;
- провести тестування та налагодження системи і оцінити результати її роботи.

Об'єктом кваліфікаційної роботи бакалавра є процеси організації та надання грумінг-послуг для домашніх тварин.

Предметом кваліфікаційної роботи бакалавра є методи, засоби та технології розробки веб-систем керування грумінг-послугами, зокрема із застосуванням Kotlin та Firebase.

Практичне значення отриманих результатів полягає у можливості використання розробленої веб-системи в діяльності закладів, що надають послуги грумінгу, для підвищення ефективності їх роботи, автоматизації процесів управління та покращення якості обслуговування клієнтів. Розроблена система може бути використана як основа для подальшого вдосконалення та розширення функціональних можливостей, а також адаптації до інших сфер надання послуг.

РОЗДІЛ 1

АНАЛІЗ ІСНУЮЧИХ РІШЕНЬ У СФЕРІ ГРУМІНГ-ПОСЛУГ І ПОСТАНОВКА ЗАВДАНЬ НА КВАЛІФІКАЦІЙНУ РОБОТУ

1.1 Аналіз сучасного стану проблеми

Аналіз сучасних наукових досліджень показує, що сфера догляду за домашніми тваринами, включаючи грумінг-послуги, є важливою складовою забезпечення здоров'я та благополуччя тварин. У дослідженні [1] встановлено, що регулярний грумінг є базовим елементом гігієнічного догляду, а його відсутність може призводити до серйозних проблем зі здоров'ям тварин, включаючи шкірні захворювання та дискомфорт. При цьому автори зазначають, що основними бар'єрами для отримання грумінг-послуг є вартість, складність доступу та організаційні труднощі, що підкреслює необхідність оптимізації сервісів у цій сфері.

У науковій роботі [2], присвяченій цифровим платформам у pet-care індустрії, зазначається, що розвиток онлайн-сервісів дозволяє вирішити проблему географічної доступності та комунікації між клієнтами і постачальниками послуг. Такі платформи забезпечують автоматизацію замовлень, регулярні платежі та персоналізовану взаємодію з користувачами, що є важливим для сучасних сервісних систем.

Крім того, у дослідженні [3], присвяченому розробці веб-орієнтованої системи управління pet-care сервісами, зазначається, що використання ручних процесів у грумінг-салонах і зоомагазинах призводить до неефективності управління записами, обліку товарів та взаємодії з клієнтами. Автори підкреслюють необхідність створення інтегрованих інформаційних систем, які об'єднують функції запису, обліку, управління послугами та аналітики.

У сучасних умовах ефективно управління записами клієнтів, ведення бази даних тварин, контроль розкладу роботи персоналу та автоматизація взаємодії з клієнтами є критично важливими для забезпечення

конкурентоспроможності підприємств. Традиційні підходи, що базуються на використанні паперових журналів, телефонних записів або електронних таблиць, не забезпечують належного рівня ефективності, що обумовлює необхідність впровадження спеціалізованих інформаційних систем.

Аналіз сучасних веб-систем показує, що на міжнародному ринку існує ряд програмних продуктів, які реалізують функції автоматизації грумінг-бізнесу. Зокрема, система MoeGo (рис. 1.1) є комплексною платформою для керування грумінг-салонами, яка забезпечує автоматизоване онлайн-бронювання, управління розкладом, комунікацію з клієнтами та зменшення кількості пропущених записів за рахунок автоматичних нагадувань. Важливою особливістю є підтримка інтелектуального планування записів, що дозволяє оптимізувати завантаженість персоналу.

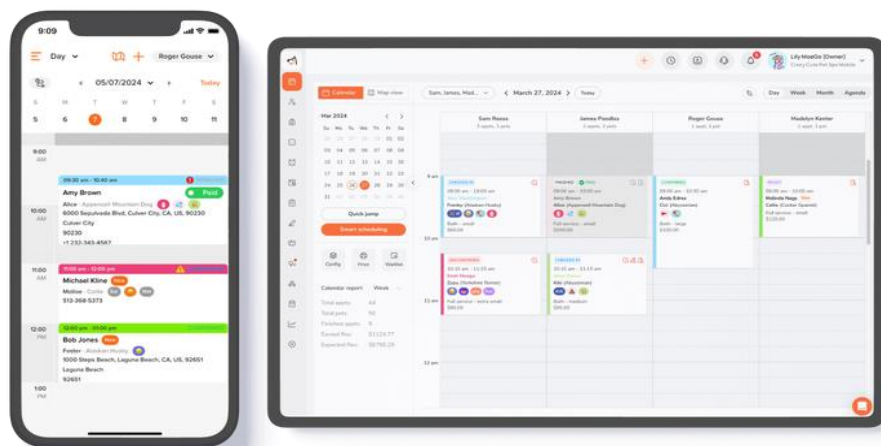


Рисунок 1.1 – Система MoeGo [4]

Система GrooMoge (рис. 1.2) також є прикладом сучасного програмного забезпечення для грумінг-бізнесу, що забезпечує управління записами, клієнтами та тваринами, автоматичні нагадування, ведення історії відвідувань та аналітику діяльності закладу. Особливістю є можливість налаштування системи під конкретні потреби бізнесу та підтримка багатофункціонального календаря.

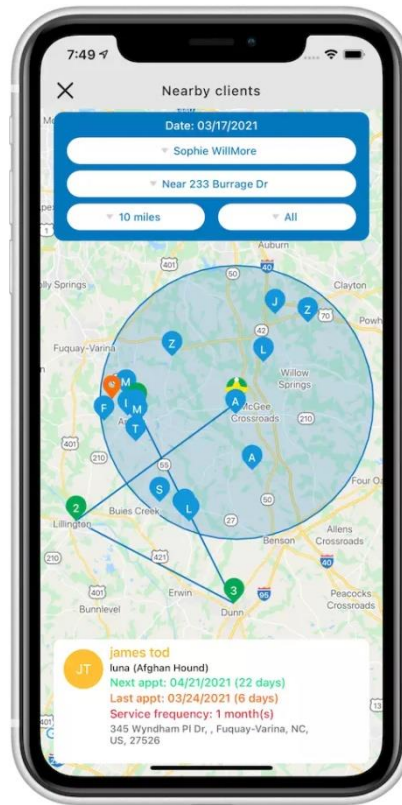


Рисунок 1.2 – Система GrooMore [5]

Узагальнений аналіз сучасних рішень показує, що ключовими функціями веб-систем у даній сфері є:

- онлайн-запис клієнтів;
- управління базою клієнтів і тварин;
- календарне планування;
- автоматичні нагадування;
- обробка платежів;
- аналітика та звітність.

Водночас існуючі рішення мають ряд недоліків: вони часто є комерційними (SaaS), орієнтованими на іноземні ринки, мають високу вартість підписки, складний інтерфейс або надлишковий функціонал для малого бізнесу. Крім того, відсутня адаптація до локальних умов, зокрема мовної підтримки та специфіки ведення бізнесу в Україні.

Проблематика автоматизації процесів обслуговування клієнтів також широко досліджується у науковій літературі. Зокрема, у роботах, присвячених системам онлайн-запису, зазначається, що впровадження цифрових механізмів бронювання дозволяє зменшити час очікування, оптимізувати використання ресурсів та підвищити задоволеність клієнтів. У дослідженні, присвяченому інформаційній системі для управління процесами у сфері догляду за тваринами, показано, що використання веб-платформ дозволяє значно підвищити ефективність обробки запитів та забезпечити масштабованість системи.

Крім того, сучасні дослідження у сфері систем планування (scheduling systems) доводять, що автоматизовані системи запису є ключовим елементом оптимізації сервісних процесів. Вони дозволяють мінімізувати простой, уникати дублювання записів та підвищувати продуктивність персоналу. Також встановлено, що використання онлайн-бронювання позитивно впливає на поведінку клієнтів, зменшуючи кількість пропущених записів і підвищуючи рівень лояльності.

Проведений аналіз сучасних веб-систем і наукових досліджень свідчить про наявність значного попиту на ефективні інформаційні системи у сфері грумінг-послуг. Незважаючи на існування комерційних рішень, залишається актуальною задача розробки спеціалізованої веб-системи, яка буде адаптована до потреб користувачів, матиме оптимальний функціонал, зручний інтерфейс та можливість подальшого розширення.

1.2 Постановка завдання на кваліфікаційну роботу бакалавра

Проведений аналіз сучасного стану проблеми у сфері надання грумінг-послуг та існуючих програмних рішень показав, що, незважаючи на наявність функціональних систем автоматизації, вони мають ряд суттєвих недоліків, зокрема високу вартість використання, складність налаштування, надлишковий функціонал або відсутність адаптації до потреб малого бізнесу

та локальних умов. Крім того, використання неінтегрованих інструментів, таких як електронні таблиці чи паперові журнали, призводить до зниження ефективності управління, втрати даних та ускладнення взаємодії з клієнтами.

У зв'язку з цим виникає необхідність розроблення спеціалізованої веб-системи керування грумінг-послугами тварин, яка забезпечить автоматизацію основних бізнес-процесів, підвищить ефективність обслуговування клієнтів та оптимізує роботу персоналу.

Для досягнення поставленої мети необхідно вирішити такі завдання:

- провести аналіз предметної області та існуючих рішень у сфері грумінг-послуг;
- визначити функціональні та нефункціональні вимоги до веб-системи;
- обґрунтувати вибір технологій і засобів розробки;
- спроектувати архітектуру системи та побудувати UML-моделі;
- розробити структуру бази даних і механізми взаємодії з нею;
- реалізувати основні функціональні модулі веб-системи;
- провести тестування та налагодження системи і оцінити результати її роботи.

РОЗДІЛ 2

СПЕЦИФІКАЦІЯ ВИМОГ ДО РОЗРОБЛЮВАНОЇ СИСТЕМИ

2.1 Аналіз, визначення вимог до розроблюваного програмного забезпечення та проектування програмного забезпечення

Процес розробки веб-системи керування грумінг-послугами тварин передбачає обов'язкове попереднє виявлення, аналіз та формалізацію вимог до програмного забезпечення, що дозволяє забезпечити відповідність функціональних можливостей системи реальним потребам користувачів і особливостям предметної області. Якісно сформовані вимоги є основою для подальшого проектування архітектури системи, визначення її функціональної структури, а також забезпечення ефективності та надійності програмного продукту в умовах практичного використання.

Предметна область дослідження охоплює процеси організації та надання грумінг-послуг, що включають взаємодію між клієнтами, персоналом салону та адміністрацією. У рамках цих процесів виконуються операції реєстрації користувачів, ведення обліку клієнтів і їхніх домашніх тварин, формування записів на обслуговування, планування графіка роботи майстрів, контроль статусу виконання послуг та збереження історії обслуговування. Аналіз існуючих підходів до організації таких процесів показує, що у більшості випадків використовуються неінтегровані або застарілі інструменти, зокрема паперові журнали чи електронні таблиці, що призводить до неефективності управління, дублювання даних і зростання ймовірності помилок. Це обумовлює необхідність створення централізованої інформаційної системи, яка забезпечить автоматизацію основних бізнес-процесів.

Виявлення вимог до розроблюваної веб-системи здійснювалося на основі системного аналізу предметної області, дослідження існуючих програмних рішень у сфері грумінг-послуг, а також моделювання типових сценаріїв взаємодії користувачів із системою. На першому етапі було

проведено аналіз бізнес-процесів, що дозволило виокремити ключові функції, які повинна підтримувати система. На другому етапі здійснено дослідження аналогів, що використовуються на ринку, з метою визначення їх функціональних можливостей, переваг і недоліків. На третьому етапі визначено основні категорії користувачів системи, до яких належать клієнти, грумери та адміністратори. На завершальному етапі було сформовано перелік функціональних і нефункціональних вимог, а також побудовано UML-моделі для візуалізації та перевірки повноти специфікації [6].

У результаті проведеного аналізу встановлено, що ключовими користувачами системи є клієнт, який здійснює запис на послуги, грумер, який виконує обслуговування, та адміністратор, який відповідає за керування процесами та даними системи. Клієнт взаємодіє із системою з метою перегляду доступних послуг, створення облікового запису, додавання інформації про домашню тварину, а також оформлення запису на обслуговування та перегляду історії наданих послуг. Грумер використовує систему для перегляду власного графіка роботи, отримання інформації про заплановані записи та зміни статусу виконання послуг. Адміністратор забезпечує керування користувачами, довідником послуг, розкладом роботи персоналу та записами клієнтів.

На основі визначених ролей користувачів сформовано функціональні вимоги до системи. Розроблювана веб-система повинна забезпечувати можливість реєстрації та автентифікації користувачів із використанням безпечних механізмів доступу, підтримку рольової моделі з розмежуванням прав доступу, ведення бази даних клієнтів і їхніх домашніх тварин, перегляд і адміністрування переліку грумінг-послуг, створення та редагування записів на обслуговування з урахуванням доступності часових інтервалів, перегляд графіка роботи персоналу, зміну статусів записів, а також збереження історії наданих послуг. Крім того, система повинна забезпечувати централізоване зберігання даних і їх узгодженість у межах усіх функціональних модулів.

Важливою складовою специфікації є нефункціональні вимоги, які визначають якісні характеристики програмного продукту. Система повинна забезпечувати зручність і зрозумілість користувацького інтерфейсу, що дозволяє користувачам швидко освоїти функціональні можливості без додаткового навчання. Швидкодія системи має забезпечувати виконання основних операцій, таких як перегляд сторінок, створення запису або авторизація, без відчутних затримок за умови стабільного мережевого з'єднання. Надійність зберігання даних передбачає відсутність втрати інформації та забезпечення її цілісності навіть у разі одночасної роботи кількох користувачів. Безпека системи реалізується через використання механізмів автентифікації та авторизації, що обмежують доступ до даних відповідно до ролей користувачів. Крім того, система повинна бути масштабованою, що дозволяє розширювати її функціональні можливості та кількість користувачів без суттєвого зниження продуктивності, а також адаптивною до різних пристроїв, що забезпечує коректну роботу як на персональних комп'ютерах, так і на мобільних пристроях.

Проектування програмного забезпечення здійснюється з використанням уніфікованої мови моделювання UML, яка дозволяє формалізувати вимоги та представити їх у вигляді графічних моделей. Зокрема, діаграма варіантів використання використовується для визначення взаємодії між користувачами та системою, діаграма діяльності – для відображення послідовності виконання основних бізнес-процесів. Використання UML-моделювання забезпечує структурованість підходу до розробки, дозволяє виявити можливі логічні помилки на ранніх етапах проектування та підвищує якість прийнятих технічних рішень.

Для формалізації взаємодії користувачів із веб-системою визначено основних акторів, які беруть участь у виконанні функціональних сценаріїв. Опис акторів та їх ролей у системі наведено в таблиці 2.1.

Таблиця 2.1 – Опис акторів веб-системи керування грумінг-послугами тварин

Актор	Опис акторів
Клієнт	Особа, яка відвідує веб-систему для ознайомлення з переліком грумінг-послуг та, за потреби, проходить реєстрацію для подальшого запису на обслуговування.
Зареєстрований клієнт	Користувач, який має обліковий запис у системі та може авторизуватися, додавати інформацію про домашню тварину, оформляти запис на послугу, переглядати та скасовувати власні записи.
Грумер	Працівник салону, який переглядає власний графік роботи, список призначених записів та змінює статус виконання послуг.
Адміністратор	Користувач із розширеними правами доступу, який керує обліковими записами, послугами, записами, графіками роботи персоналу та іншими даними системи.
Firebase-сервіс	Зовнішній програмний компонент, що забезпечує автентифікацію користувачів, зберігання даних і взаємодію з хмарною інфраструктурою.

UML-діаграма варіантів (рис. 2.1) відображає послідовність дій користувача під час оформлення запису на грумінг-послугу. Процес починається з авторизації користувача в системі, після чого він переглядає перелік доступних послуг, обирає необхідну процедуру, домашню тварину, а також дату та час обслуговування. На наступному етапі система здійснює перевірку доступності обраного часового інтервалу. У разі, якщо час є доступним, формується запис на послугу, який зберігається у базі даних, після чого користувачу відображається підтвердження. Якщо ж обраний час недоступний, система повідомляє користувача про необхідність вибору іншого інтервалу.

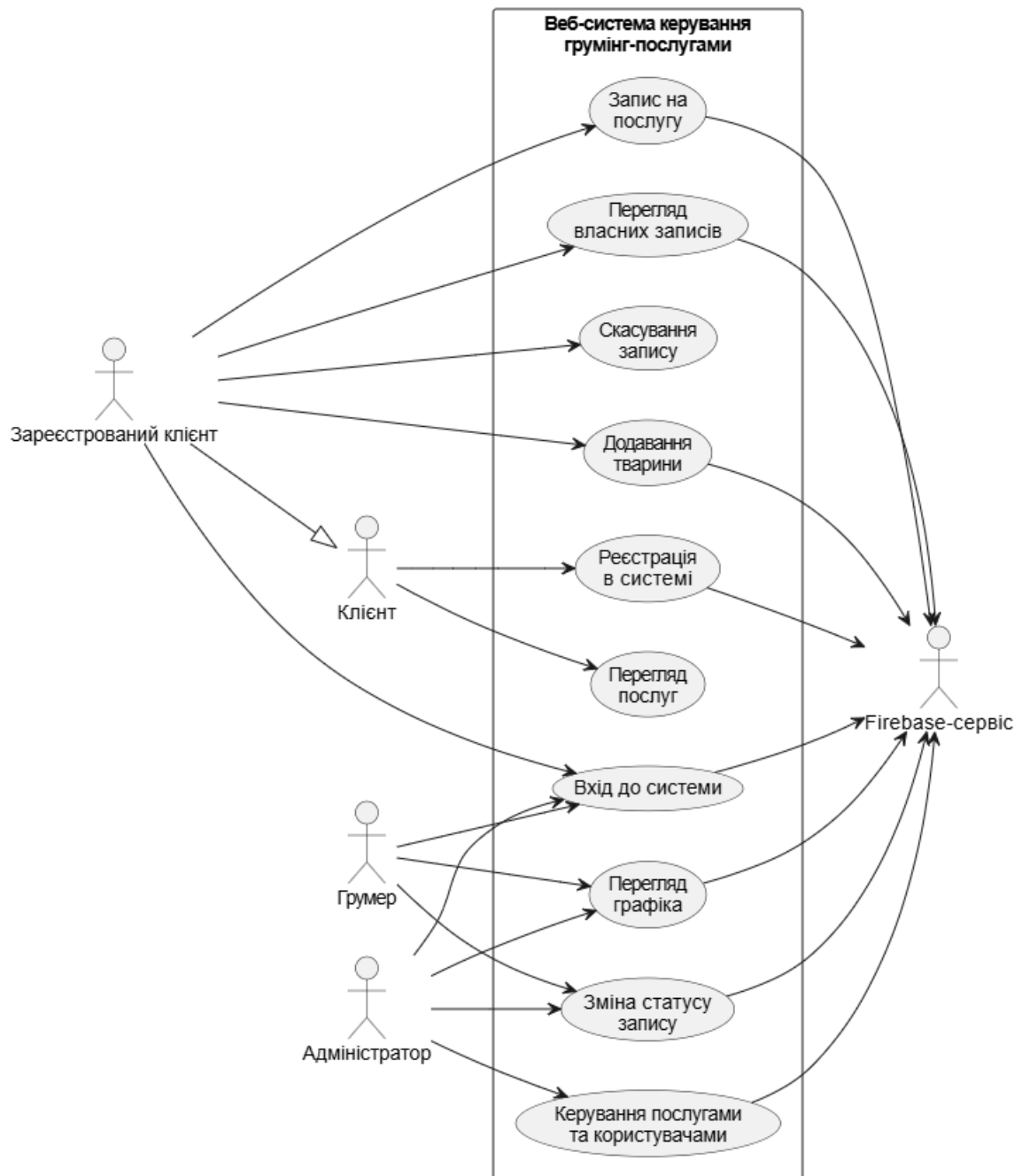


Рисунок 2.1 – UML-діаграма варіантів використання веб-системи керування грумінг-послугами

На рисунку 2.2 представлено діаграму діяльності, яка відображає процес формування запису на грумінг-послугу у веб-системі. Початковим етапом є авторизація користувача, після чого він отримує доступ до перегляду переліку доступних послуг та здійснює вибір необхідної послуги. Далі користувач обирає домашню тварину та визначає бажану дату і час обслуговування.

На наступному етапі система виконує перевірку доступності обраного часу. У разі, якщо час є доступним, відбувається створення запису на послугу, його збереження у базі даних та відображення підтвердження користувачу. Якщо ж обраний час недоступний, система формує відповідне повідомлення про неможливість запису.

Завершальним етапом процесу є об'єднання альтернативних сценаріїв та завершення виконання дії. Таким чином, діаграма відображає повний цикл взаємодії користувача із системою при створенні запису, включаючи перевірку умов та обробку можливих варіантів розвитку подій.

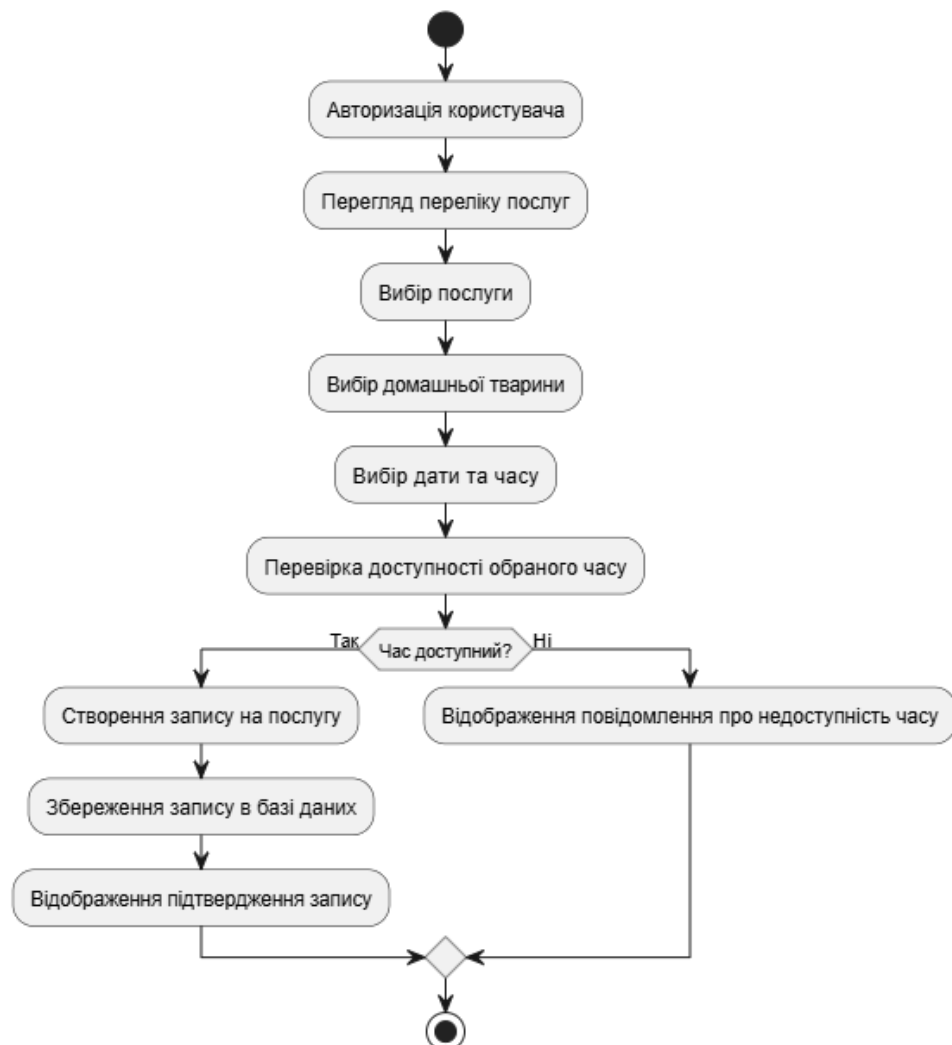


Рисунок 2.2 – Діаграма діяльності процесу створення запису на грумінг-послугу

З урахуванням визначених вимог обґрунтовано вибір технологій реалізації системи. Використання мови програмування Kotlin забезпечує сучасний підхід до розробки програмного забезпечення, підтримку об'єктно-орієнтованого та функціонального програмування, а також підвищену безпеку завдяки строгій типізації. Платформа Firebase надає засоби для реалізації серверної частини системи, зокрема автентифікацію користувачів, хмарне зберігання даних і синхронізацію в реальному часі, що дозволяє скоротити складність розробки та забезпечити надійність функціонування системи.

Таким чином, проведене виявлення та аналіз вимог дозволили сформулювати цілісне уявлення про функціональні можливості та якісні характеристики розроблюваної веб-системи керування грумінг-послугами. Отримані результати є основою для подальшого етапу проєктування архітектури системи та її практичної реалізації.

2.2 Вибір засобів, методів і алгоритмів вирішення поставленого завдання

Вибір засобів, методів і алгоритмів реалізації веб-системи керування грумінг-послугами тварин здійснюється з урахуванням особливостей предметної області та вимог до функціональності, продуктивності, масштабованості та надійності програмного продукту. Основною метою є забезпечення ефективної обробки користувацьких запитів, зручної взаємодії з системою та надійного зберігання даних.

Вибір технологій розробки здійснювався на основі таких критеріїв: швидкість створення програмного продукту, простота інтеграції, підтримка хмарної інфраструктури, безпека та можливість масштабування. У процесі аналізу розглядалися традиційні серверні рішення (Node.js, Django), а також хмарні платформи типу Firebase. Використання традиційних підходів передбачає необхідність розгортання та підтримки серверної частини, тоді як

Firebase забезпечує serverless-архітектуру, вбудовані механізми автентифікації та хмарне зберігання даних.

Для клієнтської частини системи обрано мову програмування Kotlin, яка забезпечує строгий контроль типів, підвищує надійність програмного коду та підтримує сучасні підходи до розробки. Поєднання Kotlin і Firebase дозволяє зменшити складність реалізації системи, скоротити час розробки та забезпечити ефективну взаємодію з хмарними сервісами [7, 8]

Архітектура системи побудована за принципом клієнт-серверної взаємодії з використанням проміжного рівня обробки запитів. Користувач взаємодіє із системою через веб-інтерфейс, який передає запити до програмної логіки. Обробка запитів здійснюється через механізм диспетчеризації, що забезпечує їх маршрутизацію до відповідних функціональних модулів системи. Такий підхід дозволяє підвищити гнучкість архітектури та забезпечити розділення відповідальності між компонентами.

Для вирішення прикладних задач у системі застосовуються алгоритми обробки користувацьких дій, серед яких ключовими є алгоритм створення запису на послугу та алгоритм зміни статусу запису. Алгоритм створення запису передбачає вибір послуги, визначення дати та часу, перевірку доступності інтервалу та збереження запису у базі даних, а також попередній етап пошуку відповідного грумінг-салону за заданими параметрами, такими як локація, дата та час обслуговування.

У разі недоступності обраного часу система формує повідомлення про необхідність вибору іншого інтервалу. Алгоритм зміни статусу запису включає перевірку прав доступу, вибір нового статусу, контроль допустимості переходу між станами та оновлення інформації в системі.

Система також забезпечує можливість перегляду та відстеження створених записів, що дозволяє користувачу контролювати стан обслуговування.

Для формалізації процесів взаємодії між компонентами використано UML-модельовання. Зокрема, діаграма послідовності процесу створення запису на грумінг-послугу відображає обмін повідомленнями між клієнтом, інтерфейсом, системою та базою даних (рис. 2.3). Вона демонструє послідовність виконання операцій, включаючи передачу даних, перевірку доступності часу, збереження запису та повернення результату користувачу.

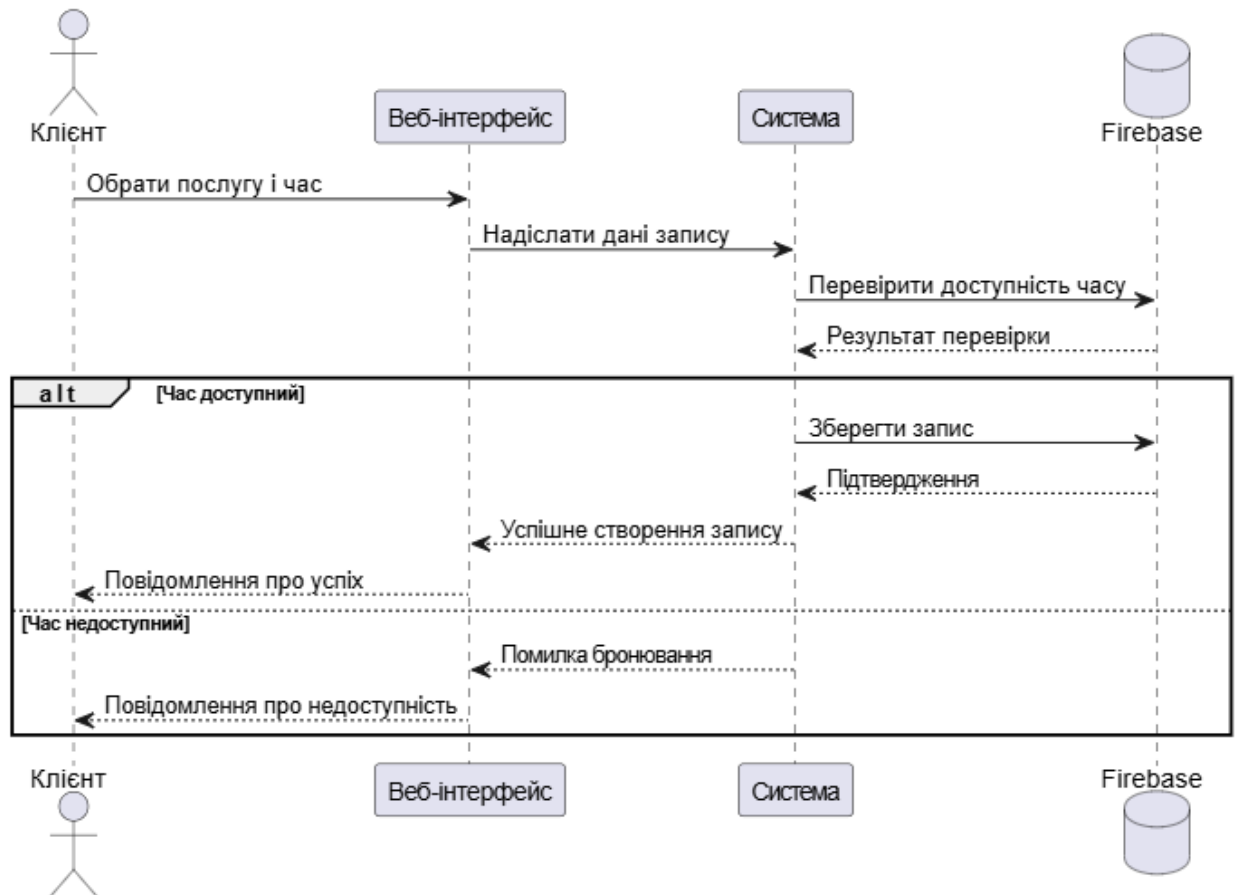


Рисунок 2.3 – UML-діаграма послідовності

Обрані засоби, методи та алгоритми забезпечують ефективну реалізацію веб-системи, дозволяють автоматизувати основні бізнес-процеси та створюють основу для подальшого розвитку і масштабування програмного продукту.

РОЗДІЛ 3

РОЗРОБКА ВЕБ-СИСТЕМИ КЕРУВАННЯ ГРУМІНГ-ПОСЛУГАМИ ТВАРИН З ВИКОРИСТАННЯМ KOTLIN ТА FIREBASE

3.1 Практична реалізація об'єкта проектування

Практична реалізація веб-системи керування грумінг-послугами здійснювалася відповідно до раніше визначених функціональних вимог, архітектурних рішень і обраного технологічного стеку. Для розробки клієнтської логіки та реалізації основних функціональних модулів використано мову програмування Kotlin, яка забезпечує строгий контроль типів, лаконічність коду та зручність підтримки програмного продукту. Для реалізації серверної інфраструктури, автентифікації користувачів і збереження даних застосовано хмарну платформу Firebase, зокрема сервіси Firebase Authentication та Cloud Firestore. Такий підхід дозволив створити цілісну інформаційну систему, у якій користувацький інтерфейс, механізми обробки дій користувача та операції з даними працюють узгоджено.

Практична реалізація системи виконувалася поетапно. На першому етапі було створено стартовий екран, який забезпечує початкову взаємодію користувача із системою. На рисунку 3.1 представлено головну сторінку, що виконує навігаційну функцію та надає доступ до основних сценаріїв роботи, зокрема до пошуку салону та переходу до авторизації. На цьому етапі засобами Kotlin було реалізовано базову логіку переходів між екранами, а також побудовано візуальні компоненти інтерфейсу. Головна сторінка виконує роль точки входу до системи та формує перше уявлення користувача про її структуру й призначення.

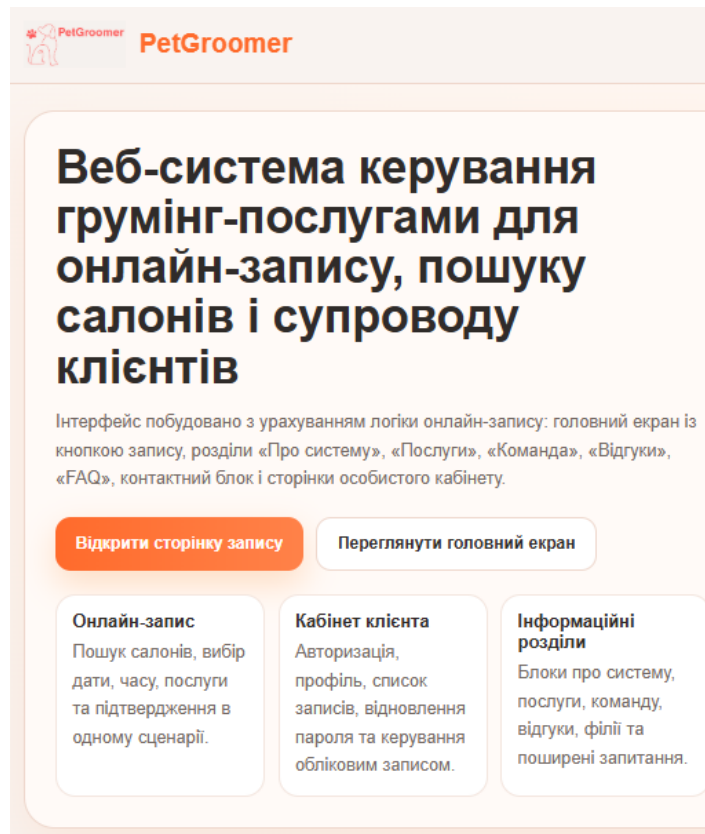
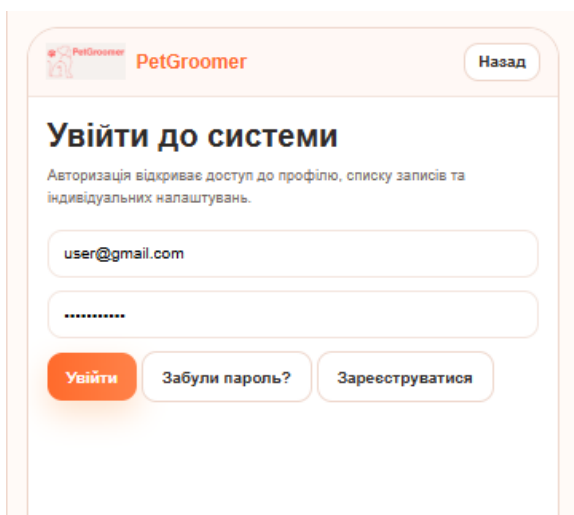


Рисунок 3.1 – Головна сторінка веб-системи керування грумінг-послугами

Наступним кроком стала реалізація механізму авторизації користувача. На рисунку 3.2 показано сторінку входу, на якій користувач вводить адресу електронної пошти та пароль для доступу до персоналізованих функцій системи. Цей модуль реалізовано із використанням сервісу Firebase Authentication, який забезпечує перевірку облікових даних та керування сеансом користувача.

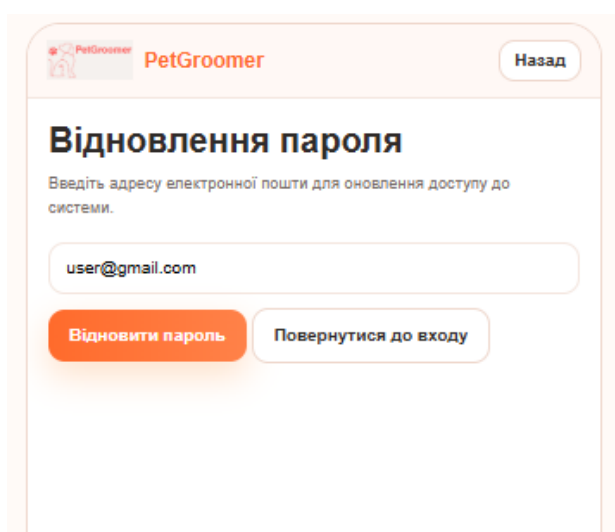
У програмній реалізації Kotlin відповідає за обробку події натискання кнопки входу, передавання введених даних до Firebase та подальше керування інтерфейсом залежно від результату авторизації. Таким чином, на цьому етапі було реалізовано не лише форму входу, а й повний ланцюг взаємодії між користувачем, клієнтською частиною та хмарним сервісом автентифікації.



The screenshot shows the login interface for 'PetGroomer'. At the top, there is a header with the PetGroomer logo and a 'Назад' (Back) button. The main heading is 'Увійти до системи' (Login to the system). Below it, a subtitle explains that authorization provides access to the profile, appointment list, and individual settings. There are two input fields: the first contains 'user@gmail.com' and the second is masked with dots. At the bottom, there are three buttons: 'Увійти' (Login) in orange, 'Забули пароль?' (Forgot password?) in white, and 'Зареєструватися' (Register) in white.

Рисунок 3.2 – Сторінка авторизації користувача

Важливою складовою практичної реалізації є коректна обробка помилок авторизації. На рисунку 3.3 представлено екран, який відображається у випадку введення некоректних облікових даних. Якщо Firebase Authentication повертає помилку, клієнтська логіка, реалізована засобами Kotlin, формує відповідне повідомлення для користувача та надає можливість повторного введення даних. Реалізація такого механізму є важливою з точки зору зручності використання системи, оскільки дозволяє користувачу своєчасно отримати зрозумілий зворотний зв'язок щодо причини невдалого входу.



The screenshot shows the password reset interface for 'PetGroomer'. At the top, there is a header with the PetGroomer logo and a 'Назад' (Back) button. The main heading is 'Відновлення пароля' (Reset password). Below it, a subtitle instructs the user to enter their email address to update access to the system. There is one input field containing 'user@gmail.com'. At the bottom, there are two buttons: 'Відновити пароль' (Reset password) in orange and 'Повернутися до входу' (Return to login) in white.

Рисунок 3.3 – Сторінка відновлення пароля

Після успішної авторизації користувач переходить до головного меню системи, яке показано на рисунку 3.4. Цей екран реалізовано як центральний вузол навігації між основними функціональними модулями. У меню передбачено переходи до профілю користувача, пошуку салонів, перегляду записів і допоміжних розділів. У межах практичної реалізації саме на цьому етапі було організовано керування станами інтерфейсу та маршрутизацію між окремими екранами. Використання Kotlin дозволило реалізувати цю логіку у вигляді структурованих функцій, що спрощує подальше супроводження та розширення системи.

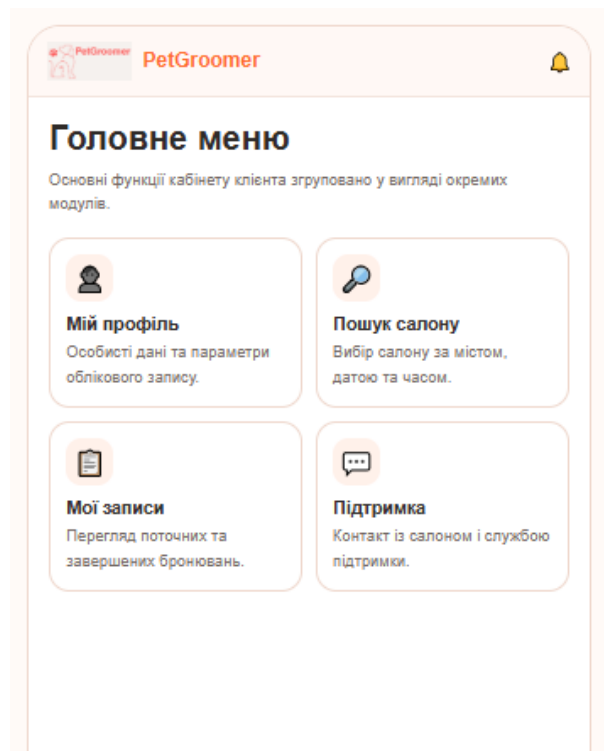


Рисунок 3.4 – Головне меню користувача

Одним із ключових модулів розробленої системи є пошук салону. На рисунку 3.5 представлено сторінку, де користувач може задати параметри пошуку, зокрема місто, дату, час та тип тварини. Реалізація цього функціоналу включає обробку введених даних на клієнтській стороні та підготовку запиту до бази даних. Для збереження й подальшого отримання

інформації про салони використовується Cloud Firestore, де дані структуровано у вигляді колекцій і документів. Kotlin у цьому модулі виконує функцію формування запиту, приймання результатів та оновлення стану інтерфейсу відповідно до отриманих даних. Окремо на цій сторінці реалізовано інтерактивний блок із картою, який покращує візуальне сприйняття результатів пошуку та розташування закладів.

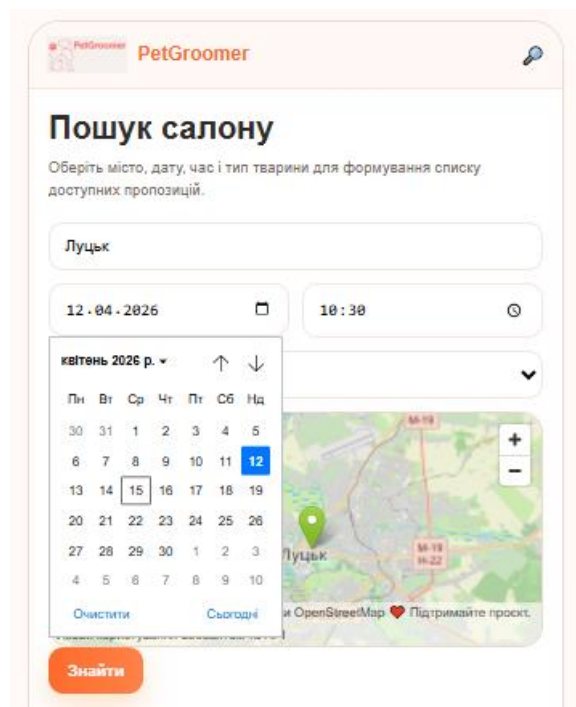


Рисунок 3.5 – Сторінка «Пошук салону»

Як показано у лістингу 3.1, функція пошуку обробляє введені параметри користувача та ініціює відображення результатів.

Лістинг 3.1 – Пошук салону

```
fun applySearch(city: String) {
    val searchSummary = city.isEmpty { "Луцьк" }
    updateSearchResults(searchSummary)
    showScreen("screen-results")
}
```

кінець лістингу 3.1

Така реалізація дозволяє забезпечити швидкий перехід до сторінки результатів пошуку без додаткових дій користувача.

Результати пошуку відображаються на окремій сторінці, що показано на рисунку 3.6. У цьому модулі система формує список салонів, які відповідають заданим критеріям. Кожна картка містить назву салону, адресу, рейтинг і короткий опис послуг. Практична реалізація цього екрану передбачає отримання даних із Cloud Firestore та їх динамічне відображення в інтерфейсі. Саме тут Kotlin забезпечує зв'язок між даними, отриманими з Firebase, і компонентами користувацького інтерфейсу. Такий підхід дозволяє формувати результати пошуку в реальному часі та забезпечує гнучкість у зміні вмісту сторінки залежно від параметрів запиту.

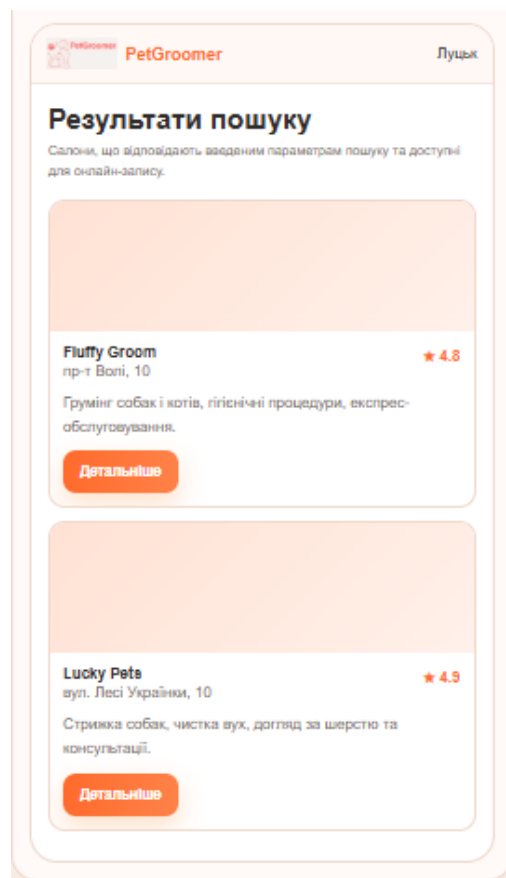


Рисунок 3.6 – Сторінка результатів пошуку салонів

Після вибору конкретного салону користувач переходить до сторінки детальної інформації, яка наведена на рисунку 3.7. На цьому екрані

реалізовано перегляд розширених відомостей про обраний заклад, перелік доступних послуг та можливість підтвердження запису. Під час практичної реалізації цей модуль було пов'язано з базою даних Firebase, що дозволяє завантажувати пов'язані дані про салон і відображати їх у структурованому вигляді. Логіка переходу до цієї сторінки, передавання ідентифікатора вибраного салону та відображення відповідних даних реалізована засобами Kotlin.

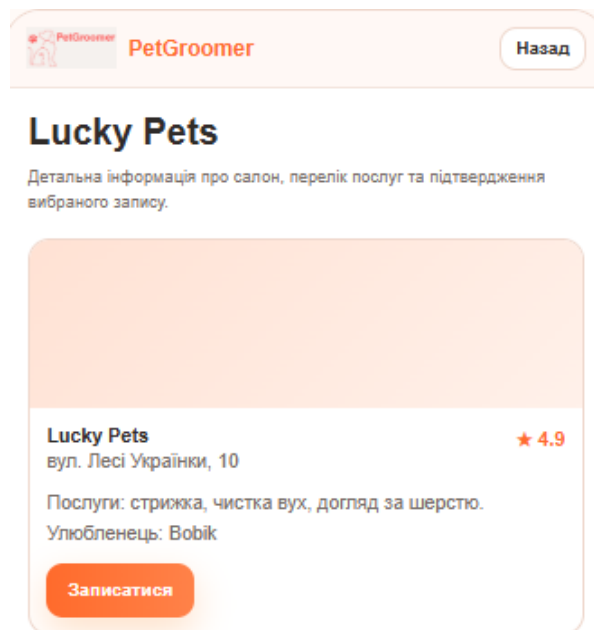


Рисунок 3.7 – Сторінка відомостей про обраний заклад

Наступним етапом є безпосереднє створення запису на послугу. Після підтвердження користувачем обраного часу та салону система формує новий запис і зберігає його у Cloud Firestore. Результат цієї операції відображається на рисунку 3.8, де показано повідомлення про успішне створення запису. У межах програмної реалізації Kotlin формує об'єкт запису, що містить інформацію про користувача, салон, послугу, дату, час та статус, а далі передає його до Firebase. Збереження даних у хмарному середовищі забезпечує доступність записів, їхню цілісність і можливість подальшого перегляду.

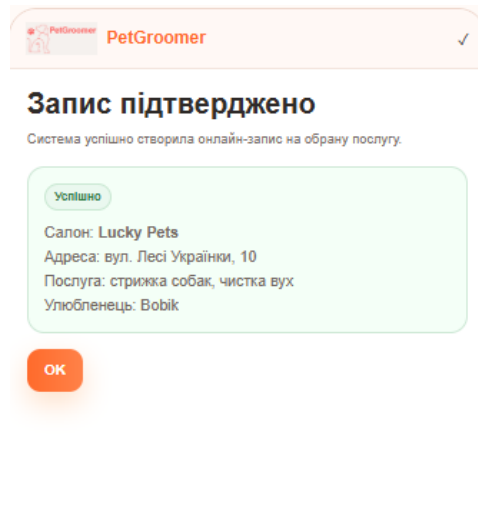


Рисунок 3.8 – Сторінка підтвердження запису

Окрему увагу під час практичної реалізації було приділено сторінці профілю користувача, яку наведено на рисунку 3.9. Цей екран містить персональні дані користувача, службову інформацію про обліковий запис, а також навігацію до пов'язаних дій, наприклад перегляду записів чи редагування профілю. Дані профілю завантажуються з Firebase і відображаються в інтерфейсі засобами Kotlin. Завдяки цьому профіль користувача виступає не лише інформаційною сторінкою, а й важливим елементом персоналізації системи.

Як наведено у лістингу 3.2, створення запису реалізується шляхом формування об'єкта та його збереження у хмарній базі даних Firebase.

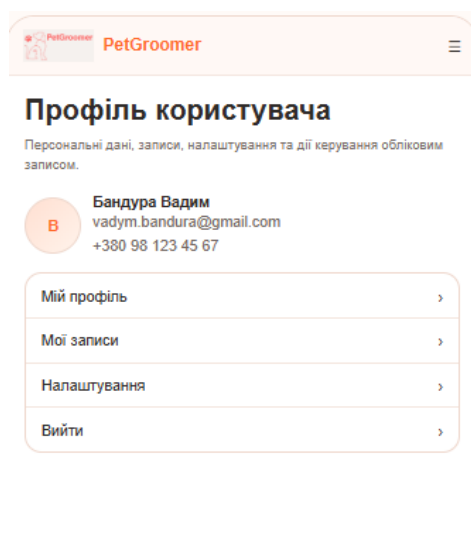


Рисунок 3.9 – Сторінка даних користувача

Лістинг 3.2 – Створення запису у Firebase

```
fun createBooking(userId: String, salonId: String, petName: String) {  
    val booking = hashMapOf(  
        "userId" to userId,  
        "salonId" to salonId,  
        "petName" to petName,  
        "status" to "confirmed"  
    )  
    FirebaseFirestore.getInstance()  
        .collection("appointments")  
        .add(booking)  
        .addOnSuccessListener {  
            showScreen("screen-success")  
        }  
}
```

кінець лістингу 3.2

Застосування хмарної бази даних забезпечує надійне збереження інформації та можливість її подальшого використання у системі.

Таким чином, практична реалізація веб-системи керування грумінг-послугами виконана як послідовна інтеграція користувацького інтерфейсу, клієнтської логіки на базі Kotlin та хмарної інфраструктури Firebase. Використання Kotlin забезпечило структурованість програмного коду, коректну обробку подій і зручність підтримки застосунку, тоді як Firebase дозволив реалізувати автентифікацію, збереження даних і хмарну взаємодію без розгортання окремої серверної частини. У результаті було створено функціонально завершену систему, яка підтримує основні сценарії роботи користувача: вхід до системи, пошук салону, вибір послуги, створення запису, перегляд профілю та обробку помилок.

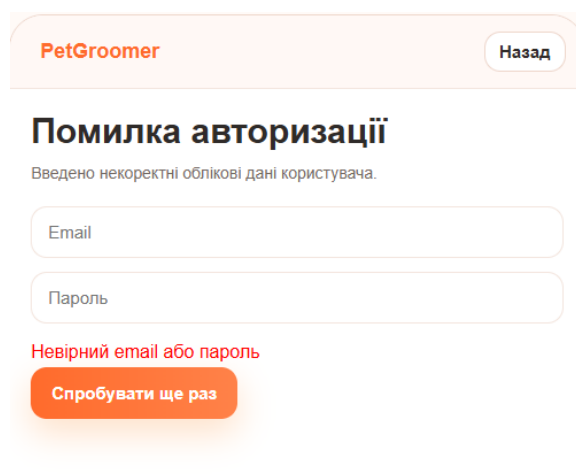
3.2 Тестування та налагодження інформаційно-комп'ютерної системи

Тестування розробленої веб-системи керування грумінг-послугами тварин є завершальним етапом її реалізації та спрямоване на перевірку коректності функціонування, відповідності вимогам, а також виявлення можливих помилок і недоліків у роботі програмного продукту.

Проведення тестування здійснювалося відповідно до базових принципів програмної інженерії, що передбачають систематичну перевірку функціональних можливостей системи, її надійності та зручності використання [9, 10].

Тестування проводилося на реальних сценаріях використання системи, зокрема авторизації користувача, створення запису на послугу та перегляду результатів пошуку. Перевірка здійснювалася шляхом введення як коректних, так і некоректних даних, що дозволило оцінити роботу механізмів валідації. Усі основні функції системи продемонстрували коректну роботу без критичних помилок.

Рисунок 3.10 демонструє обробку некоректних вхідних даних під час авторизації. Система відображає повідомлення про помилку, що підтверджує наявність механізмів валідації.



The screenshot displays a web interface for 'PetGroomer'. At the top, there is a header bar with the 'PetGroomer' logo on the left and a 'Назад' (Back) button on the right. Below the header, the main content area features a title 'Помилка авторизації' (Authentication Error) in bold. Underneath the title is a message: 'Введено некоректні облікові дані користувача.' (Incorrect login credentials entered). There are two input fields: 'Email' and 'Пароль' (Password). Below these fields, a red error message states: 'Невірний email або пароль' (Incorrect email or password). At the bottom of the form, there is an orange button labeled 'Спробувати ще раз' (Try again).

Рисунок 3.10 – Обробка помилки авторизації

На рисунку 3.11 представлено фрагмент бази даних Firebase Firestore, що ілюструє збереження запису користувача у колекції appointments. Схема відображає вибір відповідної колекції, документ запису та набір його основних полів [11]. Представлена структура підтверджує, що система зберігає ідентифікатори користувача, тварини, послуги, салону, дату та час обслуговування, а також поточний статус запису. Такий підхід забезпечує впорядковане зберігання інформації та спрощує подальшу обробку даних у системі.

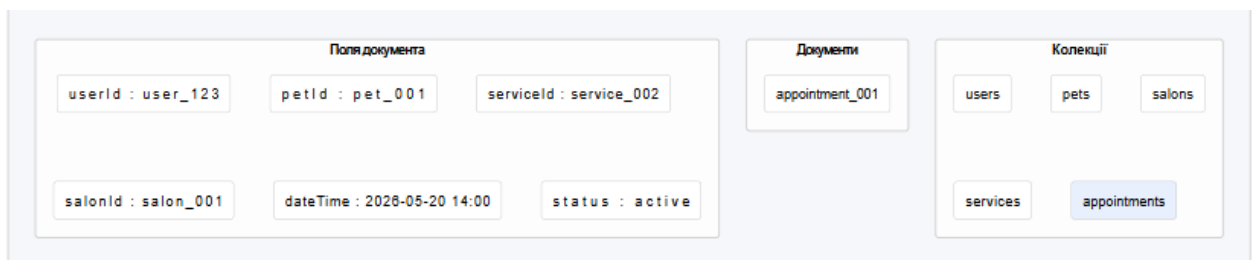


Рисунок 3.11 – Фрагмент БД Firebase Firestore з прикладом документа у колекції appointments

У результаті проведеного тестування та налагодження встановлено, що розроблена веб-система керування грумінг-послугами тварин коректно реалізує основні функціональні можливості, забезпечує стабільну роботу та належну обробку користувацьких дій. Виявлені в процесі тестування незначні недоліки були усунуті, що дозволило підвищити надійність і зручність використання системи. Отримані результати підтверджують відповідність програмного продукту поставленим вимогам та його готовність до практичного впровадження у сфері надання грумінг-послуг.

3.3 Реалізація бази даних

Реалізація бази даних веб-системи керування грумінг-послугами тварин здійснювалася з використанням хмарної NoSQL бази даних Firebase Firestore,

яка забезпечує зберігання, обробку та синхронізацію даних у режимі реального часу [12, 13]. Використання Firestore дозволяє реалізувати масштабовану, відмовостійку та гнучку структуру зберігання даних без необхідності розгортання власної серверної інфраструктури.

На відміну від реляційних баз даних, Firestore використовує документно-орієнтовану модель, де дані організовані у вигляді колекцій (collections) та документів (documents). Кожен документ містить набір полів у форматі «ключ–значення» та може включати вкладені структури.

У процесі проєктування бази даних було визначено основні сутності предметної області, які відповідають ключовим бізнес-процесам системи. До них належать:

- користувачі системи;
- грумінг-салони;
- послуги;
- записи на обслуговування;
- домашні тварини.

З урахуванням зазначених сутностей було сформовано структуру бази даних, що складається з кількох основних колекцій.

Рисунок 3.12 відображає структуру бази даних Firebase Firestore, реалізованої для зберігання даних веб-системи керування грумінг-послугами тварин. Основними колекціями є users, pets, salons, services та appointments. Центральне місце у структурі займає колекція appointments, яка забезпечує зв'язок між користувачами, тваринами, салонами та послугами. Зв'язки між сутностями реалізовано через збереження ідентифікаторів відповідних документів, що відповідає принципам проєктування NoSQL баз даних. Така структура забезпечує гнучкість, масштабованість і зручність обробки даних у межах розробленої системи.

У наведеному фрагменті коду (ліст. 3.3) реалізовано створення нового запису на грумінг-послугу. Дані передаються у вигляді асоціативного масиву,

після чого зберігаються у відповідній колекції бази даних. Такий підхід дозволяє забезпечити гнучкість структури даних та простоту їх обробки.

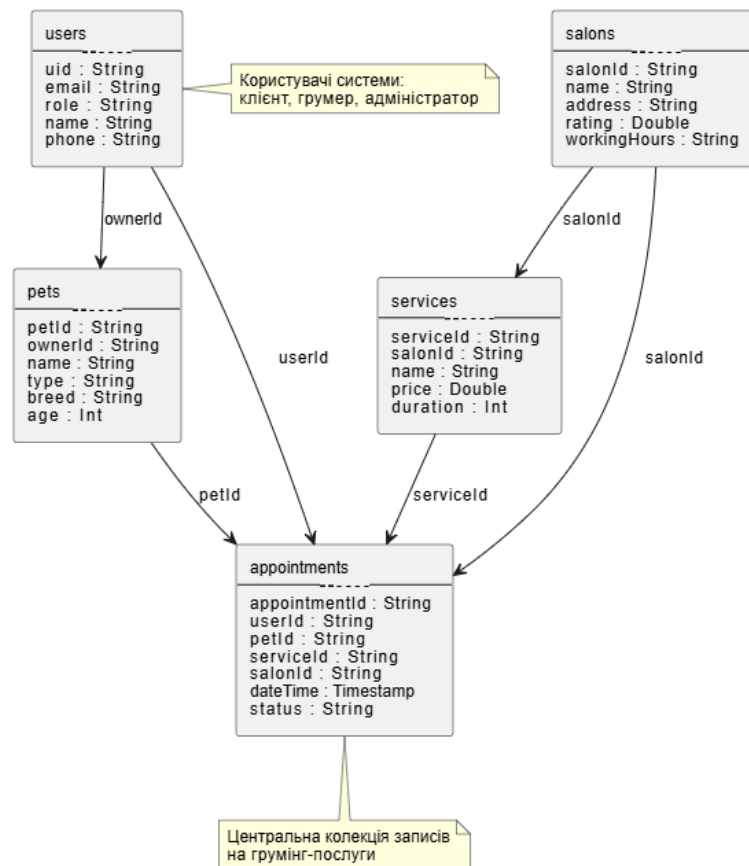


Рисунок 3.12 – Структура БД

Лістинг 3.3 – Створення запису в Firestore

```

val appointment = hashMapOf(
    "userId" to userId,
    "petId" to petId,
    "serviceId" to serviceId,
    "salonId" to salonId,
    "dateTime" to selectedDate,
    "status" to "active"
)

```

```
FirestoreFirestore.getInstance()  
    .collection("appointments")  
    .add(appointment)
```

кінець лістингу 3.3

Реалізація бази даних із використанням Firebase Firestore забезпечує ефективне зберігання та обробку даних у веб-системі керування грумінг-послугами. Обрана структура дозволяє підтримувати основні бізнес-процеси системи, забезпечує масштабованість, гнучкість та надійність функціонування програмного продукту.

3.4 Оцінка ефективності та результатів роботи веб-системи

Оцінка ефективності розробленої веб-системи керування грумінг-послугами здійснювалася з метою визначення рівня її функціональної завершеності, відповідності поставленим вимогам та практичної придатності для використання у сфері надання послуг.

У процесі оцінювання було проаналізовано основні характеристики системи, зокрема швидкодію, стабільність функціонування, зручність користування та коректність виконання ключових операцій. Проведене тестування показало, що система забезпечує виконання базових сценаріїв взаємодії користувача, таких як авторизація, пошук салонів та створення запису, без затримок і помилок за умови стабільного мережевого з'єднання.

Підтвердження коректності роботи системи можна спостерігати на прикладі успішного завершення процесу створення запису. Як показано на рисунку 3.13, після обробки введених користувачем даних система формує повідомлення про успішне бронювання послуги із зазначенням ключових параметрів, зокрема дати, часу, обраного салону, типу послуги та статусу запису. Це свідчить про узгоджену взаємодію клієнтської частини,

реалізованої засобами Kotlin, із хмарною інфраструктурою Firebase, яка забезпечує збереження даних у базі Cloud Firestore.

Крім того, на представленому рисунку відображено унікальний ідентифікатор запису та статус виконання, що підтверджує успішність операції створення запису та дозволяє використовувати ці дані для подальшого контролю та обробки. Такий підхід забезпечує цілісність даних та підвищує рівень довіри користувача до системи.

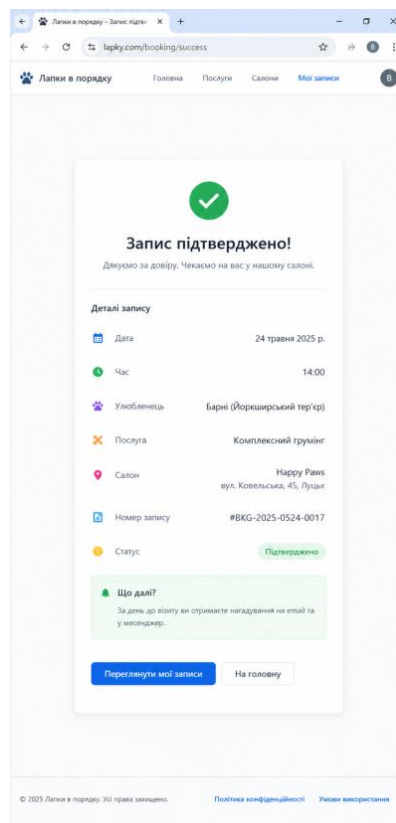


Рисунок 3.13 – Підтвердження успішного створення запису на грумінг-послугу

З точки зору зручності використання, інтерфейс сторінки підтвердження запису є інтуїтивно зрозумілим та логічно структурованим. Візуальне виділення статусу успішної операції, а також наявність кнопок переходу до перегляду записів або повернення на головну сторінку сприяє покращенню користувацького досвіду.

Результати оцінки ефективності підтверджують, що розроблена система є функціонально завершеним програмним продуктом, який забезпечує автоматизацію процесу запису на грумінг-послуги, підвищує якість обслуговування клієнтів та може бути рекомендований для практичного використання.

ВИСНОВКИ

У кваліфікаційній роботі бакалавра вирішено актуальну науково-прикладну задачу розробки веб-системи керування грумінг-послугами тварин, що спрямована на підвищення ефективності організації обслуговування клієнтів та автоматизацію основних бізнес-процесів у сфері догляду за домашніми тваринами. Актуальність теми зумовлена зростанням попиту на грумінг-послуги та необхідністю цифровізації діяльності малих і середніх підприємств, що працюють у цій сфері.

У процесі виконання роботи досягнуто поставленої мети шляхом комплексного аналізу предметної області, обґрунтування технічних рішень та реалізації функціональної веб-системи із використанням сучасних технологій програмування.

За результатами проведеного дослідження отримано такі основні результати. Здійснено аналіз предметної області та сучасних програмних рішень у сфері грумінг-послуг, зокрема систем MoeGo та GrooMore, що дозволило визначити їх функціональні можливості, переваги та недоліки, а також обґрунтувати необхідність створення адаптованої веб-системи для локального використання.

Визначено та формалізовано функціональні й нефункціональні вимоги до системи, що охоплюють процеси реєстрації користувачів, управління записами, ведення бази даних клієнтів і тварин, планування розкладу роботи та забезпечення безпечного доступу до даних.

Обґрунтовано вибір технологій розробки, зокрема використання мови програмування Kotlin та платформи Firebase, що забезпечують високу продуктивність, надійність, масштабованість системи та спрощують реалізацію серверної частини за рахунок використання хмарної інфраструктури.

Спроектовано архітектуру веб-системи з використанням клієнт-серверного підходу та побудовано UML-діаграми, які відображають

структуру системи, взаємодію компонентів та основні сценарії роботи користувачів.

Розроблено структуру бази даних на основі Firebase Firestore, визначено ключові колекції (users, pets, salons, services, appointments), що забезпечують логічну організацію даних та підтримують основні бізнес-процеси системи.

Реалізовано програмну частину веб-системи, яка включає модулі авторизації користувачів, пошуку грумінг-салонів, формування записів на послуги, управління даними та взаємодії з хмарною базою даних.

Проведено тестування та налагодження системи, що дозволило виявити та усунути помилки, пов'язані з обробкою вхідних даних, відображенням повідомлень та синхронізацією даних, а також підтвердило стабільність функціонування програмного продукту.

У результаті виконання роботи розроблено веб-систему, яка забезпечує автоматизацію процесів запису на грумінг-послуги, централізоване зберігання інформації та зручну взаємодію користувачів із системою. Використання сучасних технологій дозволило створити гнучке та масштабоване рішення, яке може бути адаптоване до потреб різних закладів сфери обслуговування.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. McDonald S.E. et al. Barriers to and facilitators of pet grooming among clients // PMC, 2022. URL: <https://doi.org/10.3389/fvets.2022.1021707> (дата звернення: 11.03.2026).
2. Yixuan Feng. Digital Pet Product & Service Platforms: The Diversity of the Industry. DOI:10.2991/aebmr.k.220307.466 (дата звернення: 11.03.2026).
3. Rheyldord Vincent Dizon, Jonathan James De Guzman, Mary Rose Cuyos, Jrkin Tacuyo, Sharmaine J. Maglapuz, Ryan Azur. A Web-Based Management System Utilizing Descriptive Analytics and Scheduling Algorithms for SJOE Pet Supply. URL: <https://doi.org/10.51584/IJRIAS.2025.1010000056> (дата звернення: 13.03.2026).
4. MoeGo: All-in-One Pet Business Software. URL: <https://www.moego.pet> (дата звернення: 13.03.2026).
5. Система GrooMore. URL: <https://www.groomore.com/> (дата звернення: 18.04.2026).
6. UML Specification. Object Management Group. URL: <https://www.omg.org/spec/UML/> (дата звернення: 21.04.2026).
7. Firebase Documentation. Cloud Firestore <https://firebase.google.com/docs/firestore> (дата звернення: 25.04.2026).
8. Firebase Authentication Documentation. URL: <https://firebase.google.com/docs/auth> (дата звернення: 25.04.2026).
9. Newman S. Building Microservices: Designing Fine-Grained Systems. 2nd Edition. O'Reilly, 2022. 618 p.
10. Pressman R., Maxim B. Software Engineering: A Practitioner's Approach. 9th ed. McGraw-Hill, 2021.
11. Google Developers. Firestore Data Model. URL: <https://firebase.google.com/docs/firestore/data-model> (дата звернення: 11.05.2026).

12. Kleppmann M. Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems.
URL: <https://dataintensive.net> (дата звернення: 01.04.2026).

13. Sadalage P., Fowler M. Evolutionary Database Design
URL: <https://martinfowler.com/articles/evodb.html> (дата звернення: 14.05.2026).