

# Petly

Petly - це застосунок, розроблений для взаємодії користувачів із притулками. Метою проєкту є спрощення процесу пошуку, допомоги та адопції тварин. Користувачі можуть зареєструватися в системі, переглядати тварин у притулках, подавати заявки на адопцію, і 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 (через тулбар):** 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
- Оновлення інформації про притулок
- Перегляд і обробка заявок на адопцію
- Додавання інформації про події або потреби притулку

## **3. Адміністратор системи**

Користувач із розширеними правами доступу. Має ті самі функціональні можливості, що і притулок, але додатково виконує контрольні функції.

Можливості:

- Підтвердження реєстрації користувачів
- Підтвердження реєстрації адміністраторів притулків
- Контроль коректності даних у системі
- Управління доступом до системи

## **Виділення підсистем системи:**

Застосунок Petly складається з кількох функціональних підсистем, кожна з яких відповідає за окремий напрям роботи системи.

### **1. Підсистема реєстрації та авторизації**

Забезпечує ідентифікацію користувачів та контроль доступу до функціоналу системи.

Функції:

- Реєстрація користувачів та адміністраторів притулків
- Авторизація в системі
- Підтвердження реєстрації адміністратором системи
- Управління правами доступу відповідно до ролі

## **2. Підсистема управління тваринами**

Забезпечує роботу з інформацією про тварин у притулках.

Функції:

- Додавання, редагування та видалення тварин (для притулків та адмінів)
- Перегляд списку тварин (для користувачів)
- Фільтрація та пошук тварин
- Відображення детальної інформації про тварину

## **3. Підсистема адопції**

Реалізує процес подання та обробки заявок на всиновлення.

Функції:

- Подання заявки на адопцію користувачем
- Перегляд заявок притулком
- Обробка та прийняття рішення щодо заявки (притулки)

## **4. Підсистема допомоги притулкам**

Забезпечує можливість підтримки притулків.

Функції:

- Перегляд актуальних потреб притулку
- Надання фінансової або матеріальної допомоги
- Інформування про способи підтримки

## **5. Інформаційна підсистема**

Містить довідкову інформацію про систему та організацію.

Функції:

- Сторінка «Про нас»
- Контактна інформація
- Загальні відомості про діяльність притулку

## **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;
- 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:
  1. “Більше” → перегляд детальної інформації про тварину;
  2. “Допомога” → можливість допомогти притулку (донат, волонтерство);
  3. “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 або забронювати зустріч у притулку.

Необхідні умови:

- Доступна кнопка “Подати заявку”;
- Форма заявки містить: контактні дані користувача, дату/час зустрічі;
- 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 “Внесок” з тулбару.

Необхідні умови:

- Відображається список притулків, доступних для донату;
- Користувач може вибрати суму для внеску;
- Доступні способи оплати (картка, онлайн-платіж);
- Підтвердження успішного донату.

9. Як користувач, я можу подати заявку на волонтерство у притулку.

Необхідні умови:

- Вкладка “Волонтерство” доступна з тулбару;
- Користувач може обрати притулок і заповнити форму заявки;
- Дані заявки зберігаються у базі та надсилаються притулку.

## 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 проєкту.
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 проєкту 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).