

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЛЬВІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ІВАНА ФРАНКА
Факультет прикладної математики та інформатики
Кафедра програмування

Проект «Petly»

Застосунок для взаємодії користувачів із притулками тварин

Виконали: студенти групи ПМІ31:
Іванько Наталія
Магоцька Діана
Карп'як Марта
Закутинський Павло
Скоробогатих Дарина

Petly — це застосунок, розроблений для взаємодії користувачів із притулками. Метою проекту є спрощення процесу пошуку, допомоги та адопції тварин. Користувачі можуть зареєструватися в системі, переглядати тварин у притулках, подавати заявки на всиновлення, а також надавати фінансову підтримку.

Основна структура сторінок і навігація:

- Реєстрація / Авторизація: після успішного входу користувач потрапляє на головну сторінку.
- Головна сторінка :
 - Перегляд списку тварин у форматі карток;
 - Кнопка «Більше» — доступ до фільтрів і пошуку;
 - Кнопка «Допомога» — перехід до сторінки підтримки притулку; – Кнопка «Адопція» — перехід до анкети на всиновлення.
- Про нас: інформація про команду, притулок і контакти.

Ролі системи:

1. Користувач

Зареєстрована особа, яка користується застосунком для взаємодії з притулками. Можливості:

- Реєстрація та авторизація в системі
- Перегляд списку тварин
- Використання пошуку та фільтрів
- Подання заявки на адопцію
- Надання допомоги притулку
- Ознайомлення з інформацією «Про нас»

2. Притулок (Адміністратор притулку)

Представник притулку, який керує інформацією у системі. Можливості:

- Додавання, редагування та видалення інформації про тварин
- Перегляд і обробка заявок на адопцію
- Додавання інформації про потреби притулку

3. Адміністратор системи

Користувач із розширеними правами доступу. Виконує контрольні функції. Можливості:

- Підтвердження реєстрації користувачів
- Підтвердження реєстрації адміністраторів притулків
- Контроль коректності даних у системі
- Видалення користувачів за потреби

Виділення підсистем системи:

Застосунок Petly складається з кількох функціональних підсистем:

1. Підсистема реєстрації та авторизації: ідентифікація та контроль доступу.
2. Підсистема управління тваринами: робота з інформацією про тварин (додавання, перегляд, фільтрація).
3. Підсистема адопції: реалізація процесу подання та обробки заявок на всиновлення.
4. Підсистема допомоги притулкам: можливість фінансової або матеріальної підтримки.
5. Інформаційна підсистема: містить сторінку «Про нас» та контактну інформацію.
6. Управління користувачами та притулками: можливість керувати доступом до системи

Функціональні вимоги:

1 Реєстрація та авторизація

1. Як користувач, можу створити новий обліковий запис, якщо ще не зареєстрований.

- *Необхідні умови:*

- Не існує такого облікового запису;
- Необхідні поля для заповнення: Ім'я/Прізвище, Email, Password.

2. Як користувач, я можу зайти у додаток “Petly” в обліковий запис.

- *Необхідні умови:*

- Користувач не авторизований у системі;
- Користувач не заблокований;
- У користувача вже є обліковий запис в базі даних;

2 Основна сторінка

3. Як користувач, я потрапляю на головну сторінку після авторизації.

- *Необхідні умови:*

- Відображається список доступних тварин з короткою інформацією (ім'я, фото, вік, порода).
- Є кнопки взаємодії:
 - * “Більше” → перегляд детальної інформації про тварину;
 - * “Допомога” → можливість допомогти притулку;
 - * “Адопція” → подання заявки на усиновлення.

4. Як користувач, я можу переглядати детальну інформацію про тварину.

- *Необхідні умови:*

- Відображаються: ім'я, вік, стать, порода, фото;
- Стан здоров'я;
- Історія тварини (як потрапила у притулок).

5. Як користувач, я можу подати заявку на адопцію тварини.

- *Необхідні умови:*

- Доступна кнопка “Подати заявку”;
- Форма заявки містить: контактні дані користувача; – Заявка зберігається у базі даних та надсилається притулку.

3 Навігація через тулбар

6. Як користувач, я можу перейти на вкладку “Про нас” з тулбару.

- *Необхідні умови:*

- Відображається інформація про додаток Petly, його місію та команду;

- Доступні контакти для зворотного зв'язку.

7. Як користувач, я можу перейти на вкладку “Профіль” з тулбару.

- *Необхідні умови:*
 - Відображається коротка інформація про користувача: Ім'я, Email;
 - Є можливість редагування даних облікового запису;
 - Перегляд історії заявок на адопцію.

8. Як користувач, я можу перейти на вкладку “Допомога” з тулбару.

- *Необхідні умови:*
 - Відображається список притулків, доступних для донату;

Нефункціональні вимоги:

Вимога	Пріоритет	Критерії прийняття
Локалізація інтерфейсу	Must	Основна мова - українська. Весь контент, сповіщення та елементи керування відображаються коректно.
База даних: SQLite	Must	Використання файлової БД SQLite (.db), що не потребує окремого сервера для запуску проекту.
Архітектура: MVC Pattern	Must	Чіткий поділ на Models (дані), Views (Razor-сторінки) та Controllers (логіка обробки запитів).
Продуктивність доступу до даних	Should	Використання асинхронних запитів (async/await) в контролерах для запобігання блокуванню потоків.
Зручність використання (UX)	Should	Наявність плейсхолдерів, валідація форм на стороні клієнта (jQuery Validation) та сервера (ModelState).
Кросплатформеність	Must	Можливість запуску проекту на Windows, macOS та Linux завдяки .NET Core та локальний БД SQLite.
Цілісність даних	Must	Коректне налаштування каскадного видалення в SQLite (наприклад, при видаленні користувача видаляються всі його плани).

Таблиця Use-Case

Підсистема		
Реєстрація та авторизація		
Роль	Use-case	Опис
Користувач	Реєстрація	Введення реєстраційних даних
	Вхід у систему	Введення логіна та пароля для авторизації
Притулок	Реєстрація адміністратора	Заповнення реєстраційної форми представником притулку
Адміністратор	Підтвердження акаунта	Перегляд та підтвердження (активація) заявок на реєстрацію від адміністраторів притулків/користувачів
Управління тваринами		
Користувач	Перегляд тварин	Відкриття каталогу доступних тварин
	Пошук тварин	Застосування пошуку та фільтрів
	Перегляд профілю тварини	Відкриття картки тварини
Притулок	Додавання тварини	Внесення інформації про тварину
	Редагування тварини	Оновлення даних про тварину
	Видалення тварини	Підтвердження видалення запису
Адміністратор	Контроль даних	Перевірка та модерація записів
Адопція		
Користувач	Подання заявки	Заповнення та надсилання заявки
	Перегляд статусу	Перегляд статусу заявки в кабінеті
Притулок	Перегляд заявок	Відкриття списку отриманих заяв
	Прийняття рішення	Схвалення або відхилення заявки
Допомога притулкам		
Користувач	Перегляд потреб	Перегляд списку потреб притулків
	Здійснення донату	Проведення оплати за реквізитами
Притулок	Додавання потреб	Створення запису про потреби притулку
Інформаційна		
Користувач	Перегляд «Про нас»	Відкриття інформаційної сторінки
Притулок		
Адміністратор	Оновлення інформації	Редагування контенту
Управління користувачами та притулками		
Адміністратор	Перегляд списку користувачів	Перегляд таблиці користувачів із ролями та датами реєстрації
	Перегляд списку адміністраторів притулків	Перегляд списку притулків, їх локацій, імен та email адміністраторів
	Видалення акаунта	Видалення облікового запису для порушників або неактивних користувачів/притулків

Use-Case діаграми із використанням UML



Рис. 1: Usecase - діаграма: Підсистема реєстрації та авторизації

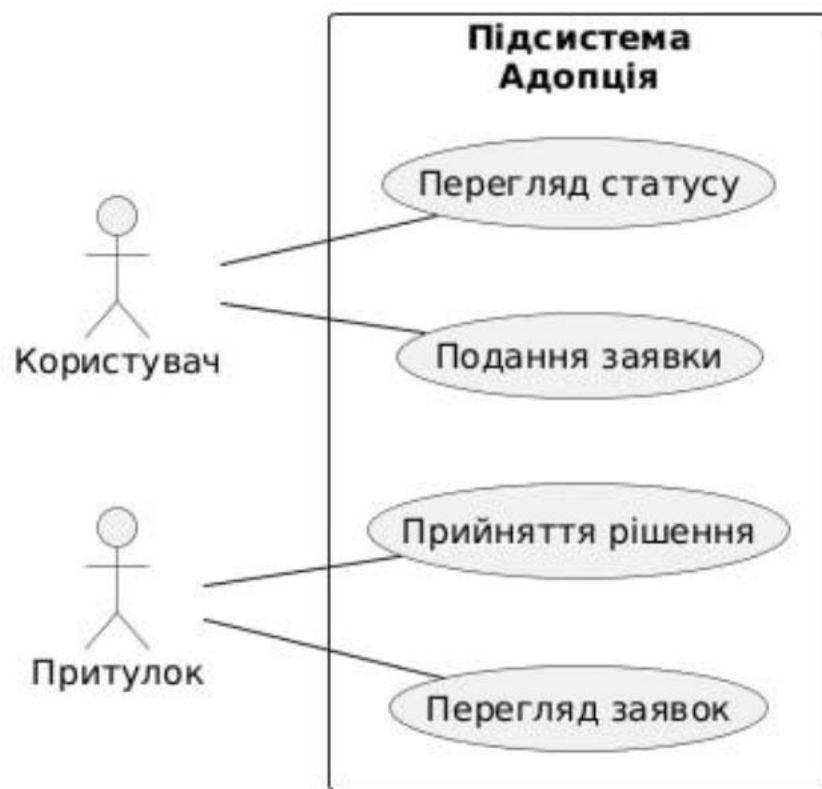


Рис. 2: Usecase - діаграма: Підсистема адопції

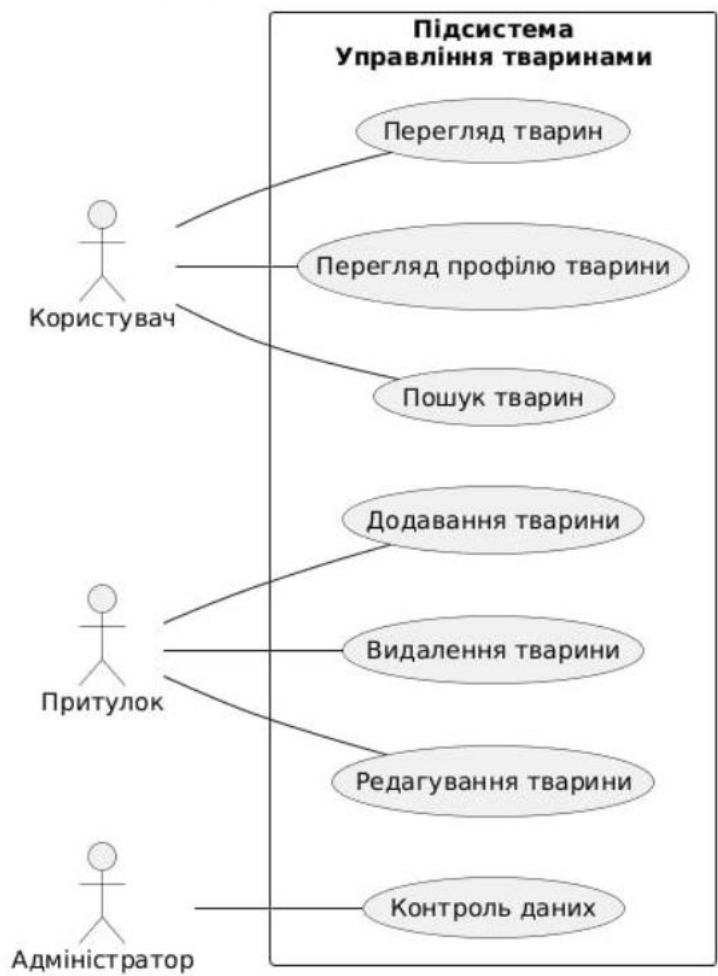


Рис. 3: Usecase - діаграма: Підсистема управління тваринами

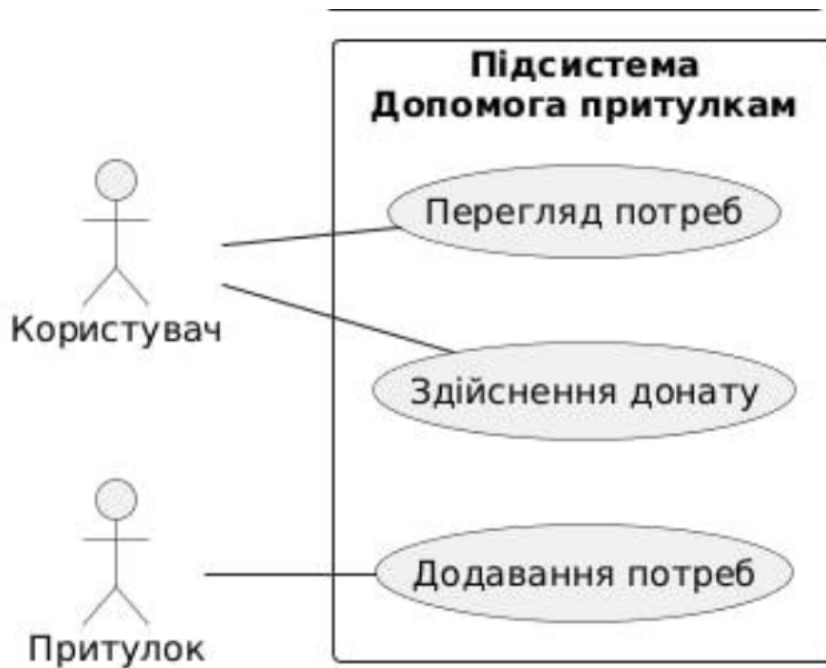


Рис. 4: Usecase - діаграма: Підсистема допомоги притулкам

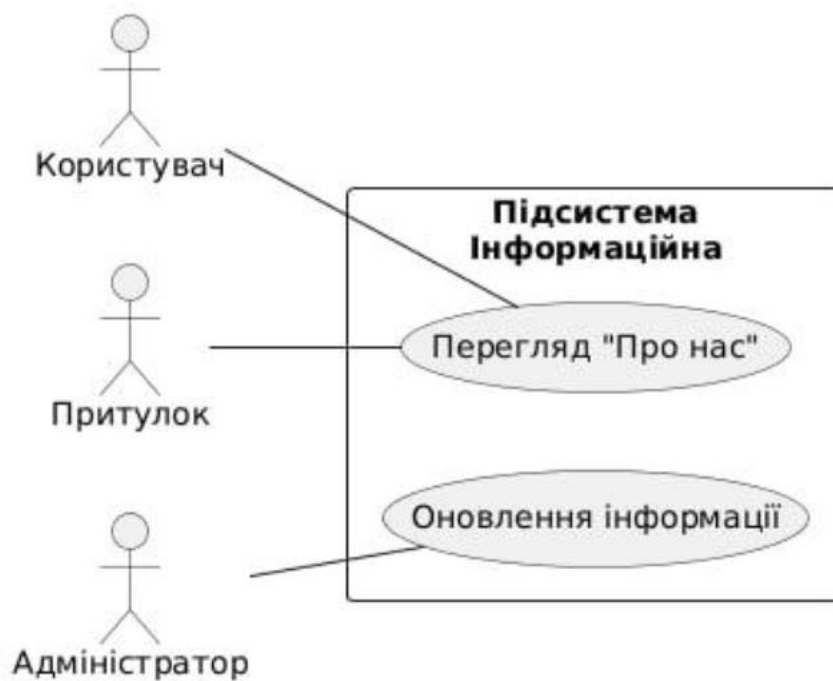


Рис. 5: Usecase - діаграма: Інформаційна підсистема



Рис. 6: Usecase - діаграма: Підсистема Управління користувачами та притулками

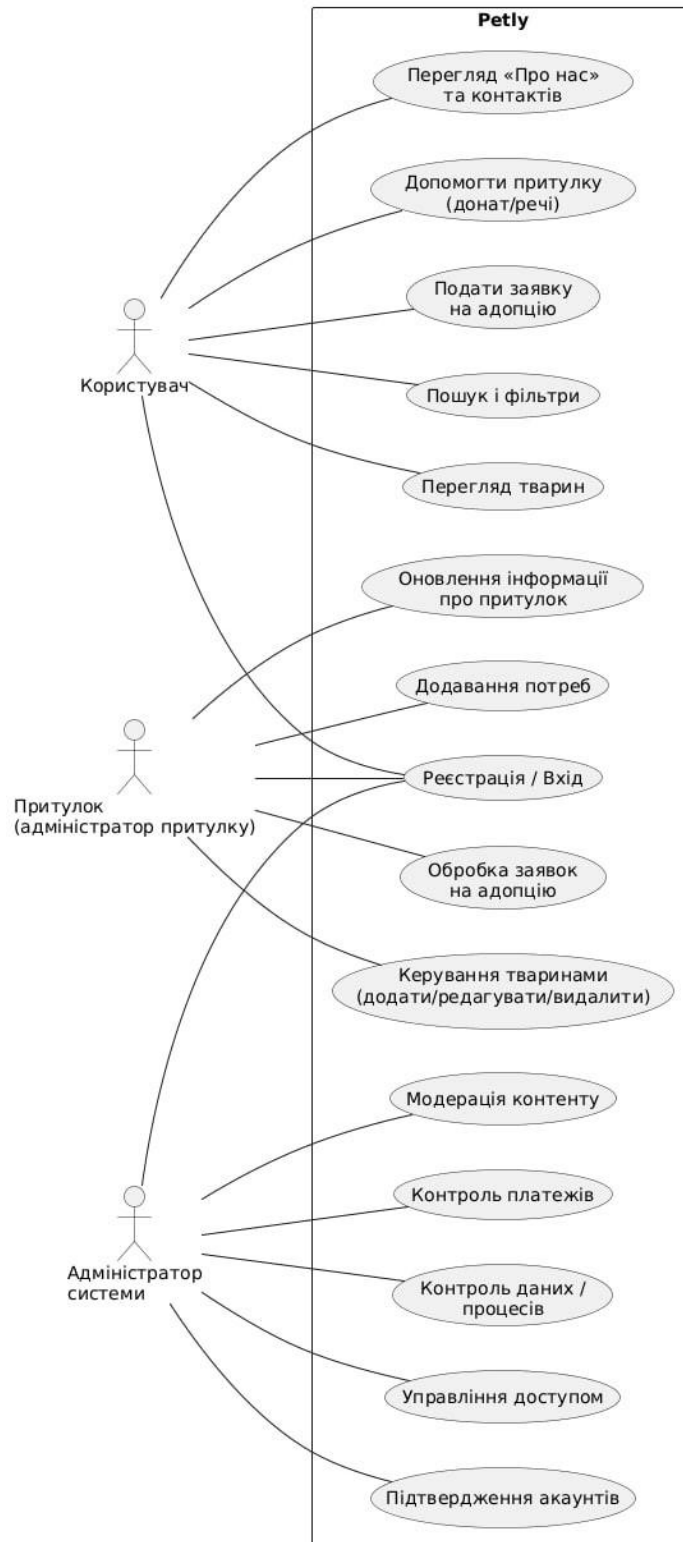


Рис. 7: Загальна Use-case діаграма системи Petly

Негативні сценарії UseCase:

Назва помилки	Актор	Опис	Результат
Помилка реєстрації	Користувач	Система відхиляє створення акаунта, якщо email вже зареєстрований або дані введені некоректно.	Відображається повідомлення про помилку, реєстрація не завершується.
Помилка авторизації	Користувач / Притулок / Адміністратор	Вхід у систему не виконується через неправильний логін або пароль.	Система повідомляє про помилку, доступ не надається.
Відмова у поданні заявки на адопцію	Користувач	Заявка не приймається, якщо форма заповнена неповністю або вже існує активна заявка на цю тварину.	Система показує повідомлення про неможливість подання заявки.
Відхилений платіж	Користувач	Донат не здійснюється через помилку платіжної системи або некоректну суму.	Платіж скасовується, користувач отримує повідомлення.
Відмова у доступі	Притулок / Користувач	Користувач намагається виконати дію, яка не відповідає його ролі.	Система блокує дію та повідомляє про відсутність прав доступу.

Userstories

Роль	User Story
Користувач	Створення облікового запису, щоб мати можливість подавати заявки на адопцію та зберігати історію своєї активності.
	Вхід у свій профіль, щоб отримати доступ до своїх даних та функцій системи.
	Перегляд список тварин, щоб побачити всіх доступних для адопції підопічних.
	Фільтрування тварин за видом, щоб швидко знайти того, хто мені підходить.
	Подання заявки на адопцію, щоб розпочати процес усиновлення тварини.
	Бачити статус заявки на адопцію в особистому кабінеті, щоб розуміти, на якому етапі знаходиться процес усиновлення.
	Подання заявки на волонтерство, щоб запропонувати свою допомогу притулку.
Адміністратор притулку	Бачити статус заявки на волонтерство в особистому кабінеті, щоб знати, чи потрібна моя допомога притулку.
	Реєстрація організації в системі, щоб отримати можливість додавати тварин та шукати для них господарів.
	Авторизація у системі, щоб отримати доступ до функцій системи.
	Додавання нових тварин у систему, щоб користувачі могли бачити їх та подавати заявки на адопцію.
	Я хочу мати можливість редагувати або видаляти інформацію про тварину, якщо дані змінилися або були введені помилково.
	Бачити всі заявки на адопцію, щоб обрати найкращу родину для тварини.
	Змінювання статусу заявки на адопцію, на «Прийнято», щоб повідомити користувача про схвалення його заявки на адопцію.
	Я хочу змінити статус заявки на адопцію, на «Відхилено», щоб повідомити користувача що його заявку на адопцію відхилили.
	Змінювання статусу тварини на «Знайшла дім», щоб вона більше не відображалася у загальному списку пошуку.
	Бачити вхідні заявки на волонтерство, щоб знати, хто пропонує допомогу.
Адміністратор системи	Змінювання статусу заявки на волонтерство, на «Прийнято», щоб зафіксувати свій намір співпрацювати з цією людиною.
	Я хочу змінити статус заявки на волонтерство, на «Відхилено», щоб позначити її як неактуальну.
	Бачити список нових реєстрацій, щоб оперативно їх перевіряти.
	Підтверджувати реєстрації нових притулків, щоб переконатися, що це реальні організації.
	Підтвердження реєстрації нових користувачів, щоб забезпечити безпеку в системі.
	Редагування даних про тварин, щоб видаляти неприпустимий контент або виправляти помилки.

Рис. 7: Таблиця UserStories системи Petly

Аналіз Use-Case діаграми

Об'єкт аналізу: Діаграма варіантів використання інформаційної системи для допомоги притулкам і адопції тварин.

1. Розподіл ролей (Актори)

У системі виділено три ключові групи користувачів, що дозволяє чітко розмежувати рівні доступу та відповідальності:

- Користувач.- основна аудиторія. Фокус на пошуку тварини, отриманні інформації та благодійності.
- Адміністратор притулку - операційний рівень. Ця роль відповідає за контент конкретного закладу (картки тварин, потреби, обробка заявок).
- Адміністратор системи - глобальний рівень. Забезпечує технічну працездатність, модерацію та фінансовий контроль усієї платформи.

2. Обґрунтування функціоналу (Use-Cases)

Логіка взаємодії побудована за принципом “шляху користувача”:

- Для Користувача реалізовано повний цикл: від знайомства з проєктом (“Про нас”) та пошуку тварин до цільової дії (“Подати заявку на адопцію”) або фінансової допомоги (“Добомога”).
- Для Притулку виділено критичні бізнес-процеси: менеджмент бази тварин та швидке реагування на потреби (додавання потреб). Це дозволяє притулкам бути автономними в межах системи.
- Для Системного адміністратора передбачено інструменти безпеки: верифікація акаунтів (щоб уникнути шахрайства під виглядом притулків).

3. Висновки щодо коректності

Діаграма відповідає вимогам до проєктування архітектури програмного забезпечення:

- Повнота: охоплено всі етапи роботи — від реєстрації до модерації даних.
- Розділення обов'язків: права доступу логічно розділені (наприклад, звичайний користувач не має доступу до керування або модерації інших користувачів).
- Масштабованість: структура дозволяє легко додавати нові функції (наприклад, чат або систему відгуків) без зміни основної логіки ролей.