

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ОДЕСЬКИЙ ДЕРЖАВНИЙ ЕКОЛОГІЧНИЙ УНІВЕРСИТЕТ

Факультет комп'ютерних наук _____

Кафедра інформаційних технологій

ДИПЛОМНА РОБОТА

Рівень вищої освіти бакалавр

на тему: Розробка веб-орієнтованої системи для центра допомоги
безпритульним тваринам

Виконав студент 4 курсу групи К-42

Напряму підготовки 6.050101

комп'ютерні науки

Сайгутіна Ірина Вікторівна

Керівник к.т.н., доц., _____

Терещенко Тетяна Михайлівна

Консультант _____

Рецензент к.т.н., доц., _____

Гнатовська Ганна Арнольдівна

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ОДЕСЬКИЙ ДЕРЖАВНИЙ ЕКОЛОГІЧНИЙ УНІВЕРСИТЕТ

Факультет Комп'ютерних наук

Кафедра Інформаційних технологій

Рівень вищої освіти бакалавр

напрямок 6.050101 комп'ютерні науки

ЗАТВЕРДЖУЮ

Завідувач кафедри _____

“ _____ ” _____ 2017 р.

З А В Д А Н Н Я
НА ДИПЛОМНУ РОБОТУ СТУДЕНТУ

Сайгутіной Ірині Вікторівні

(прізвище, ім'я, по батькові)

1. Тема роботи Розробка веб-орієнтованої системи для центра допомоги
безпритульним тваринам

керівник проекту к.т.н., доц., Терещенко Тетяна Михайлівна

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу від “ _____ ” _____ 2017 р.

№ _____

2. Строк подання студентом проекту _____

3. Вихідні дані до проекту Ресурси мережі Internet, СУБД MySQL, мова
програмування PHP, мова розмітки гіпертекстових документів HTML.

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

Аналіз предметної області

Моделювання предметної області

Вибір програмних засобів для реалізації системи

Програмна реалізація системи

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

Діаграма прецедентів

Діаграма класів

Контекстні діаграми
 Структурна схема сайту
 Структура головної сторінки

6. Консультанти розділів проекту

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

7. Дата видачі завдання “_____” _____ 201 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту	Термін виконання етапів проекту	Оцінка виконання етапу	
			у %	за 4-х бальною шкалою
1	Аналіз предметної області			
2	Моделювання предметної області			
3	Вибір програмних засобів для реалізації системи			
4	Рубіжна атестація	10.05.17		
5	Програмна реалізація інформаційної системи			
6	Оформлення пояснювальної записки			
7	Підготовка до захисту			
Інтегральна оцінка виконання етапів календарного плану (як середня по етапам)				

Студент _____

(підпис)

Сайгутіна І. В. _____

(прізвище та ініціали)

Керівник проекту _____

(підпис)

Терещенко Т. М. _____

(прізвище та ініціали)

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ І ТЕРМІНІВ	5
ВСТУП	6
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ	7
1.1 Необхідність створення інформаційної системи	7
1.2 Аналіз існуючих аналогів.....	7
1.2.1 Опис сайту «GladPet»	7
1.2.2 Опис сайту «Союз захисту тварин»	8
1.2.3 Опис сайту «Сіріус».....	10
1.3 Постановка завдання.....	11
2 МОДЕЛЮВАННЯ ПРЕДМЕТНОЇ ОБЛАСТІ	13
2.1 Діаграма прецедентів.....	13
2.2 Діаграма класів.....	14
2.3 Контекстна діаграма	16
2.4 Діаграма декомпозиції.....	18
3 ВИБІР ПРОГРАМНИХ ЗАСОБІВ ДЛЯ РЕАЛІЗАЦІЇ СИСТЕМИ.....	21
3.1 Вибір системи управління контентом CMS	21
3.1.1 Опис CMS Joomla.....	21
3.1.2 Опис CMS DataLife Engine.....	23
3.1.3 Опис CMS WordPress.....	25
3.2 Особливості використання PHP	28
3.3 Система управління базами даних MySQL	29
4 ПРОГРАМНА РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ.....	33
4.1 Етапи розробки сайту	33
4.2 Основні принципи розробки дизайну сайту.....	33
4.3 Розробка структури сайту	35
4.4 Розробка веб-інтерфейсу	37
ВИСНОВКИ.....	45
ПЕРЕЛІК ПОСИЛАНЬ	46
Д О Д А Т К И.....	47
ДОДАТОК А КОД ГОЛОВНОЇ СТОРІНКИ САЙТУ	48
ДОДАТОК Б ФОРМА ДОДАВАННЯ ТВАРИНИ	65

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ І ТЕРМІНІВ

БД – база даних – це сукупність даних, організованих відповідно до концепції, яка описує характеристики цих даних і взаємозв'язки між їх елементами.

СУБД – система управління базою даних – це комплекс програмного забезпечення, що надає можливості створення, збереження, оновлення та пошуку інформації в базах даних з контролем доступу до даних.

CMS – Content Management Systems – це програмне забезпечення для організації веб-сайтів чи інших інформаційних ресурсів в Інтернеті чи окремих комп'ютерних мережах.

DLE – Datalife Engine – комерційна система управління контентом.

PHP – Hypertext Preprocessor – скрипкова мова програмування, була створена для генерації HTML-сторінок на стороні веб-сервера.

UML – Unified Modeling Language – уніфікована мова моделювання.

WP – WordPress – некомерційна система управління контентом.

WWW – World Wide Web – найбільше всесвітнє багатомовне сховище інформації в електронному вигляді: десятки мільйонів пов'язаних між собою документів, що розташовані на комп'ютерах, розміщених на всій земній кулі.

ВСТУП

На сьогоднішній день практично кожна організація має власний веб-сайт. В умовах використання сучасних інформаційних технологій – це необхідний чинник існування, що дозволяє розширити увагу до певної діяльності.

Актуальність теми дослідження також обумовлюється тим, що проблема існування безпритульних тварин є однією з актуальних проблем міста та має несприятливі екологічні та соціальні наслідки. А також те, що безпритульність викликає великі страждання і загибель тварин, а також санітарний ризик людству. Тварини розносять ектопаразитів і ендопаразитів, збудниками кишкових захворювань і сказу. Самі вони нерідко стають жертвами жорстокості з боку людей і навіть садизму. Очевидно, що виходячи з санітарно-епідеміологічних та соціальних точок зору, не можна допускати, щоб тварини без нагляду жили та помирали на вулицях, а так само самі були причиною смертей людей.

Метою дипломного проекту є розробка веб-орієнтованої системи "GetPet", яка дозволить користувачеві додавати тварину в базу даних, здійснювати доступ до бази даних з тваринами, переглядати текстову і графічну інформацію, представлену на сайті.

В ході розробки дипломного проекту вирішуються такі завдання:

- 1) дослідження та аналіз існуючих аналогів;
- 2) визначення основних характеристик і структури інформаційної системи;
- 3) розробка дизайну сайту.

За допомогою розробленої системи можна:

- прискорити процес прилаштування тварин з притулку в сім'ї;
- зміна мислення населення міста з «купити тварину» на «взяти з притулку»;
- розмістити інформацію про тварину;
- вибрати вподобану тварину;

Пояснювальна записка до дипломної роботи містить 68 сторінок, 19 рисунків, 12 посилань.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Необхідність створення інформаційної системи

За оцінками експертів, Україна входить в десятку країн-лідерів за кількістю безпритульних тварин.

Основна причина стрімкого зростання бездомних тварин на вулиці – це неправильне поводження людей з домашніми тваринами та відсутність стерилізації. Останнім часом в Україні з'явилося безліч волонтерів, які займаються даною проблемою.

Основна ідея в тому, щоб вирішити головну проблему притулків: те, що вони переповнені. Для цього потрібно допомогти їм швидше пристроювати тварин. Для цього, в свою чергу, потрібно зробити так, щоб жителі хотіли, знали і могли в одному місці підшукати вихованця, якого вони заберуть до себе в сім'ю. Паралельно з цим можна зробити ще ряд речей, які дозволять системно вирішити проблему бездомних тварин.

Необхідність створення інформаційної системи полягає в тому, що це дозволить прискорити процес прилаштування тварин з притулків в сім'ї, а також змінити мислення населення міста з "купити тварину" на "взяти з притулку".

Крім того, така система дозволяє:

- 1) розмістити інформацію про тварину;
- 2) вибрати вподобану тварину віддалено.

Також є можливість знайти ветеринарні клініки, аптеки для тварин і надати необхідну допомогу як для притулку, так і для самих вихованців.

Для створення інформаційної системи необхідно:

- 1) дослідити аналогічні веб-сервіси;
- 2) виявити проблемні моменти;
- 3) обґрунтувати необхідність автоматизації;
- 4) обґрунтувати проектні рішення з програмного, технічного та інформаційного забезпечення.

1.2 Аналіз існуючих аналогів

1.2.1 Опис сайту «GladPet»

Сайт має досить яскравий і привабливий дизайн, який дає можливість користувачеві сконцентруватися на основному моменті – виборі вихованця.

Головна сторінка сайту дуже інформативна, на ній представлені блоки, що закликають людей взяти тварину з притулку. На сайті представлені посилання на соціальні мережі, через які користувач може поділитися інформацією, що його цікавить.

Головна сторінка сайту представлено на рис. 1.1 [1]

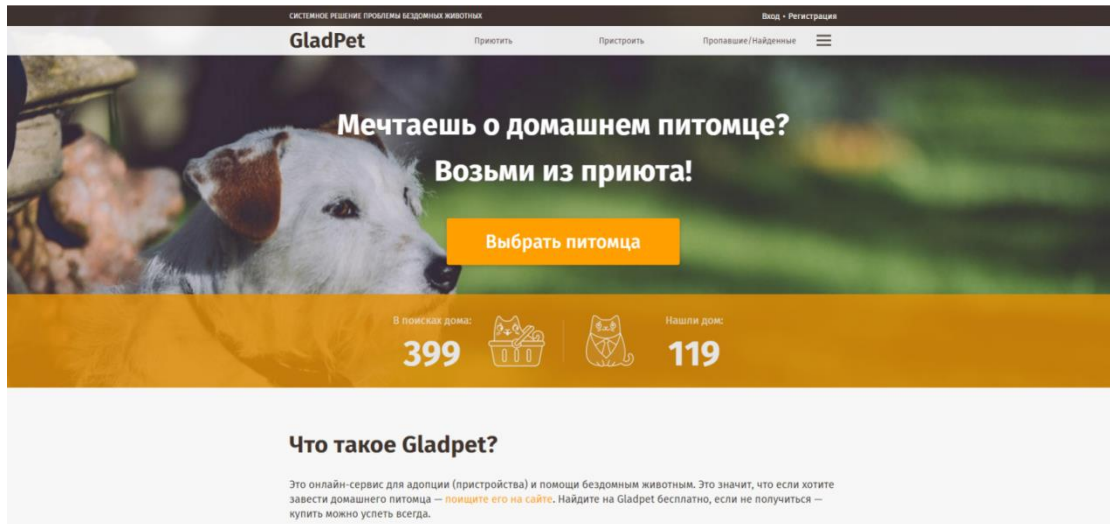


Рисунок 1.1 – Головна сторінка сайту організації «GladPet»

Сайт містить такі розділи:

1) Притулити – на даній сторінці користувач має можливість вибрати вихованця. При цьому є безліч фільтрів, таких як: вид, порода, стать, окрас, вік, організація, які допомагають при виборі тварини.

2) Прилаштувати – дана сторінка доступна тільки для авторизованих користувачів. Зареєструватися на сайті можна за допомогою соціальних мереж або ж локально. Після реєстрації користувачеві буде доступна сторінка для додавання вихованця в міжнародну базу реєстрації тварин.

3) Зниклі / Знайдені – на цій сторінці розміщена інформація про зниклих і знайдених тварин. Їх також можна вибрати за допомогою певних фільтрів: вид, порода, стать, місто, район.

1.2.2 Опис сайту «Союз захисту тварин»

Структура і дизайн сайту не дуже вдалі – через «сірості» сайту користувачеві важко орієнтуватися в розділах сайту, основні заголовки і блоки ніяк не виділені, що ускладнює роботу з пошуком інформації на даному сайті.

Перевагою даного веб-порталу є вибір мови, на якому буде відображатися сайт – російська та англійська.

На рис. 1.2 показаний зовнішній вигляд громадської організації «Союз захисту тварин» [2].

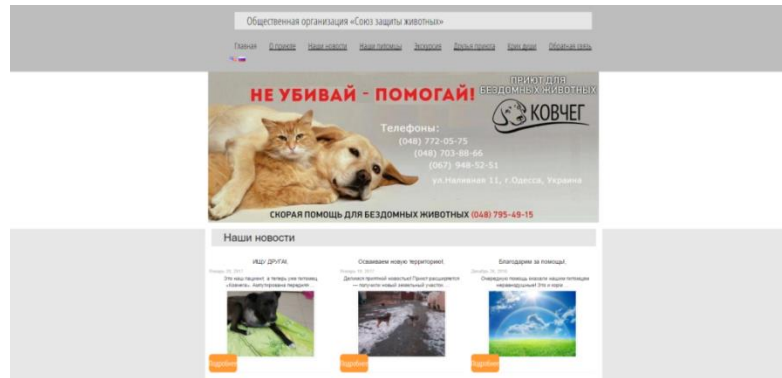


Рисунок 1.2 – Сайт організації «Союз захисту тварин»

Основний текст, представлений у всіх розділах сайту, є недоречно великий розмір шрифту. Звичайний текст важко відрізнити від заголовків.

Внутрішня сторінка сайту з вибором конкретного вихованця не має достатньої інформації про тварину. Відсутня інформація про вік, стать, породу обраної тварини. Бічне меню так само не містить належної інформації - представлений список архівів записів по місяцях і блок авторизації користувача, що є стандартними віджетами CMS WordPress.

На внутрішній сторінці (рис. 1.3) є віджет «Розповісти про цю новину», за допомогою якого користувач може розповісти про даній сторінці в соціальних мережах.

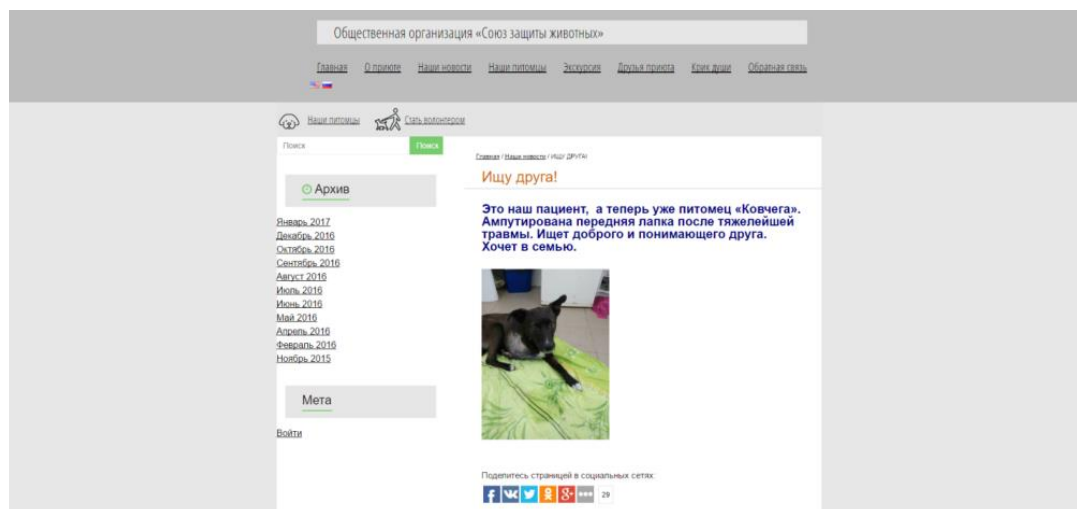


Рисунок 1.3 – Внутрішня сторінка сайту організації «Союз захисту тварин»

Сайт має наступну структуру:

- 1) Головна – на головній сторінці відображається контактна інформація з притому, а також виводяться останні записи сайту – новини та оголошення про тварин.
- 2) Про притулок – на даній сторінці описана історія створення організації, вказана контактна інформація з картою, на якій показаний цю адресу.
- 3) Наші новини – висновок останніх новин сайту.
- 4) Наші тварини – сторінка, на якій розміщені оголошення про тварин, яких можна забрати з притулку.
- 5) Екскурсія – на даній сторінці не міститься жодного матеріалу.
- 6) Друзі притулку – представлені посилання на сайти партнерів.
- 7) Зворотній зв'язок – форма зворотного зв'язку, за допомогою якої користувач може звернутися до адміністрації сайту з проханням або питанням.

1.2.3 Опис сайту «Сіріус»

Сайт має непривабливий і сірий дизайн, що не закликає користувача до дії взяти тварину з притулку.

Структура даного порталу не дозволяє відвідувачеві швидко орієнтуватися в представленій інформації. Блоки сайту мають важко дрібний шрифт, що ускладнює процес вибору тварини.

Оцінити структуру сайту притулку для бездомних тварин можна за допомогою рис. 1.4 [3]

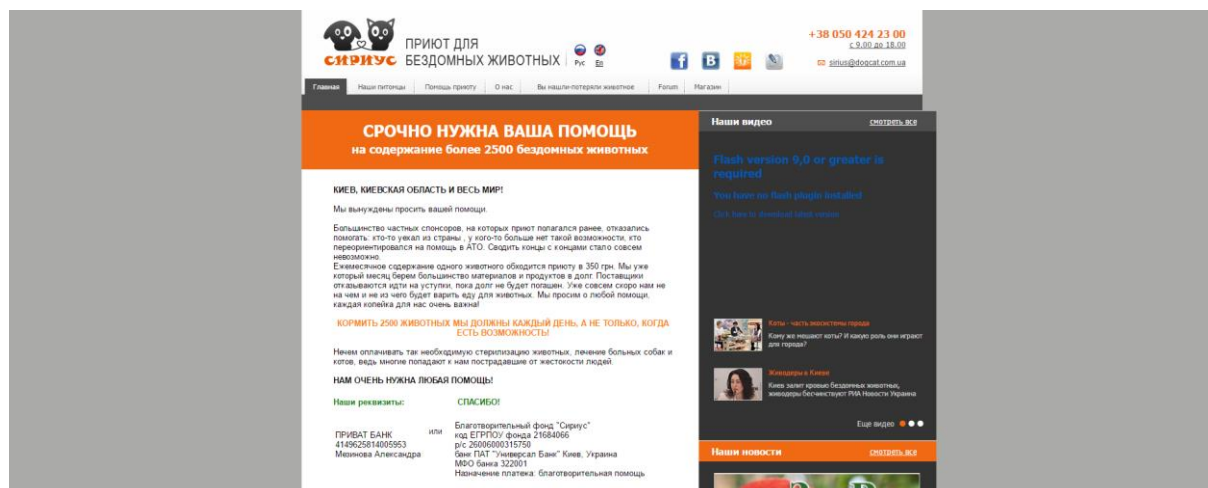


Рисунок 1.4 – Головна сторінка сайту для бездомних тварин «Сіріус»

Структура сайту:

Головна – на головній сторінці представлена інформація про реквізити для приватних спонсорів, блоки з висновком останніх новин і відео, блок вибору тварин, розбитий на дві підкатегорії – кішки і собаки.

Наші тварини – розділ сайту, який розбитий на два підрозділи: кішки і собаки. При натисканні на певний пункт меню, користувач потрапляє на відповідну сторінку сайту, де він може вибрати тварину.

Допомога притулку – даний розділ містить в собі такі підрозділи: фінансова допомога, будівництво притулку, догляд за тваринами, медикаменти, продукти. Вибравши категорію допомоги, відвідувач може зробити пожертвування на користь самих тварин або ж організації.

Про нас – в цьому розділі сайту міститься інформація про історію створення притулку, юридичний статус і контактах з організацією. Також в даному розділі є посилання на новини порталу.

Ви знайшли-втратили тварину – тут можна додати оголошення про втрату або перебування тварини.

Форум – на сайті є можливість спілкування користувачів один з одним за допомогою спеціальних розділів на форумі. На ньому можна дізнатися багато корисної інформації і поділитися своєю. Форум не має зворотної посилання на головну сторінку сайту, що є явним мінусом для зручної роботи з порталом.

Магазин – даний розділ сайту потрібен для того, щоб користувач міг не тільки фінансово підтримати проект (придбати товар з символікою даного притулку), а й стати «рекламоносієм» притулку і просто гуманного ставлення до тварин.

1.3 Постановка завдання

Завданням дипломного проекту є створення веб-орієнтованої системи для центру допомоги бездомним тваринам, за допомогою якої можна звернути увагу людей на цю проблему і сприяти прискоренню прийому тварин в сім'ї.

За допомогою даної системи люди зможуть:

- 1) повідомити про знайдений \ зниклого тваринному;

- 2) вибрати тварину з притулку віддалено;
- 3) знайти ветеринарні клініки і аптеки.

Структура такої веб-орієнтованої системи повинна мати такий вигляд:

Головна сторінка. Повинна бути оформлена у вигляді Landing Page для залучення уваги аудиторії і призову її взяти тварину з притулку. На ній будуть розташовуватися блоки із закликом до дії.

Вибрати тварину. На даній сторінці користувач зможе віддалено вибрати вподобану тварину.

Підтримати тварин. Тут користувачеві надається можливість надати допомогу притулку або самим тваринам.

Лікування. Дана сторінка містить карту, на якій представлені ветеринарні клініки і аптеки.

Про нас. На даній сторінці представлена контактна інформація.

Додати тварину. Дана сторінка містить форму, за допомогою якої користувач з певними правами зможе додати тварину на сайт. При додаванні тваринного можна вибрати його категорію: втрачене; знайдене; що знаходиться в притулку.

Для вирішення таких завдань потрібно:

- провести аналіз деяких існуючих сайтів про притулках тварин;
- розглянуть сучасні засоби розробки web-сайтів;
- виявить основні переваги та недоліки систем управління контентом;
- вибрати систему управління базами даних;
- розробити структуру майбутньої системи;
- зробити програмну реалізацію системи.

2 МОДЕЛЮВАННЯ ПРЕДМЕТНОЇ ОБЛАСТІ

2.1 Діаграма прецедентів

Діаграма прецедентів є графом, що складається з множини акторів, прецедентів, обмежених границею системи, асоціацій між акторами та прецедентами, відношень серед прецедентів, та відношень узагальнення між акторами. Діаграми прецедентів відображають елементи моделі варіантів використання.

Суть даної діаграми полягає в наступному: проєктована система представляється у вигляді безлічі сутностей чи акторів, що взаємодіють із системою за допомогою так званих варіантів використання. Варіант використання використовують для описання послуг, які система надає актору. Іншими словами, кожен варіант використання визначає деякий набір дій, який виконує система при діалозі з актором. При цьому нічого не говориться про те, яким чином буде реалізована взаємодія акторів із системою.

У мові UML є кілька стандартних видів відношень між акторами і варіантами використання:

- 1) асоціації;
- 2) включення;
- 3) розширення;
- 4) узагальнення.

При цьому загальні властивості варіантів використання можуть бути представлені трьома різними способами, а саме – за допомогою відношень включення, розширення і узагальнення.

Відношення асоціації – одне з фундаментальних понять у мові UML і в тій чи іншій мірі використовується при побудові всіх графічних моделей систем у формі канонічних діаграм.

Включення у мові UML – це різновид відношення залежності між базовим варіантом використання і його спеціальним випадком. При цьому відношенням залежності є таке відношення між двома елементами моделі, при якому зміна одного елемента приводить до зміни іншого елемента.

Відношення розширення визначає взаємозв'язок базового варіанта використання з іншим варіантом використання, функціональна поведінка якого задіюється базовим не завжди, а тільки при виконанні додаткових умов.

Частина інформації в моделі прецедентів можна усунути зазначенням зв'язків між прецедентами:

- 1) узагальнення прецеденту – стрілка з не закрашеним трикутником (трикутник ставиться у більш загального прецеденту);
- 2) включення прецеденту – пунктирна стрілка зі стереотипом «include»;
- 3) розширення прецеденту – пунктирна стрілка зі стереотипом «extend» (стрілка входить в розширюваний прецедент, в додатковому розділі якого може бути вказана точка розширення і, можливо у вигляді коментаря , умова розширення).

Діаграму прецедентів сайту «GetPet» можна побачити на рис. 2.1

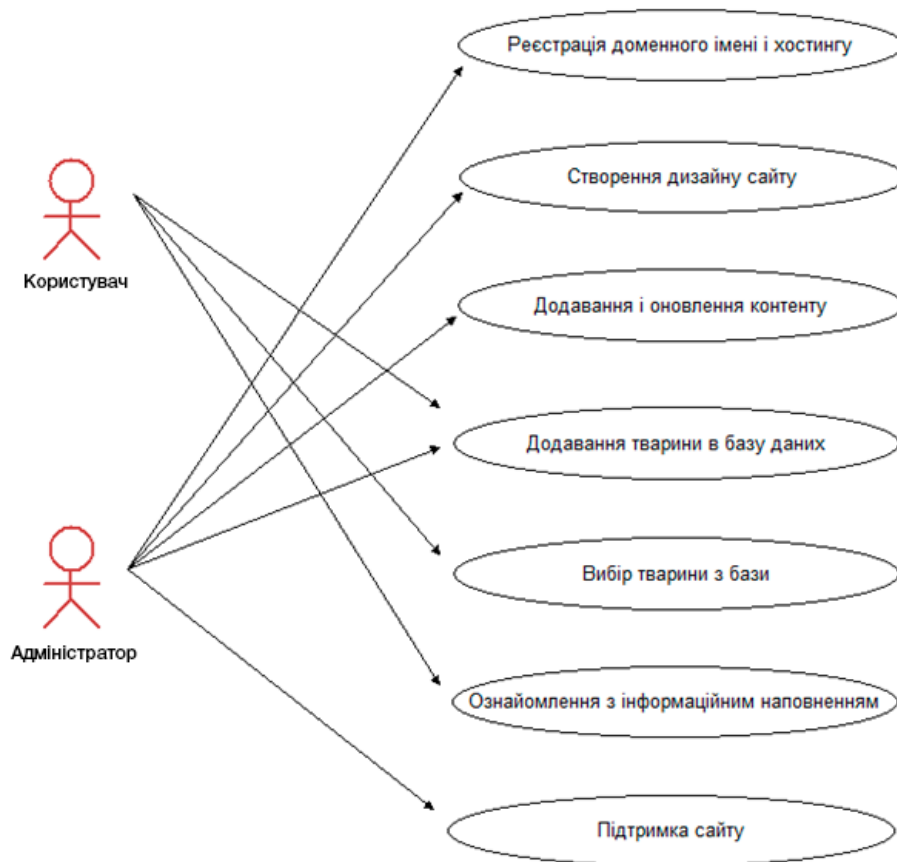


Рисунок 2.1 – Діаграма прецедентів сайту «GetPet»

2.2 Діаграма класів

Моделювання сутностей програмної системи проводять за допомогою схеми класів UML. Схема класів, створювана на початкових етапах проектування програмного продукту, дозволяє попередньо проаналізувати сутності програмної системи та їх зв'язку на досить високому рівні абстракції.

Інакше кажучи, діаграма класу може відображати лише імена класів або імена та відповідні атрибути класів, або імена, атрибути та операції класів. Класи можуть перебувати у певних відношеннях або зв'язках.

Клас представляє множину об'єктів, яку об'єднує спільна структура та поведінка. Клас є абстракцією сутностей реального світу. Реальні екземпляри сутностей (instances) це так звані об'єкти.

Атрибути класу визначають склад і структуру даних, що зберігаються в об'єктах цього класу. Кожен атрибут має ім'я і тип, який визначає, які дані він представляє.

При реалізації об'єкта в програмному коді для атрибутів буде виділена пам'ять, необхідна для зберігання всіх атрибутів, і кожен атрибут матиме конкретне значення в будь-який момент часу роботи програми. Об'єктів одного класу в програмі може бути як завгодно багато, всі вони мають однаковий набір атрибутів, описаний в класі, але значення атрибутів у кожного об'єкта свої і можуть змінюватися в ході виконання програми.

Діаграма класів відображає статичну структуру системи у термінах класів та відношень між цими класами.

Відношення:

Асоціація – взаємна залежність між об'єктами різних класів, кожен з яких є рівноправним членом залежності. Для асоціації може позначатися кількість екземплярів об'єктів кожного класу, які беруть участь у зв'язку (0 - якщо жодного, 1 – якщо один, * – якщо багато).

Можуть вказуватися мінімальна й максимальна кількість, наприклад, 0,1...* означає, що на відповідному кінці асоціації може не бути жодного екземпляра, бути один або багато.

Агрегація або відношення частина-до-цілого. Особливість цього відношення полягає в тому, що час існування об'єкта-частини збігається з часом існування об'єкта-цілого. Стрілка з ромбом на кінці, яка позначає відношення агрегації, спрямована від об'єкта-частини до об'єкта-цілого.

На діаграмі класів показані основні сутності і зв'язку між ними. У нашому випадку, основними сутностями є:

- співробітники
- тварини
- інформаційне наповнення
- користувачі сайту, які вибирають тварину

Діаграма класів представлена на рис. 2.2

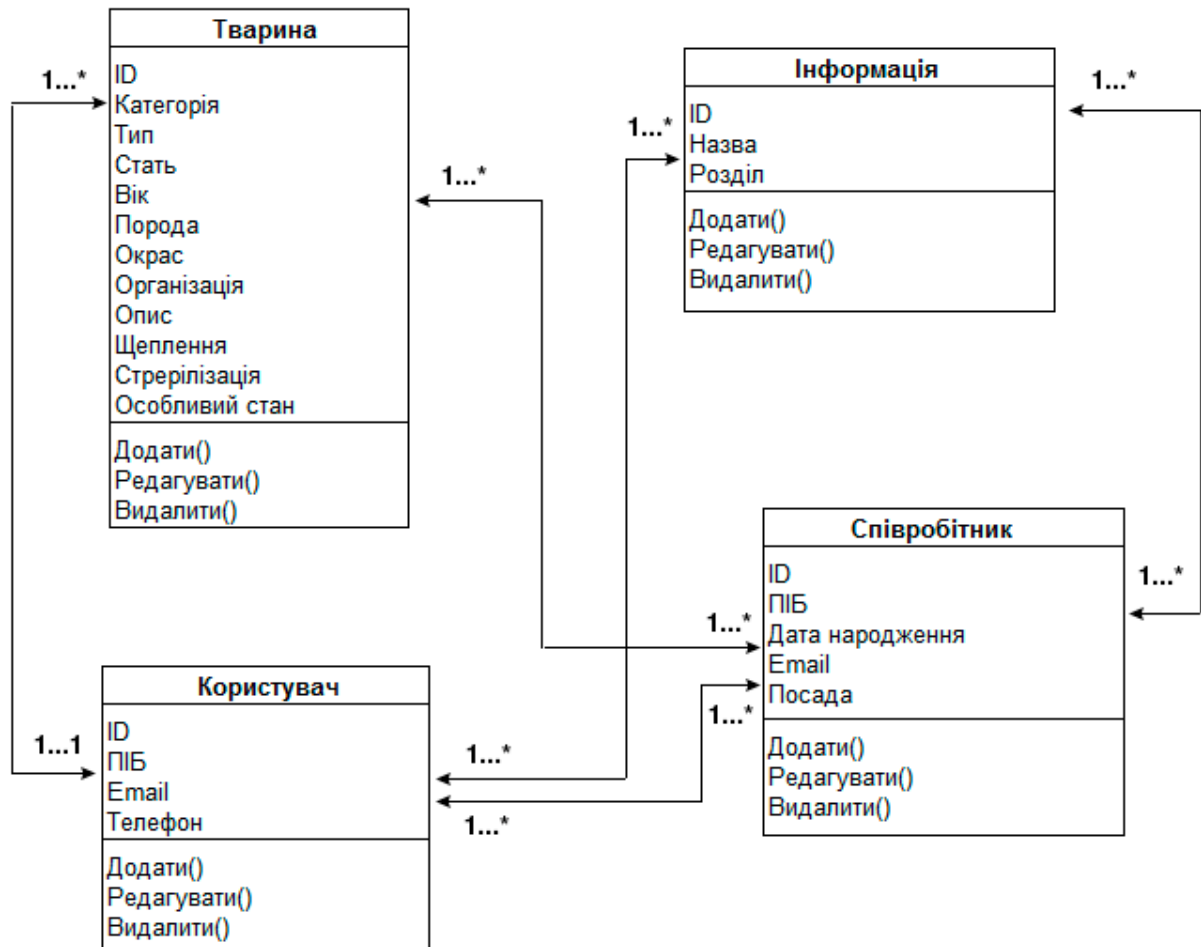


Рисунок 2.2 – Діаграма класів сайту «GetPet»

2.3 Контекстна діаграма

Побудова контекстної діаграми починається з отримання всієї системи у вигляді одного блоку і граничних стрілок, що зображують зв'язку з функціями поза системою.

Діаграма, що складається з одного блоку і стрілок, визначає межі системи і називається контекстною діаграмою моделі. При цьому блок зображує кордону системи: все, що лежить всередині нього, є частиною системи, а всі, що лежить поза ним, утворює зовнішнє середовище системи.

Стрілки на контекстній діаграмі служать для опису взаємодії системи з навколишнім світом. Такі стрілки називаються граничними. Вони можуть починатися біля кордону діаграми і закінчуватися у роботі, або навпаки, починатися у роботі і закінчуватися біля кордону діаграми.

Розрізняють п'ять типів граничних стрілок, що зв'язують систему із зовнішнім світом:

Входи – матеріали або інформація, які використовуються або перетворюються роботою для отримання результату (стрілка, що входить в лівий кордон прямокутника). Допускається, що робота може не мати жодної стрілки входу.

Управління – правила, стратегії, стандарти, якими керується робота (стрілка, що входить в верхню межу прямокутника). На відміну від вхідної інформації управління не підлягає зміні в результаті виконання роботи. Кожна робота повинна мати хоча б одну стрілку управління.

Виходи – матеріали або інформація, які виробляються роботою (стрілка, яка виходить із правої межі прямокутника). Кожна робота повинна мати хоча б один вихід, тому що робота без результату не має сенсу і не повинна моделюватися.

Механізми – ресурси, які дозволяють виконати роботу: персонал, верстати, пристрої (стрілка, що входить в нижню межу прямокутника). На розсуд аналітика стрілки механізму можуть не зображатися в моделі.

Виклик – стрілка, що вказує на іншу модель роботи (стрілка, яка виходить із нижньої межі прямокутника). Стрілка виклику використовується для вказівки того, що деяка робота виконується за межами модельованої системи.

В даному випадку використовується діаграма IDEF0.

Опис системи за допомогою IDEF0 називається функціональною моделлю. Функціональна модель призначена для опису існуючих бізнес-процесів, в якому використовуються як природний, так і графічний мови. Для передачі інформації про конкретну систему джерелом графічної мови є сама методологія IDEF0.

Функціональна модель являє з необхідним ступенем деталізації систему функцій, які відображають свої взаємини через об'єкти системи. Моделі даних є докладним описом об'єктів системи, пов'язаних з системними функціями. Повна методологія SADT підтримує створення безлічі моделей для більш точного опису складної системи.

Кожна IDEF0-діаграма містить блоки і дуги. Блоки зображують функції модельованої системи. Дуги пов'язують блоки разом і відображають взаємодії і взаємозв'язку між ними.

Діаграма для системи, що розробляється, представлена на рис. 2.3

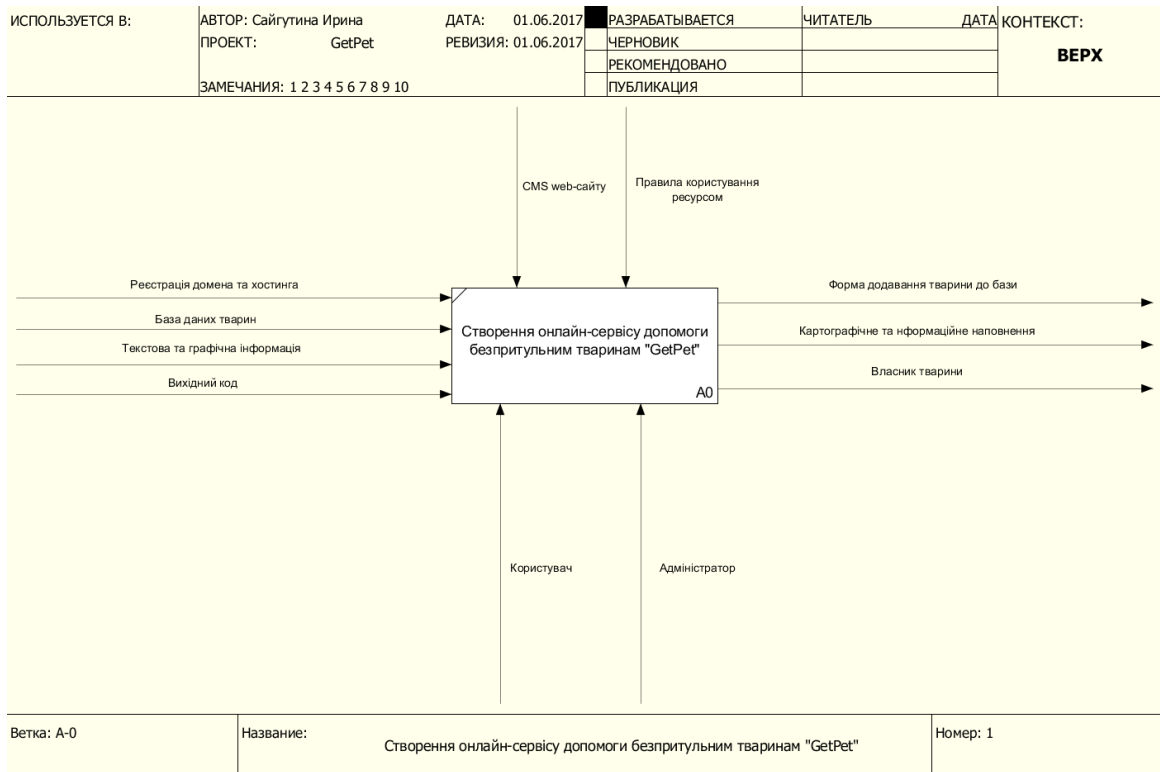


Рисунок 2.3 – Контекстна діаграма сайту «GetPet»

На контекстній діаграмі показаний основний процес у вигляді створення онлайн-сервісу для центру допомоги бездомним тваринам.

На вхід процесу надходять необхідні для створення сайту дані, тобто: інформація, база даних, доменне ім'я і хостінг.

Виходами є результат роботи, тобто активна форма додавання тваринного в базу даних, картографічне та інформаційне наповнення, а також досягнутий результат у вигляді взятого з притулку тварини.

2.4 Діаграма декомпозиції

Для головного процесу, присутнього на контекстній діаграмі, проводиться декомпозиція. На першому рівні ієрархії показуються основні внутрішні процеси системи і відповідні їм зовнішні сутності, сховища і потоки даних.

Для кожного процесу діаграми першого рівня може бути проведена декомпозиція, яка, в свою чергу, також може бути розкрита більш докладно. Декомпозиція процесів закінчується, коли досягнута необхідна ступінь деталізації або відображаються на черговому рівні діаграм процесу елементарними і не можуть бути розбиті на більш дрібні.

При проведенні декомпозиції повинно виконуватися правило балансування – при декомпозиції процесу дочірня діаграма в якості зовнішніх сутностей може мати тільки ті об'єкти (процеси, зовнішні сутності, сховища даних), з якими має інформаційний зв'язок деталізуємий процес на батьківській діаграмі.

Роботи ніколи не розміщуються на діаграмі випадковим чином. Вони розміщуються за ступенем важливості, як її розуміє автор. У методології SADT цей відносний порядок називається домінуванням.

Домінування розуміється як вплив, яке одна робота надає на інші роботи діаграми.

Самою домінуючою роботою діаграми може бути найважливіша з необхідної послідовності робіт, або яка планує або контролююча робота, що впливає на всі інші роботи.

Найбільш домінуюча робота, зазвичай, розміщується в верхньому лівому кутку діаграми, а найменш домінуюча – в правому нижньому кутку. В результаті виходить ступінчаста схема.

На діаграмі декомпозиції роботи повинні бути пронумеровані зліва направо. Номер роботи показується в правому нижньому кутку роботи.

Діаграма декомпозиції призначена для деталізації роботи. Роботи нижнього рівня – це те ж саме, що робота верхнього рівня, але в більш детальному викладі. Як наслідок цього кордону роботи верхнього рівня - це те ж саме, що кордони діаграми декомпозиції.

При декомпозиції будь-якої роботи входять в неї і виходили з неї стрілки повинні відобразитися на діаграмі декомпозиції, тобто повинна відбутися міграція стрілок.

У попередньому підрозділі диплома була побудована контекстна діаграма, що містить тільки одну роботу, яка описує діяльність в цілому, без деталізації складових цієї роботи. В даному підрозділі буде побудована діаграма декомпозиції першого рівня в нотації IDEF0.

Декомпозиція – це поділ складного об'єкта, системи, завдання на складові частини, елементи.

За допомогою діаграми декомпозиції першого рівня покажемо, з яких більш дрібних робіт складається робота «Створення онлайн-сервісу допомоги безпритульним тваринам "GetPet"» (рис. 2.4).

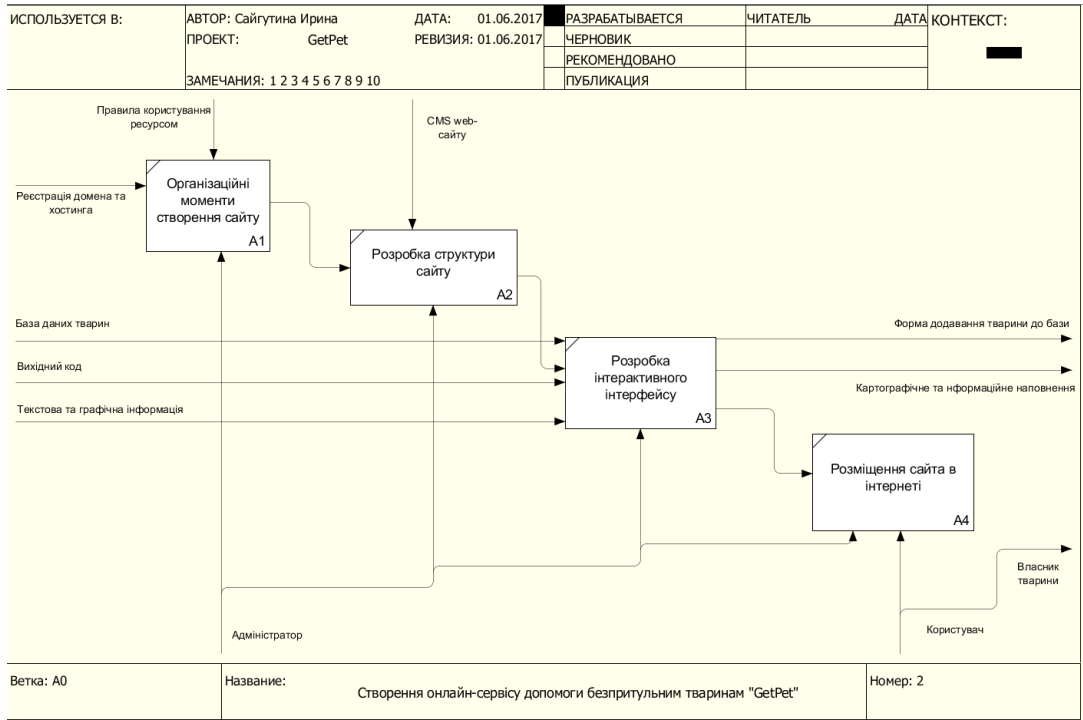


Рисунок 2.4 – Діаграма декомпозиції сайту «GetPet»

3 ВИБІР ПРОГРАМНИХ ЗАСОБІВ ДЛЯ РЕАЛІЗАЦІЇ СИСТЕМИ

3.1 Вибір системи управління контентом CMS

При виборі системи управління сайтом (CMS) потрібно враховувати кілька факторів, а саме:

- вартість даної системи;
- можливої функціональності має вистачити на реалізацію системи.

На сьогоднішній день існує безліч систем управління контентом як платних, так і безкоштовних, з досить великим функціоналом. Розглянемо основні з них: Joomla, DataLife Engine, WordPress.

3.1.1 Опис CMS Joomla

Joomla – безкоштовна система для створення веб-сайтів (рис. 3.1) [4]. Це проект з відкритим вихідним кодом, який, як і більшість подібних проектів, не стоїть на місці. Він дуже успішно розвивається, протягом уже семи років, і користується популярністю у мільйонів користувачів по всьому світу.

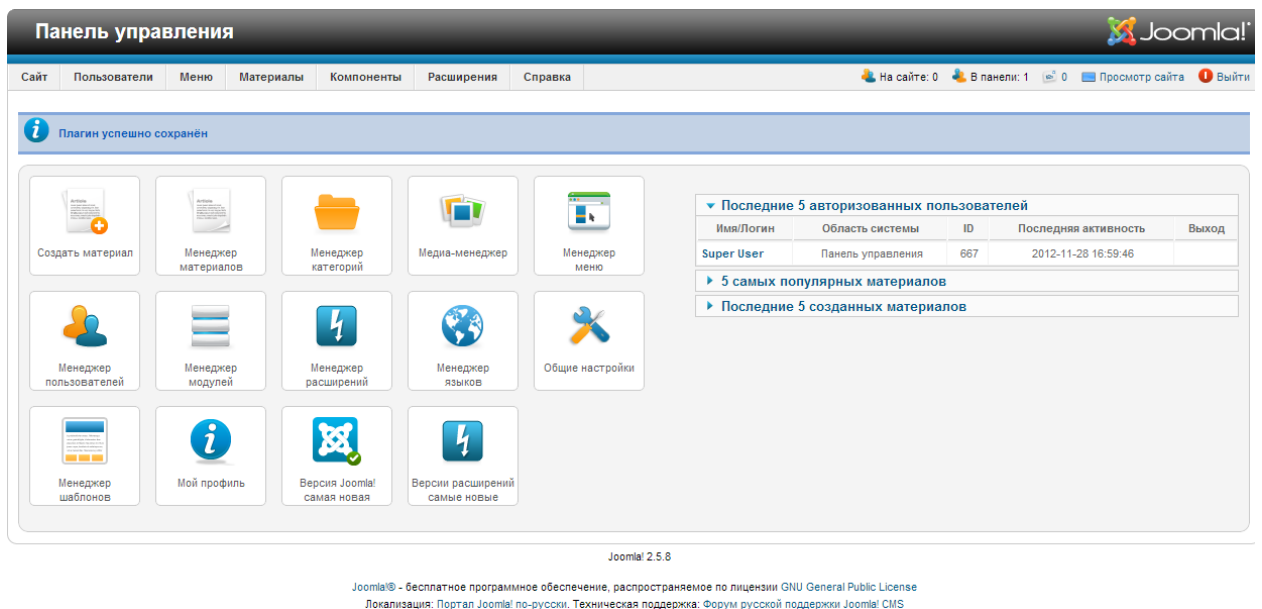


Рисунок 3.1 – Консоль CMS Joomla

Переваги використання Joomla як CMS для сайту.

Безкоштовність і відкритий вихідний код. Мабуть, для багатьох користувачів саме ціна системи відіграє вирішальну роль при виборі того чи

іншого продукту. Звичайно, як і у випадку з іншими системами, у вас може виникнути необхідність витратити деяку суму на додаткові компоненти, а також, послуги програміста і дизайнера при персоналізації сайту, але спочатку система поширюється вільно, і кожен може вносити в її код зміни за власним бажанням.

Орієнтованість на дизайнерів і розробників. Завдяки надзвичайно простий модифікації зовнішнього вигляду сайту ви можете всього в кілька кліків змінити зовнішній вигляд сайту до невпізнання. Це ще одна чудова можливість максимально персоналізувати свій сайт, підлаштувати його зовнішній вигляд під ваші потреби.

Наявність безлічі додаткових компонентів, розширень, плагінів, інструментів. Така ситуація стала майже стандартної для будь-якого популярного продукту з відкритим вихідним кодом. Завдяки тисячам додаткових компонентів ви зможете вбудувати в свій сайт практично що завгодно.

Є привабливість готового продукту. Навіть якщо у вас невеликий досвід роботи з сайтами, на виході ви отримаєте дуже професійно виглядає сайт. Це великий плюс для тих, хто вирішив створити комерційний продукт, який повинен виглядати солідно для користувача.

Постійні оновлення і підвищення якості та безпеки системи. Спільнота Joomla є одним з найбільших. Серед шанувальників цієї CMS величезна кількість розробників, які постійно створюють нові розширення, а також знаходять недоробки і способи поліпшення роботи системи. В результаті кінцевий користувач отримує все більш і більш якісну систему.

Універсальність. Як і у випадку з Drupal, можливості Joomla дозволяють створити сайт будь-якої складності і змісту з мінімальним використанням додаткових інструментів і розширень.

Недоліки використання Joomla як CMS для сайту:

Одним з недоліків Joomla можна назвати труднощі при переході на більш нову версію, коли можливе порушення деяких встановлених компонентів, причому можливо тільки послідовне оновлення через проміжні версії;

Для ранніх версій Joomla актуальна і складність самих оновлень. На щастя, в останній версії реалізований автоматичний пошук і установка оновлень;

Joomla лідирує в числі систем-улюбленців зломщиків. Сайти на цій системі досить часто піддаються атакам. Не останньою причиною тут є і

недостатньо висока захищеність системи, але ситуація постійно покращується завдяки оперативній роботі спільнот Joomla;

Складність освоєння. Доводиться визнати, що функціональність і універсальність завжди вимагають деяких зусиль від користувача для їх освоєння;

Іншим загальним недоліком, властивим для всіх безкоштовних систем, є відсутність служби підтримки, але зате користувач отримує сотні навчальних матеріалів і посібників, а також десятки майданчиків (форуми і т.д.) для обміну досвідом та вирішення виникаючих проблем.

3.1.2 Опис CMS DataLife Engine

DataLife Engine – це багатофункціональна система управління сайтом. Завдяки дуже потужній системі управління новинами, публікаціями, статтями, користувачами, DataLife Engine призначений в першу чергу для організації власних засобів масової інформації та блогів в мережі інтернет.

Велика розмаїтість базових можливостей скрипта, спочатку інтегрованих в скрипт, дозволяють DataLife Engine на крок вперед випереджати своїх найближчих конкурентів у даному напрямку створення сайтів. А завдяки точної і продуманій структурі ядра, що зводить до мінімуму вимоги до ресурсів сервера, DataLife Engine здатний з легкістю працювати з високовідвідуваними проектами с мінімальним навантаженням на сервер.

Гнучкість і легкість в налаштуванні і інтеграції власного дизайну, дозволить вам в найкоротші терміни вивчити і зрозуміти структуру шаблонів і відповідно розгорнути власні сайти, без будь-яких додаткових матеріальних витрат.

Використання просунутої технології AJAX скоротить як трафік і ресурси вашого сервера, так і трафік ваших відвідувачів, не кажучи вже про зручність використання даної технології на сайті для відвідувачів вашого сайту. Завдяки підвищеній увазі до безпеки скрипта, до зручності роботи з ним, а також динамічному розвитку, на користь використання DataLife Engine вже висловилися більше 90 000 порталів по всьому світу, успішно використовують його в роботі.

Велика увага в скрипті приділено оптимізації під пошукові системи, що дозволяє привести на ваш сайт додаткових відвідувачів.

Також як і інший популярний движок, DLE написаний на PHP і в якості СУБД використовує MySQL. DLE CMS має потужну систему публікацій,

настройки та редагування новин, ставить мінімальні вимоги до ресурсів сервера, дозволяє легко працювати з високовідвідуваними проектами.

Панель управління CMS DLE представлена на рисунку 3.2 [5].

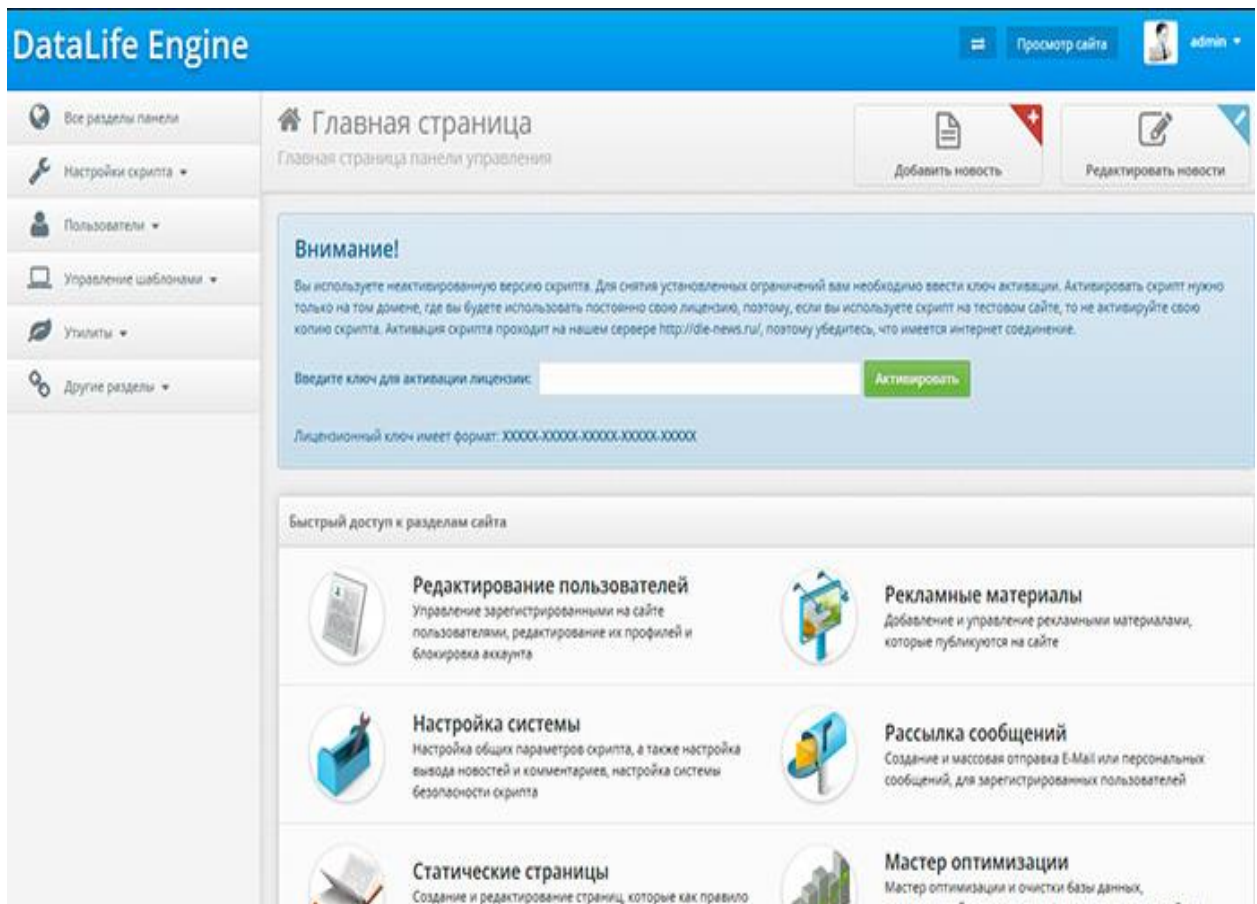


Рисунок 3.2 – Адмінпанель CMS DLE

Система управління контентом DLE для відвідувачів Вашого сайту і надає наступні можливості:

- 1) Реєстрація на сайті;
- 2) Додавання коментарів;
- 3) Зміна та видалення користувачами своїх власних коментарів;
- 4) Можливість додавання новин з боку відвідувачів;
- 5) Модерація новин;
- 6) Статистика по конкретному користувачеві (включаючи рейтинг і профіль);
- 7) Можливість завантаження фотографії в профілі користувача;
- 8) Можливість відновлення забутого пароля;
- 9) Можливість редагування новини безпосередньо на сайті.

Переваги використання DLE як CMS для сайту:

1) С допомогою численних модулів вона модифікується в каталог або галерею. Відмінні риси движка: швидкість, простота і зручність використання;

2) Навчитися керувати цим скриптом легко. Адміністративна панель інтуїтивно зрозуміла і не вимагає особливих пояснень;

3) DLE може з легкістю працювати з відвідуваними проектами, не створюючи особливих навантажень на сервер, кількість запитів до бази даних мінімально;

4) Розробники передбачили ряд інструментів, які дозволяють налаштувати DLE під свої потреби, а саме:

– розділ «Аналіз продуктивності» в адмінці видає рекомендації щодо зміни і відключення тієї чи іншої можливості скрипта з зазначенням ступеня створюваної їй навантаження. Відмовившись від непотрібних можливостей можна спробувати зменшити навантаження, не переходячи на більш дорогий тариф на хостингу;

– розділ «налагодження та оптимізація бази даних» - утиліта, здатна оптимізувати вашу базу даних, прискоривши її роботу і звільнивши місце на диску, а також зробити ремонт бази даних в разі пошкодження структури таблиць;

– розділ «шаблони сайту» дозволить вам легко впроваджувати власний дизайн в найкоротші терміни. Для кожного запису і статичної сторінки можна задати свій шаблон. Висновок даних в шаблоні гнучко настроюється (завдяки так званім тегам).

До недоліків скрипта часто відносять відсутність автоматичної установки модулів; сформовану порочну практику редагування файлів движка, чого в ідеалі не повинно бути через загрозу втрати змін при оновленні.

3.1.3 Опис CMS WordPress

WordPress – це система створення та управління веб-сайтом.

WordPress – CMS з відкритим вихідним кодом, поширювана під GNU GPL. Написана на PHP, в якості бази даних використовує MySQL. Сфера застосування - від блогів до досить складних новинних ресурсів і навіть інтернет-магазинів. Вбудована система «тем» і «плагінів» разом з вдалою архітектурою дозволяє конструювати практично будь-які проекти.

WordPress є ідеальною платформою для публікацій, орієнтованих на естетику, підтримку стандартів і зручність використання. WordPress безкоштовний і вільний до розповсюдження.

Консоль CMS WP можна побачити на рис. 3.3 [6]

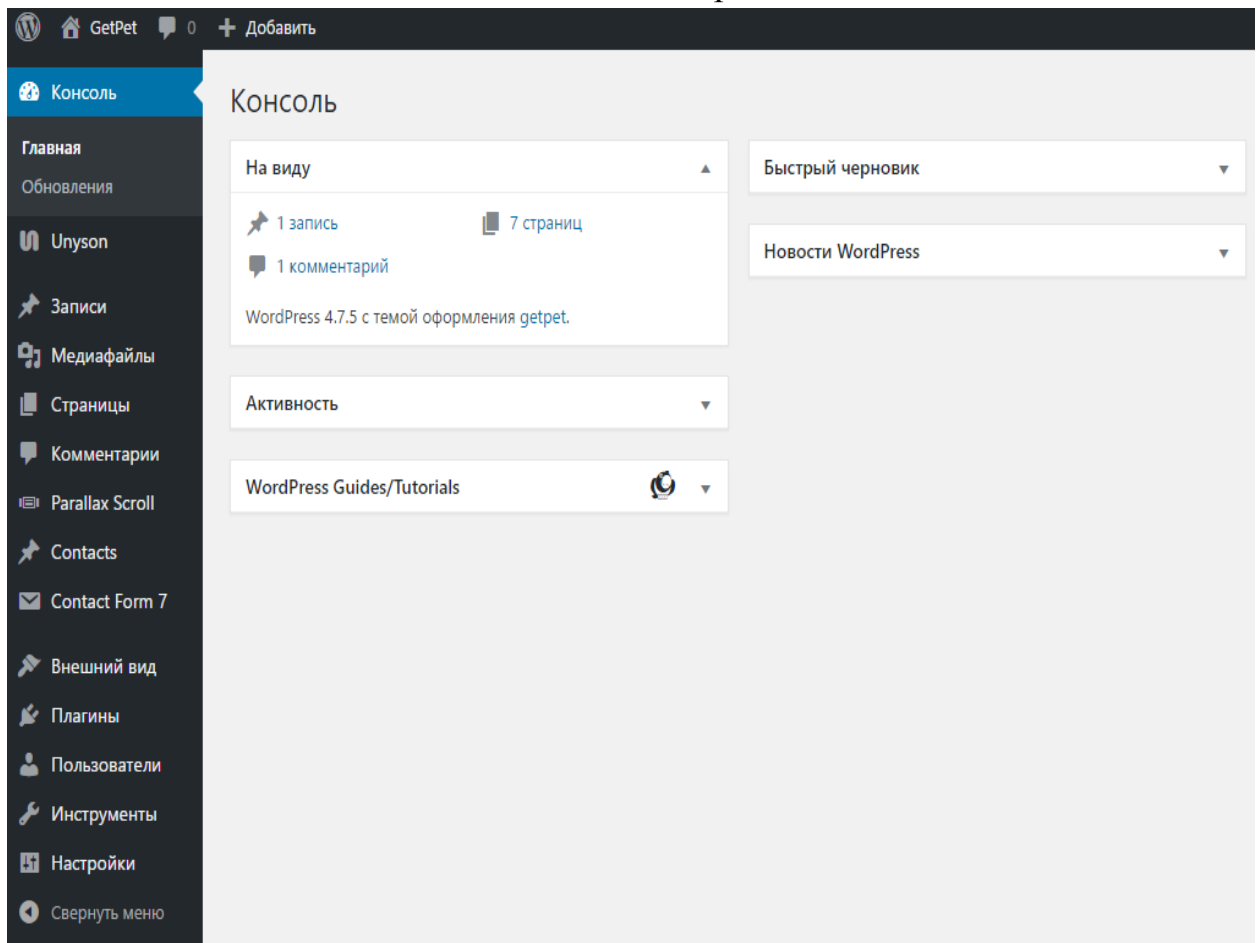


Рисунок 3.3 – Адмінпанель CMS WordPress

Основні можливості WordPress:

1) Управління та адміністрування:

WordPress розроблений для установки на персональний хостинг, що дає повний контроль над блогом, на відміну від безкоштовних блог-сервісів;

– Управління зареєстрованими користувачами;

– Особисті профілі користувачів;

– Легка установка і оновлення. Відома установка за 5 хвилин;

– Динамічне створення сторінок, не має потреби пересоздавати всі сторінки блогу при внесенні змін до нього;

– Інтернаціоналізація та локалізація WordPress "говорить" на Вашому мовою.

2) Публікація:

- RSS стрічки, публікація через протоколи RSS 1.0, RSS 2.0 і ATOM;
 - Постійні посилання.
- 3) Дизайн та розширення можливостей:
- Заснований на шаблонах дизайн;
 - Шаблонові теги, можливість просто додати зміни в шаблони без особливих знань PHP;
 - Теми, які легко змінюються вид блогу, просто скачавши і встановивши нову тему до нього;
 - Модулі. Величезна кількість плагінів (додаткових підпрограм), що розширюють функціональність блогу.
- 4) Контент:
- Захист паролем обраної інформації;
 - Запланована на певну дату публікація;
 - Багатосторінкові записи;
 - Завантаження файлів і зображень в запису.
- 5) Категорії:
- Візуальний редактор;
 - Пошук по блогу;
 - Коментування.

Як CMS для розробки веб-орієнтованої системи, спрямованої на допомогу бездомним тваринам, була обрана WordPress з кількох причин:

- 1) CMS WordPress абсолютно безкоштовна;
- 2) Велика кількість різних шаблонів;
- 3) Велика кількість плагінів;
- 4) Відкритий вихідний код;
- 5) Інтуїтивно зрозуміла адмінпанель;
- 6) Легка установка CMS на хостинге.

WordPress – це безкоштовна CMS, яка має багато переваг відносно інших систем. Наприклад, вона має дуже зручну адмінпанель, за допомогою якої можна зробити сайт будь-якої складності.

Крім того, для WordPress є величезна кількість безкоштовних шаблонів, які приємно дивують своєю якістю, різноманітність додаткових модулів під різні завдання. А велика спільнота завжди допоможе вирішити проблеми з настройками системи.

Для створення сайту вибрано CMS WordPress, так як є ряд плюсів які кращі:

- 1) CMS WordPress абсолютно безкоштовний;
- 2) Велика кількість різних шаблонів;

- 3) Велика кількість плагінів;
- 4) Відкритий вихідний код;
- 5) Адмін панель легка в розумінні;
- 6) Легка установка;
- 7) Наявність офіційних програм для пристроїв Apple і Android.

Чому не вибрано Joomla і Datalife Engine:

- Інтерфейси важчі у використанні;
- Споживають більше ресурсів і потрібно купувати більш дорогі і більш продуктивні хостинги і сервера.

3.2 Особливості використання PHP

PHP – скрипкова мова загального призначення, інтенсивно застосовується для розробки веб-додатків. В даний час підтримується переважною більшістю хостинг-провайдерів і є одним з лідерів серед мов, що застосовуються для створення динамічних веб-сайтів.

PHP має декілька недоліків, наприклад:

- 1) не підходить для створення десктопних додатків або системних компонентів;
- 2) має слабкі засоби для роботи з винятками;
- 3) глобальні параметри конфігурації впливають на базовий синтаксис мови, що ускладнює настройку сервера і розгортання додатків;
- 4) об'єкти передаються за значенням, що бентежить багатьох програмістів, які звикли до передачі об'єктів по посиланню, як це робиться в більшості інших мов;
- 5) веб-додатки, написані на PHP, часто мають проблеми з безпекою.

Але незважаючи на ці недоліки, він має свої переваги, котрі відрізняють його від інших серверних мов програмування, і найбільш підходять під задачі саме web-додатків.

- є вільним програмним забезпеченням, поширюваним під особливою ліцензією (PHP ліцензії);
- легкий в освоєнні на всіх етапах;
- підтримується великою спільнотою користувачів і розробників;
- має розвинену підтримку баз даних;
- є величезна кількість бібліотек і розширень мови;
- може використовуватися в ізольованому середовищі;
- пропонує нативні кошти організація веб-сесії, програмний інтерфейс розширення;

- є досить повною заміною пропріетарної середовища ASP (Active Server Pages) Від Microsoft;
- може бути розгорнутий майже на будь-якому сервері;
- адаптовано під велику кількість апаратних платформ і операційних систем.

Поєднавши в собі переваги інших мов програмування та обравши для себе єдине основне спрямування, PHP став універсальним, пристосованим до веб-програмування, зрозумілим та простим для вивчення засобом роботи в Інтернеті.

Як відомо, WordPress написаний і працює виключно на PHP.

3.3 Система управління базами даних MySQL

Для реалізації веб-орієнтованої системи «GetPet» в якості бази даних була обрана MySQL. Для адміністрування базою даних буде використаний веб-додаток з відкритим кодом, написаний на мові PHP, який називається PhpMyAdmin. Воно дозволяє швидко і просто створювати, редагувати і видаляти не тільки базу даних, але і окремі її елементи, такі як: таблиці, атрибути таблиць і безпосередньо самі записи. Безперечно, це значно спрощує роботу з базою даних і скорочує час на її реалізацію.

MySQL – вільна реляційна система управління базами даних. Розробка і підтримка MySQL здійснює корпорація Oracle. Продукт поширюється як під GNU General Public License, так і під власною комерційною ліцензією. Крім цього, розробники створюють функціональність за замовленням ліцензійних користувачів. Саме завдяки такому замовленню майже в найраніших версіях з'явився механізм реплікації.

MySQL є рішенням для малих і середніх додатків. Входить до складу серверів WAMP, AppServ, LAMP і в портативні збірки серверів Денвер, XAMPP, VertrigoServ. Зазвичай MySQL використовується як сервер, до якого звертаються локальні або видалені клієнти, проте в дистрибутиві входить бібліотека внутрішнього сервера, що дозволяє включати MySQL в автономні програми.

Гнучкість СУБД MySQL забезпечується підтримкою великої кількості типів таблиць: користувачі можуть вибрати як таблиці типу MyISAM, що підтримують повнотекстовий пошук, так і таблиці InnoDB, що підтримують транзакції на рівні окремих записів. Більш того, СУБД MySQL поставляється із спеціальним типом таблиць EXAMPLE, що демонструє принципи

створення нових типів таблиць. Завдяки відкритій архітектурі і GPL-ліцензуванню, в СУБД MySQL постійно з'являються нові типи таблиць.

З часом MySQL все розширювалася і зараз вона – одна з найпоширеніших систем керування базами даних.

Вона використовується, в першу чергу, для створення динамічних веб-сторінок, оскільки має впевнену підтримку з боку різноманітних мов програмування.

Переваги та недоліки MySQL.

MySQL являється компактним багатопоточним сервером баз даних та характеризується великою швидкістю, стійкістю і простотою використання.

Ця система вважається гарним рішенням для малих і середніх додатків. Вихідний код сервера компілюється на безлічі платформ. Найбільш повно можливості сервера виявляються в UNIX-системах, де є підтримка багатопоточності, що підвищує продуктивність системи в цілому. Для некомерційного використання MySQL є безкоштовним.

Можливості сервера MySQL:

- простота у встановленні та використанні;
- підтримується необмежена кількість користувачів, що одночасно працюють із БД;
- кількість рядків у таблицях може досягати 50 млн;
- висока швидкість виконання команд;
- наявність простої і ефективної системи безпеки.

Недоліками сервера MySQL залишається не реалізована підтримка транзакцій, замість якої пропонується використовувати LOCK/UNLOCK TABLE, а також відсутність підтримки для зовнішніх (foreign) ключів, тригерів, збережених процедур та представлень (VIEW). Та для розробки малих та середніх інформаційних систем, призначених для використання в межах робочих груп ці недоліки є фактично невідчутними.

Оскільки в якості CMS була обрана WordPress, то вибір СУ бази даних очевидний – це MySQL. Як було сказано раніше, WordPress працює на PHP, тому потрібно розглянути взаємодію MySQL та PHP.

Реалізація взаємодії MySQL та PHP.

Проекти на основі безкоштовного ПЗ, які вимагають повнофункціональної системи керування базами даних часто використовують MySQL. До таких проектів належать, наприклад, WordPress, phpBB, Drupal та інше програмне забезпечення, побудоване на стеку продуктів LAMP (Linux, Apache, MySQL, PHP/Perl/Python).

Популярний візуальний інтерфейс PhpMyAdmin, написаний на PHP, дозволяє працювати з MySQL в графічному режимі. Цей інтерфейс дозволяє значно спростити роботу з базами даних в MySQL.

У текстовому режимі робота з базою даних виглядає просто як введення команд у командний рядок, а результати вибірок повертаються у вигляді своєрідних таблиць, поля в яких налазять один на одного, якщо дані не вміщаються на екрані.

PhpMyAdmin дозволяє використовувати всі переваги роботи в браузері, включаючи прокрутку зображення, у випадку, коли йому для відображення бракує розмірів екрану. Більшість базових SQL-функцій роботи з даними в PhpMyAdmin зведені до інтуїтивно зрозумілих, завдяки інтерфейсу, дій, що нагадують перехід за посиланнями в мережі Інтернет.

Проте однозначною перевагою роботи в текстовому режимі залишається необмеженість функцій реалізованим інтерфейсом.

В дистрибутив самої мови PHP входить розширення, що містить вбудовані функції для роботи з базою даних MySQL. При роботі з веб-інтерфейсом для додавання інформації до бази даних користувач повинен ввести ці дані в HTML-форму та відправити їх на сервер, а все інше виконається засобами скрипта. Для перегляду вмісту таблиць достатньо перейти по посиланню і зайти на потрібну сторінку.

Процес одержання доступу до СКБД MySQL та роботи з нею можна розбити на такі чотири послідовних кроки:

- 1) З'єднання з БД:
 - `mysql_connect()` – з'єднання з MySQL;
 - `mysql_select_db()` – вибір БД.
- 2) Доступ до БД і отримання необхідних даних:
 - `mysql_query()` – виконати SQL-запит;
 - `mysql_fetch_array()`, `mysql_fetch_assoc()` – видобути наступний запис.
- 3) Звільнення пам'яті:
 - `mysql_free_result()` – звільнення пам'яті, відведеної для зберігання результату запиту.
- 4) Закриття з'єднання:
 - `mysql_close()` – закриття з'єднання; виконується автоматично після завершення виконання скрипта.

Дуже часто помилки в скриптах із доступом до БД виникають у зв'язку з некоректно сформованим SQL-виразом. Тому першим кроком у налагодженні PHP-програми, що виконує запити до MySQL повинна бути перевірка правильності SQL-інструкцій. Для цього слід вивести SQL-запит на сторінку, і, скопіювавши його, виконати безпосередньо в phpMyAdmin.

4 ПРОГРАМНА РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ

4.1 Етапи розробки сайту

Процес розробки веб-сайту можна розділити на наступні етапи:

1) Визначити цілі створення веб-ресурсу – на цьому етапі необхідно визначити, для чого потрібен сайт, тобто які завдання він повинен вирішувати: надати загальне уявлення про функції сайту.

2) Створення технічного завдання на розробку веб-сайту – необхідно описати, які функції повинен виконувати сайт, побажання по дизайну сайту, використовувані технології для його створення.

3) Створення макета для майбутнього дизайну – на цьому етапі необхідно зробити прототип майбутнього сайту, показати, яка в нього буде структура, як будуть розташовані ті чи інші блоки. На основі цього прототипу буде нарисований майбутній дизайн сайту за допомогою таких засобів, як Adobe Photoshop або Adobe Illustrator.

4) Верстання макету – за допомогою гіпертекстової розмітки (HTML) та каскадних таблиць стилів (CSS), а також мови сценаріїв (JavaScript) проводиться верстка макету.

5) «Натягування» макету на обрану CMS – з урахуванням специфіки обраної CMS, проводиться натягування розробленого макету під неї.

6) Програмування сайту – за допомогою PHP відбувається створення всіх сторінок сайту, визначається порядок роботи меню, розставляються гіперпосилання, створюється динаміка на сайті, програмуються основні складові потрібного функціоналу.

7) Наповнення сайту інформацією – додається контент сайту, тобто якась текстова та мультимедійна інформація.

8) Розташування сайту в мережі Інтернет – передбачається, що розробка сайту проводиться на локальному хості (Denwer, OpenServer), і тільки після повної розробки сайт потрібно завантажити на хостінг в Інтернеті.

9) Тестування сайту – цей етап можна здійснити як до, так і після розміщення сайту по його «рідної» адресою. На цьому етапі виявляються всі помилки і недоліки в програмуванні і написанні текстів.

4.2 Основні принципи розробки дизайну сайту

На сьогоднішній час веб-дизайн базується на принципах акцентування, контраст, балансування, вирівнювання, повторення і зручність сприйняття. Саме ці поняття будуть використовуватися для створення сайту "GetPet", тому розглянемо їх детальніше.

Акцентування – це підкреслення особливої важливості або значимості якого-небудь елемента. Щоб дотримати в процесі створення сайту принцип акцентування, необхідно проаналізувати вміст сайту і з'ясувати, яка ієрархія елементів має місце в його вмісті, тобто треба уявити собі перелік елементів, які потрібні на веб-сторінці.

Контрастування – це візуальна диференціація двох і більше елементів. Елементи з високим ступенем контрастування виглядають чітко і відокремлено, в той час, як елементи з низьким ступенем контрастування здаються такими, що зливаються і мають тенденцію змішуватися в єдиній масі. Ця характеристика допомагає швидко привертати увагу до ключових елементів, наприклад до вмісту, елементів, що дозволяє утримувати певну дію, або до тексту, що дає зрозуміти, в чому полягає призначення сайту.

Принцип балансування крутиться навколо ідеї про те, як розподілені елементи в дизайні і як вони співвідносяться із загальним розподілом візуальної навантаження в рамках веб-сторінки.

Вирівнювання – це впорядкування елементів таким чином, щоб вони максимально близько підступали до природних ліній або кордонів, які вони утворюють. Тобто, це означає, що краще впорядковувати елементи по сітці. Для створення сайту «GetPet» використовується Bootstrap сітка.

Повторення передбачає неодноразове використання в дизайні одного і того ж елемента різними способами. Дизайн, який містить повторення, стає уніфікованим. Повторення може виражатися в безлічі форм, включаючи використання того ж кольору, обрисів, лінії, шрифти, зображення і загальний підхід до стилізації.

Зручність сприйняття – свого роду стежка, якою користувач дотримується при візуальному ознайомленні з елементами дизайну. Воно головним чином залежить від того, на чому дизайнер зробив акцент і в якій мірі він забезпечив контрастування елементів між собою. Фактично зручність сприйняття цілком залежатиме від того, як дизайнер скомпонував всі елементи сторінки.

При дотриманні всіх цих правил, можна зробити простий, зручний і привабливий дизайн для сайту.

4.3 Розробка структури сайту

Структура сайту – це організація зв'язків між сторінками, розділами і різними елементами сайту.

Навігація забезпечує доступ до неї, тобто різні меню, перехресні посилання і карта сайту. На етапі планування сайту і написання технічного завдання необхідно ретельно її продумати. Зручна і зрозуміла структура сайту допомагає користувачеві легко знаходити потрібну йому інформацію.

На сайтах із дуже великим обсягом інформації є сторінки третього рівня, а якщо необхідно – то й четвертого, п'ятого тощо. Загалом виділяють три типи структур веб-сайтів – лінійну, деревоподібну та довільну. Подорожуючи сайтом із лінійною структурою, з головної сторінки ви перейдете на другу сторінку, з неї – на третю тощо. На сайті з деревоподібною структурою з головної сторінки можна потрапити на одну зі сторінок другого рівня, звідти – на одну зі сторінок третього рівня тощо. Сайт із довільною структурою видається зовсім неорганізованим, але саме в цьому й полягає принцип його створення. Подорожуючи таким сайтом, ви можете переходити з однієї його сторінки на інші в різні способи, і ваш шлях назад не обов'язково має бути таким самим.

Вибір структури визначається особливостями завдань, що розв'язуються за допомогою веб-сайту.

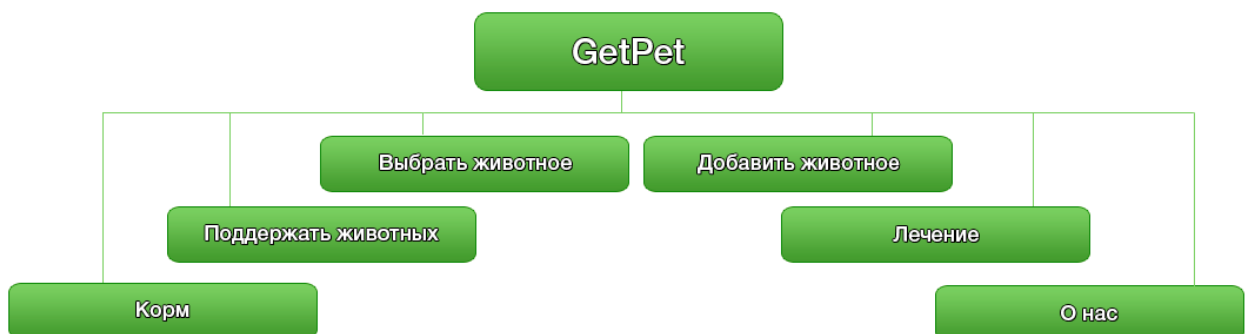


Рисунок 4.1 – Структурна схема сайту GetPet

Більшість веб-ресурсів побудовані за блоковим принципом. Все просто: нагорі шапка, зліва – меню, праворуч – блок реклами, посередині – інформація, а внизу (в підвалі сайту) написано – «Копірайт – всі права захищені».

Структура сайту може бути ускладнена – додано додаткове меню під шапкою і продубльовано над копірайтом. Також навігаційні посилання розміщують і справа замість реклами – це кому як подобається і залежить від призначення даного ресурсу. Така структура хороша для інформаційного порталу, оскільки залишено багато місця під посилання на різні розділи і лише під шапкою і над «підвалом» сайту відбувається дубляж.

Однак, в такій структурі важче шукати потрібну інформацію, тому останнім часом багато сайтобудівники використовуються техніку цільової сторінки, де інформація легко і швидко сприймається користувачами.

Landing Page (посадочна сторінка) – це веб-сторінка, основне завдання якої є збір контактних даних цільової аудиторії. Використовується для посилення ефективності сприйняття інформації, збільшення аудиторії. Цільова сторінка зазвичай містить інформацію про послугу, на яку треба звернути увагу.

Існує чотири основних типи цільових сторінок:

- 1) автономна цільова сторінка
- 2) мікросайту
- 3) головний сайт
- 4) сторінка сегментація

Структура сайту «GetPet» буде побудована по типу мікросайту.

Мікросайт – це окремих сайт з інформацією про рекламованої послуги або товар, що складається з декількох сторінок (найчастіше не більше 5). Зазвичай Мікро містить мінімум текстової інформації, велика кількість привабливого і красиві зображення товару або послуги, відео-ролики. Подібний тип цільових сторінок часто використовується великими компаніями для реклами окремих проектів.

Landing Page має наступні переваги:

- 1) зосередженість користувача;
- 2) вищий коефіцієнт конверсії;
- 3) стимуляція користувачів зробити конкретну дію;
- 4) має більш привабливий вигляд, ніж звичайний блокові сайти.

Результативність та ефективність сприйняття користувача інформації залежить від того, познайомитися чи користувач взагалі з приготованим для нього пропозиції. Тому одним з найважливіших особливостей цільової сторінки є її доступність для користувача.

Враховуючи всі переваги Landing Page, головна сторінка, тобто точка входу на сайті, зроблена саме в такому стилі.

Розглянемо структуру головної сторінки сайту на рис. 4.2



Рисунок 4.2 – Структура головної сторінки

4.4 Розробка веб-інтерфейсу

Веб-інтерфейс – це сукупність засобів, за допомогою яких користувач взаємодіє з веб-сайтом або веб-застосунком через браузер. Веб-інтерфейси

отримали широке поширення у зв'язку із зростанням популярності всесвітньої павутини і відповідно повсюдного розповсюдження веб-браузерів.

Одним з основних вимог до веб-інтерфейсів є їхній однаковий зовнішній вигляд і однакова функціональність при роботі в різних браузерах.

Головна сторінка – це точка входу користувача на сайт, тому вона повинна бути максимум інформативна, містити навігацію, і зверстана таким чином, щоб користувачеві було легко сприймати інформацію.

Як відомо, сучасні розробники користуються фреймворками (тобто інструменти і бібліотеки, які вже створила людина для того, щоб інші люди не витрачали час на тривіальних завданнях).

Для скорочення часу на створення головної сторінки сайту у вигляді Landing Page, був використаний Unyson фреймворк, який останнім часом користується великою популярністю серед розробників на WordPress, через те, що включає в себе багато вбудованих модулів, використання яких допомагає скоротити час на рішення одноманітних завдань у вигляді розташування блоків в секціях.

З усіх модулів, які присутні в Unyson фреймворку, буде використаний конкретно PageBuilder, за допомогою якого можна адаптивно розташувати секції на сторінці.

Unyson фреймворк використовує сітку Twitter Bootstrap, що робить створені секції адаптивними під будь-який дозвіл пристроїв, а це значить, що даний сайт буде правильно відображатися на різних пристроях.

Bootstrap – вільний набір інструментів для створення сайтів і веб-додатків. Включає в себе HTML і CSS-шаблони оформлення для типографіки, веб-форм, кнопок, міток, блоків навігації та інших компонентів веб-інтерфейсу, включаючи JavaScript-розширення.

Bootstrap використовує сучасні напрацювання в області CSS і HTML, тому необхідно бути уважним при підтримці старих браузерів.

До недоліків фреймворка можна віднести бідну колірну гамму стандартного набору іконок. До переваг – хорошу реалізацію grid-сітки для масштабування веб-сторінки, створення адаптивного дизайну.

Основні переваги Bootstrap:

- сітка;
- адаптивність;
- форма і інші елементи;
- динаміка;
- повна інтеграція з препроцесором CSS, наприклад, LESS;
- універсальність.

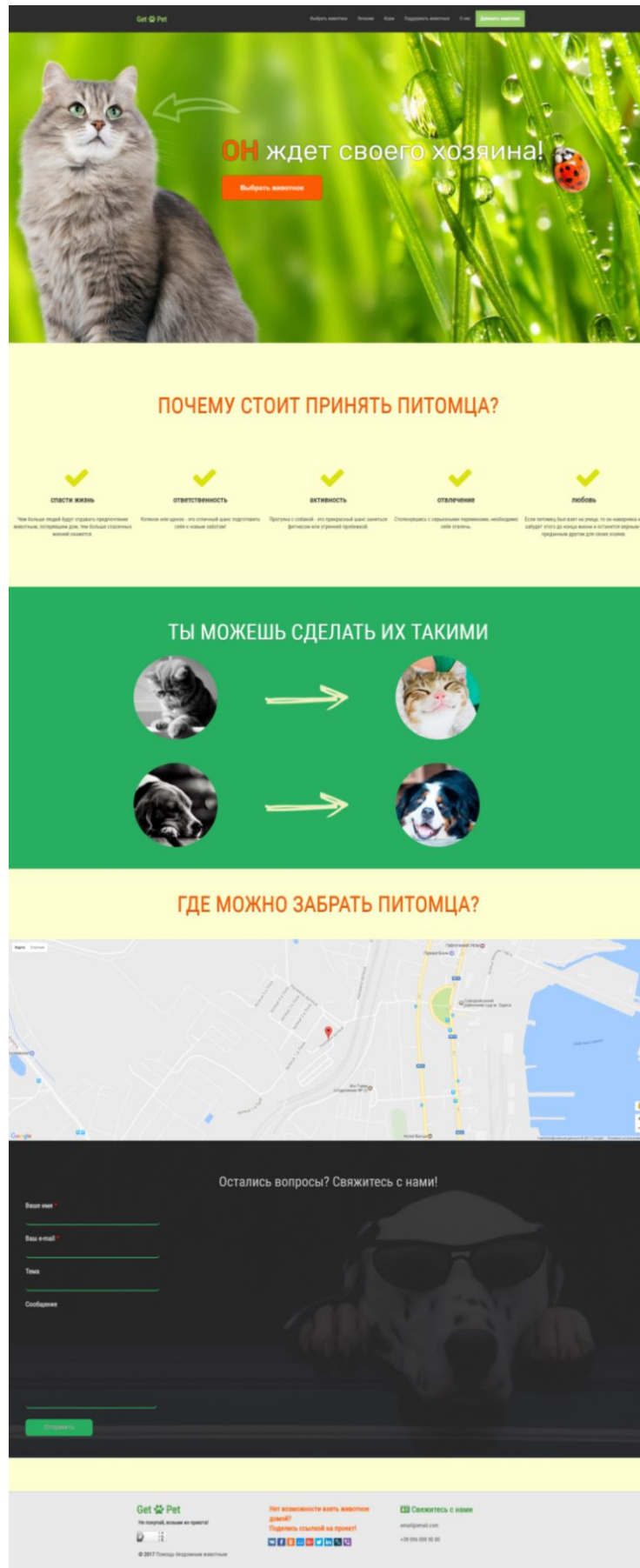


Рисунок 4. 3 – Веб-интерфейс сайту GetPet

Розглянемо більш детально інші сторінки сайту.

Get Pet

Выбрать животное Лечение Корм Поддержать животных О нас Добавить животное

Добавить животное

Форма добавления животного в базу данных

Категория:

Вид:

Пол:

Возраст:

Порода:

Окрас:

Организация:

Описание:

Фотография: Файл не выбран

Привит Стерилизован В тяжелом состоянии

Get Pet
Не покупай, возьми из приюта!

Нет возможности взять животное домой?
Поделись ссылкой на проект!

Свяжитесь с нами
email@email.com
+38 096 000 00 00

© 2017 Помощь бездомным животным

Рисунок 4. 4 – Сторінка «Додати тварину»

На даній сторінці розташовується форма, через яку користувач або адміністратор зможе додати тварину до бази даних. Тварина може бути додана в такі категорії:

- втрачена тварина;
- знайдена тварина;
- з притулку.


Get Pet

[Выбрать животное](#)
[Лечение](#)
[Корм](#)
[Поддержать животных](#)
[О нас](#)
[Добавить животное](#)

Выбрать животное

Выберите животное из списка

→ Потерянное животное



Кошка Девочка 3 месяца

Порода: Британская вислоская


Окрас: Синий

Организация: GladPet

Жирненькая кошечка потерялась...

Привет!
Стерилизован
В нормальном состоянии

→ Из приюта



Собака Мальчик 2 года

Порода: Французский бульдог


Окрас: Черный

Организация: ФЛП "Приют ОД.ЮА"

Грайливый, Ласкавый, Рухливый, Кипливый

Не привит!
Стерилизован
В нормальном состоянии

→ Из приюта



Собака Девочка 1 год

Порода: Чихуахуа


Окрас: Желтый

Организация: ФЛП "ПЕТС.РУ"

Люблячий, Рухливый, Смиливый, Пильный, Швадкий

Привет!
Не стерилизован
В нормальном состоянии

→ Из приюта



Собака Мальчик 2 года

Порода: Лабрадор

Окрас: Золотой

Организация: ФЛП "ДПК ПЕТС"

Компанійський, Тямущий, Спритний, Лагідний, Довірливий, Врівноважений, Добрий

Не привит!
Стерилизован
В нормальном состоянии

Get Pet

Не покупай, возьми из приюта!

© 2017 Помощь бездомным животным

**Нет возможности взять животное
домой?**

Поделись ссылкой на проект!

[VK](#) [FB](#) [IG](#) [GA](#) [TW](#) [IN](#) [SN](#)

Свяжитесь с нами

email@email.com

+38 096 000 00 00

Рисунок 4. 5 – Сторінка «Вибрати тварину»

На сторінці "Вибрати тварину" автоматично виводяться всі записи з бази даних, тобто раніше додані тварини.

Get Pet

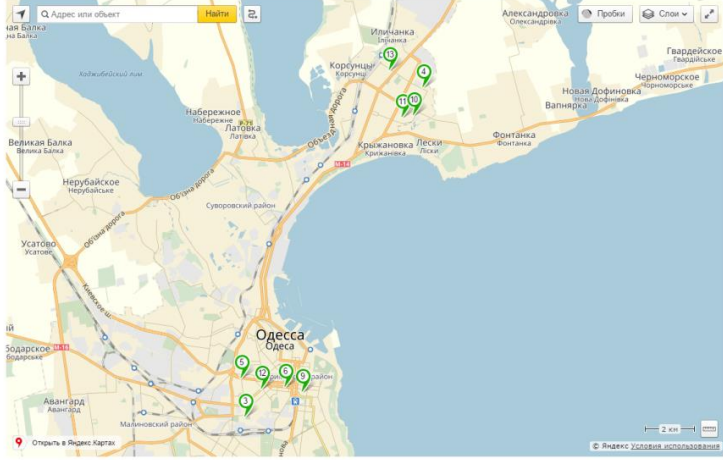
[Выбрать животное](#)
[Лечение](#)
[Корм](#)
[Поддержать животных](#)
[О нас](#)
[Добавить животное](#)

Лечение животных

Лечение

Информация! На этой странице нашего сайта вы сможете посмотреть справочную информацию по ветеринарным клиникам в городе Одесса – адреса, телефоны, официальные сайты, список продуктов и услуг.

№	Название	Адрес
1	Неотложная ветеринарная помощь	Украина, Одесса, проспект Небесной Сотни, 67
2	Габи	Украина, Одесса, проспект Академика Глушко, 22А
3	Зоолайф	Украина, Одесса, Мельницкая улица, 1
4	Евровет	Украина, Одесса, улица Генерала Бочарова, 44А
5	Центр ветеринарной медицины	Украина, Одесса, Разумовская улица, 54
6	Вита	Украина, Одесса, Малая Арнаутская улица, 74
7	Велес	Украина, Одесса, улица Леваневского, 5
8	Экввет	Украина, Одесса, проспект Академика Глушко, 21А
9	Дар	Украина, Одесса, Пантелеймоновская улица, 50
10	Ветлазарет	Украина, Одесса, улица Академика Сахарова, 20
11	Абсолют	Украина, Одесса, Марсельская улица, 42
12	Хатико	Украина, Одесса, Прохоровская улица, 8
13	Центр ветеринарных услуг	Украина, Одесса, улица Генерала Бочарова, 1



Get Pet
Не покупай, возьми из приюта!

© 2017 Помощь бездомным животным

Нет возможности взять животное домой?
Поделись ссылкой на проект!

[VK](#) [f](#) [g+](#) [in](#) [t](#) [v](#)

Свяжитесь с нами

email@email.com
+38 096 000 00 00

Рисунок 4. 6 – Сторінка «Лікування»

На даній сторінці представлені адреси основний ветеринарні клініки з відповідними позначками на карті. Для відображення адрес на карті використовується Yandex API.

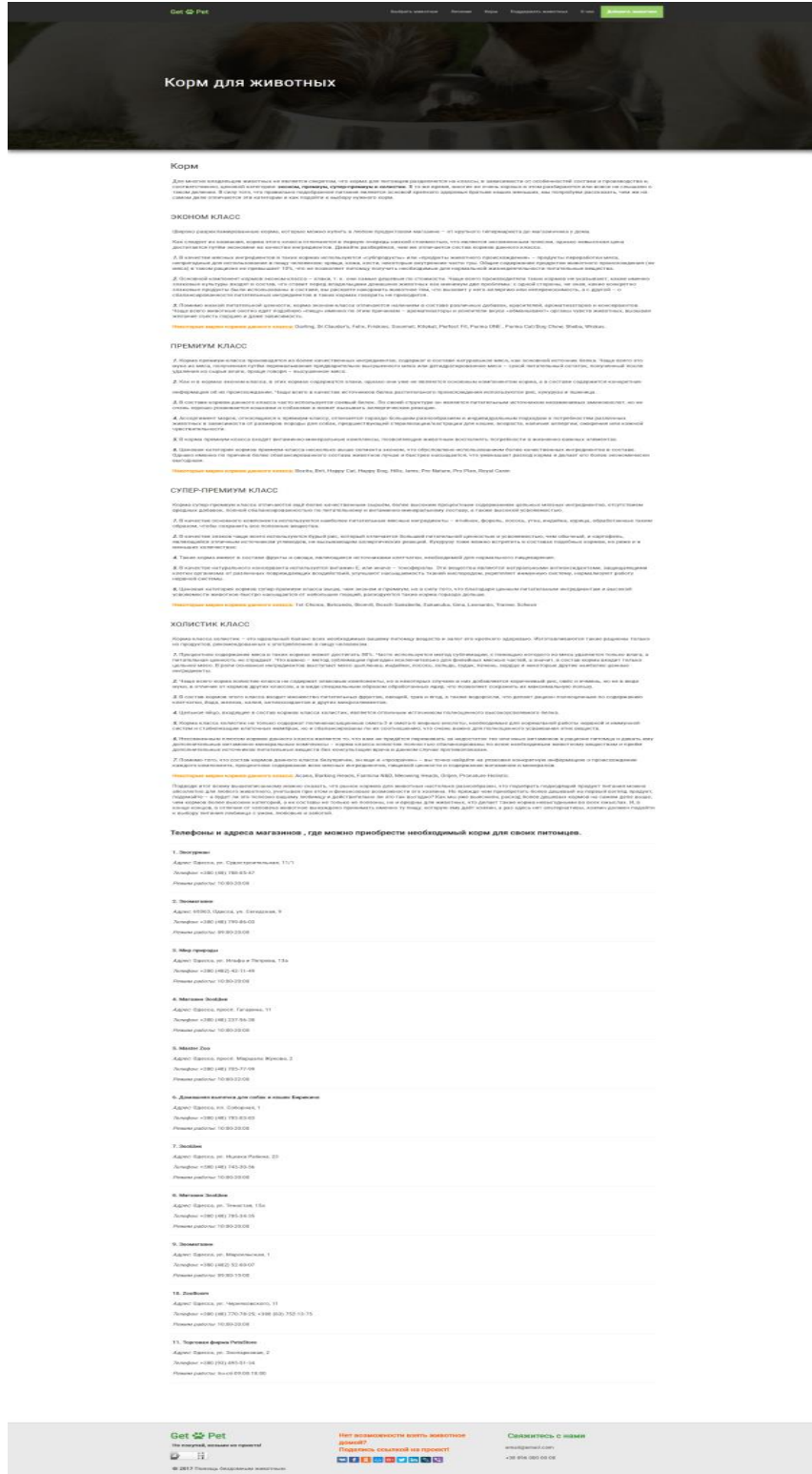


Рисунок 4. 7 – Сторінка «Корм для тварин»

На даній сторінці представлена інформація для майбутніх власників тварин про те, який корм слід вибрати для своїх вихованців. Також на сторінці є адреси, час роботи і телефони магазинів, де даний корм можна купити.

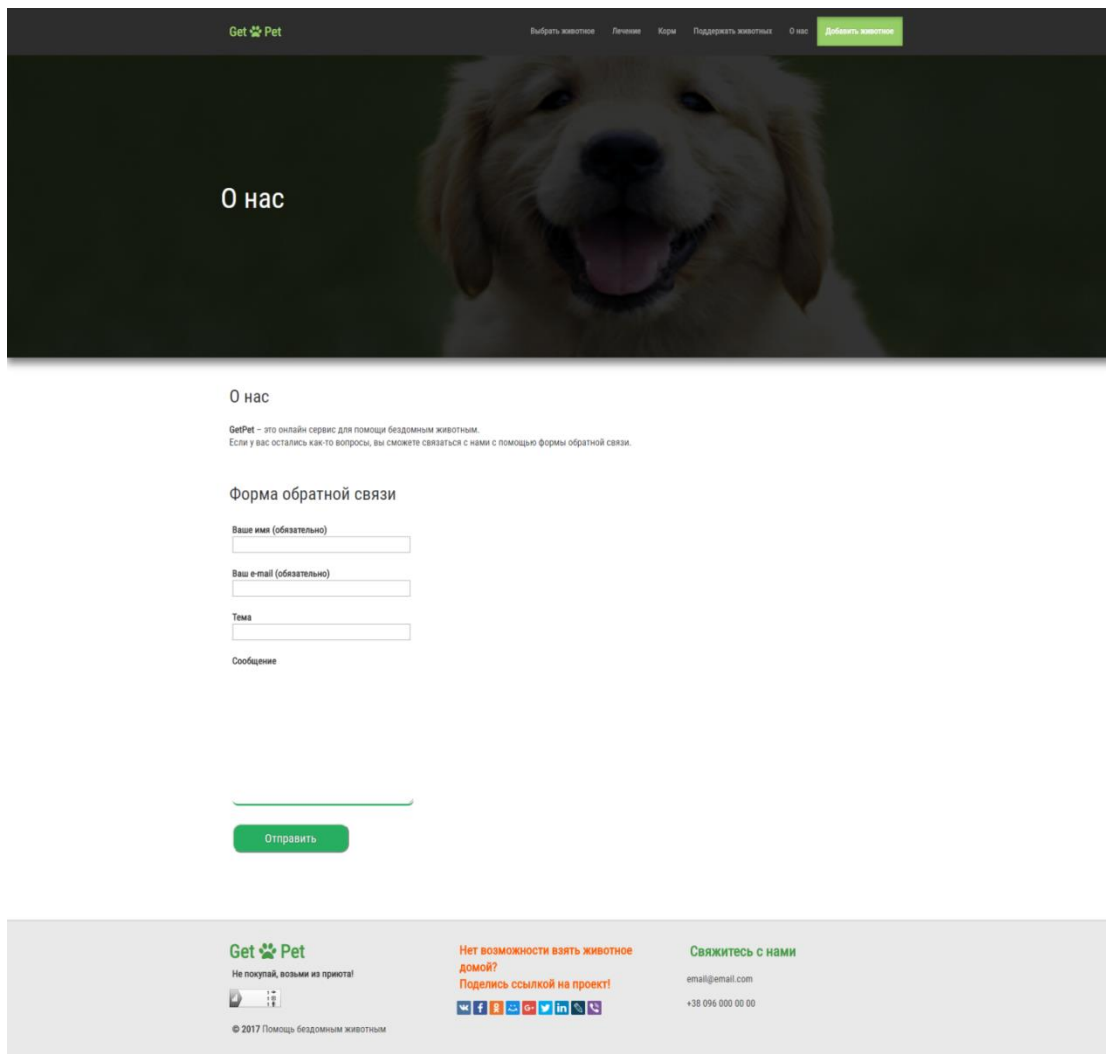


Рисунок 4. 8 – Сторінка «О нас»

На даній сторінці відображається коротка інформація про онлайн-сервіс «GetPet», а також форма зворотного зв'язку, через яку користувач зможе зв'язатися з адміністрацією сайту з будь-якого виниклого питання.

ВИСНОВКИ

За допомогою системи управління вмістом WordPress і Unyson фреймворком, був створений онлайн-сервіс допомоги тваринам GetPet. Даний сервіс дозволить людям швидко розмістити оголошення про зникнення тварини, а також, найголовніше, допоможе бездомним тваринам знайти свого власника.

Додаток може функціонувати в операційних системах будь-якого сімейства, що можуть підключатись до мережі Інтернет.

Завдяки використанню різних серверних і призначених для користувача елементів управління, застосування каскадних таблиць стилів і тем оформлення, вдалося створити привабливий, функціональний, зручний і інтуїтивно-зрозумілий інтерфейс, з можливістю швидкої модернізації в разі потреби.

Даний онлайн-сервіс дозволяє:

- 1) Вибрати вподобану тварину з будь-якої категорії;
- 2) Дати оголошення про зникнення своєї тварини;
- 3) Дати оголошення про знаходження загубленої тварини;
- 4) Швидко знайти ветеринарні клініки з відображенням їх на карті;
- 5) Отримати доступ до всієї потрібної інформації для майбутнього власника тварини;
- 6) При виникненні різних питань зв'язатися з адміністрацією сервісу за допомогою форми зворотного зв'язку;

В результаті дипломного проекту був створений онлайн-сервіс, в якому була реалізована автоматизована система обліку тварин.

ПЕРЕЛІК ПОСИЛАНЬ

1. GladPet [Электронный ресурс]. Режим доступа: <http://gladpet.org>
2. Общественная организация «Союз защиты животных» [Электронный ресурс]. Режим доступа: <http://kovcheg-odessa.od.ua>
3. Приют для бездомных животных Сириус [Электронный ресурс]. Режим доступа: <http://www.dogcat.com.ua>
4. Joomla – Find Your Database Name | mixeduperic.com [Электронный ресурс]. Режим доступа: <http://mixeduperic.com/open-source-software/joomla-find-your-database-name.html>.
5. CMS DataLife Engine – Система управления сайтами [Электронный ресурс]. Режим доступа: <https://dle-news.ru/release/page,1,3,1601-datalife-engine-v102-final-release.html>.
6. Web-icon.com.ua – сайт для Вашего бизнеса [Электронный ресурс]. Режим доступа: <http://web-icon.com.ua/articles/web-programming/twitter-bootstrap-преимущества-для-разработчика>.
7. Диго, С.М. Проектирование и эксплуатация баз данных – М.: Финансы и статистика, 1995. – 280 с.
8. Томсон Л. Разработка Web-приложений на PHP и MySQL /Л. Томсон, Л. Веллинг. – 2-е изд., испр. – СПб: ООО «ДиаСофт», 2003. – 672 с.
9. Программное обеспечение для компьютерных систем и сетей. Разработка технического задания / В.И. Дужий, А.В. Волковой, А.А. Волковая і ін. – Учеб. пособие.– Харьков: Нац. аэрокосм. ун-т «Харьк. авиац. ин-т», 2007. – 101с.
10. Воройский, Ф.С. Основные технологические принципы разработки электронных библиотек гуманитарных вузов России – М.:ТНБ – 2008.
11. Вікіпедія, вільна енциклопедія [Электронный ресурс]. Режим доступа: <https://uk.wikipedia.org/wiki/>.
12. Грачев, А. Создаем свой сайт на WordPress: работа с CMS WordPress 3 / А. Грачев. – СПб: Питер, 2011. – 282 с.

ДОДАТКИ

ДОДАТОК А КОД ГОЛОВНОЇ СТОРІНКИ САЙТУ

```

<!DOCTYPE html>
<html lang="ru-RU">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<link rel="profile" href="http://gmpg.org/xfn/11">

<!-- Bootstrap CSS & JS -->

<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.4/css/bootstrap.min.css">

<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.4/css/bootstrap-theme.min.css">

<!-- FontAwesome -->

<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/font-awesome/4.5.0/css/font-awesome.min.css">

<link rel="stylesheet" href="http://getpet.pp.ua/wp-content/themes/getpet/css/getpet.css">

<title>GetPet &#8211; Не покупай, возьми из приюта!</title>

<link rel='dns-prefetch' href='//maps.googleapis.com' />

<link rel='dns-prefetch' href='//s.w.org' />

<link rel="alternate" type="application/rss+xml" title="GetPet &raquo; Лента" href="http://getpet.pp.ua/feed/"
/>

<link rel="alternate" type="application/rss+xml" title="GetPet &raquo; Лента комментариев"
href="http://getpet.pp.ua/comments/feed/" />

<script type="text/javascript">

        window._wpemojiSettings =
        { "baseUrl": "https://s.w.org/images/core/emoji/2.2.1/72x72/", "ext": ".png", "svgUrl": "https://s.w.org/images/c
ore/emoji/2.2.1/svg/", "svgExt": ".svg", "source": { "concatemoji": "http://getpet.pp.ua/wp-includes/js/wp-emoji-
release.min.js?ver=4.7.5" } };

        !function(a,b,c){function d(a){var
b,c,d,e,f=String.fromCharCode;if(!k||!k.fillText)return!1;switch(k.clearRect(0,0,j.width,j.height),k.textBaseline="top
",k.font="600 32px Arial",a){case "flag":return
k.fillText(f(55356,56826,55356,56819),0,0),!(j.toDataURL().length<3e3)&&(k.clearRect(0,0,j.width,j.height),k.fill
Text(f(55356,57331,65039,8205,55356,57096),0,0),b=j.toDataURL(),k.clearRect(0,0,j.width,j.height),k.fillText(f(5
5356,57331,55356,57096),0,0),c=j.toDataURL(),b!==c);case "emoji4":return
k.fillText(f(55357,56425,55356,57341,8205,55357,56507),0,0),d=j.toDataURL(),k.clearRect(0,0,j.width,j.height),k.
fillText(f(55357,56425,55356,57341,55357,56507),0,0),e=j.toDataURL(),d!==e}return!1}function e(a){var
c=b.createElement("script");c.src=a,c.defer=c.type="text/javascript",b.getElementsByTagName("head")[0].appendC

```



```

hild(c)}var
f,g,h,i,j=b.createElement("canvas"),k=j.getContext&&j.getContext("2d");for(i=Array("flag","emoji4"),c.supports={
everything:!0,everythingExceptFlag:!0},h=0;h<i.length;h++)c.supports[i[h]]=d(i[h]),c.supports.everything=c.support
ts.everything&&c.supports[i[h]],"flag"!==i[h]&&(c.supports.everythingExceptFlag=c.supports.everythingExceptFla
g&&c.supports[i[h]]);c.supports.everythingExceptFlag=c.supports.everythingExceptFlag&&!c.supports.flag,c.DO
MReady=!1,c.readyCallback=function(){c.DOMReady=!0},c.supports.everything||(g=function(){c.readyCallback()
},b.addEventListener?(b.addEventListener("DOMContentLoaded",g,!1),a.addEventListener("load",g,!1)):(a.attachE
vent("onload",g),b.attachEvent("onreadystatechange",function(){ "complete"===b.readyState&&c.readyCallback()
})),f=c.source||{}},f.concatemoji?e(f.concatemoji):f.wpemoji&&f.twemoji&&(e(f.twemoji),e(f.wpemoji)))}(window,d
ocument>window._wpemojiSettings);

```

```
</script>
```

```
<style type="text/css">
```

```
img.wp-smiley,
```

```
img.emoji {
```

```
display: inline !important;
```

```
border: none !important;
```

```
box-shadow: none !important;
```

```
height: 1em !important;
```

```
width: 1em !important;
```

```
margin: 0 .07em !important;
```

```
vertical-align: -0.1em !important;
```

```
background: none !important;
```

```
padding: 0 !important;
```

```
}
```

```
</style>
```

```
<link rel='stylesheet' id='contact-form-7-css' href='http://getpet.pp.ua/wp-content/plugins/contact-form-
7/includes/css/styles.css?ver=4.8' type='text/css' media='all' />
```

```
<link rel='stylesheet' id='pirate_forms_front_styles-css' href='http://getpet.pp.ua/wp-content/plugins/pirate-
forms/css/front.css?ver=4.7.5' type='text/css' media='all' />
```

```
<link rel='stylesheet' id='getpet-style-css' href='http://getpet.pp.ua/wp-
content/themes/getpet/style.css?ver=4.7.5' type='text/css' media='all' />
```

```
<link rel='stylesheet' id='getpet-getpet-style-css' href='http://getpet.pp.ua/wp-
content/themes/getpet/css/getpet.css?ver=4.7.5' type='text/css' media='all' />
```

```
<link rel='stylesheet' id='getpet-getpet-media-css' href='http://getpet.pp.ua/wp-content/themes/getpet/css/media.css?ver=4.7.5' type='text/css' media='all' />
```

```
<link rel='stylesheet' id='fw-ext-builder-frontend-grid-css' href='http://getpet.pp.ua/wp-content/plugins/unyson/framework/extensions/builder/static/css/frontend-grid.css?ver=1.2.9' type='text/css' media='all' />
```

```
<link rel='stylesheet' id='fw-ext-forms-default-styles-css' href='http://getpet.pp.ua/wp-content/plugins/unyson/framework/extensions/forms/static/css/frontend.css?ver=2.6.16' type='text/css' media='all' />
```

```
<link rel='stylesheet' id='fw-shortcode-section-background-video-css' href='http://getpet.pp.ua/wp-content/plugins/unyson/framework/extensions/shortcodes/shortcodes/section/static/css/background.css?ver=4.7.5' type='text/css' media='all' />
```

```
<link rel='stylesheet' id='fw-shortcode-section-css' href='http://getpet.pp.ua/wp-content/plugins/unyson/framework/extensions/shortcodes/shortcodes/section/static/css/styles.css?ver=4.7.5' type='text/css' media='all' />
```

```
<link rel='stylesheet' id='fw-shortcode-button-css' href='http://getpet.pp.ua/wp-content/plugins/unyson/framework/extensions/shortcodes/shortcodes/button/static/css/styles.css?ver=4.7.5' type='text/css' media='all' />
```

```
<link rel='stylesheet' id='fw-shortcode-divider-css' href='http://getpet.pp.ua/wp-content/plugins/unyson/framework/extensions/shortcodes/shortcodes/divider/static/css/styles.css?ver=4.7.5' type='text/css' media='all' />
```

```
<link rel='stylesheet' id='font-awesome-css' href='http://getpet.pp.ua/wp-content/plugins/unyson/framework/static/libs/font-awesome/css/font-awesome.min.css?ver=2.6.16' type='text/css' media='all' />
```

```
<link rel='stylesheet' id='fw-shortcode-icon-box-css' href='http://getpet.pp.ua/wp-content/plugins/unyson/framework/extensions/shortcodes/shortcodes/icon-box/static/css/styles.css?ver=4.7.5' type='text/css' media='all' />
```

```
<link rel='stylesheet' id='fw-shortcode-map-css' href='http://getpet.pp.ua/wp-content/plugins/unyson/framework/extensions/shortcodes/shortcodes/map/static/css/styles.css?ver=4.7.5' type='text/css' media='all' />
```

```
<script type='text/javascript' src='http://getpet.pp.ua/wp-includes/js/jquery/jquery.js?ver=1.12.4'></script>
```

```
<script type='text/javascript' src='http://getpet.pp.ua/wp-includes/js/jquery/jquery-migrate.min.js?ver=1.4.1'></script>
```

```
<script type='text/javascript'>
```

```
/*  */</pre>
</div>
<div data-bbox="168 795 436 810" data-label="Text">
<pre>var pirateFormsObject = {"errors":""};</pre>
</div>
<div data-bbox="168 827 230 842" data-label="Text">
<pre>/* ]]&gt; */</pre>
</div>
<div data-bbox="168 860 235 875" data-label="Text">
<pre>&lt;/script&gt;</pre>
</div>
<div data-bbox="138 892 816 921" data-label="Text">
<pre>&lt;script type='text/javascript' src='http://getpet.pp.ua/wp-content/plugins/pirate-forms/js/scripts-general.js?ver=4.7.5'&gt;&lt;/script&gt;</pre>
</div>
```

```

<link rel='https://api.w.org/' href='http://getpet.pp.ua/wp-json/' />

<link rel="EditURI" type="application/rsd+xml" title="RSD" href="http://getpet.pp.ua/xmlrpc.php?rsd" />

<link rel="wlwmanifest" type="application/wlwmanifest+xml" href="http://getpet.pp.ua/wp-includes/wlwmanifest.xml" />

<meta name="generator" content="WordPress 4.7.5" />

<link rel="canonical" href="http://getpet.pp.ua/" />

<link rel='shortlink' href='http://getpet.pp.ua/' />

<link rel="alternate" type="application/json+oembed" href="http://getpet.pp.ua/wp-json/oembed/1.0/embed?url=http%3A%2F%2Fgetpet.pp.ua%2F" />

<link rel="alternate" type="text/xml+oembed" href="http://getpet.pp.ua/wp-json/oembed/1.0/embed?url=http%3A%2F%2Fgetpet.pp.ua%2F&#038;format=xml" />

<link rel="icon" href="http://getpet.pp.ua/wp-content/uploads/2017/03/cropped-favicon-32x32.jpg" sizes="32x32" />

<link rel="icon" href="http://getpet.pp.ua/wp-content/uploads/2017/03/cropped-favicon-192x192.jpg" sizes="192x192" />

<link rel="apple-touch-icon-precomposed" href="http://getpet.pp.ua/wp-content/uploads/2017/03/cropped-favicon-180x180.jpg" />

<meta name="msapplication-TileImage" content="http://getpet.pp.ua/wp-content/uploads/2017/03/cropped-favicon-270x270.jpg" />

</head>

<body class="home page-template-default page page-id-55">

<div class="navbar navbar-inverse navbar-fixed-top">

  <div class="container">

    <div class="navbar-header">

      <button type="button" class="navbar-toggle" data-toggle="collapse" data-target=".navbar-collapse">

        <span class="icon-bar"></span>

        <span class="icon-bar"></span>

        <span class="icon-bar"></span>

      </button>

```

```

        <a href="http://getpet.pp.ua" class="navbar-brand">Get <i class="fa fa-paw" aria-
hidden="true"></i> Pet</a>

    </div>

    <div class="navbar-collapse collapse">

        <ul class="nav navbar-nav navbar-right">

            <li><a href="http://getpet.pp.ua/выбрать-животное/">Выбрать
животное</a></li>

            <li><a href="http://getpet.pp.ua/лечение/">Лечение</a></li>

            <li><a href="http://getpet.pp.ua/корм/">Корм</a></li>

            <li><a href="http://getpet.pp.ua/поддержать-животных/">Поддержать
животных</a></li>

            <li><a href="http://getpet.pp.ua/о-нас/">О нас</a></li>

            <li class="active"><a href="http://getpet.pp.ua/добавить-
животное/">Добавить животное</a></li>

        </ul>

    </div>

</div>

</div>

</div>

<div id="page" class="site">

    <a class="skip-link screen-reader-text" href="#content">Skip to content</a>

    <header id="masthead" class="site-header" role="banner">

</header><!-- #masthead -->

    <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.4/js/bootstrap.min.js"></script>

    <div id="content" class="site-content">

        <div id="primary" class="content-area">

            <main id="main" class="site-main" role="main">

<div class="fw-page-builder-content"><section class="fw-main-row main-section" >

    <div class="fw-container-fluid">

        <div class="fw-row">

```

```

<div class="fw-col-xs-12 fw-col-sm-4">

</div>

<div class="fw-col-xs-12 fw-col-sm-8">

  <h1 class="main-title"><strong>ОН</strong> ждет своего хозяина!</h1><a href="#" target="_blank"
class="fw-btn fw-btn-1 main-button">

    <span>Выбрать животное</span>

  </a></div>

</div>

</section>

<section class="fw-main-row box-icons" >

  <div class="fw-container-fluid">

    <div class="fw-row">

      <div class="fw-col-xs-12">

        <div class="fw-divider-space" style="padding-top: 50px;"></div>

      </div>

    </div>

    <div class="fw-row">

      <div class="fw-col-xs-12">

        <p><center></p><h2 class="main-title">Почему стоит принять питомца?</h2><p></center></p></div>

      </div>

    <div class="fw-row">

      <div class="fw-col-xs-12 fw-col-sm-15">

        <div class="fw-iconbox clearfix fw-iconbox-1">

          <div class="fw-iconbox-image">

            <i class="fa fa-check"></i>

```

```

</div>

<div class="fw-iconbox-aside">

    <div class="fw-iconbox-title">

        <h3>спасти жизнь</h3>

    </div>

    <div class="fw-iconbox-text">

        <p>Чем больше людей будут отдавать предпочтение животным, потерявшим дом,
        тем больше спасенных жизней окажется.</p>

    </div>

</div>

</div></div>

<div class="fw-col-xs-12 fw-col-sm-15">

    <div class="fw-iconbox clearfix fw-iconbox-1">

        <div class="fw-iconbox-image">

            <i class="fa fa-check"></i>

        </div>

        <div class="fw-iconbox-aside">

            <div class="fw-iconbox-title">

                <h3>ответственность</h3>

            </div>

            <div class="fw-iconbox-text">

                <p>Котенок или щенок - это отличный шанс подготовить себя к новым
                заботам!</p>

            </div>

        </div>

    </div>

</div></div>

<div class="fw-col-xs-12 fw-col-sm-15">

```

```

<div class="fw-iconbox clearfix fw-iconbox-1">

<div class="fw-iconbox-image">

    <i class="fa fa-check"></i>

</div>

<div class="fw-iconbox-aside">

    <div class="fw-iconbox-title">

        <h3>активность</h3>

    </div>

    <div class="fw-iconbox-text">

        <p>Прогулка с собакой - это прекрасный шанс заняться фитнесом или утренней
пробежкой.</p>

    </div>

</div>

</div></div>

<div class="fw-col-xs-12 fw-col-sm-15">

<div class="fw-iconbox clearfix fw-iconbox-1">

<div class="fw-iconbox-image">

    <i class="fa fa-check"></i>

</div>

<div class="fw-iconbox-aside">

    <div class="fw-iconbox-title">

        <h3>отвлечение</h3>

    </div>

    <div class="fw-iconbox-text">

        <p>Столкнувшись с серьезными переменами, необходимо себя отвлечь.</p>

    </div>

</div>

```

```

</div></div>

<div class="fw-col-xs-12 fw-col-sm-15">

  <div class="fw-iconbox clearfix fw-iconbox-1">

    <div class="fw-iconbox-image">

      <i class="fa fa-check"></i>

    </div>

    <div class="fw-iconbox-aside">

      <div class="fw-iconbox-title">

        <h3>любовь</h3>

      </div>

      <div class="fw-iconbox-text">

        <p>Если питомец был взят на улице, то он наверняка не забудет этого до конца
жизни и останется верным и преданным другом для своих хозяев.</p>

      </div>

    </div>

  </div>

</div></div>

</div>

<div class="fw-row">

  <div class="fw-col-xs-12">

    <div class="fw-divider-space" style="padding-top: 100px;"></div>

  </div>

</div>

</div>

<div class="fw-container">

  <div class="fw-row">

```



```

<div class="fw-col-xs-12">

<div class="fw-divider-space" style="padding-top: 50px;"></div>

</div>

</div>

<div class="fw-row">

  <div class="fw-col-xs-12">

    <h2 class="main-title-second">Ты можешь сделать их такими</h2></div>

  </div>

</div>

<div class="fw-row">

  <div class="fw-col-xs-12 fw-col-sm-4">

    <p></p></div>

  <div class="fw-col-xs-12 fw-col-sm-4">

    <p></p></div>

  <div class="fw-col-xs-12 fw-col-sm-4">

    <p></p></div>

  </div>

</div>

<div class="fw-row">

  <div class="fw-col-xs-12">

    <div class="fw-divider-space" style="padding-top: 50px;"></div>

  </div>

</div>

<div class="fw-row">

  <div class="fw-col-xs-12 fw-col-sm-4">

    <p></p></div>

  <div class="fw-col-xs-12 fw-col-sm-4">

```

```

<p></p></div>

<div class="fw-col-xs-12 fw-col-sm-4">

  <p></p></div>

</div>

<div class="fw-row">

  <div class="fw-col-xs-12">

    <div class="fw-divider-space" style="padding-top: 50px;"></div>

  </div>

</div>

</div>

</div>

</section>

<section class="fw-main-row fw-container-fluid map-section" >

  <div class="fw-container-fluid">

    <div class="fw-row">

      <div class="fw-col-xs-12">

        <h2 class="main-title">Где можно забрать питомца?</h2></div>

      </div>

      <div class="fw-row">

        <div class="fw-col-xs-12">

          <div class="fw-map" data-
locations="{&quot;title&quot;:&quot;\u041d\u0435\u043c\u0435\u0446\u043a\u0438\u0439
\u041f\u0440\u0438\u044e\u0442 \u0414\u043b\u044f
\u0416\u0438\u0432\u043e\u0442\u0434\u044b\u0445&quot;,&quot;url&quot;:&quot;https://www.google.com.u
a/maps/place/%D0%9D%D0%B5%D0%BC%D0%B5%D1%86%D0%BA%D0%B8%D0%B9+%D0%9F%D1%
80%D0%B8%D1%8E%D1%82+%D0%94%D0%BB%D1%8F+%D0%96%D0%B8%D0%B2%D0%BE%D1%82
%D0%BD%D1%8B%D1%85/@46.4330656,30.6606445,17z/data=!3m1!4b1!4m5!3m4!1s0x40c632f29c5c0d87:0
xe7a32be66b598c78!8m2!3d46.4330656!4d30.6628332?hl=ru&quot;,&quot;thumb&quot;:false,&quot;coordinates
&quot;:{&quot;lat&quot;:46.5103872,&quot;lng&quot;:30.7162059},&quot;description&quot;:&quot;&quot;}"
data-map-type="ROADMAP" data-map-height="600" data-gmap-key="AIzaSyAYrQzGboCwzJbFJbB7BxhPw-
XKmOySdwo" data-disable-scrolling="" >

          <div class="fw-map-canvas"></div>

        </div></div>

```

```

</div>

</div>

</section>

<section class="fw-main-row fw-container-fluid feedback-section" >

  <div class="fw-container-fluid">

    <div class="fw-row">

      <div class="fw-col-xs-12">

        <div class="fw-divider-space" style="padding-top: 50px;"></div>

      </div>

    </div>

    <div class="fw-row">

      <div class="fw-col-xs-12">

        <h2 class="feedback-title">Остались вопросы? Свяжитесь с нами!</h2></div>

      </div>

    </div>

    <div class="fw-row">

      <div class="fw-col-xs-12">

        <div class="feedback-content"><div role="form" class="wpcf7" id="wpcf7-f61-p55-o1" lang="ru-RU"
dir="ltr">

          <div class="screen-reader-response"></div>

          <form action="/#wpcf7-f61-p55-o1" method="post" class="wpcf7-form" novalidate="novalidate">

            <div style="display: none;">

              <input type="hidden" name="_wpcf7" value="61" />

              <input type="hidden" name="_wpcf7_version" value="4.8" />

              <input type="hidden" name="_wpcf7_locale" value="ru_RU" />

              <input type="hidden" name="_wpcf7_unit_tag" value="wpcf7-f61-p55-o1" />

              <input type="hidden" name="_wpcf7_container_post" value="55" />

              <input type="hidden" name="_wpcf7_nonce" value="bacb961fe9" />

```

```
</div>
```

```
<p><label> Ваше имя <span style="color: red;">*</span><br />
```

```
  <span class="wpcf7-form-control-wrap your-name"><input type="text" name="your-name" value=""
size="40" class="wpcf7-form-control wpcf7-text wpcf7-validates-as-required" aria-required="true" aria-
invalid="false" /></span> </label></p>
```

```
<p><label> Ваш e-mail <span style="color: red;">*</span><br />
```

```
  <span class="wpcf7-form-control-wrap your-email"><input type="email" name="your-email" value=""
size="40" class="wpcf7-form-control wpcf7-text wpcf7-email wpcf7-validates-as-required wpcf7-validates-as-
email" aria-required="true" aria-invalid="false" /></span> </label></p>
```

```
<p><label> Тема<br />
```

```
  <span class="wpcf7-form-control-wrap your-subject"><input type="text" name="your-subject" value=""
size="40" class="wpcf7-form-control wpcf7-text" aria-invalid="false" /></span> </label></p>
```

```
<p><label> Сообщение<br />
```

```
  <span class="wpcf7-form-control-wrap your-message"><textarea name="your-message" cols="40"
rows="10" class="wpcf7-form-control wpcf7-textarea" aria-invalid="false"></textarea></span> </label></p>
```

```
<p><input type="submit" value="Отправить" class="wpcf7-form-control wpcf7-submit" /></p>
```

```
<div class="wpcf7-response-output wpcf7-display-none"></div></form></div></div></div>
```

```
</div>
```

```
<div class="fw-row">
```

```
  <div class="fw-col-xs-12">
```

```
    <div class="fw-divider-space" style="padding-top: 50px;"></div>
```

```
</div>
```

```
</div>
```

```
  </div>
```

```
</section>
```

```
</div>
```

```
</main><!-- #main -->
```

```
</div><!-- #primary -->
```

```
</div><!-- #content -->
```

```
<footer class="footer">
```

```

<div class="container">

  <div class="row">

    <div class="col-sm-4">

      <div class="footer-logo">

        <a href="http://getpet.pp.ua">Get <i class="fa fa-paw" aria-hidden="true"></i> Pet</a>

      </div>

      <p><b>Не покупай, возьми из приюта!</b></p>

      <!-- Yandex.Metrika informer -->

        <a href="https://metrika.yandex.ua/stat/?id=43378579&from=informer" target="_blank"
rel="nofollow"></a>

      <!-- /Yandex.Metrika informer -->

      <!-- Yandex.Metrika counter -->

      <script type="text/javascript">

        (function (d, w, c) {

          (w[c] = w[c] || []).push(function () {

            try {

              w.yaCounter43378579 = new Ya.Metrika({

                id: 43378579,

                clickmap: true,

                trackLinks: true,

                accurateTrackBounce: true,

                webvisor: true

              });

```

```

        } catch (e) {}

    });

    var n = d.getElementsByTagName("script")[0],

        s = d.createElement("script"),

        f = function () {

            n.parentNode.insertBefore(s, n);

        };

    s.type = "text/javascript";

    s.async = true;

    s.src = "https://mc.yandex.ru/metrika/watch.js";

    if (w.opera == "[object Opera]") {

        d.addEventListener("DOMContentLoaded", f, false);

    } else {

        f();

    }

})(document, window, "yandex_metrika_callbacks");

</script>

<noscript>

    <div></div>

</noscript>

<!-- /Yandex.Metrika counter -->

<br><br>

<p><b>© 2017</b> Помощь бездомным животным</p>

</div>

```



```
<script type='text/javascript' src='http://getpet.pp.ua/wp-content/themes/getpet/js/skip-link-focus-fix.js?ver=20151215'></script>
```

```
<script type='text/javascript' src='http://getpet.pp.ua/wp-content/plugins/unyson/framework/extensions/shortcodes/shortcodes/section/static/js/core.js?ver=4.7.5'></script>
```

```
<script type='text/javascript' src='http://getpet.pp.ua/wp-content/plugins/unyson/framework/extensions/shortcodes/shortcodes/section/static/js/transition.js?ver=4.7.5'></script>
```

```
<script type='text/javascript' src='http://getpet.pp.ua/wp-content/plugins/unyson/framework/extensions/shortcodes/shortcodes/section/static/js/background.js?ver=4.7.5'></script>
```

```
<script type='text/javascript' src='http://getpet.pp.ua/wp-content/plugins/unyson/framework/extensions/shortcodes/shortcodes/section/static/js/background.init.js?ver=4.7.5'></script>
```

```
<script type='text/javascript' src='https://maps.googleapis.com/maps/api/js?v=3.25&#038;language=ru&#038;libraries=places&#038;key=AIzaSyAYrQzGboCwzJbFJbB7BxhPw-XXKmOySdwo&#038;ver=3.25'></script>
```

```
<script type='text/javascript' src='http://getpet.pp.ua/wp-includes/js/underscore.min.js?ver=1.8.3'></script>
```

```
<script type='text/javascript' src='http://getpet.pp.ua/wp-content/plugins/unyson/framework/extensions/shortcodes/shortcodes/map/static/js/scripts.js?ver=2.6.16'></script>
```

```
<script type='text/javascript' src='http://getpet.pp.ua/wp-includes/js/wp-embed.min.js?ver=4.7.5'></script>
```

```
<script src="//yastatic.net/es5-shims/0.0.2/es5-shims.min.js"></script>
```

```
<script src="//yastatic.net/share2/share.js"></script>
```

```
</body>
```

```
</html>
```


ДОДАТОК Б ФОРМА ДОДАВАННЯ ТВАРИНИ

```

<form action="" method="post" enctype="multipart/form-data" name="add-pet">

<div class="row">

<div class="col-sm-12 add-pet-names">

<div class="col-sm-4">

<p id="category"><i id="categoryI" class="fa fa-arrow-right" style="display: none;"></i> Категорія:</p>

</div>

<div class="col-sm-4">

<select required name="category" class="form-control">

<option value="null"></option>

<option value="Потерянное животное">Потерянное животное</option>

<option value="Найденное животное">Найденное животное</option>

<option value="Из приюта">Из приюта</option>

</select>

</div>

</div>

<div class="col-sm-12 add-pet-names">

<div class="col-sm-4">

<p id="kind"><i id="kindI" class="fa fa-arrow-right" style="display: none;"></i> Вид:</p>

</div>

<div class="col-sm-4">

<select required name="kind" class="form-control">

<option value="null"></option>

<option value="Кошка">Кошка</option>

<option value="Собака">Собака</option>

</select>

```

```

</div>

</div>

<div class="col-sm-12 add-pet-names">

<div class="col-sm-4">

<p id="gender"><i id="genderI" class="fa fa-arrow-right" style="display: none;"></i> Пол:</p>

</div>

<div class="col-sm-4">

<select required name="gender" class="form-control">

<option value="null"></option>

<option value="Девочка">Девочка</option>

<option value="Мальчик">Мальчик</option>

</select>

</div>

</div>

<div class="col-sm-12 add-pet-names">

<div class="col-sm-4">

<p id="age"><i id="ageI" class="fa fa-arrow-right" style="display: none;"></i> Возраст:</p>

</div>

<div class="col-sm-4">

<input name="age" class="form-control" value="<? if (isset($_POST['age'])) echo $_POST['age'];
?>"></input>

</div>

</div>

</div>

</div>

<div class="row">

<div class="col-sm-12 add-pet-names">

<div class="col-sm-4">

```

```

<p id="breed"><i id="breedI" class="fa fa-arrow-right" style="display: none;"></i> Порода:</p>

</div>

<div class="col-sm-4">

  <input name="breed" class="form-control" value="<? if (isset($_POST['breed'])) echo $_POST['breed'];
?>"></input>

</div>

</div>

<div class="col-sm-12 add-pet-names">

<div class="col-sm-4">

<p id="color"><i id="colorI" class="fa fa-arrow-right" style="display: none;"></i> Окрас:</p>

</div>

<div class="col-sm-4">

  <input name="color" class="form-control" value="<? if (isset($_POST['color'])) echo $_POST['color'];
?>"></input>

</div>

</div>

<div class="col-sm-12 add-pet-names">

<div class="col-sm-4">

<p id="organization"><i id="organizationI" class="fa fa-arrow-right" style="display: none;"></i>
Организация:</p>

</div>

<div class="col-sm-4">

  <input name="organization" class="form-control" value="<? if (isset($_POST['organization'])) echo
$_POST['organization']; ?>"></input>

</div>

</div>

</div>

<div class="row">

<div class="col-sm-12 add-pet-names">

```

```

<div class="col-sm-4">

<p id="description"><i id="descriptionI" class="fa fa-arrow-right" style="display: none;"></i> Описание:</p>

</div>

<div class="col-sm-4">

<textarea name="description" class="form-control"><? if (isset($_POST['description'])) echo
$_POST['description']; ?></textarea>

</div>

</div>

<div class="col-sm-12 add-pet-names text-center">

<div class="col-sm-4">

<p id="image"><i id="imageI" class="fa fa-arrow-right" style="display: none;"></i> Фотография:</p>

</div>

<div class="col-sm-4">

<input name="image" type="file" class="form-control">

</div>

</div>

</div>

<div class="row">

<div class="col-sm-4">

<div class="checkbox">

<label><input name="isGraft" type="checkbox" value="0">Привит</label>

</div>

</div>

<div class="col-sm-4">

<div class="checkbox">

<label><input name="isSterilized" type="checkbox" value="0">Стерилизован</label>

</div>

```

```
</div>

<div class="col-sm-4">

<div class="checkbox">

<label><input name="isGrave" type="checkbox" value="0">В тяжелом состоянии</label>

</div>

</div>

</div>

<br>

<div class="row">

<div class="col-sm-2 col-xs-offset-4">

<button type="submit" name="send" class="btn btn-success btn-lg add-pet-button">Добавить
данные</button>

</div>

<div class="col-sm-2">

<button type="reset" name="send" class="btn btn-default btn-lg add-pet-button">Очистить форму</button>

</div>

</div>

</form>
```