

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ КІБЕРБЕЗПЕКИ, КОМП'ЮТЕРНОЇ
ТА ПРОГРАМНОЇ ІНЖЕНЕРІЇ
КАФЕДРА ПРИКЛАДНОЇ ІНФОРМАТИКИ

ДОПУСТИТИ ДО ЗАХИСТУ

Завідувач кафедри

_____ Гамаюн В.П.
(підпис) (ПІБ)

“ _____ ” _____ 2021р.

ДИПЛОМНИЙ ПРОЕКТ
(ПОЯСНЮВАЛЬНА ЗАПИСКА)

ВИПУСКНИКА ОСВІТНЬОГО СТУПЕНЯ “БАКАЛАВР”

Тема: Web-сайт ветеринарної клініки

Виконавець: _____ Куцель Євгенія Сергіївна
(підпис) (ПІБ)

Керівник: _____ Сінько Юрій Іванович
(підпис) (ПІБ)

Нормоконтролер: _____ Боровик Володимир Миколайович
(підпис) (ПІБ)

Київ 2021

НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ

Факультет кібербезпеки, комп'ютерної та програмної інженерії

Кафедра прикладної інформатики

Освітній ступень Бакалавр

Спеціальність 122 "Комп'ютерні науки"

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ Гамаюн В.П.
(підпис) (ПІБ)
" ____ " _____ 2021р.

ЗАВДАННЯ

на виконання дипломного проекту

Куцель Євгенії Сергіївни

(прізвище, ім'я, по батькові випускника в родовому відмінку)

1. Тема дипломного проекту: Web-сайт ветеринарної клініки

затверджена додатком до наказу ректора від " ____ " _____ 2021 р., № ____ /ст _____

2. Термін виконання проекту: з " ____ " _____ 2021 р. по " ____ " _____ 2021 р.

3. Вихідні дані до проекту: 1. Провести аналіз предметної області. 2. Здійснити проектування web-сайту ветеринарної клініки. 3. Здійснити реалізацію web-сайту ветеринарної клініки.

4. Зміст пояснювальної записки: 1. Аналіз предметної області. 2. Проектування web-сайту ветеринарної клініки. 3. Реалізація web-сайту ветеринарної клініки.

5. Перелік обов'язкового графічного (ілюстративного) матеріалу: 1. Діаграма варіантів використання. 2. Діаграми станів процесів програмної системи. 3. Діаграма розгортання програмної системи. 4. Структура web-сайту ветеринарної клініки. 5. Інтерфейс розділів меню. 6. Форми відправки даних.

6. Календарний план-графік

№ пор.	Завдання	Термін виконання	Відмітка про виконання
1.	Провести аналіз предметної області	10.04.21- 15.04.21	
2.	Здійснити проектування web-сайту ветеринарної клініки	16.04.21- 25.04.21	
3.	Здійснити реалізацію web-сайту ветеринарної клініки	01.05.21- 15.05.21	
4.	Оформити текстові і графічні матеріали дипломного проекту	17.05.21- 30.05.21	
5.	Підготувати презентацію по дипломному проекту	30.05.21- 31.05.21	

7. Дата видачі завдання: " ____ " _____ 2021 р.

Керівник дипломного проекту _____ Сінько Ю.І.
(підпис керівника) (ПІБ.)

Завдання прийняв до виконання _____ Куцель Є.С.
(підпис випускника) (ПІБ.)

РЕФЕРАТ

Пояснювальна записка до дипломного проекту «Web-сайт ветеринарної клініки», загальним обсягом 68 сторінок, 24 рисунки, 7 таблиць, 18 джерел літератури, 2 додатки.

ПРОГРАМНА СИСТЕМА, WEB-САЙТ, МЕТОДИ РОЗРОБКИ WEB-САЙТІВ, ЗАСОБИ РОЗРОБКИ WEB-САЙТІВ, HTML, CSS, JAVASCRIPT, PHP.

Об'єкт дослідження – діяльність ветеринарної клініки.

Предмет дослідження – засоби для створення web-сайту ветеринарної клініки.

Мета дипломного проекту – створення web-сайту для ветеринарної клініки.

Завдання дипломного проектування – провести аналіз предметної області. Здійснити проектування web-сайту ветеринарної клініки. Здійснити реалізацію web-сайту ветеринарної клініки.

Результат дипломного проектування полягає у розробці web-сайту ветеринарної клініки з повним його функціонуванням та взаємодії користувачів із ним.

ЗМІСТ

ПЕРЕЛІК ПРИЙНЯТИХ СКОРОЧЕНЬ.....	6
ВСТУП.....	7
РОЗДІЛ 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ	9
1.1 Постановка завдання	9
1.2 Класифікація web-сайтів	11
1.3 Відомі методи реалізації web-сайтів.....	13
1.4 Огляд та аналіз схожих web-сайтів.....	16
Висновки до розділу	20
РОЗДІЛ 2 ПРОЕКТУВАННЯ WEB-САЙТУ ВЕТЕРИНАРНОЇ КЛІНІКИ	21
2.1 Специфікація вимог до web-сайту	21
2.2 Розробка архітектури програмної системи.....	27
2.3 Розробка структури web-сайту.....	32
2.4 Засоби для створення web-сайту.....	34
Висновки до розділу	36
РОЗДІЛ 3 РЕАЛІЗАЦІЯ WEB-САЙТУ ВЕТЕРИНАРНОЇ КЛІНІКИ	37
3.1. Опис інтерфейсу web-сайту ветеринарної клініки.....	37
3.2. Програмна реалізація функцій web-сайту ветеринарної клініки	41
3.3. Завантаження web-сайту у мережу Інтернет	45
Висновки до розділу	46
ВИСНОВКИ	47
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	49
ДОДАТОК А	51
ДОДАТОК Б.....	55

Кафедра ПП				НАУ 21 13 61 – 000 ПЗ			
<i>Виконала</i>	<i>Куцель Є.С.</i>			Web-сайт ветеринарної клініки	Літ.	Арк.	Аркушів
<i>Керівник</i>	<i>Сінько Ю.І.</i>				5	1	
<i>Консульт.</i>					122 ТП-415Б		
<i>Нормоконт.</i>	<i>Боровик В.М.</i>						
<i>Зав. каф.</i>	<i>Гамаюн В.П.</i>						

ПЕРЕЛІК ПРИЙНЯТИХ СКОРОЧЕНЬ

ПЗ – програмне забезпечення

БД – база даних

HTML – Hyper Text Markup Language

CSS – Cascading Style Sheets

JS – JavaScript

CMS – Content Management System

UML – Unified Modeling Language

ВСТУП

Ветеринарні клініки – сфера послуг, яка ще довгий час не втратить своєї актуальності. Лікування домашніх тварин здійснюється з давніх-давен і продовжує існувати і по сьогоднішній день. Дана послуга не зміниться і в майбутньому. Зростаючий з кожним днем попит на дані послуги є показником того, що з кожним днем в будинках з'являються нові, чотириногі члени сім'ї. Домашнім улюбленцям, як і людям, часом, необхідна медична допомога. Кожен господар, прагнучи допомогти своєму вихованцеві, не економить на його здоров'ї і звертається в ветеринарні клініки. Для того, щоб полегшити пошук такої клініки створюються web-сайти з наданням необхідних послуг. Але, провівши дослідження, було виявлено, що сайтів за даною тематикою дуже мало, і якщо вони і існують, то їхній функціонал не настільки великий, щоб задовольняти вимогам клієнтів.

На сьогоднішній день неможливо уявити собі роботу підприємства або організації без свого власного сайту, адже саме він презентує вашу компанію і від нього залежить перша думка про якість роботи фірми.

В сучасному світі інформаційних технологій наявність власного сайту у підприємства є головним чинником його функціонування. Потенційні клієнти скоріше звернуть увагу на ветеринарну клініку, яка має власний сайт з усією необхідною та корисною інформацією про себе, оскільки саме достатня інформація та легкий доступ є запорукою припливу клієнтів, та підвищує відвідуваність клініки.

Тема розробки web-сайту для ветеринарної клініки є актуальною на сьогоднішній день, оскільки це підвищує ефективність роботи клініки та полегшує пошук допомоги власникам тварин у ветеринарній клініці для своїх улюбленців.

Метою даної роботи є створення web-сайту для ветеринарної клініки з повною інформацією про діяльність клініки, а також наданням користувачам достатнього функціоналу для взаємозв'язку з нею.

Для досягнення мети дипломної роботи головними етапами розробки web-сайту ветеринарної клініки є:

1. Аналіз предметної області;
2. Проектування web-сайту ветеринарної клініки;
3. Реалізація web-сайту ветеринарної клініки.

Відповідно до поставленої мети було виділено наступні завдання, які необхідно відтворити на сайті:

- наповнення сайту детальною інформацією;
- надання користувачу можливості записатися на прийом;
- здійснення пошуку послуг за ключовим словом;
- надання можливості залишити відгук;
- створення кнопки зворотнього зв'язку.

Об'єктом досліджень є ветеринарна клініка.

Предметом дослідження є засоби для створення web-сайту ветеринарної клініки.

РОЗДІЛ 1

АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Постановка завдання

Метою роботи є розробка та створення web-сайту для ветеринарної клініки, що має містити детальну інформацію про клініку, а також надавати користувачу можливість повністю взаємодіяти з ним.

Відповідно до поставленої мети, на сайті мають бути реалізовані наступні завдання:

- заповнити сайт необхідною інформацією;
- надати можливість потенційним клієнтам записатись на прийом;
- надати можливість користувачам залишити відгук;
- створити кнопку зворотнього зв'язку;
- реалізувати пошук послуг по ключовому слову.

Розглянемо детальніше вимоги до сайту, що розробляється.

Вимоги до наповненості сайту.

Web-сайт має містити необхідну для потенційних клієнтів інформацію. На головній сторінці сайту має бути розміщена основна інформація про клініку, зокрема назву клініки, основні напрямки роботи та послуги доступні клієнтам, а також головне меню, з можливістю переходу на потрібні сторінки, такі як контакти, послуги, галерея та відгуки про клініку. Також на головній сторінці клієнтам буде надана можливість записатись на прийом зі своїм домашнім улюбленцем.

У розділі Послуги користувачі зможуть відшукати необхідну послугу за допомогою пошуку, увівши ключове слово, для того щоб не ускладнювати собі роботу.

Кафедра ПП				НАУ 21 13 61 – 000 ПЗ			
Виконала	Куцель Є.С.			Аналіз предметної області	Літ.	Арк.	Аркушів
Керівник	Сінько Ю.І.					9	12
Консульт.					122 ТП-415Б		
Нормоконт.	Боровик В.М.						
Зав. каф.	Гамаюн В.П.						

У розділі Відгуки клієнти матимуть можливість залишити відгук про надані послуги, заповнивши відповідну форму.

У розділі Контакти клієнти знайдуть необхідну інформацію про місцезнаходження клініки з контактними телефонами, а також в цьому розділі буде створена кнопка для зворотнього зв'язку з клієнтом, де клієнти матимуть можливість залишити свій номер телефону та очікувати дзвінок від працівників клініки.

Вимоги до навігації сайту.

Графічний інтерфейс сайту має забезпечити наочне та інтуїтивно зрозуміле уявлення структури розміщеної на сайті інформації, а також забезпечити швидкий перехід до обраного розділу сайту. Меню навігації повинно бути зрозумілим для клієнтів, гіперпосилання на сторінки повинні мати зрозумілі заголовки. Для реалізації якісної роботи сайту має бути сформований блок, який складається зі списку гіперпосилань.

Вимоги до графічного оформлення.

Графічні елементи сайту повинні бути розроблені у відповідності до специфіки інформації, яка міститься на сторінці, а саме ветеринарна клініка. Основні елементи меню та навігації доступні на головній сторінці, а також не повинно бути надлишкової інформації на всіх сторінках сайту. Дизайн сайту повинен бути сучасним та лаконічним, а увага користувачів концентруватись на необхідній інформації. При створенні сайту варто уникати складно структурованої інформації, яка може бути важкою для сприйняття клієнтами, а також елементів, які можуть відволікати від потрібної інформації.

Загалом, сайт має бути зручним для використання, містити в собі необхідну для клієнтів інформацію, можливість взаємодіяти з сайтом, дизайн має бути лаконічним та сучасним по наповненню, а увага потенційних клієнтів фокусуватись на потрібній для них інформації та послугах.

1.2 Класифікація web-сайтів

На сьогоднішній день не існує єдиної класифікації типів сайтів. Зазвичай їх поділяють за задачами, що вирішуються сайтом. Загалом виділяють наступні варіанти класифікації:

- за доступністю сервісів;
- за природою змісту;
- за фізичним розташуванням;
- за задачею, що вирішується сайтом.

Розглянемо детальніше кожен з цих видів.

За доступністю сервісів сайти поділяють на відкриті, напіввідкриті та закриті:

- відкриті - це сайти, де всі сервіси повністю доступні для всіх відвідувачів та користувачів;
- напіввідкриті - для доступу до ресурсів чи сервісів сайту необхідна реєстрація, зазвичай безкоштовна;
- закриті - повністю закриті сайти організацій, наприклад корпоративні сайти або особисті сайти приватних осіб. Доступ, зазвичай, надається через запрошення, або ж інвайти, а також додавання адміністратором до списку допущених осіб.

За природою змісту сайти поділяють на статичні та динамічні:

- статичні сайти - це ті, зміст яких заздалегідь готується, а користувачу видаються файли в тому вигляді, в якому вони зберігаються на сервері. Тож, особливістю статичної веб-сторінки є те, що її зміст не змінюється в залежності від запиту, вона завжди залишається однаковою, якщо тільки її зміст не змінено на жорсткому диску сервера;
- динамічні сайти - це сайти, зміст яких генерується за допомогою спеціальних скриптів, або ж програм, а в основі лежать дані з будь-якого іншого джерела. Зміст сторінки постійно змінюється, в залежності від запиту користувача.

За фізичним розташуванням сайти поділяються на зовнішні та локальні:

- зовнішні сайти забезпечують доступ для будь-якого користувача мережі;

- локальні в свою чергу - доступні тільки в межах локальної мережі. До них можуть відноситись як корпоративні сайти організацій, так і сайти приватних осіб.

За задачею, що вирішується сайтом, поділяються на сайт-візитка, корпоративний сайт, Інтернет-магазин, промо-сайт, тематичний портал та веб-сервіси:

- сайт-візитка містить в собі загальні дані про власника сайту, його історію, вид діяльності, прайс-листи, контактні дані, реквізити або ж схему проїзду. Є повноцінним віртуальним ресурсом з унікальним дизайном та оформленням;

- корпоративний сайт містить в собі повну інформацію про компанію-власника, послуги або продукцію компанії, а також відображає події в житті компанії. Є обов'язковим інструментом для успішного ведення бізнесу будь-якою компанією. Від сайту-візитки його відрізняє повнота наданої інформації, наявність різних функціональних інструментів для роботи з контентом, зокрема пошук і фільтри, календар подій, фотогалерея, корпоративні блоги та форуми. Корпоративний сайт може бути інтегрований з внутрішніми інформаційними системами компанії-власника, а також містити закриті розділи для різних груп користувачів - співробітників, контрагентів, тощо;

- інтернет-магазин - це сайт з каталогом продукції, що полегшує клієнтам пошук та надає можливість замовити потрібні йому товари, обравши зручний для себе варіант розрахунку, зручний спосіб отримання товару та отримати рахунок на оплату. І найголовніше, що все це можна робити не встаючи з місця;

- промо-сайти містять інформацію про конкретну торгову марку або продукт, зазвичай на таких сайтах розміщують вичерпну інформацію про бренд або ж рекламні акції. Такий тип сайту слугує насамперед рекламним

інструментом, з метою рекламування конкретного товару чи послуги, або ж навіть якоїсь події;

– тематичний портал - великий веб-ресурс, який надає користувачу вичерпну інформацію з певної тематики, містить засоби взаємодії з користувачем та надає можливість користувачам спілкуватись в рамках порталу за допомогою форумів чи чатів;

– веб-сервіс є послугою, що створена для виконання будь-яких завдань в рамках мережі Інтернет, наприклад - дошка оголошень, каталог сайтів, пошукові сервіси, поштові сервіси, форуми, блоги, соціальні мережі та інші.

1.3 Відомі методи реалізації web-сайтів

В процесі перетворення Інтернету з набору інформаційних ресурсів в інструмент ведення бізнесу, технології створення сайтів істотно змінилися. На сьогоднішній день існує величезна кількість різних методів створення сайтів, які розрізняються в залежності від призначення і типу сайту, умінь і навичок розробника.

Кожен метод розробки сайту відрізняється один від одного і має свої особливості.

Перед тим, як створювати сайт необхідно передбачити багато деталей, і одним з головних етапів розробки web-сайту є вибір методу реалізації сайту, та правильно обрати той, за допомогою якого ви отримаєте свій перший сайт.

Можна виділити наступні основні методи створення web-сайтів:

- створення web-сайту за допомогою коду вручну;
- робота з редактором WYSIWYG;
- використання конструктора веб-сайтів;
- використання системи управління вмістом (CMS).

Створення web-сайту вручну.

Самостійна розробка сайту – це найкреативніший та вільний процес, але й трудомісткий. Написання такого сайту потребує серйозних вмінь та навичок по мовам програмування, розумінню архітектури, бізнес-процесів клієнта та

багато іншого. При цьому, створюючи сайт з нуля, клієнт отримає унікальний продукт, який буде вирішувати всі його поставлені задачі, та не витратити час на зайві процеси. Розробка сайту власноруч дозволяє створити проекти будь-якої складності та з будь-яких побажань клієнта.

Розробка сайту за допомогою редактора WYSIWYG.

WYSIWYG – «What You See Is What You Get» (з англ. – «Те що ти бачиш, це те що ти отримуєш»). Одним з відомих таких редакторів є Dreamweaver.

Ідея таких редакторів полягає в тому, що вони працюють більше як текстовий процесор або додаток для настільних видань, де ви працюєте у візуальному середовищі, тоді як програмне забезпечення пише у фоновому режимі весь необхідний код. Замість того, щоб працювати безпосередньо з кодом, користувачі можуть розробляти сайти за допомогою команд меню та діалогових вікон, зовсім не знаючи про фоновий код. Такий метод полегшує та пришвидшує нетехнічним користувачам створення веб-сайтів.

Створення сайту за допомогою конструктора.

Такий метод розробки сайту вважається одним з найпростіших, тому що для того, щоб створити web-сайт буде достатньо лише вміння користуватися браузером комп'ютера та мати почуття смаку.

Конструктор сайтів – це онлайн-сервіс, використовуючи який можна швидко створювати web-ресурси без знань основ програмування.

До переваг даного способу відносять:

1. Час створення. Завдяки конструктору сайт можна зробити за кілька годин, обравши шаблон та за потреби відредагувати його на свій погляд.
2. Низька ціна. Майже всі такі сервіси від самого початку безкоштовні, а вартість підключених модулів дуже низька.
3. Готова структура сайту. Користувачі можуть легко переміщатися по сторінкам сайту.

Недоліки цього методу є наступними:

1. Повільна швидкість завантаження. Часто конструктор генерує громіздкий HTML-код, який містить у собі багато зайвих рядків. У результаті чого, сторінка займає більшу пам'ять та через це повільно загрузається.

2. Не завжди є можливість відкоригувати код власноруч. Справа в тому, що він перенасичений тегами, різними стилями та іншими елементами. Програмісту легше зверстати новий сайт, ніж розбиратися у рядках величезного коду.

3. Структура та наповнення сайту контентом. Структура ресурсу обмежена. Ви маєте можливість створювати обмежену кількість сторінок, так як виділено невеликий об'єм пам'яті для зберігання файлів.

Створення сайту за допомогою (CMS).

Система управління вмістом (CMS) – це тип програмного забезпечення, наприклад такі, як WordPress, Joomla, Drupal, яке дозволяє створювати та керувати веб-сайтом. Вони не такі прості, як конструктори сайтів, але більш універсальні.

WordPress – це одна з найпопулярніших систем управління контентом, на якій створено більша частина сайтів в мережі Інтернет. Будь-яку CMS необхідно встановлювати самому, для цього необхідно завантажити з офіційного сайту архів останньої найновішої версії, зареєструвати хостинг (онлайн - послуга, яка дозволяє опублікувати веб-сайт або веб-додаток в Інтернет), залити завантажені файли на нього, створити БД, та все це завантажити.

Перевагами такого методу є те, що CMS надає доступ до тисяч шаблонів, які можна завантажити прямо з Адмін панелі вашого сайту. Також є можливість встановити сторонні плагіни, які надають можливість розширити функціонал, та за допомогою яких можливо провести повне налаштування сайту.

На основі CMS можливо створити як прості Landing Page, сайти-візитки, так і масштабні сайти для великих компаній.

Єдиним недоліком методу, що описується є потреба у мінімальних технічних знаннях, але з ростом популярності системи управління вмістом в Інтернеті існує багато учбового матеріалу (книги, відео уроки, довідники і т.д.).

1.4 Огляд та аналіз схожих web-сайтів

Одним з найперших кроків розробки певного сайту є здійснення пошуку його аналогів (схожих сайтів), їх огляд та аналіз з метою визначення переваг та недоліків для того, аби в подальшій розробці не допустити помилки, та використати переваги проаналізованих web-ресурсів.

Для аналізу було знайдено та обрано три web-сайти, які реалізують функції предметної області:

1. Web-сайт ветеринарного центру «Центр Современной Ветеринарной Медицины» (режим доступу: <http://csvm.com.ua/>).
2. Web-сайт ветеринарної клініки «А-Вет» (режим доступу: <https://www.avet.kiev.ua/>).
3. Web-сайт ветеринарної клініки «Ветмир» (режим доступу: <https://vetmir.com.ua/>).

Розглянемо детальніше представлені вище аналоги.

Першим розглянемо портал <http://csvm.com.ua/>. На рисунку 1.1 зображено вигляд головної сторінки веб-ресурсу <http://csvm.com.ua/>.

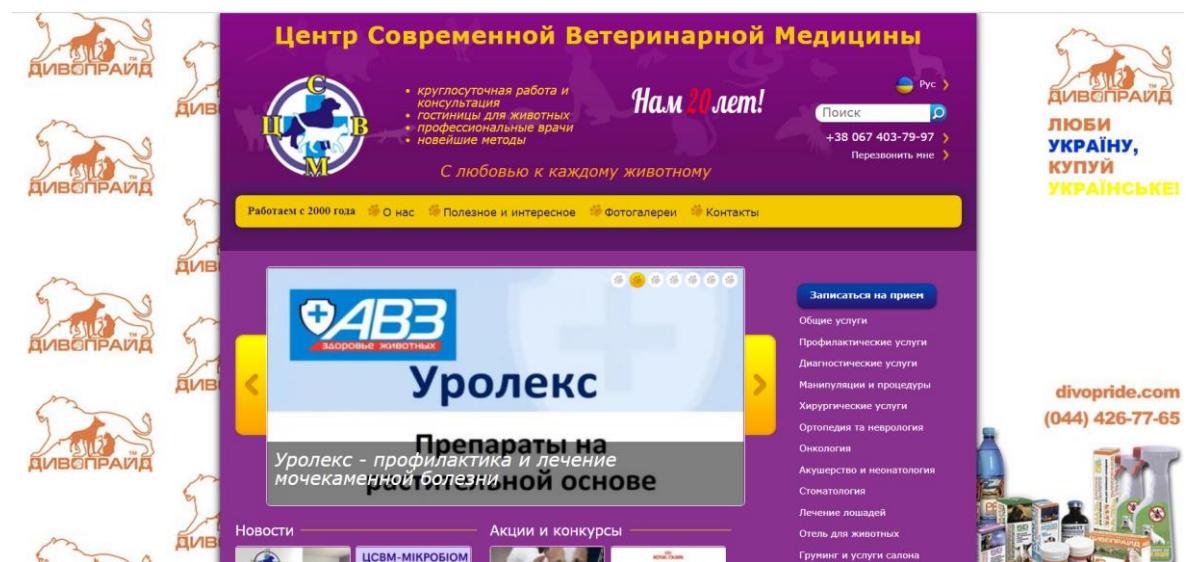


Рис. 1.1 Головна сторінка ветеринарного центру «Центр Современной Ветеринарной Медицины»

Перше, на що будуть звертати увагу клієнти – це дизайн сайту, адже саме від нього залежить те, чи залишаться користувачі на даній web-сторінці. Інтерфейс, що відображається на цьому веб-ресурсі не сучасний, має багато

елементів, які відволікають від головної наданої інформації, а головні кнопки, що відповідають за взаємодію клієнта із сайтом особливо не виділяються, через що важко їх знайти. Щодо функціональності web-сайту, є запис на прийом, існує можливість зворотнього зв'язку, можливо залишити відгук, і навіть існує пошук по сайту.

Наступним розглянемо web-сайт <https://www.avet.kiev.ua/> (див. рис. 1.2).

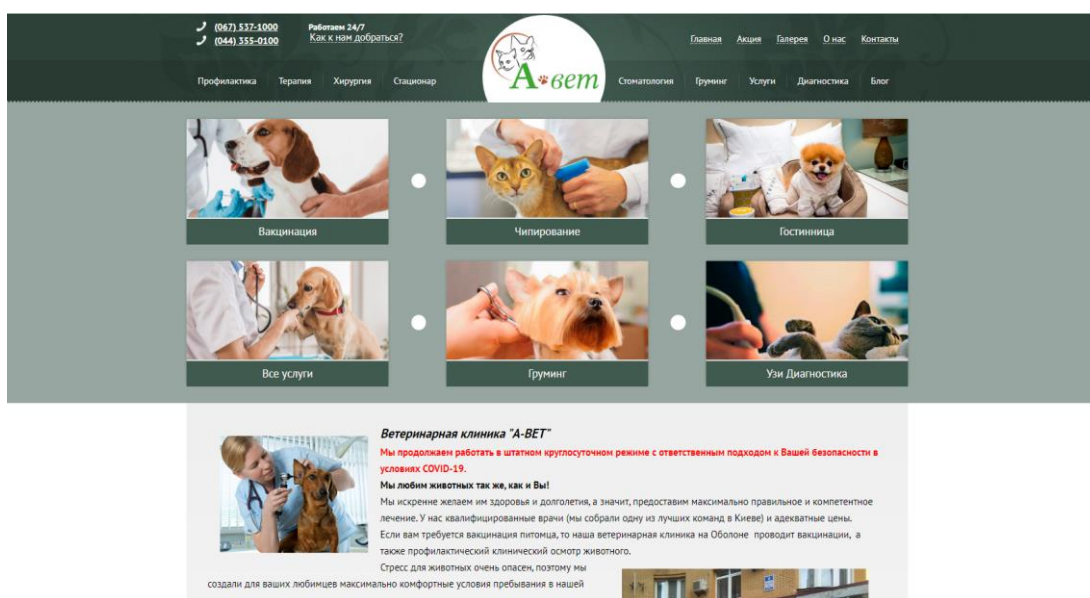


Рис.1.2 Головна сторінка сайту ветеринарної клініки «А-Вет»

Даний веб-ресурс має навігацію, яка представлена у вигляді горизонтального меню, існує опис всіх можливих послуг, але без надання інформації про ціни. Також великим недоліком є неможливість запису на прийом, відсутній зворотній зв'язок та можливість додати відгук. На даному сайті відсутня будь-яка взаємодія користувача із сайтом. Дизайн web-сайту простий, лаконічний, володіє приємною кольоровою палітрою.

Наостанок розглянемо web-сайт <https://vetmir.com.ua/> (див. рис. 1.3).



Рис. 1.3 Головна сторінка сайту ветеринарної клініки «Ветмир»

Як і в попередніх розглянутих аналогах, даний web-сайт має горизонтальне навігаційне меню, представлена вся необхідна інформація про ветеринарну клініку, а саме: послуги, ціни, акції, новини, статті та контакти. Оформлення сайту було виконано у спокійних кольорах і не відволікає клієнтів від головного. Великим недоліком є відсутність будь-якої взаємодії із сайтом: відсутня форма запису на прийом, кнопка зворотнього зв'язку.

В таблиці 1.1 наведена порівняльна характеристика програмних продуктів.

Таблиця 1.1

Порівняльна характеристика програмних продуктів

Фірма-розробник	VIS-A-VIS	Ower Studio	webalex.pro
Назва продукту	Цент Современной Ветеринарной Медицины	А-Вет	Ветмир

Функціональність	Повна взаємодія із сайтом: -запис на прийом; -зворотній зв'язок; -додавання відгуку. Надана вся необхідна інформація.	Неповна інформація. Відсутня взаємодія із web-сайтом.	Представлена повна інформація щодо клініки. Взаємодія із веб-системою відсутня.
Інтерфейс користувача	Не вдалий. Багато елементів, що ускладнює пошук необхідної інформації. Кольорова гама підібрана не вдало. Не сучасне оформлення.	Є інтуїтивним. Приємна кольорова гама. Простий, мінімалістичний, дизайн.	Є інтуїтивним. Кольорова гама підібрана вдало. Лаконічний дизайн.

Після огляду схожих web-сайтів ветеринарної клініки було проаналізовано усі переваги та недоліки представлених веб-систем. Основними перевагами є достатня функціональність та взаємодія із системою, а саме:

- можливість запису на прийом;
- можливість запиту на зворотній зв'язок;
- відправка відгуків.

Також при розробці сайту було враховано те, що інтерфейс користувача повинен бути приємним, сучасним, простим, для того, аби справити враження надійності та довіри на користувача.

Головним недоліком є неспроможність клієнта виконувати вище наведені функції та не інтуїтивний інтерфейс користувача, який ускладнює роботу із програмною системою.

Всі вище розглянуті переваги та недоліки були враховані при розробці web-сайту ветеринарної клініки.

Висновки до розділу

У ході написання розділу 1 – головним завданням було проаналізувати предметну область. Для цього було поставлено завдання, в якому були виділені головні задачі, які необхідно реалізувати у ході створення web-сайту, а також були визначені основні вимоги щодо навігації сайту, його інтерфейсу та наповненню.

Було розглянуто класифікацію сайтів, аби в подальшій розробці було зрозуміло для яких цілей створюється web-сайт.

Також було оглянуто та проаналізовано відомі методи розробки сайтів з метою обрання найзручнішого та найякіснішого. Вибір було зупинено на методі реалізації вручну, за допомогою HTML, CSS, JS, PHP. Даний метод не є найлегшим, але він надає змогу більш глибоко вивчити мови програмування та набути навичок у створенні web-сайтів.

Завершальним етапом написання розділу був огляд та аналіз схожих web-сайтів ветеринарних клінік, було визначено переваги, заради впровадження їх у web-сайт, та недоліки, аби не допустити їх у подальшій реалізації програмного продукту.

РОЗДІЛ 2

ПРОЕКТУВАННЯ WEB-САЙТУ ВЕТЕРИНАРНОЇ КЛІНІКИ

2.1 Специфікація вимог до web-сайту

Специфікація вимог до програмного продукту описує собою комплекс усіх вимог, які зв'язані з проектом. Під час розробки нової інформаційної системи або ж впроваджуючи вже існуючу необхідно визначити цю специфікацію, яка включає в себе функціональні та нефункціональні вимоги.

Під функціональними вимогами мають на увазі постановку задачі розробнику. Дані вимоги описують те, що необхідно реалізувати у продукті чи системі, а також ті дії, які повинні виконуватися користувачем при роботі з ними.

Зазвичай вони складаються з:

- User Story – показує, що очікує клієнт від команди розробки;
- Use cases – показує сценарії (варіанти) використання інформаційної системи.

User story – описує те, що робить користувач певної ролі для досягнення результату, і що потрібно зробити розробнику для відтворення поставленої задачі у реальному житті.

Як правило використовують наведений нижче шаблон:

As a/an <Назва ролі>, I want to <Ціль, Дія>, so that <Очікуваний результат>, to do <Що потрібно зробити розробнику>

Завдяки такій візуалізації взаємодії задача користувача плавно та логічно переходить у задачу для розробника, після чого настає черга варіантів використання.

Use cases описують поведінку користувача по крокам при роботі з програмним продуктом. Якщо клієнт вирішує задачу на створеній сторінці

Кафедра ПП				НАУ 21 13 61 – 000 ПЗ			
Виконала	Куцель Є.С.			Проектування web-сайту ветеринарної клініки	Літ.	Арк.	Аркушів
Керівник	Сінько Ю.І.					21	16
Консульт.					122 ТП-415Б		
Нормоконт.	Боровик В.М.						
Зав. каф.	Гамаюн В.П.						

декількома способами, то на кожне рішення повинен бути написаний свій сценарій.

Web-сайт, що розробляється має наступні варіанти використання:

1. Запис на прийом;
2. Надсилання відгуку;
3. Зворотній зв'язок;
4. Пошук послуг по назві;
5. Перегляд інформації.

Програмна система, що розробляється призначена для клієнта (гостя веб-сайту). За допомогою діаграми варіантів використання, яка наведена на рис. 2.1, більш наочно можна розглянути всю взаємодію користувача із системою.

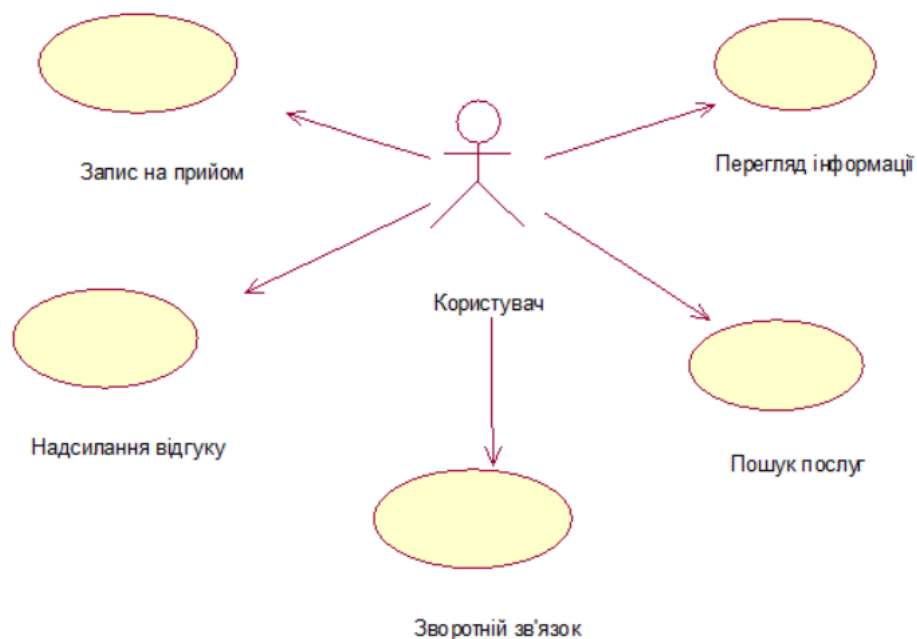


Рис. 2.1 Діаграма варіантів використання

Варіанти використання представлені у таблицях 2.1 – 2.5.

Таблиця 2.1

Варіант використання «Запис на прийом»

Контекст використання	Запис на прийом у ветеринарній клініці
Дійові особи (актори)	Користувач

Продовження таблиці 2.1

Тригер	Натискання кнопки «Запис на прийом»
Сценарій	<ol style="list-style-type: none"> 1. Заповнити поле «Ім'я» 2. Заповнити поле «Номер телефону» 3. Заповнити поле «Email» 4. Заповнити поле «Оберіть бажану дату прийому:» 5. Заповнити поле «Оберіть бажаний час прийому:» 6. Натиснути кнопку «Запис»
Пост-умова	Якщо користувач заповнив усі поля та заповним їх вірно, буде виведено повідомлення про успішне надсилання повідомлення. У протилежному випадку, будуть виведені тексти під полями з проханням ввести необхідні та коректні дані.

Таблиця 2.2

Варіант використання «Надсилання відгуку»

Контекст використання	Відправка відгуку на пошту ветеринарної клініки
Дійові особи (актори)	Користувач
Тригер	Перехід на сторінку «ВІДГУКИ»
Сценарій	<ol style="list-style-type: none"> 1. Заповнити поле «Ім'я» 2. Заповнити поле «Телефон» 3. Заповнити поле «Email» 4. Заповнити поле «Введіть текст»

Продовження таблиці 2.1

Сценарій	5. Натиснути кнопку «Запис»
Пост-умова	Якщо користувач заповнив усі поля та заповнив їх вірно, буде виведено повідомлення про успішне надсилання відгуку. У протилежному випадку, будуть виведені тексти під полями з проханням ввести необхідні та коректні дані.

Таблиця 2.3

Варіант використання «Зворотній зв'язок»

Контекст використання	Запис на прийом у ветеринарній клініці
Дійові особи (актори)	Користувач
Тригер	Перехід на сторінку «Контакти»; Натискання кнопки «Зворотній зв'язок»
Сценарій	1. Заповнити поле «Ваше ім'я:» 2. Заповнити поле «Email:» 3. Заповнити поле «Номер телефону:» 4. Заповнити поле «Ваше питання:» 5. Натиснути кнопку «Відправити»
Пост-умова	Якщо користувач заповнив усі поля та заповнив їх вірно, буде виведено повідомлення про успішне надсилання форми зворотнього зв'язку. У протилежному випадку, будуть виведені тексти під полями з

Пост-умова	проханням ввести необхідні та коректні дані.
------------	--

Таблиця 2.4

Варіант використання «Пошук послуг»

Контекст використання	Пошук послуг по назві
Дійові особи (актори)	Користувач
Тригер	Перехід на сторінку «ПОСЛУГИ»
Сценарій	1. Введення ключового слово у пошуковий рядок (можливо неповністю)
Пост-умова	Якщо користувач ввів ключове слово правильно, буде виведено список усіх послуг з даним словом, у протилежному випадку – список буде пустим.

Таблиця 2.5

Варіант використання «Перегляд інформації»

Контекст використання	Перегляд усієї інформації на web-сайті
Дійові особи (актори)	Користувач
Тригер	Перехід на сторінки «Головна», «Послуги», «Галерея», «Відгуки», «Контакти»

Наступним визначимо нефункціональні вимоги.

Під нефункціональними вимогами визначають такі вимоги, які описують як повинна працювати система, властивості та характеристики якими повинна володіти певна система або програмний продукт.

До таких вимог зазвичай відносять атрибути якості, такі як надійність, продуктивність, масштабованість та інші. Специфікація нефункціональних вимог наведена у таблиці 2.6.

Таблиця 2.6

Специфікація нефункціональних вимог

Ідентифікатори вимоги	Назва вимоги	Атрибути вимог	
		Пріоритет	Складність
1. Застосовність			
1.1	Кросбраузерність: Сайт повинен бути пристосованим до різних браузерів	Опційна	Низька
1.2	Вимоги по відповідальності стандартам графічного інтерфейсу користувача	Рекомендована	Середня
2. Надійність			
2.1	Доступність	Обов'язкова	Середня
2.2	Зручність використання	Обов'язкова	Низька
2.3	Точність	Обов'язкова	Середня
2.4	Середній час безвідмовної роботи	Обов'язкова	Низька
2.5	Безпека: базовий захист від основних атак	Обов'язкова	Низька
3. Масштабованість			
3.1	Додавання нових функцій	Рекомендована	Висока

2.2 Розробка архітектури програмної системи

Архітектура програмної системи – це сукупність основних рішень по структурній організації програмної системи, яка включає в себе:

- вибір структурних елементів та інтерфейсів за допомогою яких забезпечується їхня взаємодія та функціонування;
- з'єднання обраних елементів структури та поведінки у більш крупні системи;
- архітектурний стиль, який направляє всю організацію чи підприємство – всі елементи, інтерфейси, їхнє співробітництво та з'єднання.

Проектування архітектури – це початковий етап процесу проектування веб-сайту. На даному етапі визначаються основні процеси, підсистеми, структура управління та взаємодії. Архітектурний вигляд складається з двох компонентів: структурних елементів та зв'язками між ними.

Розглянемо детальніше архітектуру системи через опис основних бізнес-процесів, які знаходяться у програмній системі. Для цього використаємо засоби мов моделювання UML.

Для того, щоб записатися на прийом у ветеринарну клініку необхідно перебувати на головній сторінці сайту та натиснути кнопку «Запис на прийом». Процес запису на прийом зображено на діаграмі станів процесу «Запис на прийом» (рис. 2.2).

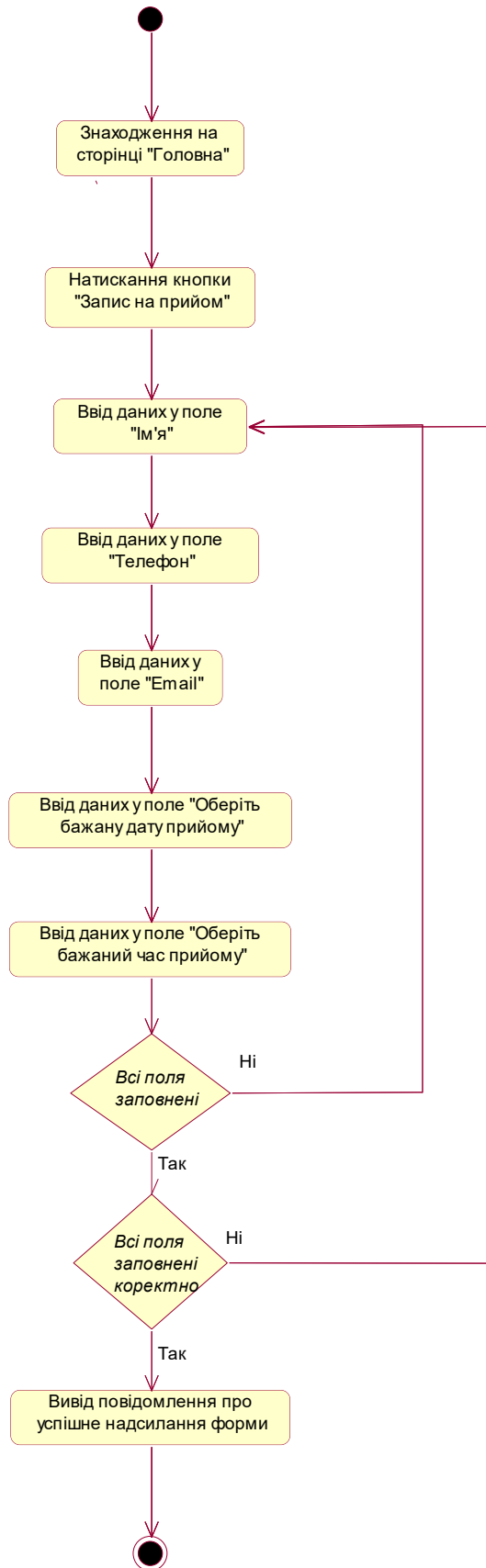


Рис. 2.2 Діаграма станів процесу «Запис на прийом»

На сайті також реалізована функція надсилання відгуків. Діаграму станів даного процесу зображено на рис. 2.3.

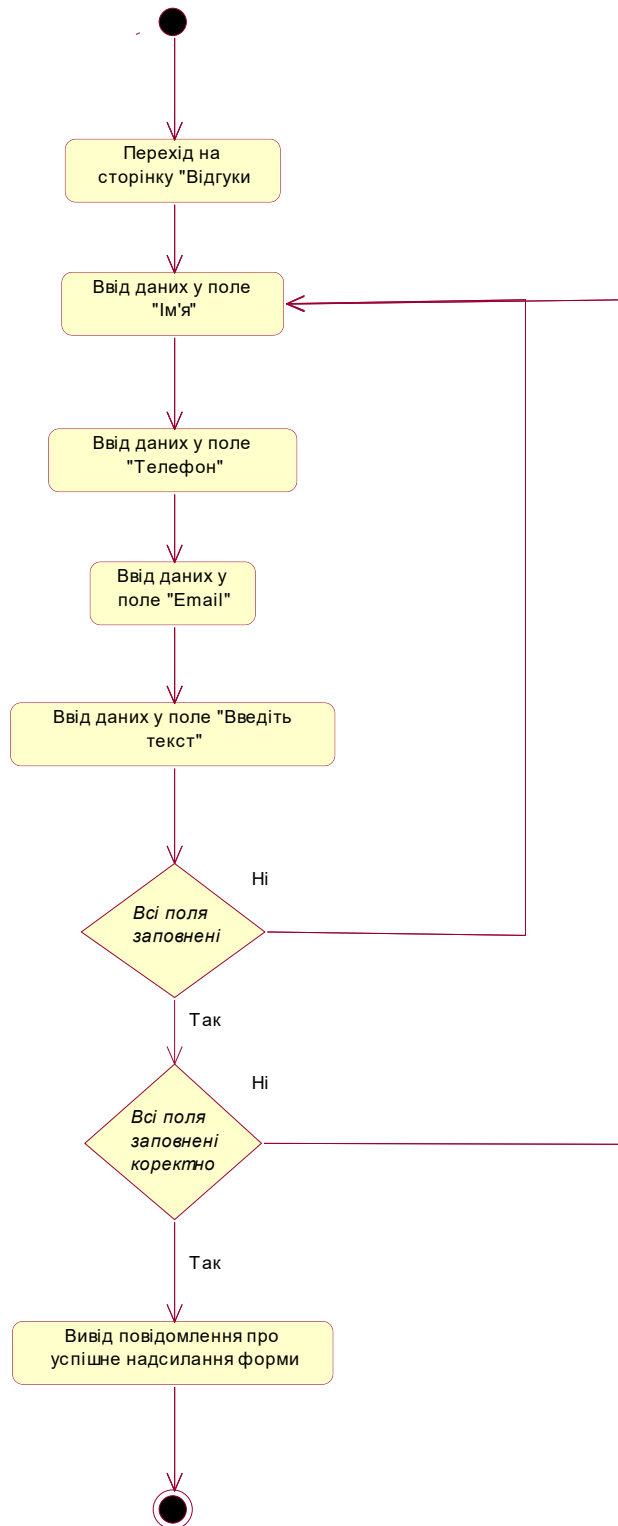


Рис. 2.3 Діаграма станів процесу «Надсилання відгуку»

Для того, щоб користувач мав змогу знайти необхідну послугу йому необхідно перейти на сторінку «Послуги» та ввести ключове слово для пошуку. Процес пошуку послуги наведено на рис. 2.4.

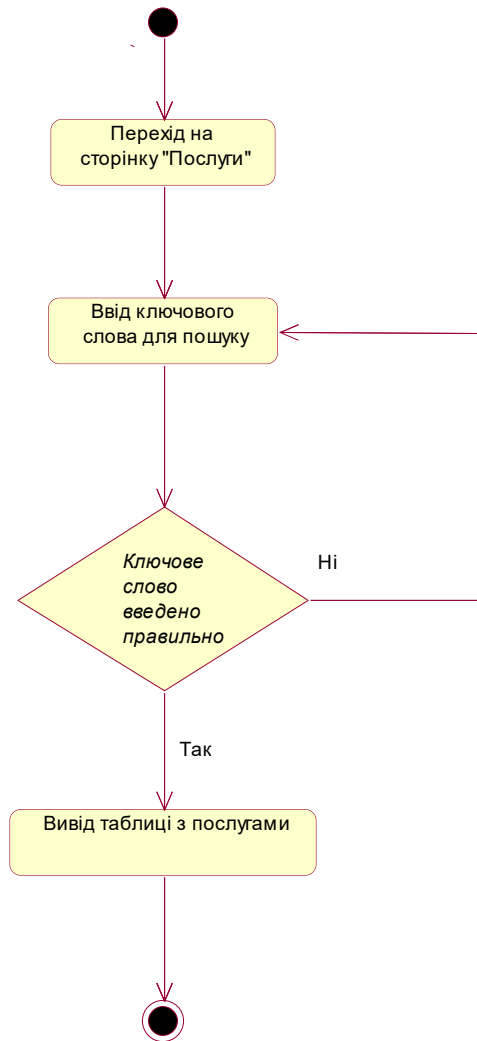


Рис. 2.4 Діаграма станів процесу «Пошук послуг»

Наступна функція передбачає змогу користувача надати запит на зворотній зв'язок, для цього необхідно заповнити надану форму на сторінці «Контакти». Даний процес функції представлено на рис. 2.5.

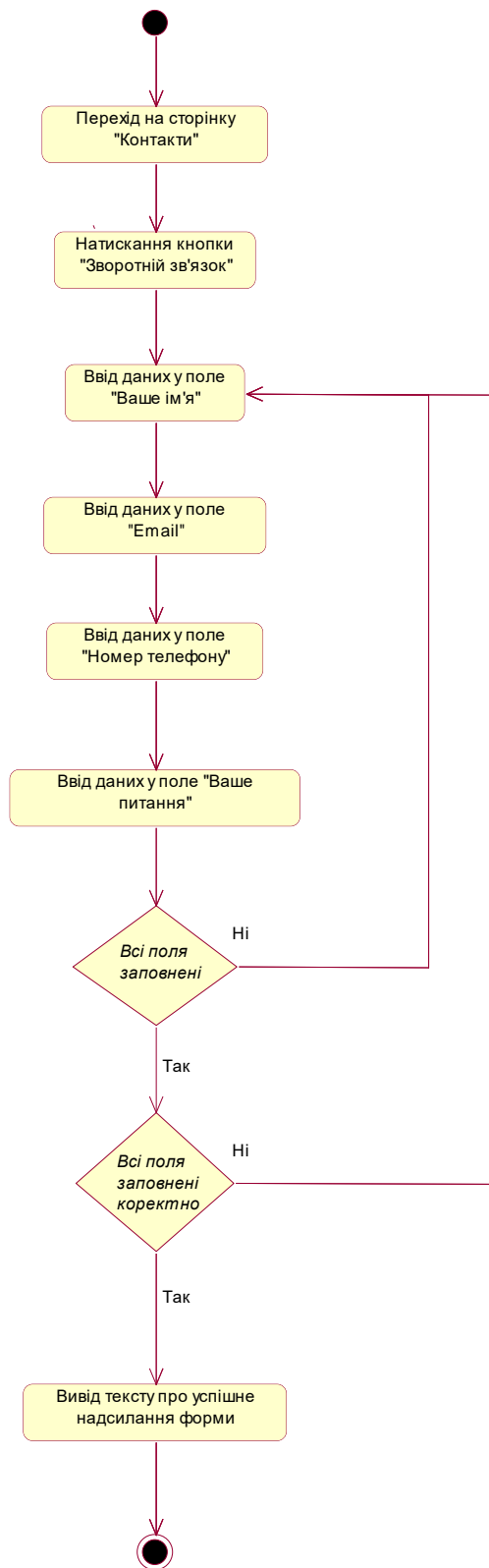


Рис. 2.5 Діаграма станів процесу «Зворотній зв'язок»

Архітектура системи являється не повною, якщо не надається інформація про платформу та на яких обчислювальних засобах вона реалізована, тому розробляється спеціальна діаграма, яка називається діаграмою розгортання. Дана діаграма відображає елементи та компоненти програми, а саме компоненти-екземпляри, які є виконуваними файлами або динамічними

бібліотеками, які існують тільки на етапі виконання. Діаграма, що описується складається з вузлів та з'єднань.

Діаграму розгортання представлено на рис. 2.6.

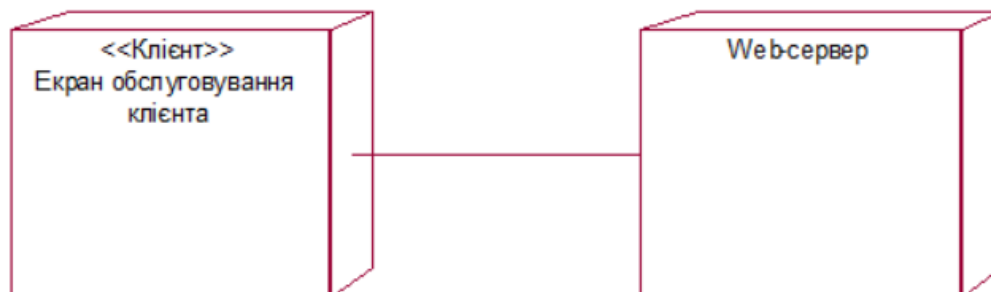


Рис. 2.6 Діаграма розгортання програмної системи

На представленій вище діаграмі зображено архітектуру клієнт-сервер. На першому рівні знаходиться web-браузер через який користувачі можуть здійснювати всі надані функції. На другому рівні – Web-сервер, на якому обробляється інформація.

2.3 Розробка структури web-сайту

Структура сайту – це планове розміщення всіх елементів сайту, як його сторінок, так і програмної частини, яка включає в себе навігацію, мережу посилань, сторінки категорій сайту та інші.

Поділяється структурне проектування сайту на зовнішнє та внутрішнє.

Зовнішня структура – це сам макет сторінки та розміщення блоків на ній.

Внутрішня структура – відображає категорії або належність окремих сторінок або матеріалів, що представлені на веб-ресурсі.

Композиція web-сайту служить для того, щоб надати користувачу можливість швидко та без проблем відшукувати потрібну інформацію, навіть якщо він вперше знаходиться на інтернет-ресурсі. Логіка розміщення повинна бути настільки зрозумілою, щоб людина не витратила більше декількох хвилин на те, щоб у ній розібратися.

Щоб почати створення web-сайту ветеринарної клініки було розроблено структуру. Головна сторінка сайту включає в себе: інформацію про ветеринарну клініку, послуги, галерею, відгуки та контактну інформацію. Дана структура показана на рис. 2.6.

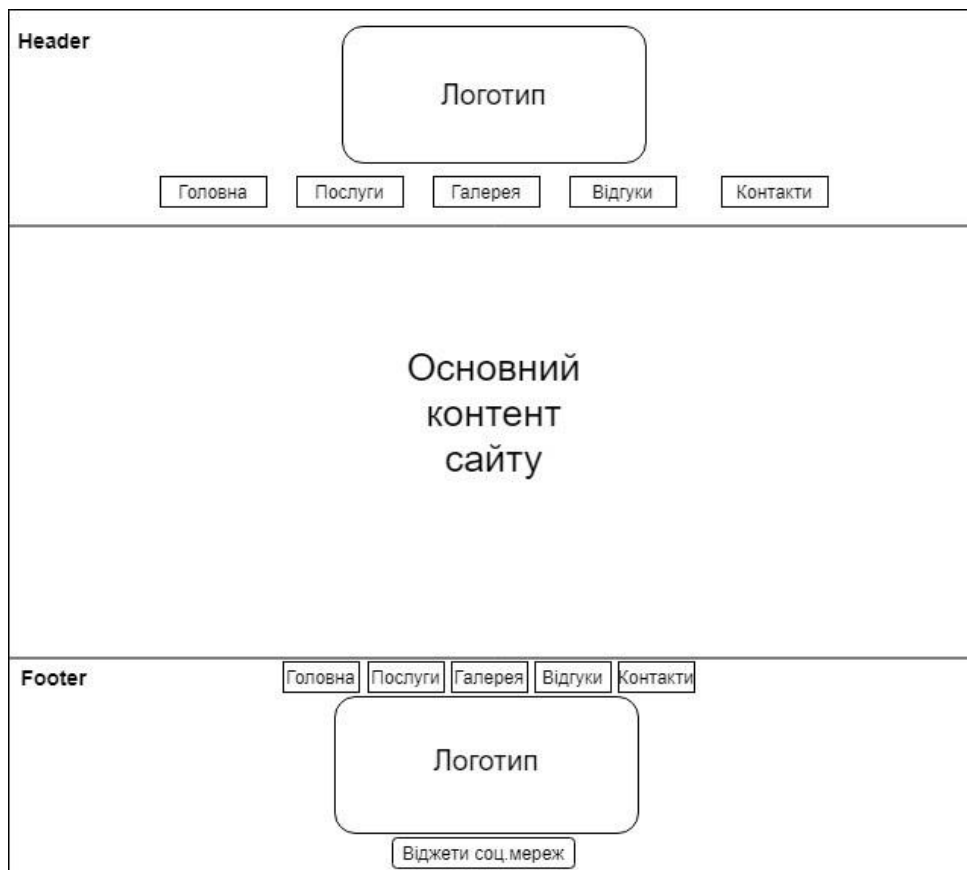


Рис. 2.6 Структура web-сайту ветеринарної клініки

Проаналізувавши попередні схожі веб-ресурси було вирішено розмістити на сайті горизонтальне меню з п'яти пунктів:

- головна;
- послуги;
- галерея;
- відгуки;
- контакти.

Верхня частина сайту, яку ще називають «header» або «шапка сайту» - містить у собі логотип ветеринарної клініки та головні пункти меню.

У центрі розміщується весь основний контент веб-ресурсу.

Нижня частина web-сайту, або як її ще називають «footer» складається з того ж логотипу, головних пунктів меню та віджетів соціальних мереж.

2.4 Засоби для створення web-сайту

Для подальшої розробки сайту необхідно визначити, які засоби для реалізації нам знадобляться відносно обраного методу. Обравши метод створення web-сайту вручну нам знадобляться такі засоби:

- HTML

HTML визначають як основну мову розмітки гіпертексту, яка дає можливість користувачу формувати та структурувати розділи, абзаци, заголовки, посилання, блоки для web-сторінок та додатків.

Головне, що потрібно розрізнити – це те, що ця мова не являється мовою програмування, тому що вона не має можливості реалізувати динамічні функції, замість цього вона дозволяє розробляти та редагувати документи, аналогічно Microsoft Word.

HTML-документами називають файли з розширенням зазвичай .html. Такі документи можна продивлятися за допомогою будь-якого веб-браузера. Браузер прочитає ці файли та відображає їхній зміст, для того, щоб будь-які користувачі змогли продивлятися інформацію, що знаходиться у даних файлах.

Кожна HTML-сторінка складається з набору тегів (так називають елементи), які створюють ієрархію, яка в свою чергу структурує зміст сторінки по розділам, параграфам, заголовкам та іншим блокам контенту.

- CSS

CSS – це спеціальний код, який використовується для додавання різних стилів до веб-сайту та його змісту та складається з великого набору різних властивостей для форматування елементів документа.

Уся структура цього коду (рис. 2.7) є набором правил, яка складається з:

- Селектора (Selector) – ім'я, об'явленого елемента у HTML-файлі. Він обирає елемент або декілька елементів для стилізації;
- Оголошення (Declaration) – це правило, яке вказує яку з характеристик елемента стилізують;
- Властивості (Properties) – способу, яким можливо стилізувати елемент, наприклад color, font-size, background та інші;

- Значення властивості (Property value) – це все, що знаходиться після знаку «:», текст після цього знаку обирає одне з можливих значень, яке присвоюється властивості.



Рис. 2.7 Структура CSS

- JavaScript

JavaScript (JS) – це одна з найвідоміших мов програмування, яка була вигадана для створення інтерактивних сайтів. Іншими словами, це мова програмування, за допомогою якої оживляють веб-сторінки. Сповіщення про нове повідомлення, завантаження нових постів, натискання на кнопку для відкриття певного вікна – це все реалізовано за допомогою JavaScript.

Код, написаний на JS називають скриптом та зберігають в окремому файлі з розширенням .js. Для того, щоб запустити готовий код у дію необхідно підключити його до сторінки. Для підключення коду даний скрипт вставляють у файл з розширенням .html наступним чином:

```
<script src="адреса_файлу"></script>
```

- Php

Мова PHP – це одна з широко використовуваних мов програмування, яка впроваджена для написання веб-додатків (сценаріїв), які виконуються безпосередньо на веб-сервері. Основним пріоритетом даної мови є представлення web-розробникам можливості швидкого створення динамічно сформованих web-сторінок. Значною відмінністю мови PHP від будь-якого коду, який виконується на стороні клієнта є те, що PHP-сценарії виконуються на стороні сервера.

Висновки до розділу

Створення будь-якого програмного продукту не може проходити без проектування. Даний етап у розробці слугує процесом, який вирішує завдання та планує їх. У розділі «Проектування web-сайту ветеринарної клініки» було розроблено специфікацію вимог до системи для повного опису поведінки web-сайту.

Також було розроблено архітектуру програмної системи, за допомогою якої було детально показано як будуть виконуватися основні функції web-сайту заради повного розуміння, як працює та чи інша функція.

Наостанок, було описано програмні засоби за допомогою яких буде розроблятися web-сайт та реалізуватися функції.

РОЗДІЛ 3

РЕАЛІЗАЦІЯ WEB-САЙТУ ВЕТЕРИНАРНОЇ КЛІНІКИ

3.1. Опис інтерфейсу web-сайту ветеринарної клініки

Після того як було здійснено аналіз предметної області, проведено аналіз схожих веб-ресурсів та розроблено проектування програмної системи можливий наступний крок у створенні сайту – це сама реалізація програмного продукту.

Після огляду деяких існуючих варіантів сайтів ветеринарних клінік було вирішено, що сайт, який розробляється повинен мати сучасний, стильний та лаконічний дизайн, з наданою основною інформацією, яка буде повністю доносити її до клієнта.

Першим етапом у реалізації web-сайту буде створення верхньої (header) та нижньої (footer) частини, оскільки вона буде незмінна до кожної наступної сторінки сайту.

У шапці сайту повинен розміщуватися логотип з назвою ветеринарної клініки, а також назви інших web-сторінок, які будуть слугувати гіперпосиланнями на них. Завдяки цим посиланням надається можливість переходити на інші сторінки сайту.

Верхня частина web-сайту ветеринарної клініки представлена на рис. 3.1.

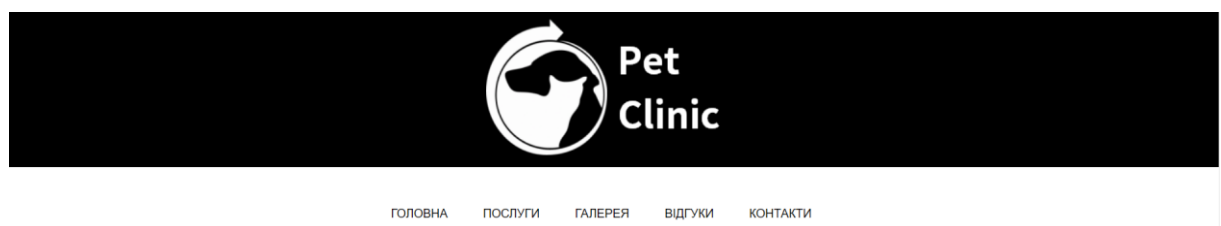


Рис. 3.1 Верхня частина web-сайту ветеринарної клініки

Нижня частина сайту буде складатися з того ж самого логотипу, що і у верхньої частини, посилань на інші сторінки веб-сайту та іконок соціальних

Кафедра ПП				НАУ 21 13 61 – 000 ПЗ			
Виконала	Куцель Є.С.			Реалізація web-сайту ветеринарної клініки	Літ.	Арк.	Аркушів
Керівник	Сінько Ю.І.					37	10
Консульт.					122 ТП-415Б		
Нормоконт.	Боровик В.М.						
Зав. каф.	Гамаюн В.П.						

віджетів.

Нижня частина web-сайту ветеринарної клініки представлена на рис. 3.2.



Рис. 3.2 Нижня частина web-сайту ветеринарної клініки

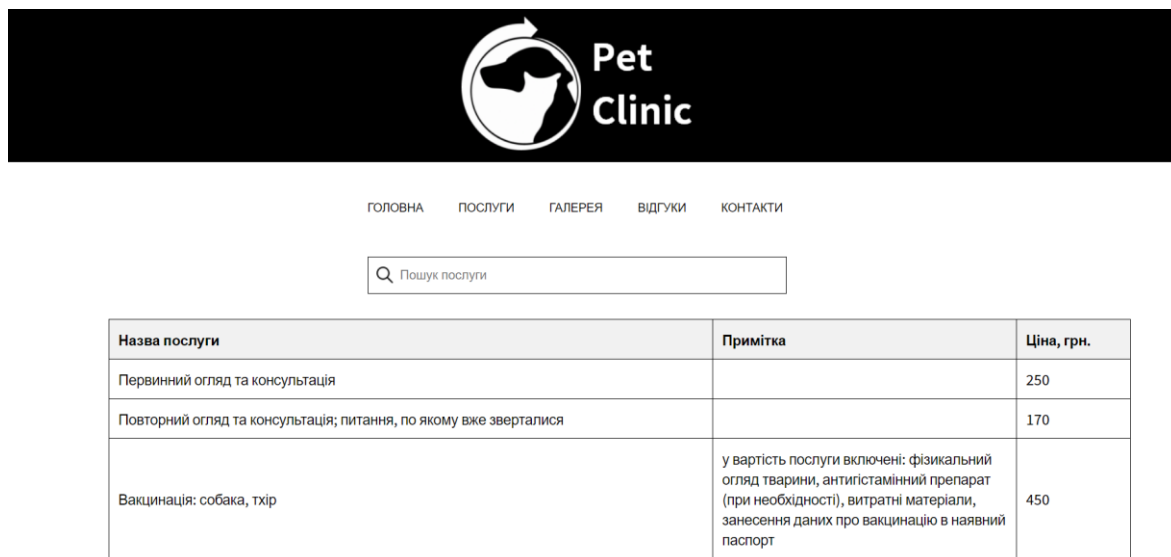
На головній сторінці між верхньою та нижньою частинами розміщено основний контент, який містить опис ветеринарної клініки, а також кнопку «Запис на прийом» за допомогою якої можливо заповнити форму запису на прийом вашої домашньої тварини до ветеринарної клініки.

Основний зміст головної сторінки web-сайту відображено на рис. 3.3.



Рис. 3.3 Основний контент головної сторінки web-сайту ветеринарної клініки

Наступним пунктом є веб-сторінка «Послуги», яка включає у себе інформацію про усі послуги, які надає ветеринарна клініка. Дана інформація представляється у вигляді таблиці, яка в свою чергу містить такі поля як «Назва послуги», «Примітка» та «Ціна, грн.». Вигляд сторінки зображено на рис. 3.5.



Назва послуги	Примітка	Ціна, грн.
Первинний огляд та консультація		250
Повторний огляд та консультація; питання, по якому вже зверталися		170
Вакцинація: собака, тхір	у вартість послуги включені: фізикальний огляд тварини, антигістамінний препарат (при необхідності), витратні матеріали, занесення даних про вакцинацію в наявний паспорт	450

Рис. 3.5 Веб-сторінка «Послуги»

Далі йде пункт «Галерея» (рис. 3.6), в якому розміщено декілька фотографій, з можливістю перегляду їх за допомогою стилізованих та інтегрованих кнопок вперед та назад.

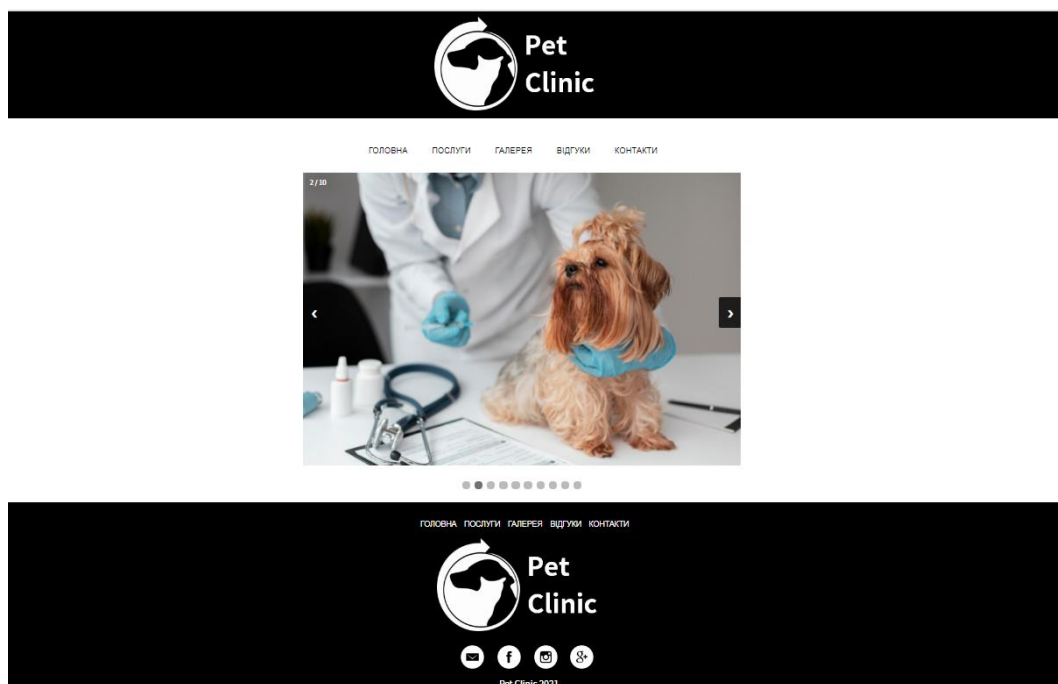


Рис. 3.6 Пункт «Галерея» web-сайту ветеринарної клініки

На наступній сторінці «Відгуки» розміщено форму для відправки на електронну пошту відгуків користувачів, які скористалися послугами ветеринарної клініки. Вигляд даної сторінки зображено на рис. 3.7.

ГОЛОВНА ПОСЛУГИ ГАЛЕРЕЯ ВІДГУКИ КОНТАКТИ

Ім'я

Телефон

Email

Введіть текст

Відправити відгук

Рис. 3.7 Вигляд web-сторінки «Відгуки»

Останнім пунктом у меню є «Контакти» (рис. 3.8). Даний пункт включає у себе детальну інформацію про місцезнаходження ветеринарної клініки; контактний номер телефону, за допомогою якого можна зв'язатися з клінікою; та електронну адресу. Також на цій веб-сторінці розміщено кнопку зворотнього зв'язку, яка надає клієнту можливість заповнити форму, яка допоможе користувачу задати питання, яке його турбує, та бути впевненим, що з ним зв'яжеться працівник клініки за кілька секунд.

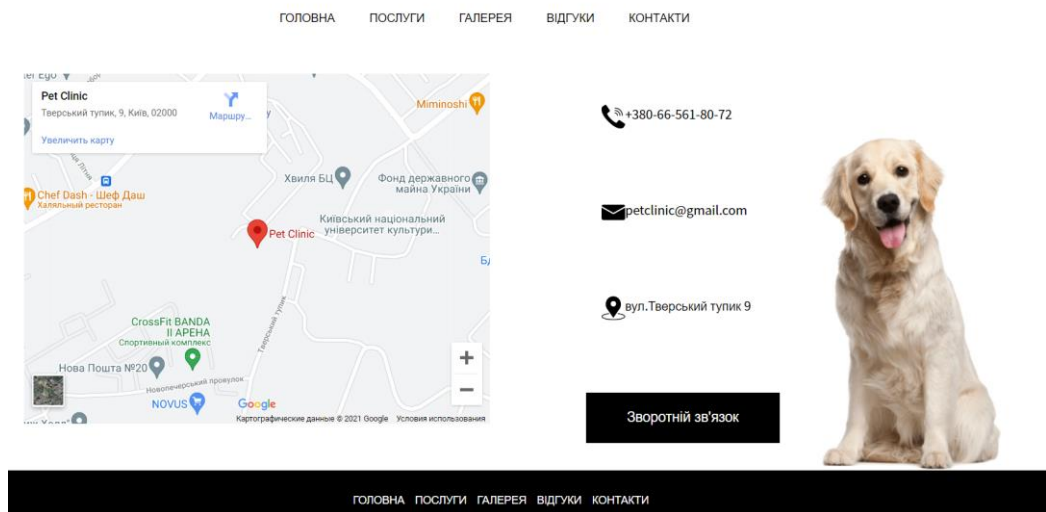


Рис. 3.8 Пункт меню «Контакти»

Контент сайту (текст, заголовки, параграфи та інше) було розроблено за допомогою коду мови гіпертекстової розмітки HTML, який представлено у додатку А.

Увесь дизайн сайту розроблявся за допомогою каскадних таблиць стилів CSS, код яких знаходиться у додатку Б.

3.2. Програмна реалізація функцій web-сайту ветеринарної клініки

Оформлення сайту – це найлегший етап у розробці, але він не дозволяє користувачу взаємодіяти із програмним продуктом повністю, єдине що зможе користувач – це переглядати інформацію на web-сторінках. Але найголовнішою задачею проекту було створення функціоналу сайту з метою надання доступу кожному, хто має бажання скористатися представленими послугами ветеринарної клініки.

Для реалізації поставлених задач було використано мови програмування JavaScript та PHP.

Однією з найважливіших функцій є запис клієнта на прийом. Для впровадження даної мети було вирішено створити форму запису на прийом (рис. 3.9), яка буде складатися з основних полів для вводу даних, після чого заповнена форма буде відправлятися на електронну пошту ветеринарної клініки.

Повідомлення про новий запис на електронній пошті представлено на рис.

3.10. Реалізація описаної функції наведено у лістингу коду нижче:

```
<?php
$name = $_POST['name'];
$phone = $_POST['phone'];
$email = $_POST['email'];
$date = $_POST['date'];
$time = $_POST['time'];

$to = "jenna.kutsel@gmail.com";
$from = $email;
$subject = "Новий запис на сайті";

$msg=""
Ім'я: $name /n
Телефон: $phone /n
Email: $email /n
Бажана дата: $date /n
Бажаний час: $time";
mail($to, $subject, $msg, "From: $from");
?>
```

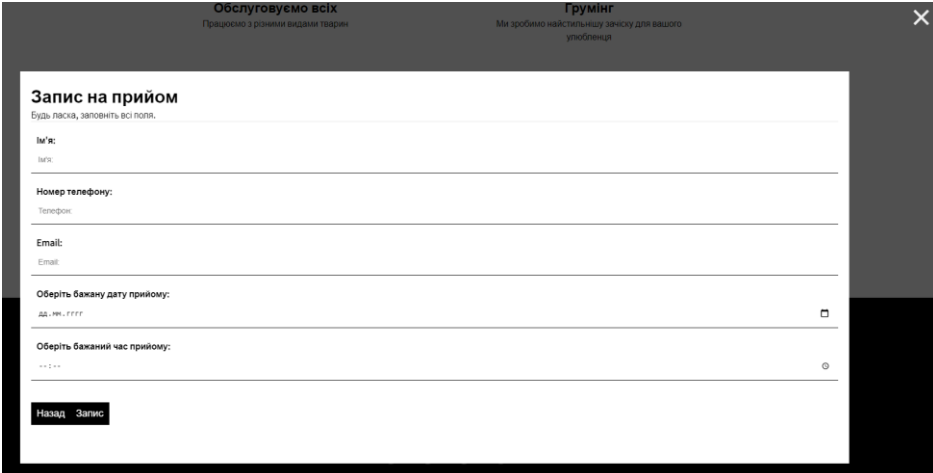
The image shows a screenshot of a web browser displaying a form titled "Запис на прийом" (Appointment). The form is set against a dark background. At the top, there are two banners: "Обслуговуємо всіх" (We serve everyone) and "Грумінг" (Grooming). The form itself has a white background and contains the following fields: "Ім'я:" (Name) with a text input field; "Номер телефону:" (Phone number) with a text input field; "Email:" with a text input field; "Оберіть бажану дату прийому:" (Select desired appointment date) with a date picker (dd.mm.yyyy); and "Оберіть бажаний час прийому:" (Select desired appointment time) with a time picker (h:i). At the bottom of the form, there are two buttons: "Назад" (Back) and "Запис" (Appointment).

Рис. 3.9 Форма запису на прийом

Новий запис на сайті Входящие x



jenna.kutsel@gmail.com через u107924.test-handyhost.ru

кому: я ▾

🇺🇦 українский ▾ > русский ▾ [Перевести сообщение](#)

Ім'я: Євгенія /n

Телефон: +380 (66) 561-80-72 /n

Email: jenna.kutsel@gmail.com /n

Бажана дата: 2021-05-31 /n

Бажаний час: 10:00

Рис. 3.10 Повідомлення про новий запис на електронній пошті

Для наступних функцій, таких як надсилання відгуку та зворотній зв'язок було використано такий же метод реалізації, як і у записі на прийом.

Форма надсилання відгуку та повідомлення про новий відгук на електронній пошті зображено на рис. 3.11-3.12.

Євгенія

+380 (66) 561-80-72

jenna.kutsel@gmail.com

Ветеринарна клініка та web-сайт працюють бездоганно! Дякую

Відправити відгук

Рис. 3.11 Форма надсилання відгуку

Новий відгук Входящие x



jenna.kutsel@gmail.com через u107924.test-handyhost.ru

кому: я ▾

🇺🇦 українский ▾ > русский ▾ [Перевести сообщение](#)

Ім'я: Євгенія /n

Телефон: +380 (66) 561-80-72 /n

Пошта: jenna.kutsel@gmail.com /n

Текст: Ветеринарна клініка та web-сайт працюють бездоганно! Дякую

Рис. 3.12 Повідомлення про новий відгук на електронній пошті

Форма зворотнього зв'язку та повідомлення про заявку на зворотній зв'язок на електронній пошті представлено на рис. 3.13-3.14.

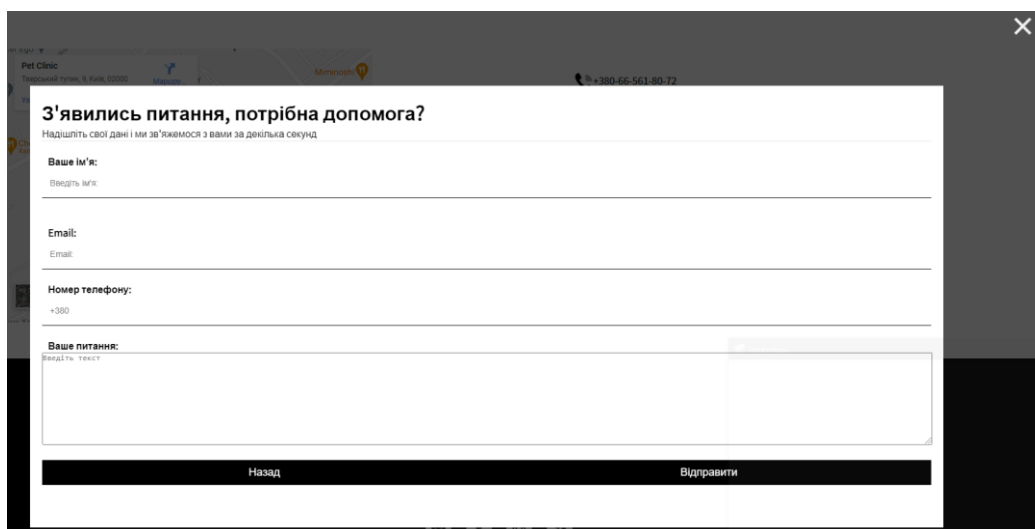


Рис. 3.13 Форма зворотнього зв'язку

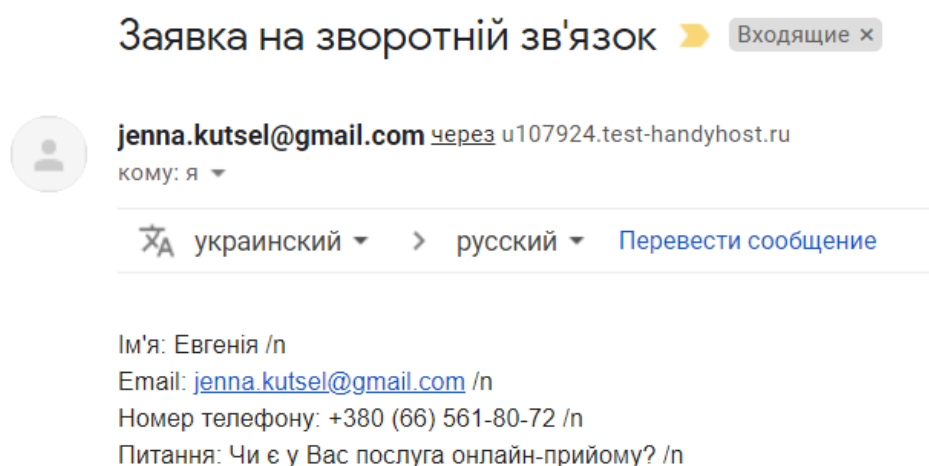


Рис. 3.14 Повідомлення про заявку на зворотній зв'язок

Остання функція «Пошук послуги» була розроблена за допомогою коду, написаний на JS, який представлений нижче:

```
function myFunction() {  
    var input, filter, table, tr, td, i;  
    input = document.getElementById("myInput");  
    filter = input.value.toUpperCase();  
    table = document.getElementById("myTable");  
    tr = table.getElementsByTagName("tr");  
    for (i = 0; i < tr.length; i++) {  
        td = tr[i].getElementsByTagName("td")[0];
```

```

if (td) {
    if (td.innerHTML.toUpperCase().indexOf(filter) > -1) {
        tr[i].style.display = "";
    } else {
        tr[i].style.display = "none";
    }
}
}
}
}

```

Дана функція виводить усі послуги, в яких існує введене користувачем ключове слово. Результат функції наведено на рис. 3.15.

ГОЛОВНА ПОСЛУГИ ГАЛЕРЕЯ ВІДГУКИ КОНТАКТИ

Назва послуги	Примітка	Ціна, грн.
Первинний огляд та консультація		250
Повторний огляд та консультація; питання, по якому вже зверталися		170

Рис. 3.15 Результат функції «Пошук послуги»

3.3. Завантаження web-сайту у мережу Інтернет

Завершальним етапом створення web-сайту є розміщення його на просторах Інтернету, адже без цього користувачам не буде надано до нього вільний доступ.

Для даної задачі нам необхідно мати домен (унікальна адреса сайту), та підключитися до серверу, так званого хостингу, за допомогою FTP-клієнта (протоколу передачі файлів, який призначений для обміну даними між клієнтом та сервером). Хостинг – це така послуга, яка надає можливість розміщати інформацію (файли) у мережу Інтернет для представлення певних ресурсів у постійному доступі до них.

У даному випадку було обрано програму FileZilla у вигляді FTP-клієнта. Все, що необхідно зробити – це вказати хост, ім'я користувача, пароль та порт, які нам були представлені після реєстрації на сайті <https://handyhost.ru/> для

доступу до акаунту хостингу, та помістити усі файли створеного web-сайту у спеціальну папку під назвою «www». Після всіх описаних дій сайт буде успішно завантажено у мережу Інтернет.

Висновки до розділу

Програмна реалізація продукту – це завершальний етап створення web-сайту, який полягає у розробці сайту за допомогою різних програмних засобів.

За допомогою мови розмітки HTML та CSS було розроблено основні веб-сторінки сайту, а також їхній дизайн.

За допомогою мови програмування JavaScript даному web-сайту було надано динамічності та реалізовано функцію «Пошук послуг».

За допомогою PHP було реалізовано відправку форм даних на електронну пошту.

Також, створений web-сайт було розміщено у мережу Інтернет з метою надання користувачам можливості бачити сайт та взаємодіяти з ним.

ВИСНОВКИ

Ветеринарні клініки – одна з найбільш актуальних сфер послуг на сьогоднішній день. Кожен власник домашнього улюбленця бажає, щоб його улюблена тварина була здорова, тому користується послугами ветеринарної клініки. Сучасні ветеринарні клініки, для покращення свого функціонування та полегшення пошуку потенційних клієнтів, мають свій web-сайт. Проте, проведені дослідження показали, що наявні сайти клінік не виконують необхідний функціонал для задоволення побажань клієнтів.

Саме тому, метою дипломної роботи було створення web-сайту. У ході виконання дипломного проекту були вирішені усі поставлені завдання, а саме:

- Проведено аналіз предметної області;
- Здійснено проектування програмної системи;
- Розроблено web-сайт ветеринарної клініки.

Для кращого розуміння концепції створюваного web-сайту, було проаналізовано схожі web-сайти, які реалізують функції предметної області, та визначено, що загальними спільними недоліками розглянутих сайтів є інтерфейс, який містить зайву інформацію та елементи, що ускладнюють пошук необхідної інформації для користувача, а також недостатня кількість функцій або їх відсутність, яка не дає змоги клієнту взаємодіяти із сайтом.

Тож, в ході роботи було досліджено та визначено основні вимоги до оформлення та вмісту web-сайту. Зокрема була визначена необхідна для потенційних клієнтів інформація, яка має міститись на головній сторінці – назва клініки, основні напрямки роботи, а також головне меню, з можливістю переходу на потрібні сторінки, такі як контакти, послуги, галерея та відгуки про клініку. Крім того, у розділі Послуги міститься інформація з усім спектром послуг, які наявні у клініці, представлені у вигляді таблиці з відповідними полями – «Назва послуги», «Примітка» та «Ціна, грн.», та надано можливість швидкого пошуку у розділі.

Також на головній сторінці клієнтам надана можливість записатись на прийом зі своїм домашнім улюбленцем. У розділі Відгуки розміщено форму для

відправки на електронну пошту відгуків користувачів, які скористалися послугами ветеринарної клініки. Розділ Контакти містить необхідну для клієнта інформацію з місцем знаходження клініки, контактними даними та створено кнопку для зворотнього зв'язку з клієнтом, де користувачі мають можливість залишити свій номер телефону та очікувати дзвінок від працівників клініки.

При розробці сайту було враховано те, що інтерфейс користувача повинен бути приємним, сучасним, простим, для того, аби справити враження надійності та довіри на користувача. Крім того, графічний інтерфейс забезпечує швидкий перехід до обраного розділу сайту, меню навігації є зрозумілим та корисним для клієнтів.

Загалом, сайт є зручним для використання, містить в собі необхідну для клієнтів інформацію з можливістю взаємодії з сайтом, дизайн є лаконічним та сучасним по наповненню, а увага потенційних клієнтів сфокусована на потрібній для них інформації та послугах.

Створений сайт надає користувачам повний доступ та взаємодію із ним, через що дозволяє охопити велику аудиторію та утримати її на довгий час.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Разработка сайта – руководство для начинающих [Электронный ресурс]: Режим доступа: <https://beseller.by/blog/>. – Название с экрана.
2. Different Ways To Build A Website – Which Of These Is Best For You? [Electronic resource]. – Access mode: <https://www.tentononline.com/different-ways-to-build-a-website/>. – Title from the screen.
3. Способы создания сайту [Электронный ресурс]: – Конструктор сайта. – Режим доступа: <https://alexsend.com.ua/blog/sposobi-sozdaniya-saita/>. – Название с экрана.
4. Что Такое Хостинг Сайтов? [Электронный ресурс]. – Режим доступа: <https://www.hostinger.com.ua/>. – Название с экрана.
5. ЧТО ТАКОЕ КОНСТРУКТОР САЙТОВ? [Электронный ресурс]: – Преимущества конструктора сайтов. – Режим доступа: <https://freehost.com.ua/faq/wiki/chto-takoe-konstruktor-sajtov/>. – Название с экрана.
6. Проектирование и архитектура программных систем : учеб. пособие /А. Г. Белик, В. Н. Цыганенко ; Минобрнауки России, ОмГТУ. – Омск : Изд-во ОмГТУ, 2016. – 96 с.
7. Как писать функциональные требования [Электронный ресурс]. – Режим доступа: <https://habr.com/ru/>. – Название с экрана.
8. Нефункциональные требования к программному обеспечению. Часть 1 [Электронный ресурс]: – Нефункциональные требования: какие они бывают. – Режим доступа: <https://habr.com/ru/post/231961/>.
9. Диаграммы состояний [Электронный ресурс]. – Режим доступа: <https://itteach.ru/rational-rose/diagrammi-sostoyaniy>. – Название с экрана.
10. Диаграммы развертывания [Электронный ресурс]: Информатика и программирование [лекции по предмету]. – Режим доступа: <https://studizba.com/lectures/10-informatika-i-programmirovanie/>.
11. Структура сайта: разработка структуры в виде схемы, типы и примеры [Электронный ресурс]: – Что такое структура сайта и зачем она нужна.

– Режим доступа: <https://postium.ru/struktura-sajta-razrabotka-v-vide-sxemy/>. – Название с экрана.

12. Архитектура сайта: 15 точечных советов для выхода в топ [Электронный ресурс]: – Режим доступа: <https://sales-generator.ru/blog/arkhitektura-sayta/>. – Название с экрана.

13. Что такое HTML? Основы языка разметки гипертекста [Электронный ресурс]. – Режим доступа: <https://www.hostinger.ru/rukovodstva/shto-takoe-html/>. – Название с экрана.

14. Самоучитель CSS [Электронный ресурс]. – Режим доступа: <http://htmlbook.ru/samcss>

15. Основы CSS [Электронный ресурс]: – Анатомия набора правил CSS. – Режим доступа: <https://developer.mozilla.org/>.

16. Зачем нужен JavaScript [Электронный ресурс]. – Режим доступа: <https://htmlacademy.ru/courses/>. – Название с экрана.

17. Что такое FTP? [Электронный ресурс]: – Что такое FTP сервер. – Режим доступа: <https://freehost.com.ua/faq/wiki/chto-takoe-ftp/>. – Название с экрана.

18. Как залить сайт на хостинг [Электронный ресурс]. – Режим доступа: <https://bitte.net.ua/blog/kak-zalit-sayt-na-hosting/>. – Название с экрана.

ДОДАТОК А

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Pet Clinic</title>
  <link rel="stylesheet" href="css/style.css">
  <link href="https://fonts.googleapis.com/css2?family=Noto+Sans+JP" rel="stylesheet">
</head>
<body>
  <header>
    <div class="logo">
      <a href="index.html"></a>
    </div>
    <nav>
      <div class="topnav" id="myTopnav">
        <a href="index.html">ГОЛОВНА</a>
        <a href="services.html">ПОСЛУГИ</a>
        <a href="gallery.html">ГАЛЕРЕЯ</a>
        <a href="about.html">ВІДГΥКИ</a>
        <a href="contacts.html">КОНТАКТИ</a>
        <a id="menu" href="#" class="icon">&#9776;</a>
      </div>
    </nav>
  </header>
  <main>
    
    <div class="advantages__container">
      <div class="advantages">
        <div class="advantages__item">
          
          <h2>Працюємо 24/7</h2>
          <p>Цілодобова турбота про ваших улюбленців</p>
        </div>
        <div class="advantages__item">
```

```

```

```
<h2>Супер-команда</h2>
```

```
<p>У нашій клініці працюють найкращі лікарі з досвідом роботи більше 10 років</p>
```

```
</div><div class="advantages__item">
```

```

```

```
<h2>Великий вибір послуг</h2>
```

```
<p>Зробимо все, аби ваш друг відчував себе здоровим та енергійним</p>
```

```
</div>
```

```
<div class="advantages__item">
```

```

```

```
<h2>Виклик на дім</h2>
```

```
<p>Якщо Ви не в змозі доїхати до нас, наші лікарі самі приїдуть на допомогу</p>
```

```
</div>
```

```
<div class="advantages__item">
```

```

```

```
<h2>Обслуговуємо всіх</h2>
```

```
<p>Працюємо з різними видами тварин</p>
```

```
</div>
```

```
<div class="advantages__item">
```

```

```

```
<h2>Грумінг</h2>
```

```
<p>Ми зробимо найстильнішу зачіску для вашого улюбленця</p>
```

```
</div>
```

```
<div class="advantages__item">
```

```

```

```
<h2>Вигул тварин</h2>
```

```
<p>Якщо у Вас немає часу на вигул друга, ми зробимо це за Вас</p>
```

```
</div>
```

```
<div class="advantages__item">
```

```

```

```
<h2>Любов та турбота</h2>
```

```
<p>Піклуємося та відносимося до тварин з особливою ласкою</p>
```

```
</div>
```

```
</div>
```

```
</div>
```

```

    <button onclick="document.getElementById('id01').style.display='block'">Запис на
прийом</button>
    <div id="id01" class="modal">
    <span onclick="document.getElementById('id01').style.display='none'" class="close"
title="Close Modal">×</span>
    <form class="modal-content" action="sender.php" method="post">
    <div class="containers">
    <h1>Запис на прийом</h1>
    <p>Будь ласка, заповніть всі поля.</p>
    <hr>
    <label for="name"><b>Ім'я:</b></label>
    <input type="text" placeholder="Ім'я:" name="name" required>
    <br>
    <label for="phone"><b>Номер телефону:</b></label>
    <input type="text" class="phone" placeholder="Телефон:" name="phone" required>
    <br>
    <label for="email"><b>Email:</b></label>
    <input type="email" placeholder="Email:" name="email">
    <br>
    <label for="date"><b>Оберіть бажану дату прийому:</b></label>
    <input type="date" id="date" placeholder="Бажана дата:" name="date" required>
    <br>
    <label for="time"><b>Оберіть бажаний час прийому:</b></label>
    <input type="time" id="time" placeholder="Бажаний час" name="time" required>
    <br>
    <div class="clearfix">
    <button type="button"
onclick="document.getElementById('id01').style.display='none'"
class="cancelbtn">Назад</button>
    <button type="button" class="signupbtn">Запис</button>
    <div class="status"></div>
    </div>
    </div>
    </form>
</div>

</main>

```

```
<footer>
  <nav>
    <a href="index.html">ГОЛОВНА</a>
    <a href="services.html">ПОСЛУГИ</a>
    <a href="gallery.html">ГАЛЕРЕЯ</a>
    <a href="about.html">ВІДГҀКИ</a>
    <a href="contacts.html">КОНТАКТИ</a>
  </nav>
  <div class="logo">
    <a href="index.html"></a>
  </div>
  <div class="social">
    <a href="#"></a>
    <a href="#"></a>
    <a href="#"></a>
    <a href="#"></a>
  </div>
  <p>Pet Clinic 2021</p>
</footer>
<script src="https://code.jquery.com/jquery-3.5.1.min.js" integrity="sha256-9/aliU8dGd2tb6OSsuzixeV4y/faTqgFtohetphbbj0=" crossorigin="anonymous"></script>
<script src="js/jquery.maskedinput.js"></script>
<script src="js/jquery.validate.min.js"></script>
<script src="js/script.js"></script>
<script src="js/script_form.js"></script>
</body>
</html>
```

ДОДАТОК Б

```
*{
    margin: 0;
    padding: 0;
}
a{
    text-decoration: none;
}
li{
    text-decoration: none;
    list-style: none;
}
body{
    font-family: 'Noto Sans JP', sans-serif;
}

.logo{
    background: #000;
    width: 100%;
    display: flex;
    justify-content: center;
}

.graficlogo{
    padding: 10px;
    max-width: 100%;
    box-sizing: border-box;
}

nav{
    margin: auto;
    width: 600px;
    height: 30px;
}

.topnav{
    background-color: #FFFFFF;
}
```

```
font-size: 20px;
margin-top: 40px;
}
.topnav a{
color:#000000FF;
text-align: center;
padding: 20px;
font-size: 15px;
}
.topnav a:hover{
border-bottom: 2px solid #000000FF;
}

.topnav .icon{
display: none;
}

.mw-100{
max-width: 100%;
display: block;
margin-left: auto;
margin-right: auto;
margin-top: 0;
}
.advantages__container{
display: flex;
justify-content: center;
}
.advantages{
margin: 10px;
margin-right: 40px;
padding: 30px 30px;
width: 50%;
display: flex;
justify-content: space-between;
text-align: center;
flex-wrap: wrap;
```



```

}
.advantages__item{
    margin-top: 50px;
    width: 40%;
}

input[type=text], input[type=date], input[type=time], input[type=email] {
    width: 100%;
    display: inline-block;
    border: none;
    color: #000;
    font-weight: 300;
    font-size: 14px;
    line-height: 22px;
    background-color: #fff;
    padding: 13px;
    margin-bottom: 20px;
    width: 100%;
    -webkit-box-sizing: border-box;
    -moz-box-sizing: border-box;
    box-sizing: border-box;
    border: 0px solid transparent;
    border-bottom: 1px solid #000;
}

/* Add a background color when the inputs get focus */
input[type=text]:focus,          input[type=date]:focus,          input[type=time]:focus,
input[type=email]:focus {
    background-color: white;
    outline: none;
}

/* Set a style for all buttons */
button {
    float: center;
    width: 100%;
    border: 0px solid transparent;

```

```

        cursor: pointer;
        background-color: #000;
        color: white;
        font-size: 18px;
        padding-top: 22px;
        padding-bottom: 22px;
        transition: all 0.3s;
        margin-top: 20px;
        margin-left: 620px;
        margin-bottom: 50px;
        width: 250px;
    }

    button:hover {
        background-color: #fff;
        color: #000;
        border: 1px solid #000;
    }

    /* Extra styles for the cancel button */
    .cancelbtn {
        background-color: #000;
    }

    /* Float cancel and signup buttons and add an equal width */
    .cancelbtn, .signupbtn, .signupbtn2 {
        float: left;
        width: 50%;
        margin-left: auto;
        margin-right: auto;
        padding: 10px;
    }

    /* Add padding to container elements */
    .containers {
        padding: 18px;
    }

```

```
/* The Modal (background) */
.modal {
  display: none; /* Hidden by default */
  position: fixed; /* Stay in place */
  z-index: 1; /* Sit on top */
  left: 0;
  top: 0;
  width: 100%; /* Full width */
  height: 100%; /* Full height */
  overflow: auto; /* Enable scroll if needed */
  background-color: #000000AF;
  padding-top: 50px;
}

/* Modal Content/Box */
.modal-content {
  background-color: #fff;
  margin: 5% auto 15% auto; /* 5% from the top, 15% from the bottom and centered */
  border: 2px solid #fff;
  width: 80%; /* Could be more or less, depending on screen size */
}

/* Style the horizontal ruler */
hr {
  border: 1px solid #f1f1f1;
  margin-bottom: 20px;
}

/* The Close Button (x) */
.close {
  position: absolute;
  right: 35px;
  top: 15px;
  font-size: 40px;
  font-weight: bold;
  color: #f1f1f1;
}
```

```

}

.close:hover,
.close:focus {
    color: #000;
    cursor: pointer;
}

/* Clear floats */
.clearfix, .clearfix2::after {
    content: "";
    clear: both;
    display: table;
}
form input.error{
    color: red;
}
form label{
    position: relative;
    margin-top: 1px;
    margin-left: 10px;
}
form label.error{
    position: relative;
    color: red;
    font-size: 12px;
    top: -10px;
}

/* Change styles for cancel button and signup button on extra small screens */
@media screen and (max-width: 300px) {
    .cancelbtn, .signupbtn, button {
        width: 100%;
    }
}

footer{

```

```

        width: 100%;
        height: auto;
        background: #000;
        margin: 0 auto;
        text-align: center;
        padding-top: 25px;
    }
    footer a{
        color: white;
        font-size: 14px;
        margin-left: 10px;
    }
    footer p{
        margin-right: -25px;
        color:white;
        text-align: center;
    }
    footer nav a:hover{
        border-bottom: 1px solid white;
    }

    #myInput {
        background-image: url('loupe.png');
        background-position: 10px 12px; /* Position the search icon */
        background-repeat: no-repeat; /* Do not repeat the icon image */
        width: 100%; /* Full-width */
        font-size: 16px; /* Increase font-size */
        padding: 12px 20px 12px 40px; /* Add some padding */
        border: 1px solid #000; /* Add a grey border */
        margin: 0;
        margin-bottom: 12px; /* Add some space below the input */
        margin-top: 50px;
        width: 533px;
        margin-left: 480px;
    }

    #myTable {

```

```

border-collapse: collapse; /* Collapse borders */
width: 100%; /* Full-width */
border: 1px solid #000; /* Add a grey border */
font-size: 18px; /* Increase font-size */
margin-top: 20px;
margin-left: 150px;
margin-right: 200px;
margin-bottom: 50px;
width: 1300px;
}

#myTable th, #myTable td {
    text-align: left; /* Left-align text */
    padding: 12px; /* Add padding */
    border: 1px solid #000;
}

#myTable tr {
    /* Add a bottom border to all table rows */
    border-bottom: 1px solid #000;
}

#myTable tr.header, #myTable tr:hover {
    /* Add a grey background color to the table header and on hover */
    background-color: #f1f1f1;
}

.social img{
    margin-top: 10px;
    margin-left: 10px;
    margin-bottom: 10px;
}

.social img:hover{
    background: #494949;
    border-radius: 50%;
}

/* Slideshow container */

```

```
.slideshow-container {
  max-width: 800px;
  position: relative;
  margin: auto;
  margin-top: 30px;
}

/* Hide the images by default */
.mySlides {
  display: none;
}

/* Next & previous buttons */
.prev, .next {
  cursor: pointer;
  position: absolute;
  top: 50%;
  width: auto;
  margin-top: -42px;
  padding: 15px;
  color: white;
  font-weight: bold;
  font-size: 18px;
  transition: 0.6s ease;
  border-radius: 0 3px 3px 0;
}

/* Position the "next button" to the right */
.next {
  right: 0;
  border-radius: 3px 0 0 3px;
}

/* On hover, add a black background color with a little bit see-through */
.prev:hover, .next:hover {
  background-color: rgba(0,0,0,0.8);
}
```

```
/* Caption text */
.text {
  color: #f2f2f2;
  font-size: 15px;
  padding: 8px 12px;
  position: absolute;
  bottom: 8px;
  width: 100%;
  text-align: center;
}

/* Number text (1/3 etc) */
.numbertext {
  color: #f2f2f2;
  font-size: 12px;
  padding: 8px 12px;
  position: absolute;
  top: 0;
}

/* The dots/bullets/indicators */
.dot {
  cursor: pointer;
  height: 15px;
  width: 15px;
  margin: 0 2px;
  margin-bottom: 20px;
  background-color: #bbb;
  border-radius: 50%;
  display: inline-block;
  transition: background-color 0.6s ease;
}

.active, .dot:hover {
  background-color: #717171;
}
```



```
/* Fading animation */
.fade {
  -webkit-animation-name: fade;
  -webkit-animation-duration: 1.5s;
  animation-name: fade;
  animation-duration: 1.5s;
}

@-webkit-keyframes fade {
  from {opacity: .4}
  to {opacity: 1}
}

@keyframes fade {
  from {opacity: .4}
  to {opacity: 1}
}

.container{
  display: flex;
}

form{
  width: 50%;
  margin: 50px auto;
}

form input.name, .phone, .email{
  color: #000;
  font-weight: 300;
  font-size: 14px;
  line-height: 22px;
  background-color: #fff;
  padding: 13px;
  margin-bottom: 10px;
  width: 100%;
  -webkit-box-sizing:border-box;
  -moz-box-sizing:border-box;
  box-sizing: border-box;
```

```
        border: 0px solid transparent;
        border-bottom: 1px solid #000;
    }
    textarea{
        width: 100%;
        height: 150px;
        resize: vertical;
    }
    form button.send-form{
        position: relative;
        float: center;
        width: 100%;
        border: 0px solid transparent;
        cursor: pointer;
        background-color: #000;
        color: white;
        font-size: 18px;
        padding-top: 22px;
        padding-bottom: 22px;
        transition: all 0.3s;
        margin-top: 3px;
        font-weight: 300;
        margin-right: 200px;
        right: 619px;
    }
    .send-form:hover{
        background-color: #fff;
        color: #000;
        border: 1px solid #000;
    }
    .ease{
        width: 0;
        height: 74px;
        background-color: #fbfbfb;
        transition: .3s ease;
    }
```

```
.map {
    display: flex;
    width: 100%;
    margin-top: 60px;
    margin-left: 150px;
    margin-bottom: 60px;
}

.contact{
width: 100%;
margin: 0;
padding: 0;

}

.contact_item{
position: relative;
text-align: left;
margin-top: 70px;
font-size: 16px;
margin-left: 60px;
padding-top: 30px;
margin-bottom: 50px;
}

.mw-200{
    position: absolute;
    margin-left: 300px;
    margin-top: -295px;
}

.question{
    position: absolute;
    margin-left: 874px;
    margin-top: -100px;
}

@media screen and (max-width: 780px){
    .topnav a:not(:first-child) {
```

```

        display: none;
    }
    .topnav a.icon{
        float: right;
        margin-top: -13px;
        display: block;
        color: #000000FF;
    }
    nav{
        width: 100%;
    }
    .topnav.responsive{
        position: relative;
    }
    .topnav.responsive a.icon{
        position: absolute;
        right: 0;
        top: 0;
    }
    .topnav.responsive a {
        float: none;
        display: block;
        text-align: left;
    }
    .advantages__item{
        width: 50%;
    }
    @media screen and (max-width: 620px){
        .advantages{
            flex-direction: column;
        }
    }
    .advantages__item{
        width: 100%;
    }
}
}

```