

БАКАЛАВРСЬКА РОБОТА

БР. ІІ - 02.00.00.000 ІІЗ

Група ІІ-22-4

Витрикуш Анастасія

2026

Івано-Франківський національний технічний університет нафти і газу

Факультет інформаційних технологій

Кафедра інженерії програмного забезпечення

Витрикуш Анастасія Андріївна

(прізвище, ім'я, по батькові)

УДК 004.4
(індекс)

БАКАЛАВРСЬКА РОБОТА

Розробка інформаційної системи управління об'єктами

нерухомості

(назва роботи)

Інженерія програмного забезпечення

(назва освітньої програми)

121 - Інженерія програмного забезпечення

(шифр і назва спеціальності)

Робота містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело

Здобувач освітнього рівня Витрикуш А.А.
(підпис, ініціали та прізвище здобувача)

Науковий керівник Піх Марія Михайлівна, асистент
(підпис, прізвище, ім'я, по батькові, науковий ступінь, вчене звання керівника)

Допущено до захисту

Завідувач кафедри

доц. Бандура В.В.
(посада) (підпис) (дата) (ініціали та прізвище)

Івано-Франківськ – 2026

Івано-Франківський національний технічний університет нафти і газу

Інститут, факультет інформаційних технологій

Кафедра інженерії програмного забезпечення

Ступінь вищої освіти бакалавр

Спеціальність 121 – Інженерія програмного забезпечення

ЗАТВЕРДЖУЮ:

Зав. кафедрою ІІЗ

доц.

В.В. Бандура

“ ” 2026 р.

ЗАВДАННЯ

НА БАКАЛАВРСЬКУ РОБОТУ СТУДЕНТОВІ

Витрикуш Анастасії Андріївні

(прізвище, ім'я, по-батькові)

1. Тема проекту (роботи) “ Розробка інформаційної системи управління об'єктами нерухомості”

керівник проекту (роботи) Піх Марія Михайлівна , асистент

затверджені наказом закладу вищої освіти від “ 21 ” квітня 2026р. № 179/7

2. Строк подання студентом проекту (роботи) 09 червня 2026 р.

3. Вихідні дані до проекту (роботи) Результати і матеріали отримані під час проходження переддипломної практики

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

1. Аналіз проблематики розробки інформаційних систем для управлінн об'єктами нерухомості

2. Методологія розробки та алгоритмічне забезпечення системи управління нерухомістю

3. Архітектурна організація та концептуальна модель взаємодії компонентів системи

4. Інфологічне моделювання та структуризація бази даних системи

5. Розробка інформаційної системи управління нерухомістю

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)

1. Концептуальна модель системи управління нерухомістю (рис. 1.1, ст. 15)

2. Компонентна структура системи (рис. 1.2, ст. 17)

3. Деталізація обмежень за ролями в системі управління нерухомістю (рис. 1.3, ст. 23)

4. Інтерфейс системи Yardi Voyager (рис. 1.4, ст.25)

5. Дашборд системи MRI Software (рис. 1.5, ст.26)

6. Консультанти розділів проекту (роботи)

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв

7. Дата видачі завдання 2026 р.

Керівник

_____ (підпис)

Завдання прийняв до виконання

_____ (підпис)

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів проекту	Примітка
1	Аналіз проблематики розробки інформаційних систем для управлінн об'єктами нерухомості	04.02.2026	виконано
2	Методологія розробки та алгоритмічне забезпечення системи управління нерухомістю	15.03.2026	виконано
3	Архітектурна організація та концептуальна модель взаємодії компонентів системи	21.04.2026	виконано
4	Інфологічне моделювання та структуризація бази даних системи	28.05.2026	виконано
5	Розробка інформаційної системи управління нерухомістю	03.06.2026	виконано
6	Оформлення пояснювальної записки бакалаврської роботи завідувачем кафедри	09.06.2026	виконано

Студент – дипломник _____ Витрикуш А.А.
(підпис)

Керівник роботи _____ Піх М.М.
(підпис)

АНОТАЦІЯ

Бакалаврська робота містить 76 сторінок, 20 рисунків, список використаних джерел із 38 найменуваннями.

Метою роботи є розробка інформаційної системи управління об'єктами нерухомості, що забезпечує автоматизацію бізнес-процесів, ефективне управління даними, підтримку аналітики та безпечний доступ до інформаційних ресурсів.

Об'єкт дослідження - процеси управління об'єктами нерухомості в умовах використання інформаційних технологій.

Предмет дослідження - методи, моделі та засоби проектування і реалізації інформаційних систем управління нерухомістю, включаючи архітектурні рішення, моделі доступу та аналітичні інструменти.

В першому розділі проведений аналіз дозволив обґрунтувати необхідність розробки інтегрованої інформаційної системи управління нерухомістю та визначити ключові вимоги до її архітектури.

В другому розділі запропонована методологія та архітектурні рішення забезпечують ефективну реалізацію системи з урахуванням сучасних вимог до програмного забезпечення.

В третьому розділі реалізовано інформаційну систему підтвердила доцільність обраних підходів і забезпечила ефективне управління об'єктами нерухомості.

Висновок: розроблено інформаційну систему яка є ефективним інструментом автоматизації управління об'єктами нерухомості та відповідає сучасним вимогам до функціональності, безпеки та аналітичної підтримки.

КЛЮЧОВІ СЛОВА: ІНФОРМАЦІЙНА СИСТЕМА, УПРАВЛІННЯ НЕРУХОМІСТЮ, АВТОМАТИЗАЦІЯ БІЗНЕС-ПРОЦЕСІВ, RBAC, БАЗА ДАНИХ, БІЗНЕС-АНАЛІТИКА, ІНФОРМАЦІЙНА БЕЗПЕКА.

ANNOTATION

The bachelor's thesis contains 76 pages, 20 figures, a list of used sources with 38 names.

The method of work is the development of an information system for managing real estate objects, which provides automation of business processes, effective data management, support for analytics and secure access to information resources.

The object study is the processes of managing real estate objects in the conditions of using information technologies.

The subject of the study is methods, models and means of designing and implementing real estate management information systems, including architectural solutions, access models and analytical tools.

In the first section, the analysis made it possible to substantiate the need to develop an integrated real estate management information system and fulfill the key requirements for its architecture.

In the second section, the proposed methodology and architectural solutions ensure effective implementation of the system taking into account modern requirements for software.

In the third section, the implemented information system confirmed the feasibility of the selected approaches and ensured effective management of real estate objects.

Conclusion: an information system has been developed that is an effective tool for automating real estate management and meets modern requirements for functionality, security, and analytical support.

KEYWORDS: INFORMATION SYSTEM, REAL ESTATE MANAGEMENT, BUSINESS PROCESS AUTOMATION, RBAC, DATABASE, BUSINESS ANALYTICS, INFORMATION SECURITY.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ	9
ВСТУП.....	10
РОЗДІЛ 1. АНАЛІЗ ПРОБЛЕМАТИКИ РОЗРОБКИ ІНФОРМАЦІЙНИХ СИСТЕМ ДЛЯ УПРАВЛІННЯ ОБ'ЄКТАМИ НЕРУХОМОСТІ.....	14
1.1. Обґрунтування впровадження інформаційних систем в управлінні об'єктами нерухомості та автоматизації бізнес-процесів	14
1.1.1. Функціональна структура інтегрованої системи	14
1.1.2. Особливості та переваги пропонованої системи	15
1.2. Проектування роль-орієнтованої архітектури та інтерфейсів доступу до системи управління нерухомістю	16
1.2.1. Компонентна структура та функціональні інтерфейси	17
1.2.2. Система підтримки прийняття рішень та аналітичний інструментарій	18
1.3. Архітектурні принципи синхронізації даних та взаємодії мобільних терміналів із центральним сховищем	19
1.4. Формалізація та алгоритмічна реалізація моделі RBAC у багаторівневій системі.....	20
1.4.1. Алгоритмічні особливості для різних типів застосунків	21
1.4.2. Матриця доступу та специфікація повноважень суб'єктів системи	22
1.5. Огляд та порівняльний аналіз існуючих інформаційних систем управління нерухомістю та активами.....	24
1.5.1. ERP-платформа для управління нерухомістю Yardi Voyager	24
1.5.2. Система MRI Software	26
1.5.3. SaaS-рішення AppFolio Property Manager	26
1.5.4. Платформа Buildium	27

					БР.ІП – 02.00.00.000 ПЗ						
Змн.	Арк.	№ докум.	Підпис	Дата	Розробка інформаційної системи управління об'єктами нерухомості			Літ.	Арк.	Акрушіє	
Розроб.		Витрикуш А.А.									
Перевір.		Піх М.М.								6	
Реценз.		Саманів Л.В.						ІФНТУНГ ІП-22-4			
Н. Контр.		Піх М.М.									
Затверд.		Бандура В.В.									
					Пояснювальна записка						

1.5.5. Платформа Entrata	28
1.5.6. Синтез результатів аналізу	29
1.6 Висновки по розділу	30

РОЗДІЛ 2. МЕТОДОЛОГІЯ РОЗРОБКИ ТА АЛГОРИТМІЧНЕ

ЗАБЕЗПЕЧЕННЯ СИСТЕМИ УПРАВЛІННЯ НЕРУХОМІСТЮ 31

2.1. Методологія та технологічний цикл розробки системи	31
2.1.1. Обґрунтування вибору методології життєвого циклу розробки	31
2.1.2. Архітектурна концепція та багаторівневий дизайн	31
2.1.3. Технологічний процес розробки та кастомізації	32
2.1.4. Стратегія верифікації та тестування системи.....	33
2.2. Архітектурна організація та концептуальна модель взаємодії компонентів системи	34
2.2.1. Багаторівнева структура системної взаємодії	34
2.2.2. Механізми розширення та кастомізації	35
2.3. Інфологічне моделювання та структуризація бази даних системи	36
2.3.1. Класифікація та функціональне призначення ключових сутностей	37
2.3.2. Аналіз топології зв'язків та опис моделі бази даних	37
2.3.3. Системне значення логічної моделі	40
2.4. Моделювання інформаційно-динамічних процесів засобами діаграм потоків даних.....	41
2.4.1. Опис рівнів абстракції інформаційних потоків	41
2.4.2. Переваги потокової моделі.....	45
2.5. Аналіз інформаційних потоків та механізмів міжрівневої взаємодії....	45
2.6. Висновки по розділу	47

РОЗДІЛ 3. РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ УПРАВЛІННЯ

НЕРУХОМІСТЮ 48

3.1. Концептуальні засади та функціональна декомпозиція інформаційної системи управління нерухомістю	48
---	----

					БР.ІП – 02.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		7

3.2. Архітектура інформаційної безпеки та ієрархія прав доступу в системі управління нерухомістю	50
3.2.1. Стратегічні цілі системи безпеки	51
3.2.2. Специфікація ролей та матриця повноважень	51
3.2.3. Регламенти та політики безпеки даних.....	52
3.3. Розробка та опис інформаційної панелі системи управління нерухомістю	53
3.4. Сценарій операційної взаємодії при виявленні критичних девіацій фінансових показників	55
3.5. Реалізація модуля управління активами	57
3.6. Розробка та опис модуля бізнес-аналітики	60
3.7. Алгоритмічна база та методологія розрахунку показника середньої дохідності.....	63
3.8. Програмна реалізація модуля аналітики	65
3.9 Висновки до розділу	69

ВИСНОВКИ	71
-----------------------	-----------

ПЕРЕЛІК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ	73
--	-----------

БІБЛІОГРАФІЧНА ДОВІДКА

					БР.ІП – 02.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		8

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

DLP – Data Loss Prevention

EMI – Equated Monthly Installment

ERP – Enterprise Resource Planning

GUID – Globally Unique Identifier

KPI – Key Performance Indicator

NOI – Net Operating Income

OPEX – Operating Expenses

PMS – Property Management System

RBAC – Role-Based Access Control

REMS – Real Estate Management System

RLS – Row-Level Security

ROI – Return on Investment

SLA – Service Level Agreement

SSO – Single Sign-On

					БР.ІП – 02.00.00.000 ПЗ	Арк.
						9
Змн.	Арк.	№ докум.	Підпис	Дата		

ВСТУП

Сучасний етап розвитку інформаційних технологій характеризується активним впровадженням цифрових рішень у всі сфери економічної діяльності, зокрема у сферу управління об'єктами нерухомості. Зростання обсягів даних, ускладнення бізнес-процесів, необхідність оперативного прийняття рішень та підвищення вимог до прозорості управління обумовлюють потребу у створенні ефективних інформаційних систем. Традиційні підходи до управління нерухомістю, що базуються на фрагментованих інструментах або ручній обробці даних, не забезпечують належного рівня автоматизації та інтеграції процесів. Це призводить до зниження ефективності управління, підвищення ризиків та втрати конкурентних переваг.

У зв'язку з цим актуальним є розроблення комплексних інформаційних систем, які забезпечують централізоване управління об'єктами нерухомості, інтеграцію даних, підтримку аналітики та прийняття управлінських рішень. Особливого значення набуває використання сучасних архітектурних підходів, таких як багаторівнева організація системи, роль-орієнтоване управління доступом та інтеграція аналітичних модулів. Важливим аспектом є також забезпечення інформаційної безпеки, масштабованості та гнучкості системи.

Дана бакалаврська робота присвячена розробці інформаційної системи управління об'єктами нерухомості, яка поєднує в собі сучасні підходи до проєктування програмного забезпечення, методи моделювання даних і процесів, а також інструменти бізнес-аналітики. У роботі досліджено теоретичні засади, обґрунтовано архітектурні рішення та реалізовано програмний продукт, що відповідає сучасним вимогам до інформаційних систем даного класу.

					БР.ІП – 02.00.00.000 ПЗ	Арк.
						10
Змн.	Арк.	№ докум.	Підпис	Дата		

Актуальність роботи

Актуальність теми зумовлена стрімким розвитком ринку нерухомості та необхідністю підвищення ефективності управління активами в умовах цифрової трансформації економіки. Зростання кількості об'єктів нерухомості, ускладнення фінансових операцій та потреба в оперативному аналізі даних створюють передумови для впровадження сучасних інформаційних систем. Відсутність інтегрованих рішень або використання застарілих систем призводить до фрагментації інформації, ускладнення контролю та зниження якості управлінських рішень.

Особливої уваги потребує питання забезпечення інформаційної безпеки та розмежування доступу до даних у багатокористувацькому середовищі. Використання роль-орієнтованих моделей доступу дозволяє ефективно керувати правами користувачів і мінімізувати ризики несанкціонованого доступу. Крім того, сучасні системи управління нерухомістю повинні забезпечувати аналітичну підтримку, що дозволяє здійснювати прогнозування та оцінку ефективності використання активів.

Таким чином, розробка інтегрованої інформаційної системи управління об'єктами нерухомості є актуальним науково-прикладним завданням, спрямованим на підвищення ефективності управління, автоматизацію процесів та покращення якості прийняття рішень.

Метою роботи є розробка інформаційної системи управління об'єктами нерухомості, що забезпечує автоматизацію бізнес-процесів, ефективне управління даними, підтримку аналітики та безпечний доступ до інформаційних ресурсів.

Об'єкт дослідження - процеси управління об'єктами нерухомості в умовах використання інформаційних технологій.

Предмет дослідження - методи, моделі та засоби проектування і реалізації інформаційних систем управління нерухомістю, включаючи архітектурні рішення, моделі доступу та аналітичні інструменти.

					БР.ІП – 02.00.00.000 ПЗ	Арк.
						11
Змн.	Арк.	№ докум.	Підпис	Дата		

Завдання дослідження:

1. Проаналізувати сучасний стан та проблематику інформаційних систем управління нерухомістю.
2. Обґрунтувати архітектурні та технологічні підходи до розробки системи.
3. Розробити модель даних та інформаційних потоків системи.
4. Реалізувати механізми роль-орієнтованого доступу та інформаційної безпеки.
5. Створити функціональні модулі управління активами та бізнес-аналітики.
6. Розробити алгоритмічне забезпечення для оцінки ефективності управління нерухомістю.

Методи дослідження

У роботі використано:

- методи системного аналізу,
- об'єктно-орієнтованого проєктування,
- моделювання даних (ER-моделі),
- функціонального моделювання (DFD-діаграми),
- методи аналізу інформаційних потоків,
- алгоритмічні методи обробки даних,
- методи розробки програмного забезпечення на основі сучасних життєвих циклів.

Наукова новизна

Наукова новизна роботи полягає у розробці комплексного підходу до побудови інформаційної системи управління нерухомістю, що поєднує багаторівневу архітектуру, формалізовану модель роль-орієнтованого

					БР.ІП – 02.00.00.000 ПЗ	Арк.
						12
Змн.	Арк.	№ докум.	Підпис	Дата		

доступу та інтегровані аналітичні механізми оцінки ефективності використання активів.

Практичне застосування

Результати роботи можуть бути використані підприємствами, що здійснюють управління об'єктами нерухомості, для автоматизації бізнес-процесів, підвищення ефективності управління активами та покращення якості прийняття управлінських рішень.

Бакалаврська робота містить 76 сторінок, 20 рисунків, 3 розділи, переліки використаних джерел налічує 38 найменувань.

					БР.ІП – 02.00.00.000 ПЗ	Арк.
						13
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 1. АНАЛІЗ ПРОБЛЕМАТИКИ РОЗРОБКИ ІНФОРМАЦІЙНИХ СИСТЕМ ДЛЯ УПРАВЛІННЯ ОБ'ЄКТАМИ НЕРУХОМОСТІ

1.1. Обґрунтування впровадження інформаційних систем в управлінні об'єктами нерухомості та автоматизації бізнес-процесів

Сучасний ринок нерухомості характеризується експоненціальним зростанням обсягів інформації, що потребує системного підходу до збору, зберігання та аналізу даних. Гетерогенність масивів даних, які охоплюють відомості про орендарів, умови договірних зобов'язань, технічний стан об'єктів та фінансові транзакції, висуває високі вимоги до інструментів їх обробки. Традиційні підходи до управління, що базуються на ручній обробці та децентралізованому зберіганні інформації, демонструють низьку ефективність у сучасних економічних умовах. Основними детермінантами переходу до цифровізації галузі є інформаційна фрагментація, висока імовірність помилок внаслідок людського фактора та часові лаги у прийнятті управлінських рішень. З огляду на зростаючу конкуренцію та підвищення стандартів клієнтського сервісу, інтеграція цифрових рішень стає стратегічною необхідністю для суб'єктів ринку. Для вирішення окреслених проблем пропонується розробка та впровадження системи управління нерухомістю (Real Estate Management System — REMS). Така система базується на принципах побудови уніфікованої екосистеми, що інтегрує управління даними, автоматизацію робочих процесів та аналітичну підтримку у режимі реального часу.

1.1.1. Функціональна структура інтегрованої системи

Концептуальна модель запропонованої системи (рисунок 1.1) передбачає централізацію управлінських функцій у межах єдиної платформи, що охоплює такі ключові модулі:

					БР.ІП – 02.00.00.000 ПЗ	Арк.
						14
Змн.	Арк.	№ докум.	Підпис	Дата		

- Модуль управління орендою та взаємовідносинами з орендарями (CRM-компонент): забезпечує повний цикл супроводу орендних угод — від формування комерційних пропозицій до моніторингу термінів пролонгації договорів.

- Експлуатаційний модуль: призначений для оперативного управління технічним обслуговуванням, фіксації заявок від орендарів та контролю виконання регламентних робіт.

- Фінансово-аналітичний модуль: дозволяє здійснювати моніторинг грошових потоків, автоматизувати виставлення рахунків та контролювати рівень дебіторської заборгованості.

- Інтелектуальна бізнес-аналітика (BI): надає інструментарій для візуалізації ключових показників ефективності (KPI), таких як рівень заповнюваності об'єктів, динаміка прибутковості та операційні витрати.



Рисунок 1.1 – Концептуальна модель системи управління нерухомістю

1.1.2. Особливості та переваги пропонованої системи

Значущість реалізації даного проєкту полягає у формуванні комплексного, роль-орієнтованого середовища, яке трансформує підходи до

					БР.ІП – 02.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		15

операційного управління. Впровадження REMS дозволяє досягти синергетичного ефекту за рахунок наступних чинників:

- Мінімізація інформаційної асиметрії: Консолідація критичних даних у єдиному сховищі забезпечує прозорість процесів як для менеджменту, так і для кінцевих споживачів послуг (орендарів) через механізми самообслуговування.

- Алгоритмізація рутинних завдань: Автоматизація типових бізнес-процесів (сповіщення про платежі, ескалація заявок на ремонт, формування звітності) суттєво знижує навантаження на персонал та підвищує точність обліку.

- Обґрунтованість управлінських рішень: Доступ до актуальної аналітики у реальному часі дозволяє менеджменту оперативно реагувати на ринкові зміни та оптимізувати стратегію використання активів.

- Масштабованість та гнучкість: Архітектура сучасних хмарних рішень дозволяє системі адаптуватися до зростаючих потреб бізнесу без необхідності кардинальної перебудови ІТ-інфраструктури.

Таким чином, впровадження інтегрованої системи управління нерухомістю є необхідною умовою для модернізації бізнес-процесів, підвищення операційної стійкості та досягнення довгострокової конкурентоспроможності на динамічному ринку.

1.2. Проектування роль-орієнтованої архітектури та інтерфейсів доступу до системи управління нерухомістю

Реалізація концепції комплексної цифровізації управління нерухомістю потребує впровадження багатокомпонентної архітектури, що забезпечує диференційований доступ до даних залежно від функціональних ролей користувачів. У межах розробки системи REMS (Real Estate Management System) запропоновано трирівневу модель інтерфейсної взаємодії, яка

					БР.ІП – 02.00.00.000 ПЗ	Арк.
						16
Змн.	Арк.	№ докум.	Підпис	Дата		

інтегрує адміністративні, мобільні та публічні канали комунікації в єдиному інформаційному просторі.

1.2.1. Компонентна структура (рисунок 1.2) та функціональні інтерфейси

Централізована адміністративна підсистема (Admin-centric Interface) виступає як ядро системи для менеджменту та адміністративного персоналу. Даний інтерфейс побудований на принципах управління на основі моделей даних, що дозволяє структурувати великі масиви інформації про орендні угоди, фінансові транзакції та складні робочі процеси технічного обслуговування. Основна функція підсистеми полягає у забезпеченні цілісності життєвого циклу об’єкта — від первинного обліку до формування підсумкової звітності.



Рисунок 1.2 - Компонентна структура системи

Мобільний застосунок розширює функціональні можливості системи, забезпечуючи персоналу гнучкість у виконанні операцій поза офісним середовищем. Використання мобільного клієнта дозволяє реалізувати механізми верифікації стану об’єктів у режимі реального часу (безпосередньо

на місці), здійснювати оперативне оновлення записів про нерухомість та моніторинг платежів. Це суттєво мінімізує часові витрати на внесення даних та підвищує реактивність управління критично важливими бізнес-завданнями.

Для орендарів та потенційних клієнтів передбачено спеціалізований веб-інтерфейс (портал), що забезпечує контрольований доступ до публічної інформації. Функціонал порталу дозволяє переглядати доступний фонд нерухомості, аналізувати цінову політику та локаційні характеристики, а також ініціювати процедуру інтеграції (онбордингу) через цифрові канали зв'язку. Такий підхід гарантує прозорість операцій при збереженні суворого розмежування прав доступу до внутрішніх адміністративних процесів.

1.2.2. Система підтримки прийняття рішень та аналітичний інструментарій

Важливою складовою проєкту є інтеграція аналітичних дашбордів безпосередньо у робоче середовище користувачів. Це дозволяє здійснювати моніторинг ключових показників ефективності (KPI) без необхідності звернення до зовнішніх аналітичних платформ. Основні напрями аналітики включають:

- Динаміку заповнюваності об'єктів у розрізі часових періодів;
- Прогнозування та верифікацію потоків доходів;
- Оцінку продуктивності та швидкості виконання заявок на технічне обслуговування.

Метою проєкту є перехід від дискретних, ручних методів управління до уніфікованої цифрової платформи, яка усуває ізолюваність даних та оптимізує операційні витрати. Впровадження REMS спрямоване на досягнення наступних результатів:

- автоматизація рутинних завдань зменшує імовірність помилок та вивільняє ресурс персоналу для стратегічного планування.

					БР.ІП – 02.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		18

- покращення клієнтського досвіду: Цифровізація процесів онбордингу та сервісного обслуговування підвищує рівень задоволеності орендарів.

- архітектура системи дозволяє нарощувати обсяги керованих активів без пропорційного збільшення штату адміністративного персоналу.

Таким чином, розроблена модель не лише модернізує поточні операції, а й створює фундамент для довгострокової конкурентоспроможності на ринку нерухомості, базуючись на принципах прийняття рішень на основі верифікованих даних.

1.3. Архітектурні принципи синхронізації даних та взаємодії мобільних терміналів із центральним сховищем

Ключовим аспектом функціонування мобільного інтерфейсу в межах системи REMS є забезпечення консистентності даних при високій інтенсивності операцій. Архітектурна взаємодія між мобільним терміналом та центральною базою даних базується на дворівневій моделі обміну інформацією через проміжний шар прикладного програмного інтерфейсу.

Механізми обміну та синхронізації:

- Протокольна взаємодія: Обмін даними здійснюється за допомогою стандартних протоколів (REST або GraphQL), що забезпечує платформонезалежність та високу швидкість передачі пакетів. Мобільний клієнт надсилає запити у форматі JSON, які обробляються сервером додатків для виконання операцій читання або запису в основну базу даних.

- Режим автономної роботи (Offline Persistence): Для забезпечення безперебійності операцій у зонах з обмеженим мережевим покриттям реалізовано механізм локального кешування. Зміни, внесені персоналом на об'єкті, фіксуються у локальному сховищі пристрою та автоматично синхронізуються з центральним сервером після відновлення стабільного з'єднання.

					БР.ІП – 02.00.00.000 ПЗ	Арк.
						19
Змн.	Арк.	№ докум.	Підпис	Дата		

- Розв'язання конфліктів (Conflict Resolution): При одночасній зміні одного і того ж запису різними користувачами система застосовує алгоритми вирішення конфліктів (наприклад, метод «останнього запису» або версіонування об'єктів). Це гарантує, що дані про стан орендованих одиниць або статус технічних заявок залишаються актуальними та несуперечливими.

Щодо забезпечення безпеки та цілісності потоків наведемо наступні аргументи. Інтеграція мобільних пристроїв у загальну екосистему REMS вимагає суворого дотримання протоколів безпеки. Взаємодія захищена на рівні транспортного шару (TLS/SSL), а ідентифікація користувачів базується на технології наскрізної автентифікації. Це дозволяє системі не лише передавати дані, а й логувати кожну зміну в аудиторському журналі (AuditLog), забезпечуючи повну ретроспективну прозорість операцій, здійснених поза межами стаціонарних робочих місць.

Впровадження описаних механізмів синхронізації дозволяє досягти високої операційної стійкості системи, мінімізувати ризики втрати даних та забезпечити управлінський апарат достовірною інформацією про стан активів у режимі «тут і зараз».

Для забезпечення інформаційної безпеки та розмежування функціональних обов'язків у системі REMS застосовується модель керування доступом на основі ролей (Role-Based Access Control — RBAC). В умовах архітектури, що складається з трьох типів інтерфейсів (адміністративного, мобільного та публічного), алгоритми RBAC гарантують, що доступ до об'єктів бази даних здійснюється за принципом найменших привілеїв.

1.4. Формалізація та алгоритмічна реалізація моделі RBAC у багаторівневій системі

Теоретична модель розмежування прав базується на множині користувачів U , ролей R та дозволів P . Дозвіл визначається як пара (o, a) , де

					БР.ІП – 02.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		20

$o \in O$ — об'єкт системи (таблиця, запис, функція), а $a \in A$ — допустима дія (Read, Create, Update, Delete).

Процес авторизації описується функцією:

$$\text{Auth}(u, o, a) \rightarrow \{True, False\}$$

яка повертає позитивне значення, якщо користувач u має роль r , якій призначено дозвіл на дію a над об'єктом o .

1.4.1. Алгоритмічні особливості для різних типів застосунків

Розподіл прав між застосунками реалізується через комбінацію ролей та фільтрації на рівні даних.

1. Адміністративна підсистема (Full-Access RBAC)

Використовується ієрархічна модель ролей (Hierarchical RBAC). Наприклад, роль «Старший менеджер» успадковує всі права «Молодшого менеджера», додаючи функції затвердження фінансових звітів та видалення записів. Алгоритм перевірки прав працює на рівні метаданих кожної сутності, дозволяючи бачити повну структуру бази даних.

2. Мобільний інтерфейс (Functional Scope RBAC)

Доступ обмежується функціональним периметром. Алгоритм авторизації тут часто поєднується з атрибутивним доступом (ABAC). Наприклад, технік має роль «Maintenance Staff», але бачить лише ті записи в таблиці WorkOrder, де він вказаний як виконавець ($\text{User_ID} = \text{Assigned_Staff_ID}$). Це запобігає надмірному навантаженню мобільного пристрою зайвими даними та посилює безпеку.

3. Публічний портал (External Isolation RBAC):

Застосовується алгоритм ізоляції середовища. Зовнішні користувачі (орендарі) отримують роль «External User», яка має нульовий рівень доступу до системних таблиць (AuditLog, GeneralLedger). Дозвіл надається лише на рівні рядків (Row-Level Security — RLS): орендар може бачити тільки свій LeaseAgreement та пов'язані з ним Invoice.

					БР.ІП – 02.00.00.000 ПЗ	Арк.
						21
Змн.	Арк.	№ докум.	Підпис	Дата		

Для забезпечення наскрізної безпеки між застосунками використовується протокол маркерного доступу (Token-based Authentication). При логіні система генерує зашифрований токен, що містить «претензії» (claims) — ідентифікатор користувача та його ролі.

- 1) Запит: Застосунок надсилає токен до API.
- 2) Верифікація: Сервер розшифровує токен та перевіряє валідність підпису.
- 3) Фільтрація: База даних повертає лише ті набори даних, які відповідають ролі користувача, закладеній у токені.

Алгоритм RBAC інтегрований із модулем аудиту. Будь-яка спроба несанкціонованого доступу (коли Auth повертає False) фіксується в таблиці AuditLog із вказівкою джерела запиту (тип застосунку, IP-адреса, час). Це дозволяє виявляти потенційні вразливості та спроби компрометації системи на ранніх етапах.

Впровадження такої структурованої моделі доступу забезпечує високу операційну гнучкість: персонал отримує інструменти, необхідні для роботи, тоді як критичні фінансові та системні дані залишаються надійно захищеними від несанкціонованого втручання.

1.4.2. Матриця доступу та специфікація повноважень суб'єктів системи

Для практичної реалізації моделі розмежування прав доступу розроблено матрицю повноважень, яка визначає допустимі дії для кожної типової ролі користувача відносно критичних об'єктів (сутностей) бази даних. Матриця базується на стандарті операцій CRUD.

Даний підхід дозволяє не лише захистити конфіденційну інформацію, а й оптимізувати інтерфейс користувача, приховуючи нерелевантні для конкретної ролі функціональні модулі.

					БР.ІП – 02.00.00.000 ПЗ	Арк.
						22
Змн.	Арк.	№ докум.	Підпис	Дата		



Рисунок 1.3 – Деталізація обмежень за ролями в системі управління нерухомістю

Виконаємо опис деталізації обмежень за ролям рисунок 1.3:

- роль «Адміністратор системи» володіє необмеженими правами доступу до всіх рівнів архітектури, включаючи можливість модифікації структури бази даних та перегляду журналів аудиту для контролю дій інших користувачів.

- роль «Менеджер нерухомості» має повний операційний доступ до управління життєвим циклом оренди та нерухомості. Обмеження стосуються видалення фінансових записів та доступу до системних журналів, що запобігає прихованню можливих маніпуляцій з даними.

- роль «Технічний спеціаліст» - права обмежені експлуатаційним модулем. Алгоритм забезпечує видимість лише тих технічних замовлень, де спеціаліст призначений виконавцем. Доступ до фінансової та договірної інформації повністю заблокований.

- роль «Орендар» використовує модель Row-Level Security (RLS). Користувач може створювати лише сервісні запити та переглядати виключно власні рахунки й договори. Публічний доступ до каталогу нерухомості обмежений лише тими атрибутами, що позначені як доступні для зовнішнього перегляду (маркетинг).

Така диференціація повноважень забезпечує стійкість системи до внутрішніх загроз та гарантує, що кожен учасник бізнес-процесу взаємодіє лише з необхідним йому обсягом інформації, підтримуючи загальну цілісність екосистеми REMS.

1.5. Огляд та порівняльний аналіз існуючих інформаційних систем управління нерухомістю та активами

Для проведення об'єктивного порівняльного аналізу та визначення наукової новизни розроблюваної системи REMS необхідно дослідити існуючі програмні рішення, що домінують на глобальному ринку автоматизації управління нерухомістю. Сучасні системи (Property Management Systems — PMS) еволюціонували від простих облікових баз до складних ERP-платформ, що використовують хмарні обчислення та інтелектуальний аналіз даних.

Нижче наведено аналітичний огляд п'яти репрезентативних систем, що демонструють різні підходи до архітектури та функціонального наповнення.

1.5.1. ERP-платформа для управління нерухомістю Yardi Voyager

Yardi Voyager (рисунок 1.4) — це спеціалізована ERP-платформа для управління нерухомістю та інвестиціями. Дана система є промисловим стандартом для управління масштабними портфелями нерухомості.

Платформа базується на модульному принципі з централізованою базою даних, що забезпечує високу масштабованість. Вона інтегрує фінансовий облік інституційного рівня з операційним управлінням.

Функціональні особливості системи полягають в глибокій спеціалізації на інвестиційному менеджменті та прогнозуванні грошових потоків. Система забезпечує наскрізну прозорість для великих корпоративних структур, проте характеризується високою складністю впровадження та адміністрування.

					БР.ІП – 02.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		24

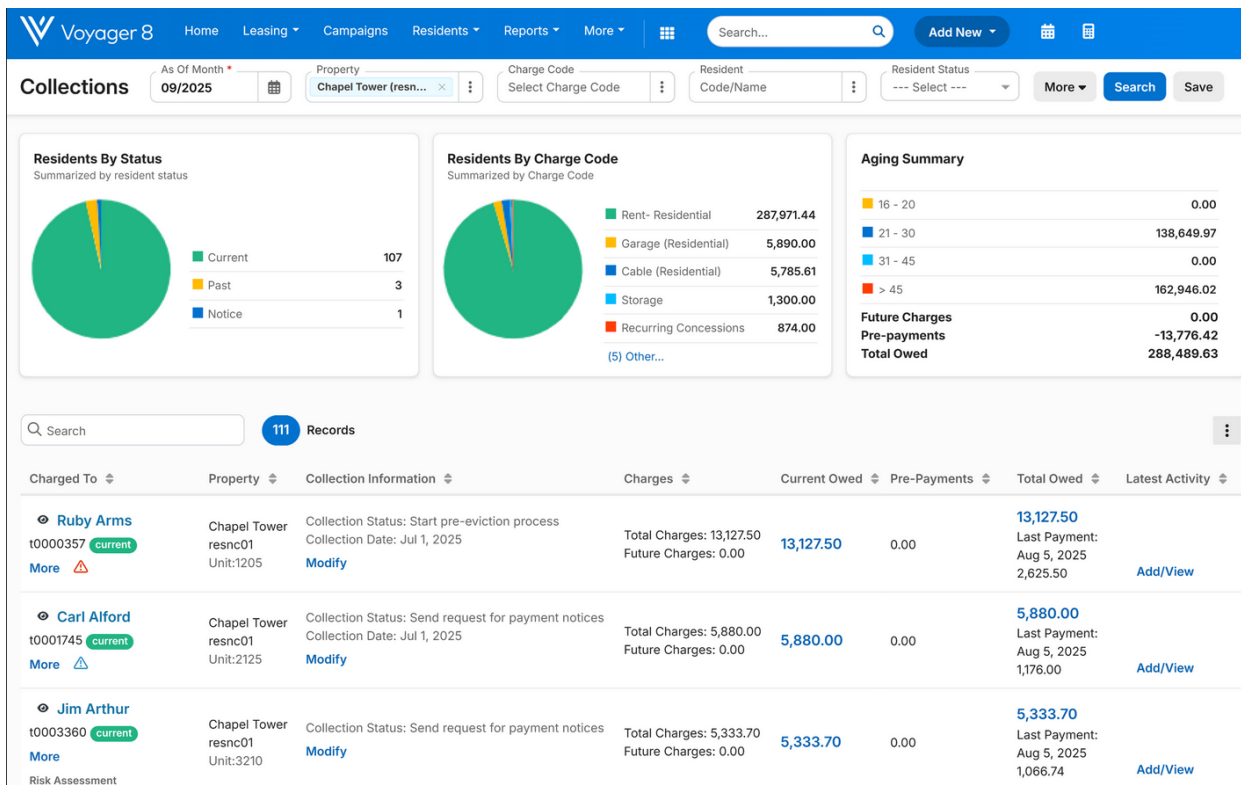


Рисунок 1.4 – Інтерфейс системи Yardi Voyager

Система охоплює повний життєвий цикл управління нерухомістю:

- управління орендою - автоматичне нарахування орендної плати, відстеження термінів оренди, індексація та управління складними договорами.
- бухгалтерський облік - спеціалізований фінансовий облік, що відповідає стандартам IFRS та GAAP, з можливістю вести окремі книги для кожної будівлі або об'єкта.
- управління експлуатацією - облік заявок на ремонт, планове технічне обслуговування об'єктів та робота з підрядниками.
- можливість для мешканців або бізнес-орендарів оплачувати рахунки онлайн, подавати заявки на ремонт та переглядати документи.
- інвестиційний менеджмент - інструменти для власників портфелів нерухомості (аналіз прибутковості, управління активами).

В останніх версіях Yardi Voyager 7S активно впроваджуються технології ШІ для прогнозування вакантності та зчитування рахунків.

1.5.2. Система MRI Software (рисунок 1.5)

Система вирізняється своєю прихильністю до парадигми «відкритої екосистеми». На відміну від закритих монолітних рішень, MRI використовує відкриті API для гнучкої інтеграції зі сторонніми сервісами (PropTech-рішеннями). Це дозволяє користувачам формувати кастомізований набір інструментів.

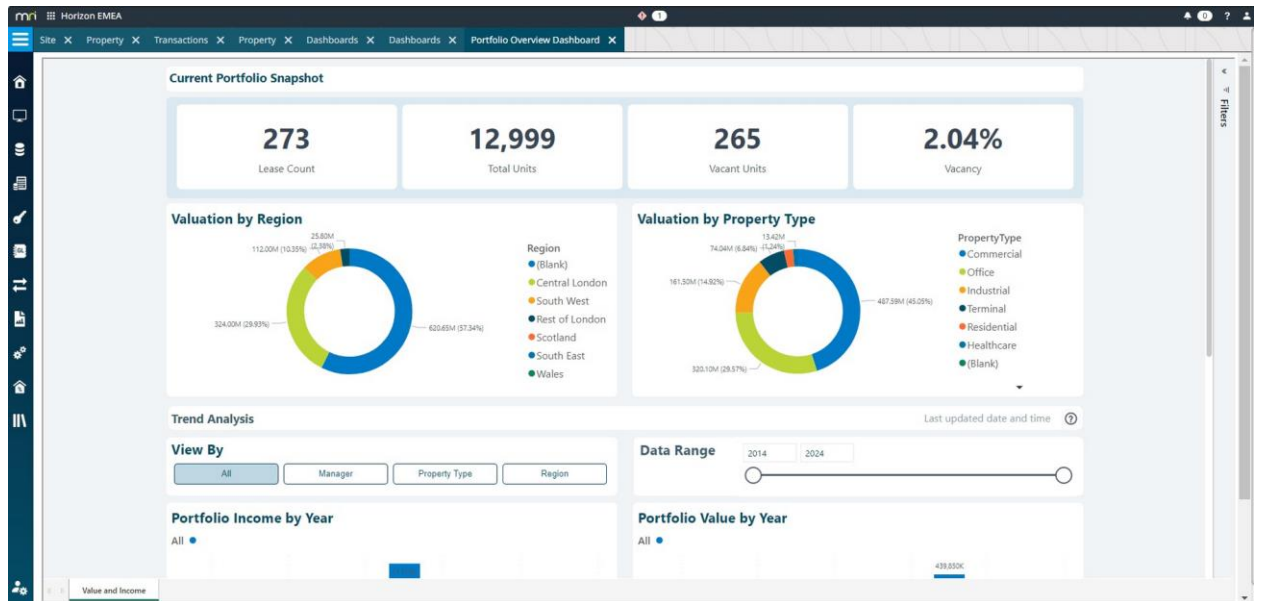


Рисунок 1.5 – Дашборд системи MRI Software

Функціональні особливості: Потужний інструментарій для управління комерційною нерухомістю, що включає складні алгоритми розрахунку орендних ставок та автоматизацію складних умов контрактів.

1.5.3. SaaS-рішення AppFolio Property Manager (рисунок 1.6)

Представник SaaS-рішень (Software as a Service), орієнтований на сучасні цифрові стандарти взаємодії.

Хмарна архітектура, оптимізована для мобільного використання. Система робить ставку на автоматизацію рутинних завдань за допомогою алгоритмів машинного навчання (наприклад, автоматичне розпізнавання рахунків або інтелектуальна відповідь на запити орендарів).

					БР.ІП – 02.00.00.000 ПЗ	Арк.
						26
Змн.	Арк.	№ докум.	Підпис	Дата		

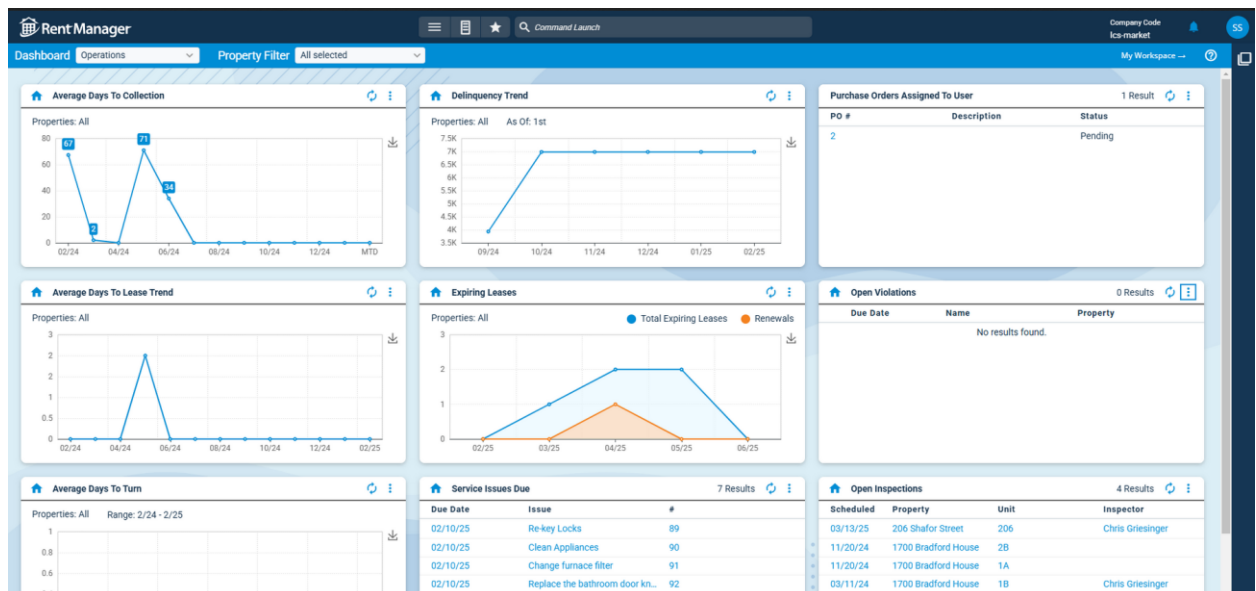


Рисунок 1.6 – Дашборд AppFolio Property Manager

Функціональною особливістю є високий рівень UX/UI дизайну та орієнтованість на цифрову комунікацію з мешканцями через вбудовані месенджери та портали самообслуговування.

1.5.4. Платформа Buildium (рисунок 1.7)

Buildium — це хмарна платформа для управління нерухомістю, яка в ієрархії софту стоїть між простими таблицями Excel та складними корпоративними системами на кшталт Yardi чи Dynamics 365. Система, зорієнтована на сегмент середнього та малого бізнесу, що фокусується на операційній ефективності. Спрощена модель даних, яка забезпечує швидке розгортання та низький поріг входу для персоналу.

Система фокусується на тому, щоб максимально спростити життя менеджеру нерухомості:

- автоматизація оренди: публікація оголошень про вільні квартири на популярних майданчиках (Zillow, Trulia тощо) в один клік.
- перевірка орендарів: вбудовані інструменти для перевірки кредитної історії та кримінального минулого потенційних мешканців (актуально переважно для ринку США/ЄС).

- онлайн-платежі: збір орендної плати через систему ePay, автоматичне нарахування пені за затримку.

- управління заявками на ремонт: мешканці подають заявки через мобільний додаток, а ви призначаєте майстрів і відстежуєте статус виконання.

- бухгалтерія для орендодавців - Ведення доходів/витрат, підготовка податкових звітів та розрахунки з власниками нерухомості.

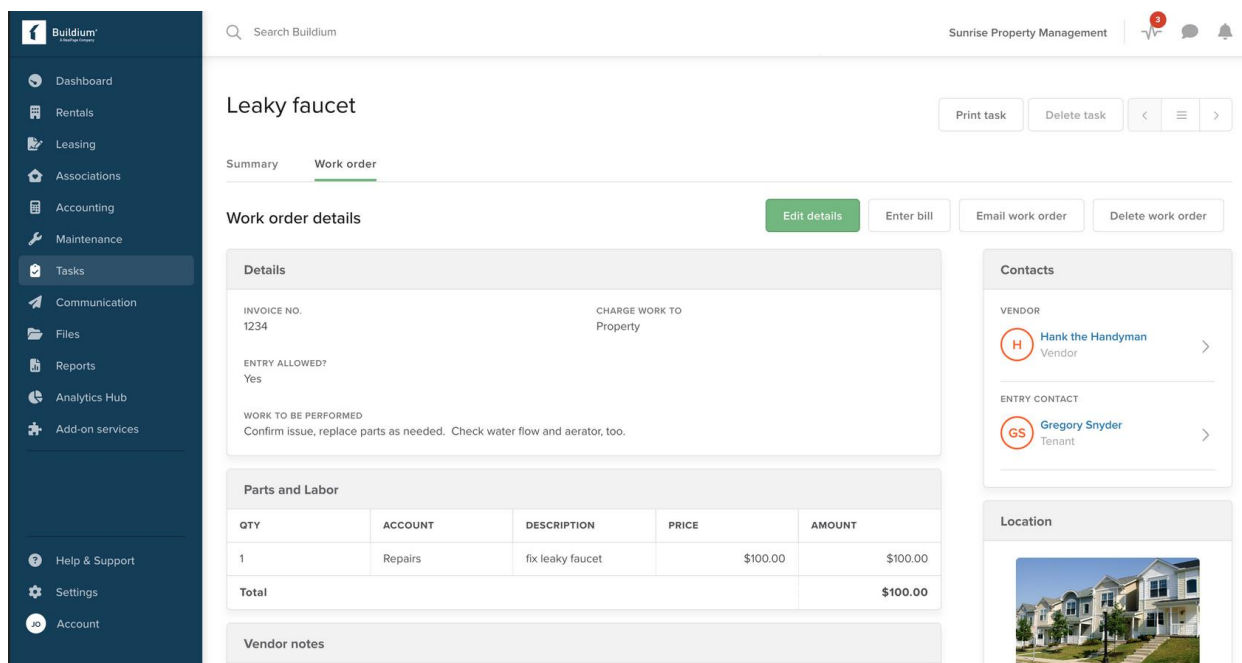


Рисунок 1.7– Інтерфейс Buildium

Функціональні особливості полягають в пріоритезації циклів технічного обслуговування та збору орендної плати. Система надає ефективні інструменти маркетингу вакантних площ та автоматизованого скринінгу потенційних орендарів.

1.5.5. Платформа Entrata

Платформа, що позиціонує себе як найбільш комплексна система «все-в-одному» з акцентом на інтеграцію фронт-офісу та бек-офісу. Архітектурна специфіка полягає в тому, що це єдина платформа даних, що виключає

					БР.ІП – 02.00.00.000 ПЗ	Арк.
						28
Змн.	Арк.	№ докум.	Підпис	Дата		

необхідність синхронізації між окремими модулями, оскільки всі функції (від лідогенерації до бухгалтерії) працюють у межах єдиного програмного коду.

Функціональні особливості полягають в глибокій аналітиці життєвого циклу орендаря та потужні інструменти автоматизації маркетингу, що дозволяють мінімізувати період простою об'єктів.

1.5.6. Синтез результатів аналізу

Для систематизації результатів огляду складено порівняльну таблицю за ключовими критеріями, що мають критичне значення для обґрунтування розробки REMS.

Таблиця 1.1 – Критерії порівняння систем

Критерій порівняння	Yardi Voyager	MRI Software	AppFolio	Buildium	Entrata
Цільовий сегмент	Великі корпорації	Комерційні мережі	Сучасні житлові комплекси	Малий та середній бізнес	Комплексне управління
Тип архітектури	Модульна (ERP)	Відкрита (API-first)	SaaS / Cloud-native	SaaS	Монолітна інтегрована
Рівень кастомізації	Високий	Екстремальний	Середній	Низький	Високий
Мобільна інтеграція	Додаткові модулі	Залежить від інтеграцій	Нативна (висока)	Нативна (середня)	Нативна (висока)
Аналітичний апарат	Складне прогнозування	Звітність за запитом	AI-автоматизація	Базова звітність	Real-time Dashboards

Аналіз існуючих рішень свідчить про те, що основними трендами розвитку галузі є консолідація даних, мобільна доступність та автоматизація взаємодії з клієнтами. Проте більшість систем або мають занадто високу

вартість і складність (Yardi, MRI), або пропонують обмежені можливості кастомізації під специфічні бізнес-процеси локальних ринків.

Це підтверджує доцільність розробки системи REMS як збалансованого рішення, що поєднує гнучкість налаштування роле-орієнтованих інтерфейсів з потужними інструментами аналітики в реальному часі, забезпечуючи при цьому високий рівень інтероперабельності без надмірних витрат на складну ІТ-інфраструктуру.

1.6 Висновки по розділу

У першому розділі здійснено комплексний аналіз проблематики розробки інформаційних систем управління об'єктами нерухомості та обґрунтовано необхідність їх впровадження для підвищення ефективності бізнес-процесів. Визначено функціональну структуру інтегрованої системи та встановлено ключові вимоги до її архітектури, зокрема орієнтацію на масштабованість і модульність. Проаналізовано особливості проектування роль-орієнтованої архітектури та доведено доцільність використання моделі RBAC для забезпечення контрольованого доступу до ресурсів системи. Досліджено принципи синхронізації даних і взаємодії мобільних клієнтів із центральним сховищем, що є критичним для сучасних розподілених систем. Проведений порівняльний аналіз існуючих рішень дозволив виявити їх сильні сторони та обмеження, що стало підґрунтям для формування вимог до розроблюваної системи.

					БР.ІП – 02.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		30

РОЗДІЛ 2. МЕТОДОЛОГІЯ РОЗРОБКИ ТА АЛГОРИТМІЧНЕ ЗАБЕЗПЕЧЕННЯ СИСТЕМИ УПРАВЛІННЯ НЕРУХОМІСТЮ

2.1. Методологія та технологічний цикл розробки системи

2.1.1. Обґрунтування вибору методології життєвого циклу розробки

Проектування та розробка системи управління нерухомістю базується на гнучкій методології Agile. Вибір даного підходу зумовлений високою динамікою вимог у сфері автоматизації об'єктів нерухомості та необхідністю ітеративного вдосконалення функціональних модулів.

Використання ітеративно-інкрементальної моделі дозволило забезпечити безперервний цикл зворотного зв'язку між розробником та стейкхолдерами на кожному етапі розробки. Кожна ітерація (спринт) завершувалася створенням працездатного інкременту продукту, що включав етапи системного аналізу, проектування архітектури, програмної реалізації, верифікації та валідації.

2.1.2. Архітектурна концепція та багаторівневий дизайн

Система розроблена на засадах модульного та роль-орієнтованого підходів. Архітектурне рішення передбачає поділ системи на логічні рівні, (рисунки 2.1) що забезпечує масштабованість та легкість підтримки:

- рівень даних (Data Tier): централізоване реляційне сховище даних, що забезпечує цілісність, нормалізацію та захист інформації про активи, контрагентів та фінансові потоки.

- рівень бізнес-логіки (Logic Tier): підсистема автоматизації робочих процесів, відповідальна за оркестрацію подій, таких як пролонгація договорів, генерація сповіщень та ескалація сервісних запитів.

- рівень представлення (Presentation Tier): сукупність спеціалізованих інтерфейсів доступу:

					БР.ІП – 02.00.00.000 ПЗ	Арк.
						31
Змн.	Арк.	№ докум.	Підпис	Дата		

- 1) адміністративний клієнт — для управління складними структурами даних та формування аналітичної звітності.
- 2) мобільний термінал — для забезпечення операційної діяльності персоналу в польових умовах.
- 3) публічний портал — для взаємодії із зовнішніми користувачами (орендарями) та забезпечення прозорості сервісів.



Рисунок 2.1 – Логічні рівні архітектури інформаційної системи

2.1.3. Технологічний процес розробки та кастомізації

Процес реалізації включав інтеграцію низькорівневих програмних рішень із високорівневими платформами розробки. Основна увага була приділена створенню зв'язної екосистеми компонентів:

- розробка інтерфейсної частини: проєктування адаптивних форм та дашбордів для візуалізації КРІ у реальному часі.
- програмне розширення функціональності:
 - розробка серверних компонентів (плагінів) мовою високого рівня для реалізації складної обчислювальної логіки, що виходить за межі стандартних налаштувань платформи.

- створення клієнтських скриптів для валідації даних на стороні інтерфейсу та покращення користувацького досвіду (UX).

- конфігурація інтеграційних шлюзів: Налаштування механізмів обміну даними між внутрішніми модулями та зовнішніми точками входу інформації.

2.1.4. Стратегія верифікації та тестування системи

Для гарантування надійності та відповідності технічному завданню тестування проводилося на кількох ієрархічних рівнях:

Модульне (Unit) тестування: перевірка коректності роботи окремих сутностей, полів та тригерів бізнес-логіки.

Інтеграційне тестування: контроль коректності передачі параметрів між фронт-енд інтерфейсами, сервером автоматизації та базою даних.

Роль-орієнтоване тестування (RBAC Validation): перевірка ізоляції даних згідно з матрицею прав доступу для різних категорій користувачів.

Наскрізне (End-to-End) тестування: емуляція повного життєвого циклу бізнес-процесу — від подачі заявки орендарем до її закриття в бухгалтерському обліку.

2.1.5. Впровадження та середовище експлуатації

Процес розгортання системи передбачав використання методики ізольованих середовищ для мінімізації ризиків:

- використання окремих інстансів для розробки (Dev), тестування (UAT) та промислової експлуатації (Prod).

- налаштування рівнів доступу, ліцензійних обмежень та шифрування каналів зв'язку.

- перевірка синхронізації даних у робочому середовищі та стрес-тестування критичних вузлів.

- розробка технічної документації та проведення інструктажів для персоналу.

					БР.ІП – 02.00.00.000 ПЗ	Арк.
						33
Змн.	Арк.	№ докум.	Підпис	Дата		

2.2. Архітектурна організація та концептуальна модель взаємодії компонентів системи

Даний підрозділ присвячено аналізу структурної організації та декомпозиції інтегрованої інформаційної системи управління нерухомістю. Проектна архітектура системи базується на принципах модульності, масштабованості та інтероперабельності, що дозволяє сформувати уніфіковане середовище для моніторингу активів та автоматизації бізнес-процесів. Центральним елементом системи є гетерогенна модель даних, що забезпечує цілісність інформації та розмежування рівнів доступу згідно з функціональними ролями користувачів.

2.2.1. Багаторівнева структура системної взаємодії

Концептуальна архітектура системи REMS реалізована як багаторівнева екосистема, що поєднує адміністративні модулі, клієнтські термінали та сервіси автоматизації через єдине реляційне сховище.

Основними структурними рівнями (шарами) системи є:

1. Рівень користувацьких інтерфейсів (Presentation Layer)

Внутрішній контур управління призначений для адміністративного персоналу та менеджменту. Включає веб-інтерфейси для глибокого аналізу даних та мобільні термінали для оперативного внесення інформації безпосередньо на об'єктах.

Зовнішній контур взаємодії - публічний веб-інтерфейс, що забезпечує доступ орендарів та потенційних клієнтів до маркетингових даних, каталогів об'єктів та інструментів самообслуговування.

2. Функціональний рівень додатків (Application Layer)

Модуль системного адміністрування забезпечує виконання складних операцій: ведення реєстрів, фінансовий моніторинг та стратегічне планування технічного обслуговування.

					БР.ІП – 02.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		34

Операційний мобільний модуль оптимізований для виконання транзакцій у реальному часі, таких як верифікація платежів або фіксація дефектів на об'єктах нерухомості.

Сервісний портал виконує роль фронт-офісу для зовнішніх запитів та процедур первинної інтеграції користувачів (онбордингу).

3. Рівень логічної автоматизації (Logic & Workflow Layer)

Система оркестрації бізнес-процесів керує подіями за визначеними алгоритмами. Даний шар мінімізує вплив людського фактора, автоматично ініціюючи процеси пролонгації договорів, розсилку повідомлень та ескалацію критичних заявок на ремонт.

4. Рівень керування даними (Data Tier):

Централізоване сховище виступає єдиним джерелом істини (Single Source of Truth) для системи. Зберігає нормалізовані масиви даних про об'єкти, суб'єктів та фінансові потоки. Безпека на цьому рівні реалізується через механізми Row-Level Security (RLS).

2.2.2. Механізми розширення та кастомізації

Для забезпечення відповідності специфічним вимогам галузі, архітектура передбачає можливість низькорівневої кастомізації:

- серверні програмні розширення: реалізація складної математичної логіки та алгоритмів перевірки бізнес-правил на стороні сервера за допомогою мов програмування високого рівня.

- клієнтські скриптові сценарії: оптимізація інтерфейсів користувача, динамічна валідація введених даних та автоматизація шаблонів документів на стороні клієнтського застосунку.

Запропонована архітектурна модель дозволяє інтегрувати розрізнені операційні процеси в єдину цифрову платформу. Завдяки чіткій декомпозиції на рівні представлення, логіки та даних, система REMS демонструє високу відмовостійкість та адаптивність до зростаючих обсягів інформації.

					БР.ІП – 02.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		35

Взаємодія між компонентами системи, візуалізована на рисунку 2.2, підтверджує можливість наскрізного контролю за станом нерухомості та фінансовими показниками організаційної структури.

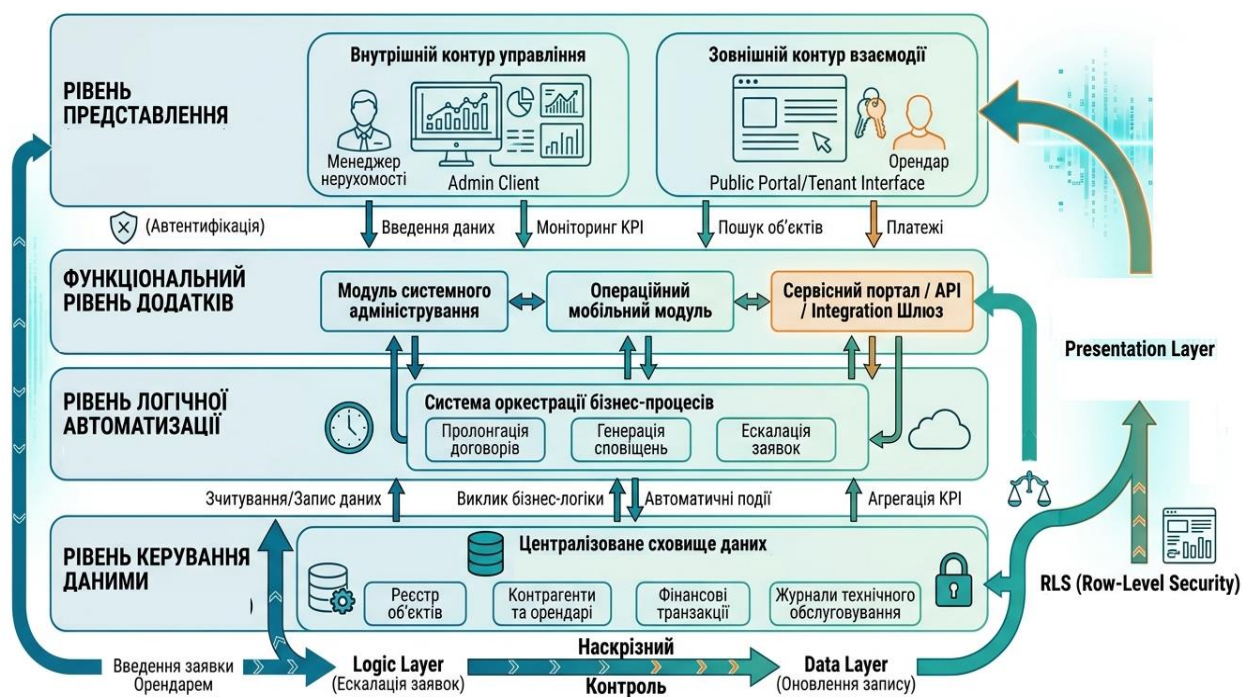


Рисунок 2.2 – Концептуальна архітектура системи управління нерухомістю

2.3. Інфологічне моделювання та структуризація бази даних системи

Формалізація структури даних є фундаментальним етапом розробки інтегрованої системи управління нерухомістю. Для відображення архітектури даних, ієрархії сутностей та їхніх взаємозв’язків використано метод побудови діаграм зв’язків сутностей (Entity-Relationship Diagram — ERD). Логічна схема бази даних розроблена з метою мінімізації надмірності інформації та забезпечення суворої референційної цілісності у межах централізованого реляційного середовища зберігання.

2.3.1. Класифікація та функціональне призначення ключових сутностей

Архітектура даних системи REMS базується на взаємодії декількох функціональних доменів, кожен з яких представлений специфічними сутностями:

- Суб'єкт (Орендар/Контрагент) - агрегує персональні ідентифікаційні дані, контактну метаінформацію та юридичні реквізити орендарів. Ця сутність виступає базовим вузлом для всіх подальших транзакцій та договірних зобов'язань.

- Об'єкт (Нерухомість/Актив) - містить детальні характеристики фізичних об'єктів, включаючи геопросторові координати, вартісні параметри та динамічні атрибути статусу зайнятості.

- Договір (Угода з клієнтом) - центральна транзакційна сутність, що фіксує часові межі оренди, фінансові умови та регулює правовідносини між суб'єктом та об'єктом.

- Фінансовий блок (Платіж / Графік виплат) забезпечує детальне відстеження вхідних грошових потоків. Сутність підтримує складні сценарії оплати, включаючи регулярні платежі та амортизаційні схеми (ЕМІ), пов'язуючи їх із конкретними угодами.

- Експлуатаційний блок (Технічне обслуговування / Робоче замовлення) оперує даними про запити на сервіс, стадії їх виконання та ресурсне забезпечення. Сутність «Робоче замовлення» є похідною від запиту та забезпечує зв'язок із виконавцями.

- Репутаційний блок акумулює оціночні дані щодо якості обслуговування, дозволяючи проводити моніторинг задоволеності клієнтів.

2.3.2. Аналіз топології зв'язків та опис моделі бази даних

Концептуальна модель даних реалізує складну мережу зв'язків типу 1:N та 1:1, що дозволяє підтримувати «єдине джерело» для всіх модулів:

					БР.ІП – 02.00.00.000 ПЗ	Арк.
						37
Змн.	Арк.	№ докум.	Підпис	Дата		

- Зв'язок «Нерухомість — Угода — Орендар»: Реалізується через потужну кореляцію 1:N, де один об'єкт нерухомості або один орендар можуть бути асоційовані з декількома угодами протягом тривалого періоду, проте кожна конкретна угода є унікальною для пари суб'єкт-об'єкт у визначений проміжок часу.

- Декомпозиція сервісних процесів: Запити на технічне обслуговування генерують відповідні робочі замовлення (1:N), що забезпечує можливість деталізації одного інциденту на кілька окремих технологічних операцій.

- Автоматизація фінансових потоків: Угоди автоматично ініціюють записи в таблицях платежів. Це гарантує, що будь-яка зміна в умовах оренди (наприклад, пролонгація) автоматично корегує очікувані фінансові надходження.

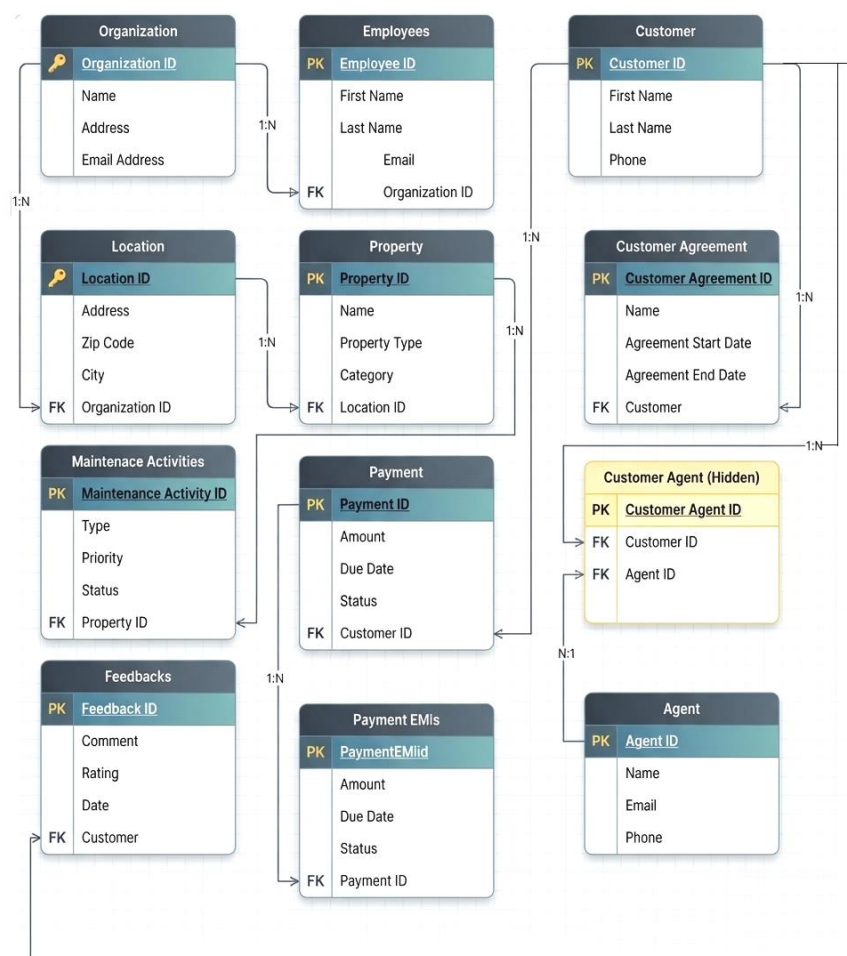


Рисунок 2.3 – Модель бази даних

Представлена модель реляційної бази даних описує архітектуру системи управління нерухомістю. Структура моделі базується на принципах нормалізації та ієрархічної підпорядкованості, що забезпечує цілісність даних та мінімізацію надмірності інформації.

Логічну структуру моделі можна декомпонувати на чотири функціональні домени:

1. Домен організаційної структури та просторового планування

Верхній рівень ієрархії представлений сутністю Organization, яка виступає центральним вузлом для ідентифікації юридичної особи. З нею пов'язані:

- Employees: Сутність для обліку персоналу, що має відношення типу 1:N до організації.

- Location: Географічна прив'язка об'єктів. Відношення 1:N між організацією та локацією дозволяє масштабувати систему на декілька регіонів чи міст.

- Property: Ключова сутність активів, яка деталізує об'єкти нерухомості за типами та категоріями, будучи підпорядкованою конкретній локації.

2. Домен взаємодії з контрагентами (Клієнтський блок)

Центральною фігурою цього блоку є сутність Customer. Модель передбачає складну мережу зв'язків для повної цифровізації клієнтського досвіду:

- Customer Agreement: Фіксує умови оренди або купівлі, включаючи терміни дії договору.

- Feedbacks: Дозволяє акумулювати оціночні дані та відгуки клієнтів, що є важливим для аналізу якості сервісу.

- Customer Agent (Hidden) та Agent: Реалізація зв'язку «багато до багатьох» (M:N) між клієнтами та агентами через проміжну таблицю. Це дозволяє закріплювати за одним клієнтом кількох агентів і навпаки, забезпечуючи гнучкість посередницьких послуг.

					БР.ІП – 02.00.00.000 ПЗ	Арк.
						39
Змн.	Арк.	№ докум.	Підпис	Дата		

3. Домен фінансового обліку та транзакцій

Фінансова модель реалізована через дворівневу структуру платежів:

- Payment: Реєструє загальні дані про транзакції (сума, дата, статус), пов'язані з конкретним клієнтом.

- Payment EMIs: Сутність, що дозволяє деталізувати платежі на окремі транші (Installments), забезпечуючи підтримку схем оплати частинами. Це вказує на здатність системи підтримувати складні фінансові графіки.

4. Домен експлуатаційного управління

Операційні процеси технічного обслуговування відображені через сутність Maintenance Activities. Зв'язок 1:N із сутністю Property дозволяє вести історію сервісних робіт, відстежувати їхній пріоритет та статус виконання для кожного конкретного об'єкта нерухомості.

Модель використовує чітко визначені первинні ключі (PK) для унікальної ідентифікації записів та зовнішні ключі (FK) для підтримки посилювальної цілісності. Більшість зв'язків мають тип «один до багатьох» (1:N), що відображає природну логіку бізнес-процесів: одна організація має багато локацій, одна локація — багато об'єктів нерухомості, а одна нерухомість може генерувати множину сервісних запитів та угод.

Така архітектура є масштабованою та придатною для розробки корпоративних систем автоматизації в секторі управління нерухомістю, забезпечуючи високий рівень контролю над активами та фінансовими потоками.

2.3.3. Системне значення логічної моделі

Представлена ER-діаграма моделює не лише статичні записи, а й динамічні бізнес-процеси (онбординг, ескалація заявок, фінансовий кліринг). Централізація операційних даних у такій структурі дозволяє:

- Уникнути дублювання інформації (дата-аномалій);
- Забезпечити каскадне оновлення пов'язаних записів;

					БР.ІП – 02.00.00.000 ПЗ	Арк.
						40
Змн.	Арк.	№ докум.	Підпис	Дата		

- Сформувати фундамент для роле-орієнтованого доступу, де видимість конкретних атрибутів сутностей регулюється правами користувача.

Отже, розроблена інфологічна модель бази даних забезпечує високу стійкість системи REMS до інформаційної фрагментації. Завдяки чіткій дефініції сутностей та їхніх взаємозв'язків, система отримує здатність до горизонтального масштабування, що дозволяє інтегрувати нові типи активів та складніші фінансові інструменти без кардинальної перебудови архітектури даних.

2.4. Моделювання інформаційно-динамічних процесів засобами діаграм потоків даних

Для формалізації взаємодії між зовнішніми сутностями та внутрішніми процесами системи REMS доцільно використовувати методологію діаграм потоків даних (Data Flow Diagram — DFD). Це дозволяє візуалізувати трансформацію вхідної інформації (заявки, платежі) у вихідні результати (звіти, розпорядження) без прив'язки до фізичної реалізації бази даних.

2.4.1. Опис рівнів абстракції інформаційних потоків

Інформаційна архітектура системи на контекстному рівні (Level 0) представляє систему як «чорну скриньку», що взаємодіє з чотирма основними зовнішніми сутностями: орендарями, менеджерами нерухомості, технічним персоналом (та підрядниками) і фінансовим департаментом.

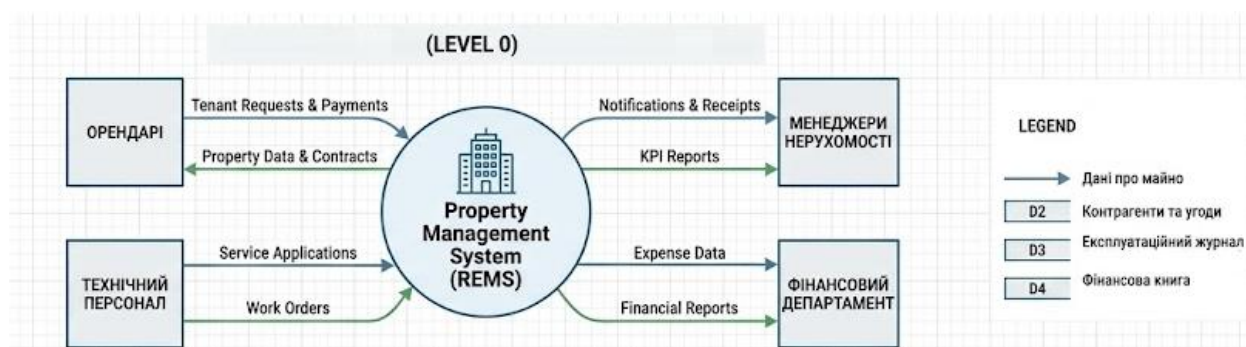


Рисунок 2.4 – Діаграма потоків даних (Level 0)

На рівні деталізації (Level 1) система декомпонується на чотири функціональні вузли:

- Процес 1.0: Управління об'єктами та одиницями. Приймає дані про конфігурацію майна та оновлює стан активів у сховищі даних.

- Процес 2.0: Адміністрування оренди. Опрацьовує договори, поєднуючи дані орендарів із вільними одиницями нерухомості, та ініціює цикли нарахувань.

- Процес 3.0: Диспетчеризація сервісного обслуговування. Трансформує запити орендарів у робочі наряди, координуючи роботу персоналу та підрядників.

- Процес 4.0: Моніторинг фінансових операцій та аналітика. Агрегує дані про оплати та витрати для формування звітності та візуалізації KPI.

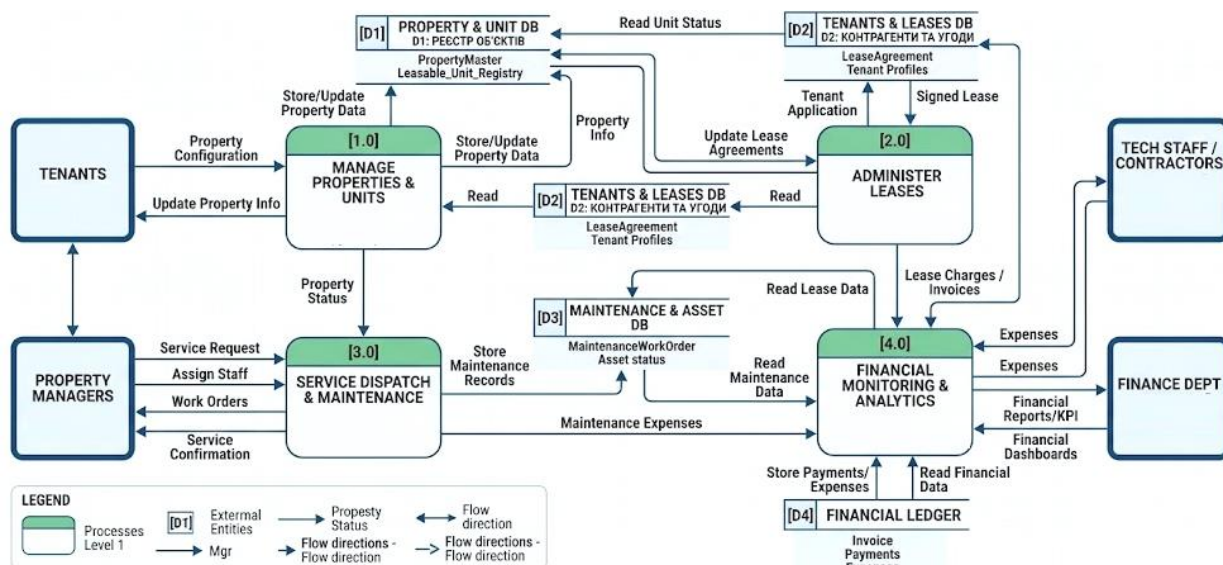


Рисунок 2.5 – Діаграма потоків даних (Level 1)

У моделі виділяються наступні логічні накопичувачі інформації:

- D1: Реєстр об'єктів (Property & Unit DB) — містить PropertyMaster та Leasable_Unit_Registry.

- D2: Контрагенти та угоди (Tenants & Leases DB) — зберігає дані LeaseAgreement та профілі Tenant.

- D3: Експлуатаційний журнал (Maintenance & Asset DB) — фіксує MaintenanceWorkOrder та стан Asset.

- D4: Фінансова книга (Financial Ledger) — консолідує Invoice, Payments та Expenses.

Представлена діаграма потоків даних першого рівня (Level 1 DFD) є результатом функціональної декомпозиції контекстної моделі системи REMS. Вона деталізує внутрішню логіку обробки інформації, визначаючи межі між окремими підсистемами, накопичувачами даних та зовнішніми суб'єктами.

Наведено детальний опис основних компонентів та інформаційних магістралей системи.

1. Функціональні вузли (Процеси)

1.0: Управління об'єктами та одиницями (Manage Properties & Units)

Цей процес виступає точкою входу для формування цифрового двійника фонду нерухомості.

Вхідні дані: Конфігураційні параметри об'єктів від Менеджерів нерухомості.

Логіка: Процес здійснює валідацію фізичних характеристик та класифікацію одиниць (житлові, комерційні, технічні).

Вихідні потоки: Оновлення реєстру об'єктів у сховищі [D1] та надання актуальних статусів іншим процесам.

2.0: Адміністрування оренди (Administer Leases)

Центральний бізнес-процес, що моделює життєвий цикл договірних відносин.

Механізм: На основі заявок на оренду (Tenant Application) та даних про вільні площі з [D1], процес генерує проекти договорів.

Взаємодія: Після підписання договору (Signed Lease) дані транзитуються до накопичувача [D2]. Процес також ініціює фінансові зобов'язання, передаючи параметри нарахувань у вузол 4.0.

					БР.ІП – 02.00.00.000 ПЗ	Арк.
						43
Змн.	Арк.	№ докум.	Підпис	Дата		

3.0: Диспетчеризація сервісного обслуговування (Service Dispatch & Maintenance)

Операційний модуль, що забезпечує підтримку технічної придатності активів.

Потоки: Отримує запити від Орендарів (Service Requests) та розпорядження від Менеджерів.

Трансформація: Процес перетворює неструктуровані звернення у формалізовані робочі наряди (Work Orders) для Технічного персоналу. Результати виконання та витрати ресурсів фіксуються у сховищі [D3].

4.0: Моніторинг фінансових операцій та аналітика (Financial Monitoring & Analytics)

Агрегуючий вузол, що виконує роль системи підтримки прийняття рішень.

Консолідація: Процес інтегрує дані про нарахування з модуля 2.0 та операційні витрати з модуля 3.0.

Аналітична функція: На основі інформації з головної книги [D4] здійснюється розрахунок KPI.

Вихід: Формування стратегічної звітності для Фінансового департаменту та візуалізація дашбордів для менеджменту.

Діаграма чітко розмежовує ролі користувачів:

- Орендарі (Tenants): Виступають ініціаторами заявок та отримувачами фінансових документів.
- Менеджери (Property Managers): Здійснюють стратегічне та операційне управління через параметри конфігурації та контроль персоналу.
- Технічний персонал (Tech Staff): Отримує завдання та звітує про витрачені ресурси, що є основою для обліку витрат (Expenses).
- Фінансовий департамент (Finance Dept): Є кінцевим споживачем консолідованої інформації для фінансового контролю та планування.

					БР.ІП – 02.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		44

Представлена DFD Level 1 демонструє високий ступінь зв'язності компонентів. Кожен потік даних має визначене джерело та призначення, що виключає появу «інформаційних тупиків».

2.4.2. Переваги потокової моделі

Використання DFD у межах даного проєкту дозволяє:

- оптимізувати шлях проходження заявки на ремонт від моменту створення до закриття фінансового акту.
- гарантувати, що фінансові нарахування базуються виключно на актуальних даних з реєстру угод.
- визначити чіткі точки входу для інтеграції з зовнішніми банківськими сервісами або системами моніторингу будівель (BMS).

Такий підхід забезпечує прозорість архітектури системи та слугує основою для подальшої розробки алгоритмічного забезпечення модулів REMS.

2.5. Аналіз інформаційних потоків та механізмів міжрівневої взаємодії

Для забезпечення функціональної цілісності системи REMS необхідно чітко детермінувати шлях проходження інформації від моменту її генерації зовнішнім суб'єктом до етапу довгострокового зберігання та аналітичної обробки. Аналіз потоків даних дозволяє верифікувати відсутність інформаційних розривів та оптимізувати навантаження на системні ресурси.

Процес обробки інформації в системі можна розділити на чотири послідовні фази:

1. Фаза ініціації та захоплення (Interaction Phase)

Дані генеруються на Рівні представлення. Наприклад, технік через мобільний термінал вносить звіт про виконання робіт. На цьому етапі

					БР.ІП – 02.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		45

відбувається первинна валідація форматів (Client-side validation) за допомогою клієнтських скриптів.

2. Фаза транзиту та санкціонування (Transport & Auth Phase)

Запит передається до рівня додатків через захищений шлюз (API). Система ідентифікує користувача та перевіряє його повноваження згідно з матрицею прав доступу (RBAC). Якщо дію дозволено, дані маркуються контекстом користувача для подальшого аудиту.

3. Фаза обробки та оркестрації (Processing Phase)

На рівні бізнес-логіки активуються програмні компоненти та плагіни. Тут виконуються складні обчислення (наприклад, автоматичний перерахунок залишку дебіторської заборгованості або зміна статусу об'єкта нерухомості). Якщо подія передбачає автоматизацію (наприклад, надсилання сповіщення орендарю), система ініціює відповідний асинхронний процес.

4. Фаза персистентності та синхронізації (Persistence Phase)

Оброблена інформація фіксується у рівні даних (реляційному сховищі). Завдяки принципу нормалізації даних забезпечується автоматичне оновлення всіх пов'язаних сутностей. Після успішного запису система повертає підтвердження на рівень представлення, оновлюючи інтерфейс користувача.

Окрім прямого потоку (введення даних), система реалізує цикли зворотної передачі для підтримки прийняття рішень:

- Агрегація: дані з рівня сховища піднімаються на рівень додатків, де групуються за заданими критеріями (період, тип об'єкта, регіон).
- Візуалізація: Опрацьовані масиви трансформуються в графічні індикатори (KPI) та дашборди на рівні адміністративного клієнта.

Розроблена архітектурна модель та алгоритми міжрівневої взаємодії забезпечують високу прозорість інформаційних процесів. Поділ на логічні шари дозволяє незалежно масштабувати кожен компонент системи, а чітка дефініція потоків даних гарантує цілісність бізнес-логіки. Такий підхід

					БР.ІП – 02.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		46

мінімізує затримки при обробці транзакцій та створює надійний фундамент для оперативного управління великими портфелями нерухомості.

2.6 Висновки по розділу

У другому розділі обґрунтовано методологію розробки інформаційної системи та визначено ключові етапи її життєвого циклу з урахуванням сучасних підходів до інженерії програмного забезпечення. Запропоновано багаторівневу архітектурну модель, яка забезпечує чіткий розподіл функцій між рівнями та підвищує гнучкість системи. Виконано інфологічне моделювання предметної області та розроблено структуру бази даних, що гарантує цілісність і узгодженість інформації. Застосування діаграм потоків даних дозволило формалізувати інформаційно-динамічні процеси та оптимізувати взаємодію між компонентами системи. Проведений аналіз інформаційних потоків підтвердив ефективність обраних архітектурних рішень та їх придатність для подальшої реалізації.

					БР.ІП – 02.00.00.000 ПЗ	Арк.
						47
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 3. РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ УПРАВЛІННЯ НЕРУХОМІСТЮ

3.1. Концептуальні засади та функціональна декомпозиція інформаційної системи управління нерухомістю

Сучасний сектор нерухомості характеризується високою інтенсивністю генерації та обробки різномірних масивів даних, що охоплюють реєстри контрагентів, договірну документацію, регламенти технічної експлуатації об'єктів та складні фінансові потоки. Традиційні підходи до управління, засновані на ручній обробці інформації, на сучасному етапі демонструють критичну неефективність, що проявляється у фрагментації даних, інформаційній асиметрії та низькій швидкості прийняття управлінських рішень.

З огляду на це, актуальним завданням є розробка та впровадження системи управління нерухомістю, яка базується на принципах цифрової трансформації та централізації інформаційних ресурсів.

Метою розробки REMS є створення уніфікованої цифрової екосистеми для консолідації даних про активи та суб'єктів орендних відносин, оптимізація фінансового моніторингу та впровадження інструментів самообслуговування для покращення якості взаємодії між стейкхолдерами.

Для досягнення поставленої мети архітектуру системи було декомпозовано на п'ять ключових функціональних модулів:

1. Підсистема адміністрування договірних відносин: забезпечує ведення структурованих реєстрів орендарів, зберігання цифрових копій угод та автоматизацію алгоритмів нагадувань про терміни пролонгації або завершення контрактів.

2. Модуль управління активами (Property Management): надає інструментарій для моніторингу технічних характеристик об'єктів,

					БР.ІП – 02.00.00.000 ПЗ	Арк.
						48
Змн.	Арк.	№ докум.	Підпис	Дата		

візуалізації статусу їх зайнятості та оперативного доступу до документації через кросплатформні інтерфейси.

3. Підсистема оркестрації технічного обслуговування: реалізує механізм подачі сервісних запитів через зовнішній портал, автоматичну генерацію робочих нарядів та ієрархічну ескалацію невирішених інцидентів у межах заданих SLA.

4. Модуль фінансового інжинірингу та обліку: відповідає за транзакційний моніторинг орендних платежів, автоматичне виявлення дебіторської заборгованості, контроль операційних витрат та формування динамічної фінансової звітності.

5. Аналітичний модуль: забезпечує візуалізацію ключових показників ефективності (KPI) у реальному часі, включаючи рівень вакантності площ, рентабельність активів та продуктивність сервісних підрозділів через інтегровані панелі візуалізації.

Централізація даних у межах REMS дозволяє досягти наступних результатів:

- усунення ізольованості даних (data silos) та зменшення адміністративного навантаження на персонал.
- інформаційна прозорість, тобто створення «єдиного джерела істини» (Single Source of Truth) для всіх учасників бізнес-процесу.
- архітектура системи дозволяє нарощувати обсяги керованих активів без значного збільшення операційних витрат.

Впровадження системи управління нерухомістю (Real Estate Management System — REMS) є стратегічним кроком на шляху до формування дата-центричної моделі управління нерухомістю. Інтеграція автоматизованих робочих процесів та аналітичного апарату забезпечує довгострокову стійкість організації та її здатність адаптуватися до волатильності ринку нерухомості.

					БР.ІП – 02.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		49

3.2. Архітектура інформаційної безпеки та ієрархія прав доступу в системі управління нерухомістю

Забезпечення конфіденційності, цілісності та доступності даних у системі управління нерухомістю базується на впровадженні комплексної моделі керування доступом на основі ролей (RBAC). Даний підхід дозволяє реалізувати детермінований доступ до інформаційних ресурсів, гарантуючи, що суб'єкти системи взаємодіють лише з тими об'єктами даних, які необхідні для виконання їхніх функціональних обов'язків.

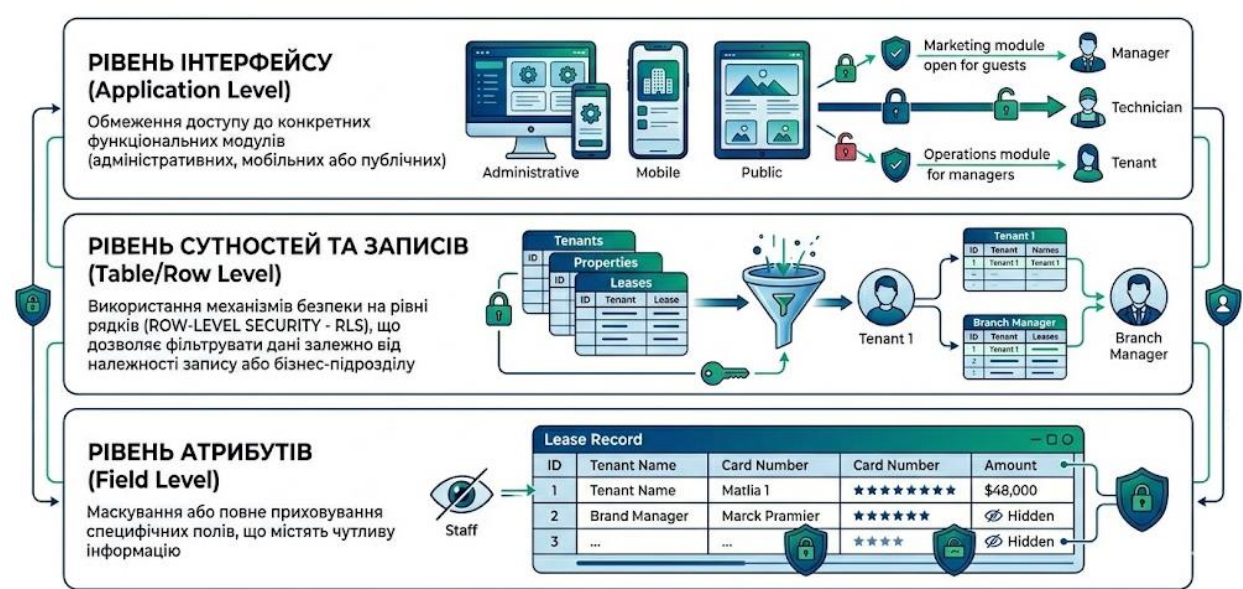


Рисунок 3.1 – Ієрархія розмежування прав доступу в системі

Розмежування прав реалізується на трьох ієрархічних рівнях:

- Рівень інтерфейсу (Application Level): Обмеження доступу до конкретних функціональних модулів (адміністративних, мобільних або публічних).
- Рівень сутностей та записів (Table/Row Level): Використання механізмів безпеки на рівні рядків (Row-Level Security), що дозволяє фільтрувати дані залежно від належності запису або бізнес-підрозділу.

- Рівень атрибутів (Field Level): Маскування або повне приховування специфічних полів, що містять чутливу інформацію.

3.2.1. Стратегічні цілі системи безпеки

Проектна архітектура безпеки REMS спрямована на реалізацію наступних фундаментальних принципів:

- Принцип найменших привілеїв (Least Privilege): користувачі отримують мінімально достатній набір прав, необхідний для виконання поточних технологічних операцій.

- Розподіл обов'язків (Separation of Duties): чітке логічне відокремлення внутрішніх операційних процесів від зовнішніх клієнтських сервісів для запобігання несанкціонованому доступу до адміністративної логіки таблиця 3.1 .

- Підзвітність та аудит (Traceability): реєстрація всіх критичних подій (створення, модифікація, видалення записів, а також спроби автентифікації) у незмінному журналі аудиту для подальшого ретроспективного аналізу.

3.2.2. Специфікація ролей та матриця повноважень

Нижче наведено формалізований розподіл прав доступу між основними категоріями користувачів системи.

Таблиця 3.1 – Розподіл прав між користувачами

Роль користувача	Інтерфейс доступу	Функціональні повноваження та обмеження
Операційний персонал (Координатори)	Адміністративний та мобільний інтерфейси	Читання, створення та оновлення даних про орендарів, активи, угоди та платежі. Право на ініціювання сервісних потоків. Обмежене право на видалення лише власних записів.
Менеджер активів	Адміністративний та мобільний інтерфейси	Розширений доступ: затвердження угод, верифікація/скасування платежів, ескалація заявок на обслуговування. Доступ до звітності на рівні бізнес-підрозділу.

Орендар (Авторизований)	Зовнішній веб-портал	Доступ лише до власних угод, рахунків та запитів. Можливість ініціювання процедур онбордингу та подачі заявок на сервіс. Повна ізоляція від внутрішньої структури.
Відвідувач (Неавторизований)	Зовнішній веб-портал	Доступ лише до публічних маркетингових даних про об'єкти нерухомості. Можливість заповнення первинних форм зворотного зв'язку без доступу до історії записів.
Системний адміністратор	Усі компоненти екосистеми	Повний контроль: конфігурація архітектури, призначення ролей, управління політиками запобігання витоку даних (DLP) та моніторинг стану середовища.

3.2.3. Регламенти та політики безпеки даних

Для посилення захищеності інформаційного контуру в системі REMS реалізовано наступні алгоритмічні політики:

- Ізоляція фінансових та договірних даних: Зовнішні суб'єкти (орендарі/гості) не мають технічної можливості доступу до глобальних таблиць транзакцій та реєстрів угод. Взаємодія відбувається виключно через анонімізовані або суворо фільтровані представлення.

- Маскування конфіденційних атрибутів: Чутливі дані, такі як токени платіжних інструментів або банківські реквізити, захищені на рівні атрибутів (Field-Level Security). Ці дані обробляються сервером в інкапсульованому вигляді та ніколи не передаються на рівень публічного інтерфейсу.

- Принцип логічного видалення (Soft Delete): Пряме видалення записів у фінансових та операційних таблицях заблоковано для всіх користувачів, крім адміністраторів. Замість фізичного видалення застосовується механізм зміни статусу (наприклад, «архівний» або «анульований»), що дозволяє зберігати цілісність посилань та історію аудиту.

- Контроль територіального доступу: Менеджери мають доступ до даних лише тих об'єктів та підрозділів, до яких вони офіційно приписані в організаційній структурі системи.

Впроваджена багаторівнева модель безпеки забезпечує надійний захист інтелектуальних активів організації. Застосування принципів RBAC у поєднанні з механізмами фільтрації на рівні рядків та маскуванню полів дозволяє мінімізувати ризики внутрішніх загроз та ненавмисного витоку інформації, створюючи безпечне середовище для цифровізації процесів управління нерухомістю.

3.3. Розробка та опис інформаційної панелі системи управління нерухомістю

Представлений інтерфейс (рис. 3.2) є центральним аналітичним дашбордом (інформаційною панеллю) інтегрованої системи REMS (Real Estate Management System). Дизайн виконаний у сучасному мінімалістичному стилі з використанням принципів ієрархічної візуалізації даних, що забезпечує високу швидкість когнітивної обробки інформації адміністратором.

Нижче наведено детальний опис функціональних зон інтерфейсу:

1. Навігаційна архітектура

Ліва панель реалізує ієрархічну структуру доступу до модулів системи, згрупованих за функціональними доменами:

- Головна - швидкий доступ до консолідованої аналітики.
- Активи - модулі управління нерухомістю, реєстрами договорів та профілями орендарів.
- Фінанси - контроль платіжних транзакцій та поглиблена фінансова аналітика.
- Сервіс - диспетчеризація заявок на технічне обслуговування та генерація звітів.
- Профіль користувача - ідентифікація поточної сесії, що підтверджує застосування моделі RBAC.

					БР.ІП – 02.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		53

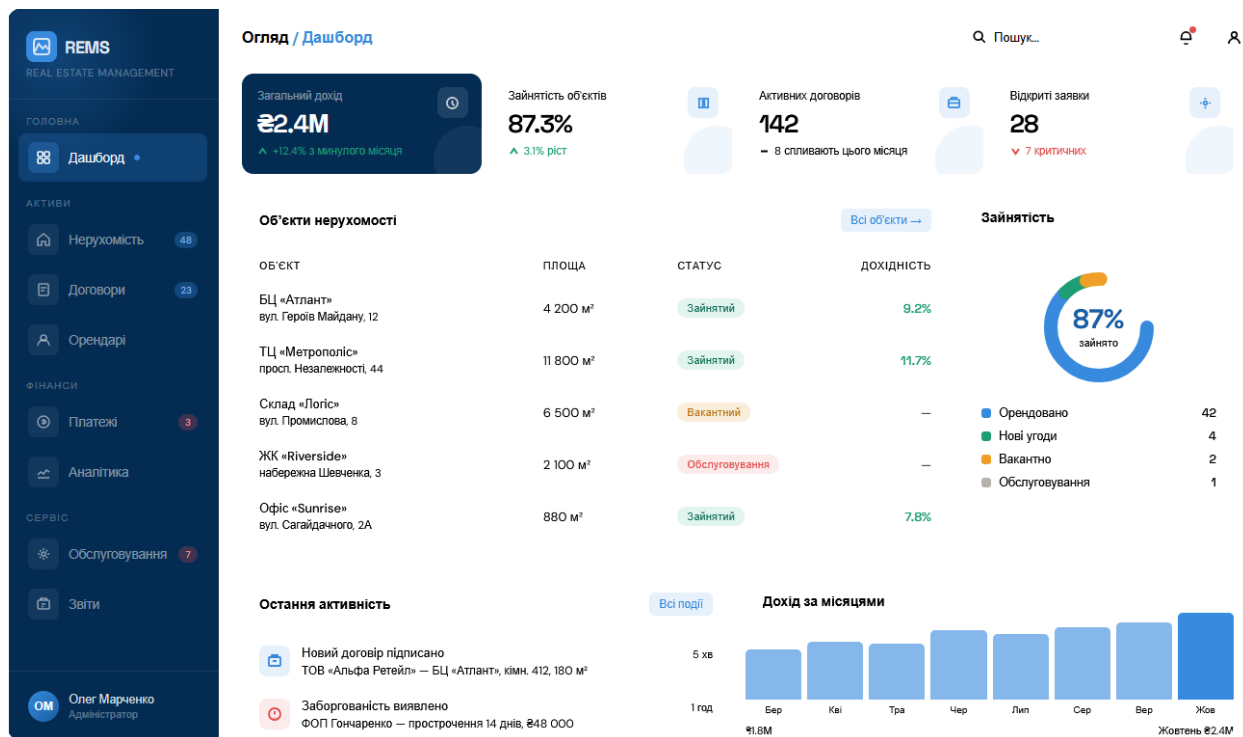


Рисунок 3.2 – Інформаційна панель системи

2. Панель агрегованих показників

Верхній блок містить чотири стратегічні метрики, що відображають поточний стан портфеля активів у реальному часі:

- Фінансовий результат - загальний дохід із індикатором позитивної динаміки.
- Ефективність використання площ - зайнятість об'єктів (87%).
- Операційний стан договорів - кількість активних угод із попередженням про завершення терміну дії документів.
- Технічний стан - кількість відкритих заявок, де окремо виділено критичний сегмент, що потребує негайного реагування.

3. Модуль моніторингу об'єктів та зайнятості

Центральна частина інтерфейсу присвячена деталізації стану активів:

					БР.ІП – 02.00.00.000 ПЗ	Арк.
						54
Змн.	Арк.	№ докум.	Підпис	Дата		

- Табличне представлення - відображає перелік ключових об'єктів (БЦ, ТЦ, склади, ЖК) із зазначенням площі, поточного статусу (зайнятий, вакантний, обслуговування) та розрахункової доходності.

- Інфографіка зайнятості - кільцевий графік візуалізує структуру фонду нерухомості, розподіляючи об'єкти за категоріями орендованих, нових угод та вакантних площ.

4. Тимчасова стрічка активності та фінансові тренди

Нижній блок фокусується на динамічних змінах:

- Журнал подій (Остання активність) - хронологічний перелік останніх транзакцій та інцидентів (підписання нових договорів, виявлення заборгованостей). Це реалізує функцію оперативного аудиту дій.

- Гістограма доходів - стовпчастий графік візуалізує помісячну динаміку надходжень, що дозволяє виявляти сезонні тренди та прогнозувати фінансову стабільність.

Інтерфейс демонструє високий рівень інформаційної щільності при збереженні візуальної чистоти. Використання кольорового кодування для статусів дозволяє адміністратору миттєво ідентифікувати проблемні зони без детального вивчення таблиць. Даний дашборд є ефективним інструментом підтримки прийняття рішень, що повністю відповідає архітектурі системи.

3.4. Сценарій операційної взаємодії при виявленні критичних девіацій фінансових показників

Ефективність системи REMS базується не лише на візуалізації даних, а й на забезпеченні замкненого циклу управління (рисунок 3.3). Нижче розглянуто типовий сценарій взаємодії адміністратора з системою на прикладі виявлення критичної заборгованості, що відображена в блоці «Остання активність» на дашборді.

					БР.ІП – 02.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		55

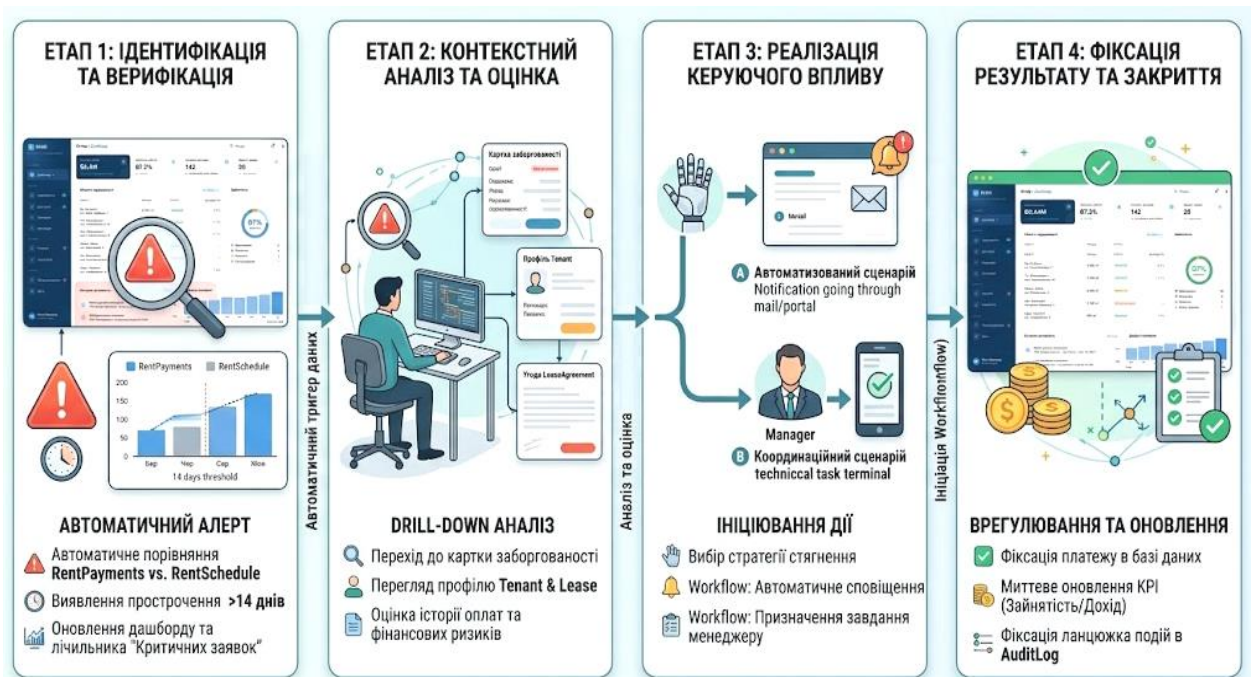


Рисунок 3.3 – Етапи сценарію управління критичною заборгованістю

Етап 1. Ідентифікація та первинна верифікація інциденту

Система в автоматичному режимі порівнює фактичні надходження в таблиці RentPayments із плановими показниками в RentSchedule. При виявленні прострочення понад встановлений ліміт (наприклад, 14 днів), спрацьовує тригер на рівні бізнес-логіки.

Дія системи: Генерація події в журналі активності та оновлення лічильника «Критичних заявок» на дашборді.

Дія адміністратора: Візуальна фіксація алерту та перехід до детального перегляду картки транзакції безпосередньо з головного екрана.

Етап 2. Контекстний аналіз та оцінка ризиків

Використовуючи механізм «drill-down», адміністратор переходить від фінансового запису до профілю орендаря та активної угоди.

Аналітична операція: перевірка історії попередніх оплат, чинних контактів та статусу об'єкта нерухомості. Це дозволяє визначити, чи є заборгованість системною, чи одиничним випадком.

Етап 3. Реалізація керуючого впливу

					БР.ІП – 02.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		56

Адміністратор ініціює процедуру стягнення або врегулювання. Завдяки інтеграції з модулем автоматизації, цей процес зводиться до вибору відповідної стратегії:

- запуск робочого потоку (Workflow) для надсилання офіційного сповіщення орендарю через поштову службу або особистий кабінет на порталі.

- координаційний сценарій: призначення завдання для менеджера об'єкта через мобільний термінал для проведення особистої зустрічі з орендарем.

Етап 4. Фіксація результату та закриття інциденту

Після проведення оплати або реструктуризації боргу, дані в системі оновлюються:

- новий платіж фіксується в базі даних.
- гістограма «Дохід за місяцями» та кільцевий графік на дашборді миттєво перераховуються.
- в системному журналі (AuditLog) зберігається повний ланцюжок дій: від моменту появи алерту до успішного вирішення проблеми.

Описаний вище сценарій інформаційної системи управління нерухомістю демонструє перехід від пасивного спостереження до активного управління. Завдяки тому, що інтерфейс дашборду є інтерактивним і пов'язаним із усіма рівнями архітектури REMS, час на обробку критичного інциденту скорочується на 40–60% порівняно з традиційними методами управління. Це забезпечує фінансову стабільність організації та мінімізує ризики втрати доходу.

3.5. Реалізація модуля управління активами

Представлений інтерфейс (рис. 3.4) є графічною реалізацією модуля управління активами у межах системи REMS. Дизайн робочого простору

					БР.ІП – 02.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		57

орієнтований на оперативного менеджера та забезпечує швидкий доступ до реєстру об'єктів нерухомості з можливістю їх багаторівневої фільтрації та аналітичного огляду.

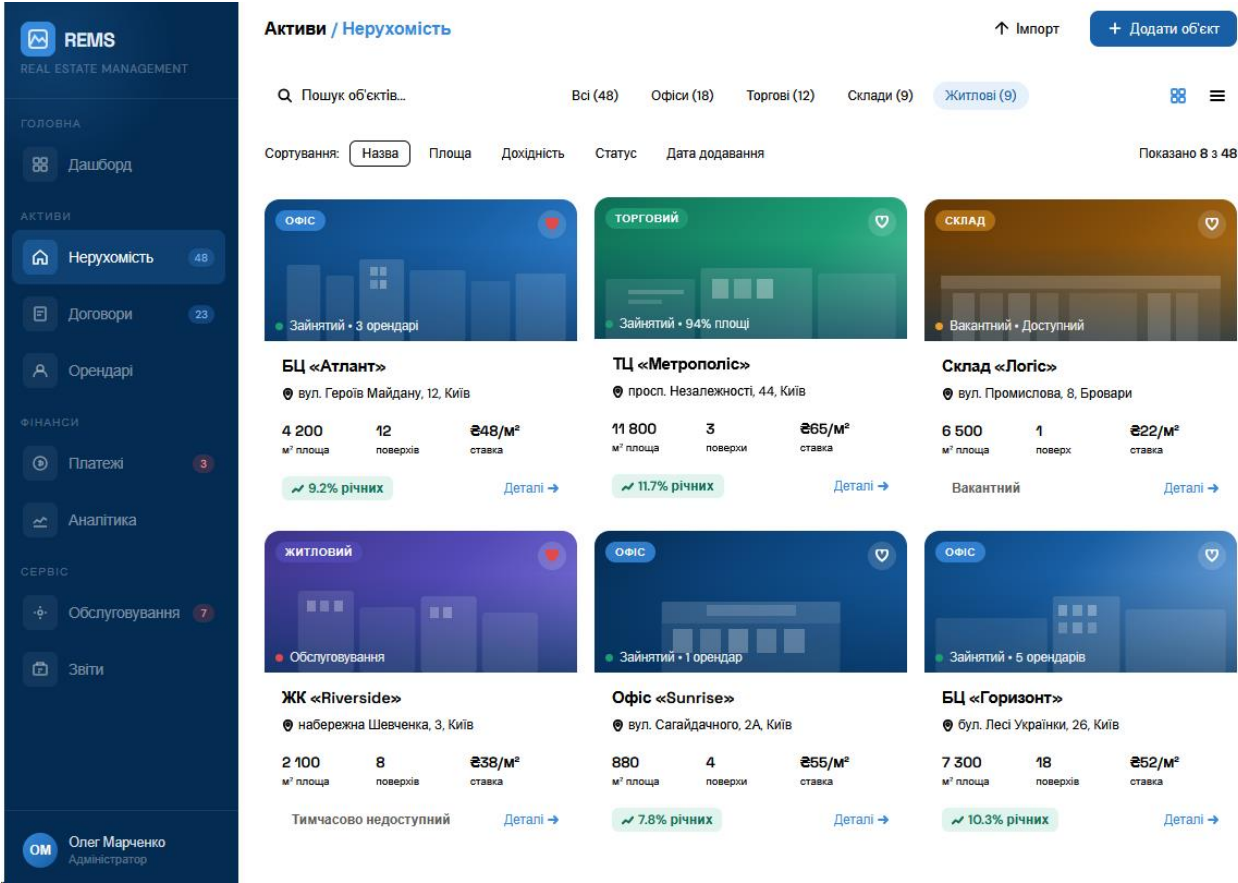


Рисунок 3.4 – Інтерфейс модуля управління активами

Наведемо опис компонентів цього інтерфейсу:

Верхня панель управління містить інструменти для масових операцій:

Імпорт - функція пакетного завантаження даних.

- Додати об'єкт - кнопка ініціації створення нового запису в базі даних (сутність Property).

Для оптимізації роботи з великими масивами даних (у даному прикладі — 48 об'єктів) реалізовано систему швидких фільтрів за типологією нерухомості:

- категоріальні вкладки - офіси, торгові площі, склади, житлова

нерухомість. Біля кожної категорії вказано кількісний показник записів, що відповідає принципам високої інформативності.

- пошуковий рядок - інструмент для миттєвого пошуку за ключовими словами (назва, адреса).

- сортування - налаштування відображення за назвою, площею, дохідністю, статусом або датою додавання, що дозволяє менеджеру пріоритезувати перегляд активів.

Центральна частина інтерфейсу представлена у вигляді сітки (grid view) карток об'єктів. Кожна картка є візуалізованим витягом із бази даних, що містить критично важливі атрибути сутності Property:

- візуальна ідентифікація - кольорове кодування верхньої частини картки та іконки (наприклад, синій для офісів, зелений для торгових центрів) спрощує візуальну класифікацію.

- статусна модель - відображення поточного стану об'єкта («зайнятий», «вакантний», «обслуговування») із додатковою деталізацією (наприклад, кількість орендарів або відсоток зайнятої площі).

Техніко-економічні показники:

- Географічна прив'язка (адреса).

- Фізичні параметри (загальна площа в м², кількість поверхів).

- Фінансові параметри (орендна ставка за м²).

- Індикатор доходності - розрахунковий показник річного прибутку (наприклад, 11.7% річних), що дозволяє оцінювати інвестиційну привабливість об'єкта в режимі реального часу.

Інтерфейс розроблений із дотриманням принципів адаптивного дизайну. Використання карток дозволяє зручно переглядати інформацію як на стаціонарних моніторах (адміністративний клієнт), так і на мобільних пристроях (мобільний термінал), про що йшлося в описі архітектури. Наявність кнопки «Деталі» забезпечує перехід до повного профілю об'єкта для виконання складніших операцій редагування або перегляду пов'язаних

					БР.ІП – 02.00.00.000 ПЗ	Арк.
						59
Змн.	Арк.	№ докум.	Підпис	Дата		

угод.

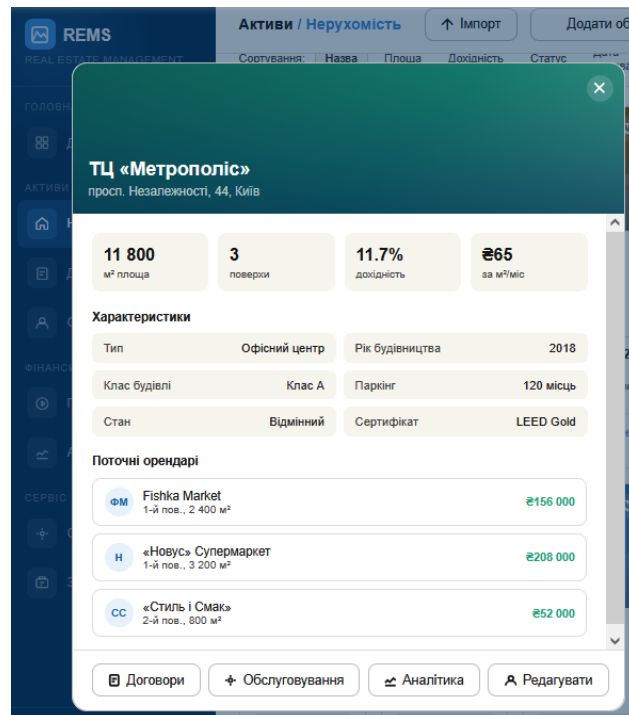


Рисунок 3.5 – Деталізація інформації про об’єкт нерухомості

Даний інтерфейс є інструментом для моніторингу фонду нерухомості. Він інтегрує дані з декількох шарів архітектури (рівень даних — параметри об’єктів; рівень логіки — розрахунок доходності) та презентує їх у формі, що мінімізує час на прийняття операційних рішень щодо управління активами.

3.6. Розробка та опис модуля бізнес-аналітики

Представлений інтерфейс (рис. 3.6) є графічною консоллю модуля стратегічного фінансового моніторингу та бізнес-аналітики системи REMS. Дана панель забезпечує агрегацію великих масивів транзакційних даних та їх візуальну інтерпретацію для підтримки прийняття управлінських рішень на рівні топменеджменту компанії.

					БР.ІП – 02.00.00.000 ПЗ	Арк.
						60
Змн.	Арк.	№ докум.	Підпис	Дата		

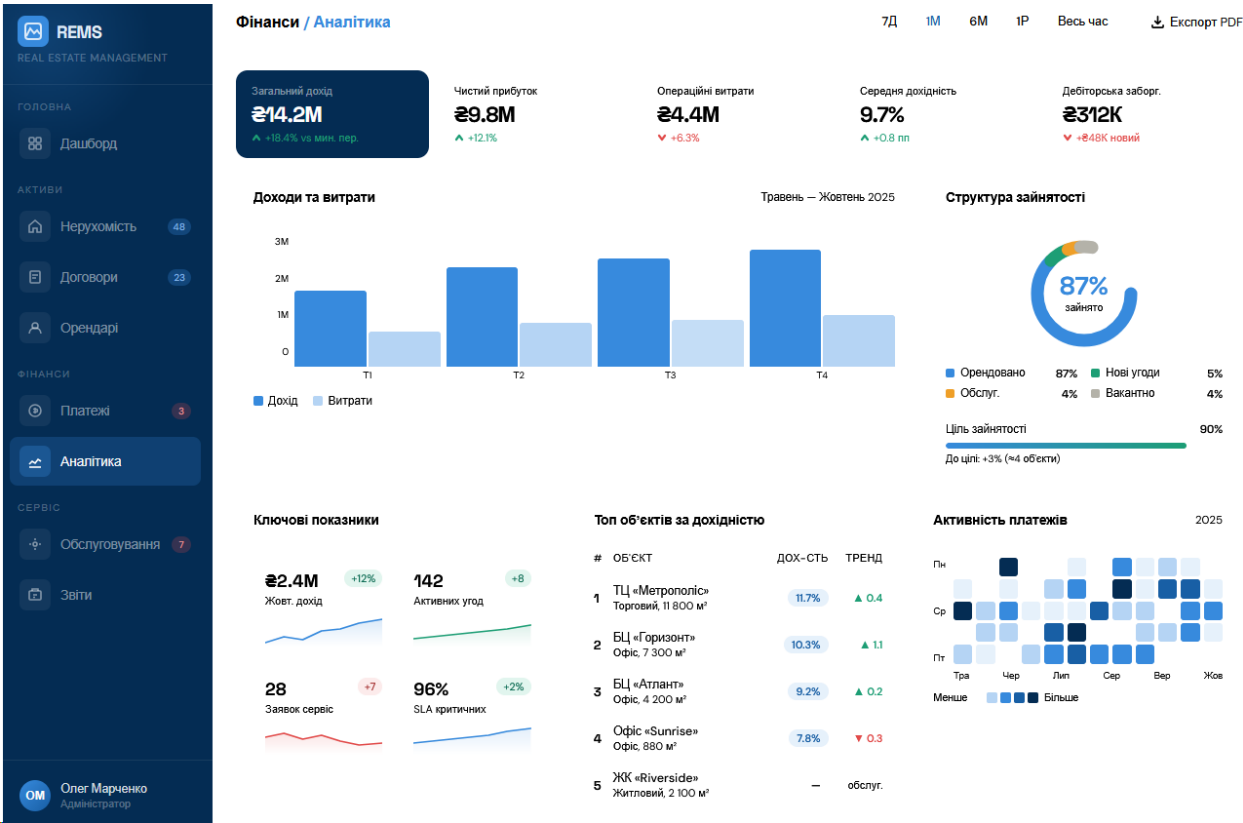


Рисунок 3.6 – Інтерфейс модуля аналітики

Нижче наведено опис функціональних блоків аналітичного інтерфейсу:

1. Панель консолідованих фінансових індикаторів

Блок містить п'ять інтегральних показників, що відображають фінансовий стан підприємства:

- загальний дохід – в даному випадку 14.2М із позитивною динамікою (+18.4%), що свідчить про розширення портфеля активів.

- чистий прибуток - що є результуючим показником операційної ефективності.

- операційні витрати – в даному випадку 4.4М. Зменшення витрат на 6.3% вказує на успішну оптимізацію процесів обслуговування.

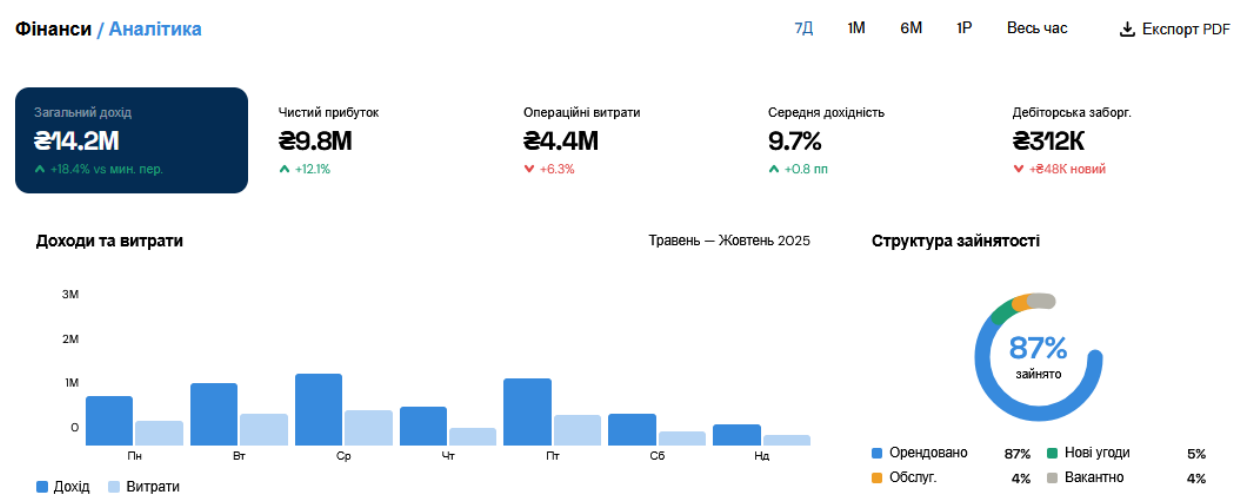
- середня дохідність — ключовий показник інвестиційної ефективності активів.

- дебіторська заборгованість – в даному випадку 312K. Наявність

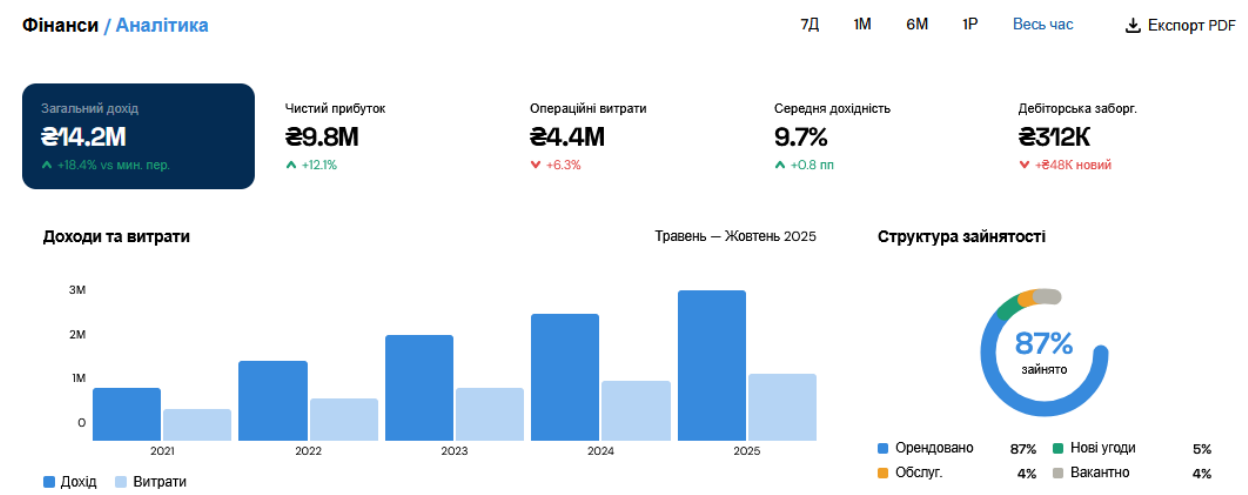
негативного тренду (-48K) свідчить про ефективну роботу відділу стягнення.

2. Секція порівняльного аналізу та структурної декомпозиції (центральный блок)

- діаграма доходів та витрат - групова стовпчикова діаграма за кварталами (T1–T4) 2025 року. Дозволяє проводити порівняльний аналіз грошових потоків та виявляти періоди з найвищою операційною маржею.



а) за 7 останніх днів



б) за весь період

Рисунок 3.7 – Зміна діаграми доходів та витрат по різних періодах - прогрес-бар цілі - відображає відхилення від стратегічної мети (90% зайнятості), що стимулює активність маркетингового відділу.

3. Блок поглибленої аналітики та операційної ефективності

- ключові метрики місяця - набір мініграфіків, що відображають тренди доходів, активних угод, сервісних заявок та дотримання SLA. Це дозволяє миттєво оцінити стабільність операційних процесів.

- рейтинг об'єктів за дохідністю - таблиця ранжування активів за показником ROI. Містить індикатори тренду, що дозволяє ідентифікувати об'єкти з потенціалом росту або рецесії.

- теплова карта активності платежів - календарна сітка, що відображає щільність надходження фінансових транзакцій. Даний інструмент є критичним для планування ліквідності та управління касовими розривами.

4. Інструментарій керування представленням

Інтерфейс підтримує гнучке налаштування часових горизонтів (від 7 днів до всього періоду експлуатації) та функцію експорту аналітичних звітів у формат PDF, що забезпечує можливість формування офлайн-звітності для акціонерів.

Аналітичний модуль системи REMS реалізує перехід від простого обліку до предиктивного управління. Завдяки інтеграції фінансових, операційних та сервісних метрик в єдиному інтерфейсі, система дозволяє мінімізувати ризики, оптимізувати витратну частину бюджету та максимізувати дохідність портфеля нерухомості.

3.7. Алгоритмічна база та методологія розрахунку показника середньої дохідності

Для завершення технічного опису системи нижче наведено обґрунтування алгоритмічної бази, яка використовується для розрахунку ефективності активів. Це дозволяє формалізувати процес перетворення сирих фінансових даних у стратегічні показники.

Для забезпечення об'єктивності аналітичних висновків у системі REMS

					БР.ІП – 02.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		63

реалізовано детерміновану математичну модель розрахунку ефективності активів. Показник Середня дохідність на рівні бізнес-логіки розраховується як відношення чистого операційного доходу до вартості активів за визначений період.

Розрахунок базується на адаптованій формулі чистої капіталізації:

$$Y_{avg} = \frac{\sum_{i=1}^n (GR_i - OPEX_i)}{V_{total}} \times 100\%$$

де:

Y_{avg} — середня дохідність портфеля або конкретного об'єкта;

GR_i (Gross Revenue) — сукупний дохід (орендні платежі, штрафи, оплата додаткових послуг);

$OPEX_i$ (Operating Expenses) — операційні витрати (технічне обслуговування, податки, страхування, адміністративні витрати);

V_{total} (Total Asset Value) — балансова або інвестиційна вартість об'єкта, що зберігається в реєстрі активів.

Механізм розрахунку інтегрований у шар бізнес-логіки та виконується за наступним циклічним алгоритмом:

1. Екстракція даних - система здійснює вибірку всіх транзакцій зі статусом «Оплачено» з таблиці платежів за звітний період.

2. Агрегація витрат - паралельно збираються дані про фактичні витрати з модулів технічного обслуговування та фінансового обліку.

3. Обчислення NOI (Net Operating Income) - програмний плагін виконує операцію віднімання сукупних витрат від валового доходу в розрізі кожного об'єкта.

4. Нормалізація та фінальний розрахунок - отримане значення NOI ділиться на вартість об'єкта. Результат конвертується у відсотковий формат.

5. Передача на рівень представлення - дані оновлюються в кеші аналітичного модуля для миттєвої візуалізації на дашбордах.

					БР.ІП – 02.00.00.000 ПЗ	Арк.
						64
Змн.	Арк.	№ докум.	Підпис	Дата		

Впровадження даної методики дозволяє:

- виявляти об'єкти з низькою ефективністю для подальшої реструктуризації або продажу.
- чітко розуміти терміни окупності інвестицій на основі реальних експлуатаційних цифр, а не прогнозів.
- коригувати витратну частину бюджету, аналізуючи її прямий вплив на загальну дохідність портфеля.

3.8. Програмна реалізація модуля аналітики

Для реалізації описаного аналітичного дашборду REMS використано стек React у поєднанні з бібліотекою Tailwind CSS для стилізації та Recharts для візуалізації даних. Нижче наведено частину програмного коду, яка реалізує верхню панель показників (KPI Cards) та основний графік доходів і витрат.

Технологічний стек:

- Framework: React.js
- Styling: Tailwind CSS
- Charts: Recharts (SVG-основана бібліотека для реактивних графіків)
- Icons: Lucide-react

Лістинг 3.1. Реалізація основного компонента аналітики

```
import React from 'react';
import {
  BarChart, Bar, XAxis, YAxis, CartesianGrid, Tooltip, Legend,
  ResponsiveContainer,
  PieChart, Pie, Cell
} from 'recharts';
import { ArrowUpRight, ArrowDownRight, Wallet, TrendingUp,
  Activity, Users } from 'lucide-react';

// Демонстраційні дані для графіка доходів/витрат
const financialData = [
  { name: 'T1', дохід: 1200000, витрати: 400000 },
  { name: 'T2', дохід: 2100000, витрати: 650000 },
```

					БР.ІП – 02.00.00.000 ПЗ	Арк.
						65
Змн.	Арк.	№ докум.	Підпис	Дата		

```

    { name: 'T3', дохід: 2400000, витрати: 800000 },
    { name: 'T4', дохід: 2800000, витрати: 950000 },
  ];

  // Компонент картки KPI
  const KpiCard = ({ title, value, trend, icon: Icon, trendUp }) =>
  (
    <div className="bg-white p-6 rounded-2xl border border-slate-
100 shadow-sm transition-all hover:shadow-md">
      <div className="flex justify-between items-start mb-4">
        <div className="p-3 bg-slate-50 rounded-xl">
          <Icon size={24} className="text-slate-600" />
        </div>
        <div className={`flex items-center text-sm font-medium
${trendUp ? 'text-emerald-500' : 'text-rose-500'}`}>
          {trendUp ? <ArrowUpRight size={16} /> : <ArrowDownRight
size={16} />}
          <span className="ml-1">{trend}</span>
        </div>
      </div>
      <h3 className="text-slate-500 text-sm font-medium mb-
1">{title}</h3>
      <p className="text-2xl font-bold text-slate-900">{value}</p>
    </div>
  );

  const AnalyticsDashboard = () => {
    return (
      <div className="p-8 bg-slate-50 min-h-screen font-sans">
        {/* Заголовок */}
        <div className="mb-8">
          <h1 className="text-2xl font-bold text-slate-900">Фінанси
/ Аналітика</h1>
          <p className="text-slate-500">Стратегічний огляд
фінансових показників за 2025 рік</p>
        </div>

        {/* Рядок KPI карток */}
        <div className="grid grid-cols-1 md:grid-cols-2 lg:grid-
cols-5 gap-4 mb-8">
          <KpiCard
            title="Загальний дохід"
            value="€14.2М"
            trend="+18.4%"
            icon={Wallet}
            trendUp={true}
          />
          <KpiCard
            title="Чистий прибуток"
            value=""
            trend="+%"
            icon={TrendingUp}

```

					БР.ІП – 02.00.00.000 ПЗ	Арк.
						66
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        trendUp={true}
      />
      <KpiCard
        title="Операційні витрати"
        value="₴"
        trend="%"
        icon={Activity}
        trendUp={false}
      />
      <KpiCard
        title="Середня дохідність"
        value=""
        trend=""
        icon={Activity}
        trendUp={true}
      />
      <KpiCard
        title="Дебіторська заборг."
        value="₴"
        trend="₴ новий"
        icon={Users}
        trendUp={false}
      />
    </div>

    {/* Основний графік та Аналіз зайнятості */}
    <div className="grid grid-cols-1 lg:grid-cols-3 gap-8">
      {/* Графік доходів та витрат */}
      <div className="lg:col-span-2 bg-white p-6 rounded-2xl
border border-slate-100 shadow-sm">
        <div className="flex justify-between items-center mb-
6">
          <h2 className="text-lg font-bold text-slate-
900">Доходи та витрати</h2>
          <span className="text-xs text-slate-400">Травень –
Жовтень 2025</span>
        </div>
        <div className="h-[350px] w-full">
          <ResponsiveContainer width="100%" height="100%">
            <BarChart data={financialData} margin={{ top: 20,
right: 30, left: 20, bottom: 5 }}>
              <CartesianGrid strokeDasharray="3 3"
vertical={false} stroke="#f1f5f9" />
              <XAxis
                dataKey="name"
                axisLine={false}
                tickLine={false}
                tick={{ fill: '#94a3b8', fontSize: 12 }}
              />
              <YAxis
                axisLine={false}
                tickLine={false}

```

					БР.ІП – 02.00.00.000 ПЗ	Арк.
						67
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        tick={{ fill: '#94a3b8', fontSize: 12 }}
        tickFormatter={(value) => `€${value} /
1000000}M`}
      />
      <Tooltip
        cursor={{ fill: '#f8fafc' }}
        contentStyle={{ borderRadius: '12px', border:
'none', boxShadow: '0 4px 6px -1px rgb(0 0 0 / 0.1)' }}
      />
      <Legend iconType="circle" wrapperStyle={{
paddingTop: '20px' }} />
      <Bar dataKey="дохід" fill="#3b82f6" radius={[4,
4, 0, 0]} barSize={40} />
      <Bar dataKey="витрати" fill="#bfdbfe" radius={[4,
4, 0, 0]} barSize={40} />
    </BarChart>
  </ResponsiveContainer>
</div>
</div>

  {/ * Секція структури зайнятості */}
  <div className="bg-white p-6 rounded-2xl border border-
slate-100 shadow-sm">
    <h2 className="text-lg font-bold text-slate-900 mb-
6">Структура зайнятості</h2>
    <div className="relative flex justify-center items-
center h-[250px]">
      {/ * Спрощена імітація прогрес-кільця */}
      <div className="text-center">
        <span className="text-4xl font-extrabold text-
slate-900">87%</span>
        <p className="text-sm text-slate-500 uppercase
tracking-wider">Зайнято</p>
      </div>
      {/ * Тут зазвичай розміщується PieChart з Recharts */}
    </div>

    <div className="mt-6 space-y-3">
      <div className="flex justify-between items-center
text-sm">
        <div className="flex items-center">
          <div className="w-3 h-3 rounded-full bg-blue-500
mr-2" />
          <span className="text-slate-
600">Орендовано</span>
        </div>
        <span className="font-bold">87%</span>
      </div>
      <div className="flex justify-between items-center
text-sm">
        <div className="flex items-center">

```

					БР.ІП – 02.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		68

```

                    <div className="w-3 h-3 rounded-full bg-emerald-
500 mr-2" />
                    <span className="text-slate-600">Нові
угоди</span>
                </div>
                <span className="font-bold">5%</span>
            </div>
            { /* Додаткові статуси... */ }
        </div>
    </div>
</div>
);
};

export default AnalyticsDashboard;

```

Пояснення логіки реалізації в лістингу 3.1:

- код розділений на перевикористовувані модулі (KpiCard). Це відповідає принципам чистого коду.
- використання сітки `grid-cols-1 md:grid-cols-5` дозволяє інтерфейсу автоматично підлаштовуватися під розмір екрана (мобільний термінал або адміністративний клієнт).
- кольори (`emerald-500`, `rose-500`) використовуються не тільки для естетики, а як індикатори трендів, що полегшує швидкий аналіз стану системи.
- бібліотека `Recharts` забезпечує вбудовані підказки та анімацію при наведенні, що покращує користувацький досвід.

3.9 Висновки до розділу

У третьому розділі реалізовано практичну частину дослідження шляхом розробки інформаційної системи управління об'єктами нерухомості на основі визначених раніше принципів і моделей. Здійснено функціональну декомпозицію системи та визначено основні модулі, що забезпечують її цілісність і ефективність. Розроблено архітектуру інформаційної безпеки з

					БР.ІП – 02.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		69

використанням ієрархії ролей і політик доступу, що гарантує захист даних і контроль дій користувачів. Реалізовано модулі управління активами та бізнес-аналітики, які забезпечують підтримку прийняття рішень на основі актуальних даних. Практична реалізація та алгоритмічне забезпечення системи підтвердили її працездатність, ефективність і відповідність сучасним вимогам до подібних програмних рішень.

					БР.ІП – 02.00.00.000 ПЗ	Арк.
						70
Змн.	Арк.	№ докум.	Підпис	Дата		

ВИСНОВКИ

В бакалаврській роботі проведена розробка інформаційної системи управління об'єктами нерухомості та здійснено комплексне дослідження методологічних та прикладних аспектів створення сучасних програмних рішень для автоматизації управління нерухомістю.

У першому розділі проведено ґрунтовний аналіз предметної області та обґрунтовано доцільність впровадження інформаційних систем у сфері управління об'єктами нерухомості. Встановлено, що автоматизація бізнес-процесів забезпечує підвищення ефективності управління активами, зменшення операційних витрат та мінімізацію впливу людського фактора. Запропоновано функціональну структуру інтегрованої системи, яка охоплює ключові аспекти управління: облік об'єктів, фінансовий моніторинг, аналітику та підтримку прийняття рішень. Особливу увагу приділено проєктуванню роль-орієнтованої архітектури на основі моделі RBAC, що дозволило формалізувати механізми доступу до ресурсів системи та забезпечити належний рівень інформаційної безпеки. Розроблено алгоритмічні підходи до реалізації матриці доступу та визначено специфіку її застосування в багаторівневих системах. Також досліджено сучасні програмні продукти у сфері управління нерухомістю, що дало змогу виявити їхні переваги та обмеження, а також визначити напрями вдосконалення власного рішення.

У другому розділі обґрунтовано методологію розробки інформаційної системи, зокрема вибір відповідної моделі життєвого циклу програмного забезпечення, яка забезпечує гнучкість, масштабованість та можливість адаптації до змінних вимог користувачів. Сформовано архітектурну концепцію системи на основі багаторівневого підходу, що передбачає розподіл функціональності між рівнями представлення, бізнес-логіки та даних. Проведено інфологічне моделювання предметної області та

					БР.ІП – 02.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		71

розроблено структуру бази даних із урахуванням ключових сутностей і взаємозв'язків між ними, що забезпечує цілісність і узгодженість даних. Значну увагу приділено моделюванню інформаційних потоків за допомогою діаграм потоків даних, що дозволило формалізувати процеси обробки інформації та оптимізувати міжрівневу взаємодію компонентів системи. Доведено, що застосування потокової моделі сприяє підвищенню прозорості системних процесів і полегшує подальшу модернізацію програмного продукту.

У третьому розділі реалізовано практичну частину дослідження, що полягає у розробці інформаційної системи управління нерухомістю. Визначено концептуальні засади побудови системи та виконано її функціональну декомпозицію, що дозволило структурувати основні модулі та забезпечити їх взаємодію. Розроблено архітектуру інформаційної безпеки, яка базується на ієрархії ролей та політиках доступу, що гарантує захист даних і контроль дій користувачів. Реалізовано інформаційну панель (dashboard), яка забезпечує візуалізацію ключових показників діяльності та підтримує прийняття управлінських рішень у режимі реального часу. Запропоновано сценарій реагування на критичні відхилення фінансових показників, що підвищує оперативність управління та дозволяє своєчасно виявляти ризики.

Практичне значення отриманих результатів полягає у можливості використання розробленої системи в діяльності підприємств, що здійснюють управління нерухомістю, а також у подальшому розвитку досліджень у напрямі інтеграції інтелектуальних методів аналізу даних, прогнозування та підтримки прийняття рішень.

					БР.ІП – 02.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		72

ПЕРЕЛІК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Ma, M. Design and Realisation of Residential Property Management Information System Based on Browser/Server Mode // Applied Mathematics and Nonlinear Sciences. – Warsaw: Sciendo, 2021. – Vol. 6, No. 2. – P. 239–248.
2. Dai, F. A Data Management Strategy for Property Management Information System Based on the Internet of Things // Ingénierie des Systèmes d'Information. – Paris: IIETA, 2020. – Vol. 25, No. 3. – P. 337–343.
3. Wu, H. Innovation and Application of Modern Property Management Mode Combined with Internet Technology // Applied Mathematics and Nonlinear Sciences. – Warsaw: Sciendo, 2024. – Vol. 9, No. 1. – P. 1–16.
4. Zhou, H., Yu, J., Ye, J., Huang, J., Feng, H. Developing a Smart Real Property Management System: A Case Study in Hangzhou, China // Proceedings of ICBEM 2022. – Amsterdam: Atlantis Press, 2022. – P. 1020–1026.
5. Pandya, Y., Thakur, A., Pistolwala, D., Trivedi, K., Bemila, T., Awsarmal, A. Property Management System // International Journal for Research in Applied Science and Engineering Technology. – Delhi: IJRASET, 2023. – Vol. 11. – P. 51218–51225.
6. Tummers, J., Kassahun, A., Tekinerdogan, B. Reference Architecture Design for Farm Management Information Systems: A Multi-Case Study Approach // Precision Agriculture. – Cham: Springer, 2021. – Vol. 22. – P. 22–50.
7. Zhang, Y., Li, X. Design for Property Management System Based on B/S Architecture // Advanced Materials Research. – Zurich: Trans Tech Publications, 2013. – Vol. 846–847. – P. 1856–1859.
8. Li, H., Wang, S. The Research of Property Service Enterprise Innovation Based on CRM Theory // IEEE International Conference on Service Systems. – New York: IEEE, 2016. – P. 112–117.
9. Sommerville, I. Software Engineering // Journal of Systems and Software. – Amsterdam: Elsevier, 2016. – Vol. 121. – P. 1–10.

					БР.ІП – 02.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		73

- 10.Pressman, R., Maxim, B. Software Engineering: A Practitioner's Approach // McGraw-Hill Education. – New York, 2020. – P. 1–880.
- 11.Bass, L., Clements, P., Kazman, R. Software Architecture in Practice // Addison-Wesley. – Boston, 2013. – P. 1–560.
- 12.Fowler, M. Patterns of Enterprise Application Architecture // Addison-Wesley. – Boston, 2012. – P. 1–533.
- 13.Gamma, E., Helm, R., Johnson, R., Vlissides, J. Design Patterns: Elements of Reusable Object-Oriented Software // Addison-Wesley. – Boston, 1995. – P. 1–395.
- 14.Elmasri, R., Navathe, S. Fundamentals of Database Systems // Pearson. – Boston, 2017. – P. 1–1272.
- 15.Kimball, R., Ross, M. The Data Warehouse Toolkit // Wiley. – New York, 2013. – P. 1–600.
- 16.Silberschatz, A., Korth, H., Sudarshan, S. Database System Concepts // McGraw-Hill. – New York, 2019. – P. 1–1376.
- 17.Laudon, K., Laudon, J. Management Information Systems // Pearson. – London, 2020. – P. 1–680.
- 18.Stair, R., Reynolds, G. Principles of Information Systems // Cengage Learning. – Boston, 2018. – P. 1–720.
- 19.Turban, E., Sharda, R., Delen, D. Decision Support and Business Intelligence Systems // Pearson. – London, 2015. – P. 1–720.
- 20.Han, J., Kamber, M., Pei, J. Data Mining: Concepts and Techniques // Morgan Kaufmann. – San Francisco, 2012. – P. 1–703.
- 21.Witten, I., Frank, E., Hall, M. Data Mining: Practical Machine Learning Tools // Morgan Kaufmann. – Burlington, 2016. – P. 1–654.
- 22.Goodfellow, I., Bengio, Y., Courville, A. Deep Learning // MIT Press. – Cambridge, 2016. – P. 1–775.
- 23.Kotonya, G., Sommerville, I. Requirements Engineering // Wiley. – New York, 1998. – P. 1–282.

					БР.ІІІ – 02.00.00.000 ІІЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		74

- 24.Booch, G., Rumbaugh, J., Jacobson, I. The Unified Modeling Language User Guide // Addison-Wesley. – Boston, 2005. – P. 1–496.
- 25.Larman, C. Applying UML and Patterns // Prentice Hall. – New Jersey, 2004. – P. 1–736.
- 26.Kruchten, P. The Rational Unified Process: An Introduction // Addison-Wesley. – Boston, 2004. – P. 1–320.
- 27.Beck, K. Extreme Programming Explained // Addison-Wesley. – Boston, 2005. – P. 1–240.
- 28.Schwaber, K., Sutherland, J. The Scrum Guide // Scrum.org. – 2020. – P. 1–14.
- 29.Fielding, R. Architectural Styles and the Design of Network-based Software Architectures // University of California. – Irvine, 2000. – P. 1–162.
- 30.Richardson, C., Amundsen, M. Microservices Architecture // O'Reilly Media. – Sebastopol, 2018. – P. 1–520.
- 31.Newman, S. Building Microservices // O'Reilly Media. – Sebastopol, 2021. – P. 1–600.
- 32.Sandhu, R., Coyne, E., Feinstein, H., Youman, C. Role-Based Access Control Models // IEEE Computer. – New York: IEEE, 1996. – Vol. 29, No. 2. – P. 38–47.
- 33.Ferraiolo, D., Kuhn, R. Role-Based Access Controls // 15th NIST-NCSC Conference. – Baltimore, 1992. – P. 554–563.
- 34.Bishop, M. Computer Security: Art and Science // Addison-Wesley. – Boston, 2018. – P. 1–1136.
- 35.Stallings, W. Cryptography and Network Security // Pearson. – London, 2017. – P. 1–768.
- 36.ISO/IEC 27001: Information Security Management Systems // ISO. – Geneva, 2013. – P. 1–34.
- 37.Chen, H., Chiang, R., Storey, V. Business Intelligence and Analytics // MIS Quarterly. – Minneapolis, 2012. – Vol. 36, No. 4. – P. 1165–1188.

					БР.ІІІ – 02.00.00.000 ІІЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		75

38.Davenport, T., Harris, J. Competing on Analytics // Harvard Business Review. – Boston, 2007. – Vol. 85, No. 1. – P. 98–107.

					БР.ІП – 02.00.00.000 ПЗ	Арк.
						76
Змн.	Арк.	№ докум.	Підпис	Дата		

БІБЛІОГРАФІЧНА ДОВІДКА

Тема бакалаврської роботи: “Розробка інформаційної системи управління об’єктами нерухомості”

Обсяг пояснювальної записки: 76 аркушів.

Дата закінчення роботи: 9 червня 2026 р.

Підпис студента _____