

# ІНФОРМАЦІЙНА СИСТЕМА ДЛЯ АВТОМАТИЗАЦІЇ ПРОЦЕСІВ ПРОДАЖУ ОБ'ЄКТІВ НЕРУХОМОСТІ

Вінницький національний технічний університет

## *Анотація*

*У роботі представлено проєкт розроблення програмного додатка для інформаційної системи продажу нерухомості. Реалізовано модулі керування об'єктами нерухомості, обліку клієнтів, укладання угод, пошуку та аналізу вартості. Програму створено мовою С# із використанням Windows Forms та принципів об'єктно-орієнтованого програмування. Використання об'єктно-орієнтованого підходу дало змогу чітко структурувати предметну область, забезпечити узгоджену взаємодію між ключовими елементами системи та створити гнучке програмне рішення, яке легко розширювати новими функціональними можливостями.*

**Ключові слова:** інформаційна система, програмний додаток, об'єкт нерухомості, продажі, об'єктно-орієнтоване програмування, С#.

## *Abstract*

*The paper presents a project to develop a software application for a real estate sales information system. Modules for managing real estate objects, accounting for customers, concluding transactions, searching and analyzing costs have been implemented. The program was created in C# using Windows Forms and the principles of object-oriented programming. The use of an object-oriented approach made it possible to clearly structure the subject area, ensure consistent interaction between key elements of the system, and create a flexible software solution that is easy to expand with new functionality.*

**Keywords:** information system, software application, real estate, sales, object-oriented programming, C#.

## **Вступ**

У сучасних умовах розвитку ринку нерухомості зростає потреба в ефективних програмних засобах для автоматизації процесів обліку та управління даними. Використання інформаційних систем дозволяє підвищити продуктивність роботи агентств нерухомості, спростити пошук об'єктів та покращити якість обслуговування клієнтів.

Актуальність дослідження зумовлена необхідністю створення програмного рішення, яке забезпечує зручне керування інформацією про об'єкти нерухомості, клієнтів і угоди без залучення складної серверної інфраструктури.

Метою роботи є розробка архітектури та програмна реалізація інформаційної системи для продажу нерухомості на основі принципів об'єктно-орієнтованого програмування [1]. Для реалізації застосовано платформу .NET [2] та мову програмування С# [3].

## **Результати дослідження**

У ході розроблення інформаційної системи було здійснено дослідження предметної області, визначено основні функціональні вимоги та проаналізовано типові сценарії взаємодії користувачів із програмними продуктами у сфері нерухомості. Отримані результати стали підґрунтям для формування архітектури системи, визначення її модульної структури та організації зберігання даних.

Аналіз існуючих програмних засобів показав, що значна частина з них має низку недоліків. Зокрема, спостерігається складність користувацького інтерфейсу, перевантаженість функціоналу, обмежені можливості пошуку та фільтрації об'єктів, а також недостатня структурованість даних. Крім того, деякі рішення потребують додаткових витрат для розширення функціональних можливостей.

Урахування виявлених недоліків дозволило сформулювати вимоги до інформаційної системи, яка повинна забезпечувати зручний доступ до інформації, ефективний пошук об'єктів нерухомості, можливість створення та редагування угод, а також аналіз вартості.

Додаток призначений для введення, редагування, видалення та зберігання даних про об'єкт нерухомості. Користувач може вводити такі дані, як: адреса, площа, ціна, тип нерухомості, поверх (для квартир).

Розроблений додаток дозволяє:

- редагувати дані про об'єкт;
- видаляти об'єкт зі списку;
- зберігати інформацію до бази даних;
- формувати договір про продаж об'єкту;
- формувати звіт про продані об'єкти;

Для відображення функціональних можливостей системи побудовано діаграму варіантів використання (рис. 1), що відповідає стандартам UML-моделювання [4]. Вона демонструє основні сценарії роботи користувача з системою.

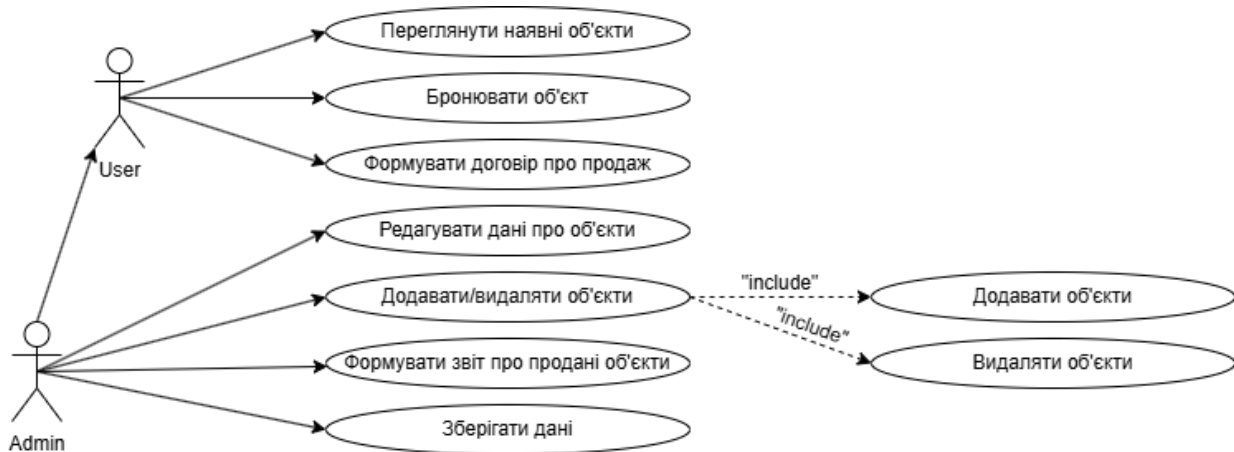


Рис. 1. Use case діаграма

Алгоритм функціонування програмного додатку включає такі основні етапи:

- запуск додатку;
- ініціалізацію компонентів та завантаження даних;
- перегляд наявних об'єктів нерухомості;
- редагування даних про об'єкти;
- бронювання потрібного об'єкту;
- формування договору про продаж об'єкту;
- збереження даних;
- формування звіту про продані об'єкти;
- завершення роботи додатку.

Після запуску додатку користувач переходить до головного вікна з переліком об'єктів нерухомості, де має можливість взаємодіяти з каталогом. Адміністратор, у свою чергу, може здійснювати керування об'єктами та переглядати інформацію про них. Після виконання необхідних дій робота з додатком завершується закриттям програми.

У процесі проєктування додатку було виділено такі основні функціональні модулі:

- модуль роботи з даними, що забезпечує створення, редагування, видалення та збереження інформації про об'єкти нерухомості;
- модуль оформлення договору купівлі-продажу, який відповідає за введення даних про покупця та агента, а також за обчислення загальної вартості угоди;
- модуль формування звітів, призначений для підрахунку загальної суми продажів і відображення дати реалізації об'єктів.

Поділ функціональності на окремі модулі забезпечує впорядковану структуру програмного забезпечення, полегшує його супроводження та створює можливості для подальшого розширення.

На рисунку 2 представлена блок-схема алгоритму створення договору про продаж об'єкту нерухомості.

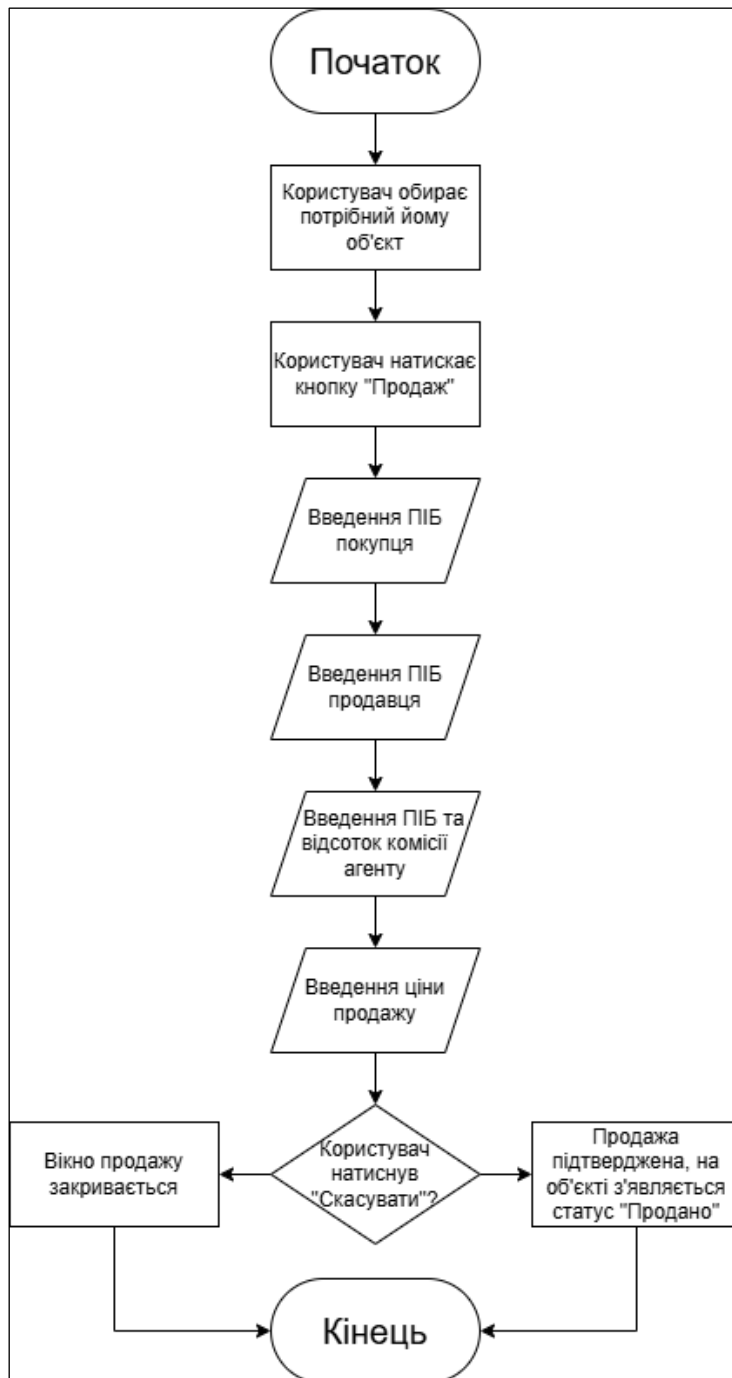


Рис. 2. Блок-схема алгоритму створення договору про продаж об'єкту нерухомості

Архітектуру програмного продукту реалізовано із застосуванням технології Windows Forms [5].

Діаграма класів відображає статичну структуру програмної системи, включаючи основні класи та зв'язки між ними. У розробленому додатку можна виокремити такі ключові класи (рис. 3): Property, Agent, Buyer, Seller та Transaction.

Property (Нерухомість) – абстрактний клас, що містить основні властивості об'єкта нерухомості, зокрема адресу, площу, ціну, статус та тип. Від нього наслідуються класи Apartment (Квартира) та House (Будинок), які розширюють базовий функціонал додатковими характеристиками. Також у системі представлені класи Buyer (Покупець), Seller (Продавець) та Agent (Агент), що описують учасників угод, а клас Transaction (Угода) використовується для зберігання інформації про процес купівлі-продажу нерухомості.

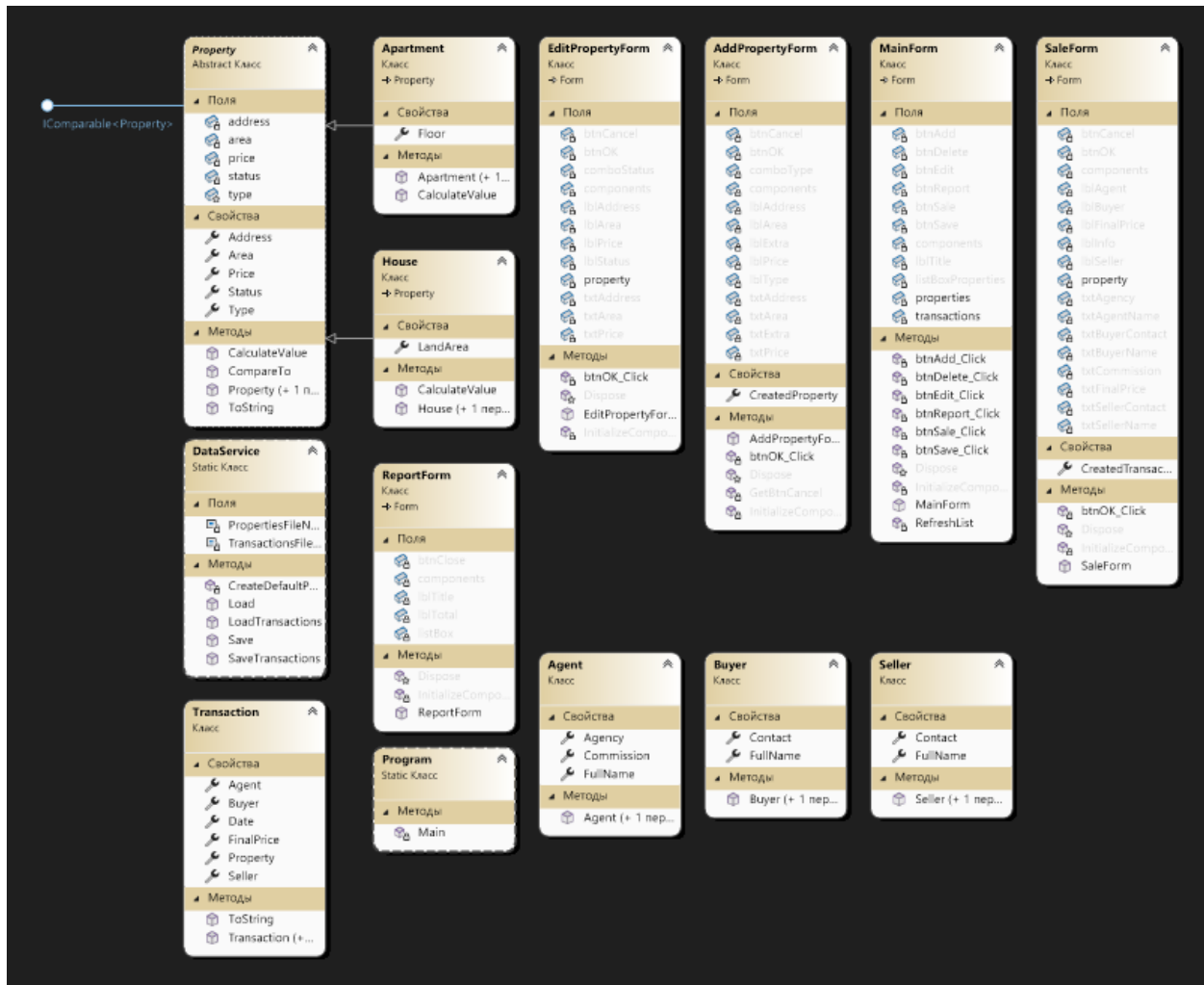


Рис. 3. Діаграма класів програмного додатку

Програмний продукт розроблено з використанням середовища Visual Studio. В основу додатку покладено клас MainForm, який є головною формою програми та забезпечує роботу з даними про об'єкти нерухомості й операції купівлі-продажу.

Клас MainForm відповідає за відображення списку об'єктів нерухомості, керування ними та виконання основних операцій. У межах класу використовуються колекції properties та transactions, які зберігають відповідно дані про нерухомість і здійснені угоди. Завантаження та збереження даних виконується за допомогою сервісу DataService.

Основні методи класу MainForm:

- RefreshList() — оновлює відображення списку об'єктів нерухомості у компоненті listBoxProperties.
- btnAdd\_Click — обробляє подію додавання нового об'єкта нерухомості шляхом відкриття форми AddPropertyForm.
- btnEdit\_Click — забезпечує редагування вибраного об'єкта нерухомості за допомогою форми EditPropertyForm.
- btnDelete\_Click — виконує видалення вибраного об'єкта нерухомості після підтвердження користувача.
- btnSale\_Click — відкриває форму SaleForm для оформлення угоди продажу обраного об'єкта нерухомості та змінює його статус на «Продано».
- btnSave\_Click — зберігає дані про нерухомість і транзакції у файл за допомогою методів DataService.
- btnReport\_Click — відкриває форму ReportForm для перегляду звіту по здійснених угодах.

Клас Property — базовий клас, який описує загальні характеристики об'єктів нерухомості. Він містить основні властивості, такі як адреса, тип об'єкта, вартість та статус, і слугує основою для подальшої роботи з нерухомістю в системі.

Клас Transaction — відповідає за представлення угод купівлі-продажу нерухомості. У ньому зберігається інформація про проданий об'єкт, сторони угоди, дату здійснення та інші важливі параметри, що дозволяє вести облік операцій.

Клас AddPropertyForm — реалізує логіку додавання нового об'єкта нерухомості до системи, надаючи користувачеві зручний інтерфейс для введення необхідних даних.

Клас EditPropertyForm — призначений для редагування інформації про вже існуючий об'єкт нерухомості, що дозволяє актуалізувати дані без створення нового запису.

Клас SaleForm — відповідає за оформлення угоди продажу вибраного об'єкта нерухомості та створення відповідної транзакції.

Клас ReportForm — забезпечує формування та відображення звітів на основі здійснених транзакцій, що дозволяє аналізувати результати діяльності.

Клас DataService — виконує функції збереження та завантаження даних про об'єкти нерухомості та транзакції, забезпечуючи їхню цілісність і постійність.

Такий підхід до побудови програмного додатку дозволяє чітко розмежувати відповідальність між компонентами, підвищити читабельність і підтримуваність коду, а також спростити подальше розширення функціональних можливостей системи.

Використання принципів об'єктно-орієнтованого програмування дозволило забезпечити гнучкість і масштабованість системи. Застосування механізмів наслідування та поліморфізму дало змогу реалізувати різні типи об'єктів нерухомості.

Для організації зберігання даних використано серіалізацію у формат XML, що забезпечує автономність роботи системи та спрощує її впровадження.

Інтерфейс користувача реалізовано у вигляді настільного додатку з логічною структурою та зручною навігацією. Система забезпечує виконання основних операцій: роботу з об'єктами нерухомості, облік клієнтів, оформлення угод, пошук та фільтрацію інформації, а також збереження і завантаження даних (рис. 4).

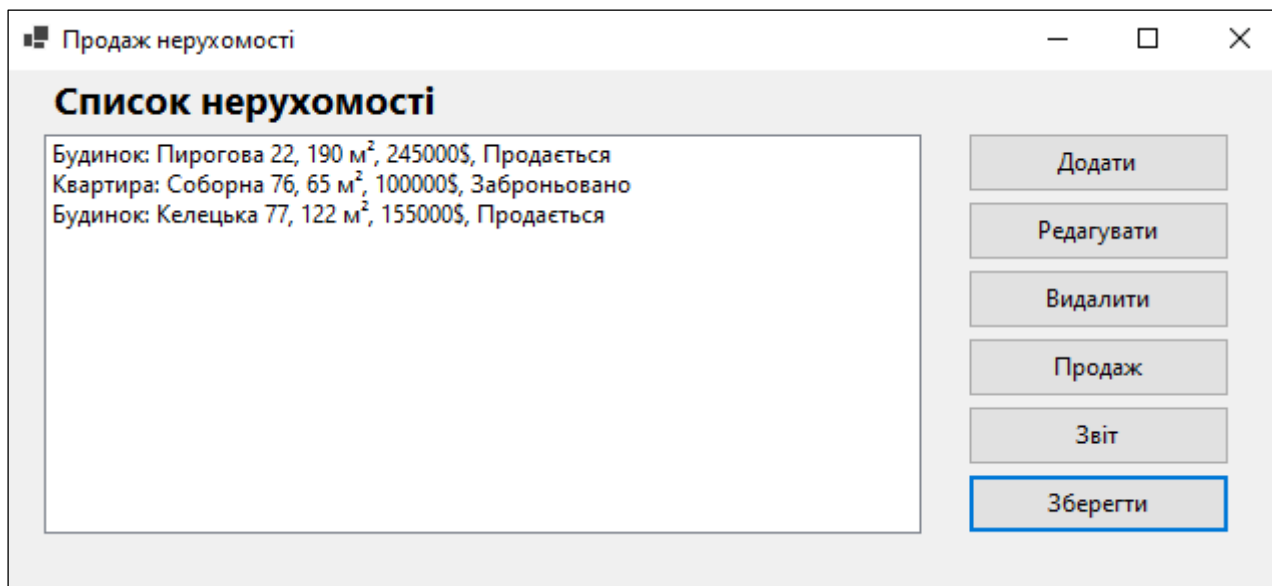


Рис. 4. Головне вікно програмного додатку

## Висновки

У процесі дослідження було проаналізовано особливості організації діяльності у сфері продажу нерухомості та визначено ключові вимоги до інформаційної системи. Встановлено, що впорядковане представлення даних і логічна організація процесів сприяють підвищенню ефективності роботи та зменшенню кількості помилок.

Аналіз існуючих програмних продуктів дозволив виявити їх основні недоліки, зокрема складність використання, обмеженість функціоналу та недостатню гнучкість. Урахування цих аспектів дало змогу розробити ефективну структуру системи.

У результаті створено інформаційну систему, яка забезпечує управління об'єктами нерухомості, облік клієнтів, оформлення угод і пошук інформації. Архітектура системи базується на принципах об'єктно-орієнтованого програмування, що гарантує її модульність і розширюваність.

Розроблений програмний продукт реалізовано мовою C# із використанням Windows Forms. Система дозволяє автоматизувати ключові бізнес-процеси без застосування зовнішніх баз даних, що спрощує її використання в локальних умовах.

Отримані результати можуть бути використані для подальшого розвитку системи, розширення її функціоналу та інтеграції з іншими програмними рішеннями.

## СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Object-Oriented Programming Concepts [Електронний ресурс] – URL: <https://learn.microsoft.com/en-us/dotnet/standard/design-guidelines/>
2. .NET Platform [Електронний ресурс] – URL: <https://dotnet.microsoft.com/en-us/>
3. C# Programming Guide [Електронний ресурс] – URL: <https://learn.microsoft.com/en-us/dotnet/csharp/>
4. UML Diagrams Overview [Електронний ресурс] – URL: <https://www.uml-diagrams.org/>
5. Windows Forms Documentation [Електронний ресурс] – URL: <https://learn.microsoft.com/enus/dotnet/desktop/winforms/>

**Харьков Віктор Олексійович** – студент групи 2ІСТ-24б, факультет інтелектуальних інформаційних технологій та автоматизації, Вінницький національний технічний університет, м. Вінниця, e-mail: [viktor7kharkov4@gmail.com](mailto:viktor7kharkov4@gmail.com)

**Войцеховська Ольга Олександрівна** – PhD, доцент кафедри системного аналізу та інформаційних технологій, Вінницький національний технічний університет, м. Вінниця, e-mail: [olgav1085@gmail.com](mailto:olgav1085@gmail.com)

**Kharkov Viktor O.** – student of group 2IST-24b, Faculty of Intelligent Information Technologies and Automation, Vinnytsia National Technical University, Vinnytsia, e-mail: [viktor7kharkov4@gmail.com](mailto:viktor7kharkov4@gmail.com)

**Voitsekhovska Olha O.** – PhD, Associate Professor of the Department of System Analysis and Information Technologies, Vinnytsia National Technical University, Vinnytsia, e-mail: [olgav1085@gmail.com](mailto:olgav1085@gmail.com)