

Пояснювальна записка

до дипломної роботи

бакалавр

(освітньо-кваліфікаційний рівень)

на тему «Розробка веб-додатка для написання видавничої продукції»

Виконала: студентка 4 курсу, групи ІІЗ-15д
спеціальності 121 „Інженерія програмного
забезпечення”

_____ Пашенко Р.М.

(підпис)

Керівник,

професор, д.т.н. _____ Марченко Д.М.

(підпис)

Рецензент,

доцент, к.т.н. _____ Митрохін С.О.

(підпис)

ЛИСТ ПОГОДЖЕННЯ І ОЦІНЮВАННЯ
дипломної роботи студентки гр. ІПЗ-15д Пащенко Р.М.

Науковий керівник

Професор, д.т.н. _____ Марченко Д.М.

Оцінка наукового керівника: _____

Рецензент Митрохін С.О., доцент каф. ПМ СНУ ім. В. Даля

ПБ, місто роботи, посада

Оцінка рецензента: _____

Кінцева оцінка за результатами захисту: _____

Голова ЕК,
Зав. Кафедри ПМ
д.т.н., доцент

_____ Лифар В.О.
підпис

Факультет інформаційних технологій та електроніки

Кафедра програмування та математики

Освітньо-кваліфікаційний рівень бакалавр

Спеціальність 121 „Інженерія програмного забезпечення”

ЗАТВЕРДЖУЮ

Завідувач кафедри ПМ,

д.т.н., доцент

_____ Лифар В.

«___» _____ 2019 р.

З А В Д А Н Н Я НА ДИПЛОМНУ РОБОТУ СТУДЕНТКИ ПАЩЕНКО РАЇСИ МИКОЛАЇВНИ

1. Тема роботи Розробка веб-додатка для написання видавничої продукції.

керівник роботи Марченко Д.М.

затверджені наказом вищого навчального закладу від “23” квітня 2019 року
№ 68/14.04

2. Строк подання студентом роботи 07 травня 2019 р.

3. Вихідні дані до роботи

Об'єктом даної роботи є розробка веб-додатка для написання видавничої продукції.

3.1 Літературні джерела:

Гутманс Е. РНР 5. Професійне програмування : Гутманс Е. — М: 2006. — 1418 с.

Давид Скляр. РНР. Рецепты программирования. 3-е изд.: Скляр Д. — М: 2017. — 369 с.

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

4.1 Вступ

4.2 Аналіз предметної галузі (огляд літератури), з висвітленням наступних питань:

Аналіз предметної області.

Діаграма варіантів та вимоги до програми.

4.3 Основна частина, в якій висвітлити:

Система повинна мати такий функціонал.

Розробка структури веб-ресурсу.

Опис режимів роботи.

4.4 Висновки

4.4 Перелік використаних джерел

5. Перелік графічного матеріалу немає

6. Дата видачі завдання 12 лютого 2019 року.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломної роботи	Строк виконання етапів роботи	Примітка
1	Одержання завдання на виконання роботи	12.02.19	
2	Укладання і погодження з керівником плану і етапів виконання роботи	20.02.19	
3	Узагальнення даних літературних джерел, укладання розділу «Аналіз предметної галузі»	1.03.19	
4	Аналіз шляхів виконання завдання. Вибір і погодження з керівником оптимального шляху	10.03.19	
3	Проектування інфологічної моделі задачі що реалізується.	01.05.19	
5	Укладання та тестування програмного продукту	10.05.19	
6	Укладання, оформлення та погодження пояснювальної записки з керівником	20.05.19	
7	Укладання, оформлення та погодження з консультантом розділу «Охорона праці»	15.05.19	
7	Здача готової пояснювальної записки на кафедрі	07.06.19	
8	Укладання доповіді і презентації	10.06.19	

Студент

(підпис)

Пащенко Р.М.

Керівник роботи

(підпис)

Марченко Д.М.

РЕФЕРАТ

Текст-69, рисунків-10, таблиць-8 , джерел літератури-27, додаток-1.

Метою роботи – є розробка веб-додатка для написання видавничої продукції, який допоможе окремим видавцям зосередитися на своєму бізнесі та позбутися рутинної роботи з написанням книг.

За допомогою веб-технологій такі як PHP – для розробки серверної частини сайту, HTML 5 – в якості базової розмітки та Bootstrap 3 – для оформлення вебресурсу і його адаптивності на різних пристроях, було розроблено веб-додаток. На підставі виконаної роботи, можна зробити висновок про те, що поставлені до даного проекту вимоги виконані.

САЙТ. СТВОРЕННЯ САЙТУ. КНИГА. АВТОР. ЗАМОВНИК.
ЗАМОВЛЕННЯ.

ЗМІСТ	
ВСТУП.....	7
1 АНАЛІТИЧНИЙ ОГЛЯД.....	9
1.1 Аналіз предметної області	9
1.2 Логічна структура сайту	10
1.3 Вимога до програми	11
1.4 Аналіз існуючих систем	13
1.5 Діаграма прецедентів	15
2 ПОСТАНОВКА ЗАДАЧІ	18
2.1 Функціональні та нефункціональні вимоги до веб-додатка	18
2.2 Система повинна мати такий функціонал	18
2.3 Нефункціональні вимоги	20
2.4 Посадові обов'язки адміністратора веб-додатка	20
3 ПРОГРАМНА РЕАЛІЗАЦІЯ	22
3.1 Вибір сучасних веб-технологій для розробки веб-ресурсів	22
3.2 Вибір систем управління базами даних	28
3.3 Вибір середовища розробки	31
3.4 Розробка структури веб-ресурсу	33
3.5 Проектування бази даних	35
3.6 Фізична структура	37
3.7 Реалізація основних функцій	37
3.8 Опис режимів роботи	44
ВИСНОВКИ	54
СПИСОК ЛІТЕРАТУРИ	55
ДОДАТОК	57

Кожна організація має власний веб-сайт. В умовах використання сучасних інформаційних технологій - це необхідний чинник існування, що дозволяє розширити поле рекламної діяльності і притягнути тим самим додаткових клієнтів. Веб-дизайнери швидко зрозуміли ефективність нового засобу масової інформації. Сьогодні наявність власного сайту вважається критерієм сучасного підприємства або фірми.

Веб-сайтом називається сукупність будь-яких електронних документів, об'єднаних під одним доменним ім'ям в мережі. Якісний сайт працює на імідж компанії. У його функції входить надання інформації мільйонам відвідувачів двадцять чотири години на добу. Створити якісний сайт непросто, для цього будуть потрібні не тільки навички дизайнера для красивого і стильного оформлення, а й досвід програмування. Складність і обсяг програм, необхідних для створення того чи іншого сайту, залежить від безлічі факторів, таких, як структура, мета, необхідна функціональність, забезпечення надійності. Створення веб-сайтів за допомогою новітніх технологій дозволяє не тільки рекламувати свій бізнес в інтернеті, а також ефективно просувати його через всесвітню павутину, знаходити нових клієнтів і вести співпрацю по всьому світу.

Багато людей мріють написати і випустити хоча б одну книгу. А деякі перетворюють письменницьке ремесло в основне джерело стабільного доходу. Причому для цього зовсім необов'язково мати багату фантазію і вміти вибудовувати хвацько закручені сюжетні лінії.

Метою даного проекту розробити веб – додаток для написання видавничої продукції, який допоможе окремим видавцям зосередитися на своєму бізнесі та позбутися рутинної роботи з написанням книг. З цього ми розробили веб - додаток для оформлення замовлення для написання книжок, і для їх редагування.

Об'єктом дослідження: є ефективне подання матеріалу за допомогою сучасних веб-технологій та основні процеси електронної комерції.

Предметом дослідження: є принципи та методи розробки сучасного веб-додатку.

Мета дослідження: розробка веб-додатка для написання видавничої продукції.

Методологічна та теоретична основа дослідження: розробка веб-додатка.

1.1 Аналіз предметної області

Веб - додаток буде призначений для такої роботи як, допомогти клієнтам з написанням книг та для редагуванні книжок, він буде називатися «Getitwritten.io».

Автори з веб – додатку зможуть на замовлення писати книжки будь якого жанру, будь-то комп'ютерні технології, бізнес та гроші, освіта та викладання, інженерія та транспорт, та багато інших жанрів. Ми будемо ретельно збирати і аналізувати матеріал, підбирати приклади, перевіряти факти і кілька разів редагувати написаний текст. До нас звертатимуться люди, які занадто зайняті, щоб писати самостійно.

Виставляємо рекламу по якій можна перейти на наш веб - додаток. Замовник заходить на наш сервіс де може ознайомиться з нашими послугам, будь то написання книги, або редагування книги. Як будується робота з нашими клієнтами. Подивитися нашу статистику і почитати відгуки про нашу виконану роботу з замовниками які до нас зверталися. Подивитись в яких жанрах ми пишемо книги.

Розрахувати самостійно ціну через наш калькулятор який знаходиться на веб - додатку. Де потрібно вибрати жанр який потрібен , ввести кількість сторінок, і калькулятор розрахує скільки це буде коштувати. Ціна розраховується в доларах. Ми будемо працювати при передоплаті п'ятдесят відсотків.

Після чого замовник залишає нам своє замовлення. Для початку роботи нам знадобиться детально заповнений бриф і всі напрацювання, які є. Вони можуть бути в будь-якому вигляді: аудіо- та відеофайли, рукопис або роздруківки, в загальному, все, що необхідно включити в книгу. До рукопису обов'язково додається синопсис - короткий виклад змісту книги і невеликий опис. Якщо мова йде не про художній твір, то в синопсисі вказується кваліфікація автора докази його компетенції, короткий зміст, вказівка можливої цільової аудиторії і додаткові пояснення при необхідності. Ми розглядаємо заказ, зв'язуємося з замовником для підтвердження деталей і

переступаємо до роботи. Книги, написані на замовлення, виходять або під іменами авторів, реально написали книги, або - під ім'ям замовника написаної книги. Авторство книги, написаної на замовлення, визначає сам замовник - видання книги можливо під ім'ям одного учасника або групи авторів.

Після виконаної роботи, адміністратор здає її замовнику, дотримуючись конфіденційності та анонімності.

В середньому автору для написання книги в 200-500 сторінок знадобиться від 3 місяців до півроку.

1.2 Логічна структура сайту

Логічна структура сайту може бути представлена у вигляді чотирьох основних моделей: лінійна модель, моделі "решітка", "дерево" (ієрархія) і "павутина". Крім того, існують різні комбінації на базі основних моделей, що дозволяють реалізувати будь-яку логічну структуру сайту.

Строго лінійна структура сайту використовується досить рідко, оскільки дозволяє відвідувачам сайту переходити тільки від попередньої сторінки до наступної, і ніяк інакше. [1]

Строго лінійна структура сайту використовується досить рідко, оскільки дозволяє відвідувачам сайту переходити тільки від попередньої сторінки до наступної, і ніяк інакше.

Застосовується лінійна структура сайту для різних онлайн-презентацій, опису покрокових процесів. Основна перевага застосування такої моделі організації сайту полягає в передбачуваності поведінки відвідувача на сайті, завдяки чому можливо передбачити дії відвідувачів ще на етапі створення сайту.

Лінійна структура сайту не підходить для сайтів з великим об'ємом інформації. Для більш зручного представлення великих обсягів інформації використовуються інші моделі логічної структури, наприклад, «грати».

Логічна структура сайту "дерево" - це найбільш часто зустрічається модель організації сайту. "Дерево" - ієрархічна структура - дозволяє відвідувачам сайту за своїм бажанням керувати глибиною відвідування

сайту. Відвідувачі можуть заходити лише на сторінки верхніх рівнів ієрархії або ж «спуститися» до нижніх рівнів. При цьому свобода вибору відвідувачів залежить від «ширини дерева».

Якщо відвідувачам для досягнення кінцевої мети необхідно зробити дуже багато клацань миші, то ієрархічна структура сайтів може виявитися в даному випадку занадто вузькою. Відвідувачів дратує нескінченне «клацання», що не приносить очікуваних результатів.

У той же час, занадто широке «дерево», ґрунтуючись на дуже великій кількості варіантів вибору, може змусити відвідувачів витратити багато часу на вивчення наданих варіантів. Це також призводить до подразнення відвідувачів і вони залишають сайт.

Якщо логічна структура сайту заплутана і незрозуміла, то таку модель організації сайту прийнято називати "павутиною". Така структура сайту може з'явитися внаслідок неправильного проектування сайту або ж навмисно, для забезпечення більш виразного представлення інформації. Наприклад, статті про технічні виробы можуть посилатися на схеми цих виробів, норми і вимоги до виробництва виробів, ряд інших документів. Такий підхід дозволяє відвідувачам отримати максимально повну інформацію.

Якщо модель організації сайту обрана вірно, а логічна структура сайту відповідає потребам відвідувачів, то юзабіліті і ефективність такого сайту будуть високими. Тому до розробки структури сайту слід підійти дуже ретельно.

1.3 Вимога до програми

Вимоги до функціональних характеристик для забезпечення надійності даного веб-додатка для компанії «Getitwritten.io» по написанню книг:

- розробити зручний для користувача інтерфейс, щоб навіть людина, не працював з комп'ютером досить, швидко міг розібратися в цьому веб-додатку;

- дизайн повинен бути виконаний не в збиток зручності навігації по сайту, тобто надавати велике навантаження на ресурси ПК і каналів зв'язку, що дозволить збільшити швидкість роботи веб-додатку;
- зручний призначений для користувача інтерфейс;
- зручна навігація, основні розділи сайту краще розмістити зверху;
- контакти це обов'язкова умова, зв'язок з компанією повинна бути зручною і самої прямої;
- сумісність браузерів. Сайт повинен бути сумісний з усіма браузерами та платформами;
- калькулятор який повинен розраховувати ціну.

Вимоги до надійності веб-додатка. Веб-інтерфейс повинен бути правильним, якісним, тобто повинен безпомилково виконувати необхідні функції, описані вище, і бути придатним для експлуатації. Програмний засіб має бути стійким, тобто обчислювальна система, що реалізує її повинна функціонувати успішно, незважаючи на випадкові відхилення, як на зовнішніх умовах, так і всередині самої програми. Таким чином, для забезпечення надійності необхідно:

- наявність головного розділу, тобто головного модуля, з якого відбувається перехід на необхідний розділ;
- всі дані повинні бути введені правильно і коректно;
- розмежування доступу до певних розділів.

Вимоги до надійності інформації. Для забезпечення надійності інформації, що зберігається в таблицях бази даних, база даних знаходиться тільки на веб-сервері. Доступ до таких даних має тільки адміністратор сайту.

Вимоги до інтерфейсу. Для зацікавленості користувача при роботі з веб-сайтом для «Getitwritten.io», необхідно створити зручний і прийнятний інтерфейс, який відповідає таким умовам:

- веб-додаток повинен містити конкретну інформацію і велика кількість матеріалів певної тематики. Що дозволить зібрати велику аудиторію його відвідувачів;

- веб-додаток повинен бути оформлений на високому рівні. При оформленні необхідно враховувати розмір графічних файлів, виправданість застосування деяких технологій, правильність і сумісність колірного балансу, грамотність пропорцій і перспектив за обсягом і висвітлення, правильне асоціативне сприйняття графічного оформлення, відповідно до тематики сайту і колірного рішення;
- веб-додаток повинен бути ретельно продуманий. Користувач повинен легко розуміти структуру сайту по його меню, а інструменти навігації сайту повинні бути зрозумілі;
- в веб-додатку не повинно бути помилок, помилок і поганий літературної стилістики в тексті. Не дотримання правил верстки тексту, підбору шрифтів і їх розміру - також знижують його оцінку користувачем;
- для того щоб користувачі постійно поверталися, сайт повинен постійно оновлюватися. Оновлення застарілої інформації і поповнення нової повинне проводитися регулярно;
- веб-додаток не повинен містити зайвий код, а програмні рішення повинні працювати без помилок.

Для експлуатації веб-додатка необхідно мати на комп'ютері будь-яку операційну систему, але браузер повинен підтримувати Java, HTML5, CSS3.

1.4 Аналіз існуючих систем

У даний момент актуальні сайти з написанням книжок в інтернеті є:

1. Pr-kniga;
2. White-pr;
3. Чернильный гусь.

1. «Pr-kniga» <http://www.pr-kniga.com/>;

Переваги сайту:

- **багаторічний досвід написання книг на замовлення;**
- зручний інтерфейс;
- доступні ціни.

Недоліки сайту:

- пишуть тільки ділові та науково-популярні книги;
- відсутня форма зворотного зв'язку.

2. «White-pr» <http://white-pr.com.ua>;

Переваги сайту:

- доступні приклади текстів;
- постійним клієнтам знижка.

Недоліки сайту:

- не зручний інтерфейс для користувача;
- відсутні відгуки від клієнтів.

3. «Чернильный гусь» <http://inkgoose.ru>.

Переваги сайту:

- доступні відгуки від клієнтів;
- сучасний і зручний інтерфейс.

Недоліки сайту:

- маленький вибір жанрів;
- немає можливості підписатися для розсилки.

1.5 Діаграма прецедентів

Діаграма прецедентів в UML - діаграма, що відображає відносини між акторами і прецедентами і є складовою частиною моделі прецедентів, що дозволяє описати систему на концептуальному рівні. [2]

Прецедент - можливість модельованої системи (частина її функціональності), завдяки якій користувач може отримати конкретний, вимірний і потрібний йому результат. Прецедент відповідає окремому сервісу системи, визначає один із варіантів її використання і описує типовий спосіб взаємодії користувача з системою. Варіанти використання зазвичай застосовуються для специфікації зовнішніх вимог до системи.

Діаграма прецедентів є [графом](#), що складається з множини акторів, прецедентів (варіантів використання) обмежених границею системи (прямокутник), асоціацій між акторами та прецедентами, відношень серед прецедентів, та відношень узагальнення між акторами.

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 ([англ.](#) use case) 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:

- асоціації ([англ.](#) association relationship)
- включення ([англ.](#) include relationship)
- розширення ([англ.](#) extend relationship)
- узагальнення ([англ.](#) generalization relationship)

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 ([англ. include](#)) у мові UML — це різновид відношення залежності між базовим варіантом використання і його спеціальним випадком. При цьому відношенням залежності ([англ. dependency](#)) є таке відношення між двома елементами моделі, при якому зміна одного елемента (незалежного) приводить до зміни іншого елемента (залежного).

Відношення розширення ([англ. extend](#)) визначає взаємозв'язок базового варіанта використання з іншим варіантом використання, функціональна поведінка якого задіюється базовим не завжди, а тільки при виконанні додаткових умов. Діаграми прецедентів відображають елементи моделі варіантів використання. [2]

На діаграмі прецедентів (рис. 1.1) представлені три «актора».

- Адміністратор;
- Автор;
- Замовник.

Адміністратор має процеси: передавати замовлення; зв'язуватися для підтвердження наданих відомостей для замовлення; розробка, підтримка концепції веб - додатка, покращення його роботи; удосконалення та виправлення поточної структури веб-ресурсу; зворотній зв'язок з відвідувачами ресурсу, робота з коментарями, відповіді на повідомлення; аналітика відвідуваності, залучення відвідувачів, конверсії, розробка планів подальших публікацій; віддати готову книгу; зробити замовлення; передати книгу для перевірки і підтвердження замовлення.

Автор має процеси: передати замовлення; написання книги; редагування книги; шукає інформацію за матеріалом; віддати готову книгу.

Замовник має процеси: розрахувати ціну; зробити замовлення; зв'язуватися для підтвердження наданих відомостей замовлення; підписатися

на поновлення; залишити відгуки; прийняти книгу для перевірки і підтвердження замовлення.



Рисунок 1.1 — Діаграма прецедентів

Велика перевага веб-додатків у тому, що організація їх роботи можлива з використанням opens source продуктів. Так, розгорнувши на комп'ютері операційну систему Linux (наприклад Debian), веб сервер Apache, інтерпретатор мови програмування PHP та сервер даних MySQL можна влаштувати роботу веб-додатку. Велика кількість веб застосунків працює з використанням саме таких програмних продуктів. Для розробки та функціонування веб сайтів необхідна мінімальна програмна платформа, що дуже відрізняє цей тип програмних продуктів від решти інших.

2.1 Функціональні та нефункціональні вимоги до веб-додатка

Функціональні вимоги ідентифікують завдання або дії, які повинні бути виконані. Функціональні вимоги визначають дії, які система повинна бути здатною виконати, зв'язок входу виходу в поведінці системи.

Нефункціональні вимоги - це вимоги, які визначають критерії роботи системи в цілому, а не окремі сценарії поведінки. Нefункціональні вимоги визначають системні властивості, такі як продуктивність, зручність супроводу, розширюваність, надійність, середовищні чинники експлуатації.

Була поставлена задача – розробити веб-додаток для написання видавничої продукції.

2.2 Система повинна мати такий функціонал

- опис чим ми займаємося;

Важливо використовувати ключові слова в описі веб-додатка, ресурс буде більш авторитетним і завоює вищий рейтинг в пошукових системах, якщо використовувати більше посилань з ключовими фразами.

Це зробить популярним наш веб – додаток в інтернеті.

- прайс-калькулятор;

Дозволить розрахувати ціну за написання книги, або редагування її. Обираєм тип послуги, категорію, і вводимо кількість сторінок.

Ця можливість дозволить клієнтам самостійно розраховувати ціни без допомоги адміністратора.

- оформлення замовлення клієнтом;

Замовник заповнює форму де потрібно вести свій e-mail, і заповнює поле де пише про замовлення, з додатковими даними якщо потрібно.

Ця можливість дозволяє зробити замовлення.

- при замовленні можна отримати знижку двадцять відсотків;

На веб – додатку буде доступна форма зі знижкою двадцять відсотків, котру зможе отримати кожен.

Ця можливість допоможе в залученнях нових клієнтів, підвищення обсягів заказів, стимулювання постійних клієнтів на замовлення книги.

- замовник зв'язується з нами через e-mail;

Буде доступна форма де можна буде задавати питання, клієнт присилає запитання на нашу пошту на яке відповідь адміністратор.

Ця можливість це зручний інструмент комунікації між співробітниками і клієнтами, це ще й потужний інструмент для розвитку та реклами вашого бізнесу.

- можливість подивитись статистику і відгуки клієнтів;

Буде відображатись відгуки попередніх клієнтів, та статистика наших написаних книг та відредагованих. Статистика буде оновлюватися кожен раз після зробленої роботи. Знизу буде кнопка, де при натисканні можна зразу перейти до оформлення замовлення.

Ця можливість відгуків це одна з форм прояву соціального схвалення. Відгуки клієнтів володіють неймовірною силою переконання, що допоможе залучити більше клієнтів. можливість подивитися як працює система замовлення;

- зображення як працює веб - додаток;

Буде зображення як працює наш веб - додаток з клієнтами, для наглядного прикладу.

Ця можливість з зображенням на сайті викликає інтерес і занурюють клієнта в контекст. Разом з текстом фотографія працює на головну мету - донести до аудиторії повідомлення про сервіс.

- підписка на наш сайт;

Форма де клієнт заповнює туди свій e-mail, і йому приходить на пошту наші оновлення, або якісь пропозиції.

Ця можливість дає відстежити реакцію користувачів на ваш лист - відсоток відкриття листів, актуальність тематики, попит на пропозиції і корисність пропонованих заглядів.

2.3 Нефункціональні вимоги

- для пошуку по сайту, користувач переміщається по сайту по кнопках.
- система працює на будь-якій ОС в якій присутній мережу Інтернет.
- зручний і зрозумілий інтерфейс - на прохання постановника завдання інтерфейс не повинен виділятися яскравими квітами, бажано, щоб мав стандартні кольори, шрифти та елементи.

2.4 Посадові обов'язки адміністратора веб-додатка

Адміністратор сайту виконує такі посадові обов'язки:

1. Всіляко сприяє просуванню і популяризації сайту в Інтернеті.
2. Розробляє і підтримує основну концепцію сайту, вносить пропозиції щодо доопрацювання концепції та змісту сайту, по введенню нового сервісу для відвідувачів сайту.
3. Стежить за текстовим наповненням сайту, постійним оновленням інформації.
4. Переглядає всі опубліковані на сайті документи і матеріали, видаляє і редагує матеріали, що не відповідають загальній концепції і правилам користування сайтом.

5. Виконує роботи по редагуванні і коректури матеріалів і їх структурування, включаючи дизайн оформлення текстів і так далі.

6. Досліджує потреби і запити відвідувачів сайту.

7. Відстежує роботу конкурентів, тобто сайтів зі схожою тематикою, концепцією і змістом.

Для розробки веб-додатка було обрано таке програмне середовище як: HTML 5 і CSS 3, PHP, Sublime Text 3, Bootstrap 3 , СУБД MySQL.

Ще використовуються такі додатки як Facebook, Instagram, Twitter. Наш продукт вимагає інтеграції з цими додатками.

3.1 Вибір сучасних веб-технологій для розробки веб-ресурсів

HTML5 нова відкрита платформа, призначена для створення вебдодатків застосовуючи аудіо, відео, графіку, анімацію та багато іншого.

Мета розробки HTML5 - поліпшення рівня підтримки мультимедіа технологій з одночасним збереженням зворотної сумісності, зручності читання коду для людини і простоти аналізу для парсерів .

HTML5 був створений як єдина мова розмітки, який міг би поєднувати синтаксичні норми HTML і XHTML. Він розширює, покращує і раціоналізує розмітку документів, а також додає єдиний API для складних веб-додатків

Ряд нових елементів в HTML5 дозволяє позбутися від необхідності використання для відображення сайту в браузері сторонніх розширень . До таких належать елементи audio і video, що забезпечують інтеграцію мультимедійних даних і здійснюють відтворення мультимедійного контенту в браузері без програвача Flash Player. [3]

Основні переваги HTML5:

1. Структура (семантика) документа. В HTML нарешті з'явився цілий ряд нових семантичних тегів, які служать для створення більш змістовної організації веб-сторінок. Це такі теги, як: `<header>`, `<article>`, `<footer>`, `<nav>`, `<aside>`, `<section>`. Деякі теги HTML4 визнали застарівшими, деякі змінили своє значення і атрибути.

2. Графічні можливості. Новий елемент створений `<canvas>` створення для роботи з 2D графікою, що створює безліч нових можливостей для

впровадження на сторінки. <canvas> являє собою динамічну поверхню для програмного малювання та різних операцій над графікою.

3. Мультимедіа. Додана підтримка мультимедійного контенту за допомогою тегів <video> та <radio>.

4. Геолокація. Зчитування геолокації стало можливо завдяки функціям `getCurrentPosition` і `watchPosition`. Щоправда браузер буде запитувати дозволу перед проведенням операції, оскільки в специфікації вказано, що без дозволу браузери не мають права відправляти геодані.

5. JavaScript API. API для роботи з графікою і мультимедіа в новому HTML5, перетягування об'єктів(Drag & Drop методи) та інші можливості. 6. Нові елементи веб-форм. З'явилися також нові елементи для вебформ, які знову ж таки розширюють функціонал сторінки і поле для фантазії кодера.

7. Локальне сховище. Кешування даних не лишилось не поміченим, куки, які вже морально застаріли і не всюди годяться, тепер можуть бути замінені на локальне сховище на стороні клієнта. Інформацію можна зберігати на пристрої клієнта в набагато більшому розмірі і з ширшими можливостями, ніж у куків. [4]

HTML 5 дозволяє прискорити завантаження сторінок сайту , додає нові можливості на сайт. Якщо ж якась із функцій HTML 5 не підтримується браузером, то це не завдає жодної шкоди сайту, адже HTML 5 ґрунтується на HTML 4 й не підтримувані елементи просто ігноруються. [5]

PHP — скриптова мова програмування, була створена для генерації HTML-сторінок на стороні веб-серверу. PHP є однією з найпоширеніших мов, що використовуються у сфері веб-розробок. PHP підтримується переважною більшістю хостинг-провайдерів. Проект за яким був створений PHP — проект з відкритими програмними кодами. [6]

PHP інтерпретується веб-сервером в HTML-код, який передається на сторону клієнта. На відміну від таких скриптових мов програмування, як JavaScript, користувач не має доступу до PHP-коду, що є перевагою з точки зору безпеки але значно погіршує інтерактивність сторінок. Але ніщо не забороняє використовувати PHP для генерування і JavaScript-кодів які виконуються вже на стороні клієнта. [7]

PHP - мова, яка може бути вбудована безпосередньо в html-код сторінок, які, в свою чергу коректно будуть оброблені PHP -інтерпретатором. Механізм PHP просто починає виконувати код після першої екрануючої послідовності (<?) і продовжує виконання до того моменту, коли він зустрине парну екрануючу послідовність (?>). [6]

Велика різноманітність функцій PHP дають можливість уникнути написання багаторядкових призначених для користувача функцій на C або Pascal. [9]

1. Наявність інтерфейсів до багатьох баз даних - в PHP вбудовані бібліотеки для роботи з MySQL, PostgreSQL, mSQL, Oracle, dbm, Hyperware, Informix, InterBase, Sybase. - через стандарт відкритого інтерфейсу зв'язку з базами даних (Open Database Connectivity Standard — ODBC) можна підключатися до всіх баз даних, до яких існує драйвер.

2. Традиційність. Мова PHP здаватиметься знайомою програмістам, що працюють в різних областях. Багато конструкцій мови запозичені з C, Perl. Код PHP дуже схожий на той, який зустрічається в типових програмах на C або Pascal. Це помітно знижує початкові зусилля при вивченні PHP. PHP - мова, що поєднує переваги Perl і C і спеціально спрямована на роботу в Інтернеті, мова з універсальним і зрозумілим синтаксисом. І хоча PHP є досить молодою мовою, вона здобула таку популярність серед web-програмістів, що на даний момент є мало не найпопулярнішою мовою для створення web-додатків (скриптів).

3. Наявність вихідного коду та безкоштовність. Стратегія Open Source, і розповсюдження початкових текстів програм в масах, безсумнівно справили благотворний вплив на багато проектів, в першу чергу - Linux хоч і успіх проекту Apache сильно підкріпив позиції прихильників Open Source. Сказане 30 відноситься і до історії створення PHP, оскільки підтримка користувачів зі всього світу виявилася дуже важливим чинником в розвитку проекту PHP. Ухвалення стратегії Open Source і безкоштовне розповсюдження початкових текстів PHP надало неоціниму послугу користувачам. Додатково, користувачі PHP по всьому світу є свого роду колективною службою

підтримки, і в популярних електронних конференціях можна знайти відповіді навіть на найскладніші питання.

4. Ефективність. Ефективність є дуже важливим чинником при програмуванні для середовищ розрахованих на багато користувачів, до яких належить і web. Важливою перевагою PHP є те, що ця мова належить до інтерпретованих. Це дозволяє обробляти сценарії з достатньо високою швидкістю. За деякими оцінками, більшість PHP-сценаріїв (особливо не дуже великих розмірів) обробляються швидше за аналогічні їм програми, написані на Perl. Проте, щоб не робили розробники PHP, виконувані файли, отримані за допомогою компіляції, працюватимуть значно швидше - в десятки, а іноді і в сотні разів. Але продуктивність PHP цілком достатня для створення цілком серйозних web-додатків. [7]

Bootstrap - фреймворк, набір HTML + CSS інструментів і шаблонів для верстки і більш ефективного і швидкого створення сайтів і веб-додатків більш ефективно і швидко. Бутстрап - сучасний помічник розробників інтерфейсів, дизайнерів і вебмайстрів, доступний для використання з відкритою ліцензією. [12] Бутстрап зараз на піку популярності і можна впевнено сказати, що на його основі можна зробити будь-який веб-інтерфейс.

У ньому є такі шаблони:

- шрифти;
- кнопки;
- форми;
- мітки;
- навігація;
- сітка;
- javascript;
- розширення;
- інше.

Основні переваги бутстрапа:

1. Економія часу (а значить і грошей) - ви економите зусилля бо використовуєте вже готові класи і дизайн.

2. Адаптивність (mobile first), висока швидкість і оптимізація, стандартизація інтерфейсів - динамічні макети бутстрапа якісно відображаються на самих різних пристроях без необхідності внесення змін в розмітку.

3. Дизайн - єдині шаблони і стильове оформлення елементів макета і всіх сторінок на сайті в цілому. І при цьому Bootstrap крос-браузерний і добре відображається у всіх браузерах.

4. Простота і відкритість - використовувати бутстрап настільки просто, що з ним справляються навіть початківці веб-розробники, а відкритий вихідний код дозволяє самому брати участь в розробці, модифікувати під свої потреби або просто користуватися хорошим безкоштовним рішенням.

При цьому код HTML, JavaScript і CSS в бутстрап продуманий і розглянутий під мікроскопом сотнями розробників з усього світу - все для того, щоб пересічні вебмайстера і верстальники могли легко і просто налаштувати сітку сайту або вбудувати необхідні елементи в інтерфейс. Також, в Bootstrap використовується динамічна мова стилів LESS, яка розширює можливості CSS.

JavaScript - це об'єктно скриптова мова, розроблена компанією Netscape, яка використовується мільйонами веб-сторінок і серверних додатків по всьому світу. JavaScript є розширенням мови ECMA-262 Edition 3 (ECMAScript) з невеликими відмінностями від опублікованого стандарту. [8]

Всупереч поширеній думці, JavaScript не є "інтерпритатором Java". JavaScript - це динамічна скриптова мова, що підтримує прототип створення об'єктів. Базовий синтаксис навмисно схожий на Java і C ++, щоб зменшити число нових концепцій, необхідних для вивчення мови. Такі мовні конструкції, як if, for, while, switch, try ... catch схожі на конструкції цих мов.

JavaScript може функціонувати і як процедурна, і як об'єктноорієнтована мова. На відміну від синтаксичних визначень класів у компільованих мовах, таких як C ++ чи Java, в JavaScript об'єкти можуть бути створені програмно під час виконання. Після того, як об'єкт був створений,

він може бути використаний як креслення (або прототип) для створення схожих об'єктів.

Динамічні можливості JavaScript включають: створення об'єктів під час виконання, змінне число параметрів, динамічне створення скриптів (за допомогою eval), інтроспекція об'єктів (за допомогою for ... in), відновлення вихідного коду (програми на JavaScript можуть декомпілювати тіла функцій назад у вихідний код).

Для того, щоб повідомити браузеру, що в документі HTML вмонтований сценарій JavaScript, використовують дескриптор. В початковому коді HTML ця команда повинна бути вказана між дескрипторами мови.

Сучасні гіпертекстові інформаційні системи умовно можна уявити у вигляді сукупності декількох комбінацій:

- системи збереження гіпертекстових об'єктів;
- системи відображення гіпертекстових об'єктів;
- системи підготовки гіпертекстових об'єктів;
- системи програмування переглядом сукупності гіпертекстових об'єктів.

Програми перегляду гіпертекстових сторінок традиційно називають скриптами (scripts). Як це було раніше і в локальних системах, в програмуванні перегляду гіпертекстових документів web існує два методи: 1. створення скриптів, які інтерпретуються програмою перегляду (технологія JavaScript). [13]компіляція байткоду (технологія Java). При першому методі для розробки гіпертекстової сторінки потрібний тільки звичайних текстовий редактор і сам і сам гіпертекстовий документ повинен легко читатися людиною-оператором (користувачем). Другий підхід дозволяє збільшити ефективність виконання програм та захист кодів від несанкціонованих модифікацій. Байткоди або мобільні коди забезпечують технологію програмування на Java.

Таблиця 3.1 – Порівняльна характеристика сучасних веб-технологій (п'ятибальна система)

Критерії	HTML	PHP	JavaScript
Об'єктно орієнтована	0	1	1
Простота у використанні	1	0	0
Чутливість до реєстру змінних	0	1	1
Чутливість до реєстру функцій	0	0	1
Оновлюваність файлів на сервері	0	1	0
Підтримка фреймворків	1	1	1
Підтримка БД	0	1	0
Сумарний коефіцієнт	2	5	3

3.2 Вибір систем управління базами даних

СУБД MySQL - є однією з найвідоміших, надійніших і швидших зі всього сімейства існуючих СУБД. Однією з причин є правила її розповсюдження – що є безкоштовна і розповсюджується вона разом зі своїми початковими текстами, інша причина - це те, що MySQL відносно швидка СУБД. PostgreSQL, наприклад, також розповсюджується під подібною ліцензією, але вона не набула такого широкого поширення. Одна з причин - це помітна повільність. Отже, дві головні причини популярності MYSQL: ціна і продуктивність. [13]

MySQL написаний для близько десяти видів операційних систем. Це і FreeBSD, OPENBSD, MACOS, OS/2, SUNOS, WinXP і Linux. Сьогодні MySQL особливо поширена на платформах Linux і Windows. Причому на останніх зустрічається набагато рідше. [14]

Принцип роботи СУБД MySQL аналогічний принципу роботи будь-якої СУБД, що використовує SQL (Structured Query Language, мова структурованих запитів) як командна мова для створення та видалення баз даних, таблиць, для поповнення таблиць даними, для здійснення вибірки даних. [15]

Характеристики СУБД MySQL:

- Написаний на мовах C і C++;
- Протестований на широкому спектрі різних компіляторів;
- Працює на безлічі різних платформ;
- Для забезпечення переносимості використовує інструменти GNU;

- Доступні API-інтерфейси для C, C++, Eiffel, Java, Perl, PHP, Python, Ruby;
- Повністю багато потоковий з використанням потоків ядра. Може працювати в багатопроцесорних системах;
- Забезпечує транзакційний і нетранзакційний механізми зберігання;
- Використовує дуже швидкі дискові таблиці (MYLSAM) із стисненням індексів;
- Дуже швидка система розподілу пам'яті, заснована на потоках;
- Код MySQL протестований за допомогою інструментів пошуку витоків пам'яті;
- Сервер доступний як окрема програма для використання в клієнтсерверному мережевому середовищі. Мережева зв'язність СУБД MySQL:
- Клієнти можуть підключатися до сервера MYSQL, використовуючи сокети TCP/IP на будь-якій платформі. У Windows-системах клієнти можуть підключатися з використанням іменованих каналів. У системах на базі UNIX клієнти можуть підключатися через файли сокетів UNIX-доменів.
- Інтерфейс CONNECTOR/ODBC дозволяє MySQL підтримувати клієнтські програми, які використовують ODBC-з'єднання. Наприклад, для підключення до сервера MySQL можна використовувати MS Access. Клієнтське програмне забезпечення може виконуватися під управлінням Windows або UNIX. Початкові тексти інтерфейсу CONNECTOR/ODBC доступні. Підтримуються всі функції ODBC, так само як і множина інших.
- Інтерфейс CONNECTOR/JDBC дозволяє MYSQL взаємодіяти з клієнтськими програмами на Java, в яких використовуються JDBCпідключення.
- Клієнтське програмне забезпечення може виконуватися під управлінням Windows або UNIX. Початкові тексти інтерфейсу CONNECTOR/JDBC доступні.

Для програмування для PHP на сервері реалізована програмна сторінка для створення баз даних та таблиць phpMyAdmin. Phpmyadmin - це інструмент, написаний на PHP, який дозволяє адмініструвати базу даних MYSQL. В реалізованому інструменті можливо запускати команди SQL, працювати з полями (додавати, редагувати, видаляти), працювати з таблицями (створювати, змінювати), створювати додаткові бази даних, і багато що інше. [18]

Oracle Database - перша в світі база даних, розроблена спеціально для роботи в мережах розподілених обчислень Grid, призначена для ефективного розгортання на базі різних типів устаткування, від невеликих серверів до могутніх симетричних багатопроцесорних серверних систем, від окремих кластерів до корпоративних розподілених обчислювальних систем. СУБД надає можливість автоматичної настройки і управління, яка робить її використання простим і економічно вигідним.

Oracle Developer Suite - повний набір інтегрованих засобів для розробки web-додатків. Включає зручне інтегроване середовище розробки із засобами моделювання, програмування, розробки компонентів, бізнес-аналіз і складання звітів. Всі ці засоби використовують загальні ресурси, що дозволяє спільно працювати над одним проектом групі розробників. Маючи одночасний доступ до ресурсів бази даних не заважаючи один одному, а вносячи паралельні зміни до однієї і тієї ж бази даних. [16]

Сервер ORACLE забезпечує ефективні і дієві рішення для основних засобів баз даних: управління великими базами даних і контроль управління простором ORACLE підтримує найбільші бази даних, потенційного розміру до сотень гігабайт. Щоб забезпечити дієвий контроль за використанням дорогих дискових пристроїв, він надає повний контроль розподілу простору. Підтримує велике число користувачів, що одночасно виконують різноманітні застосування, які оперують одними і тими ж даними. Сервер мінімізує суперництво за дані і гарантує узгодженість даних.

У комп'ютерному оточенні, сполучених мережами, ORACLE комбінує дані, що фізично знаходяться на різних комп'ютерах у одну логічну базу даних, до якої мають доступ все користувачі мережі. Розподілені системи

володіють такими ж ступенем прозорості для користувачів і узгодженості даних, що і нерозподілені системи, надаючи в те ж час переваги управління локальною базою даних. ORACLE автоматично підтримує цілісність даних, дотримуючи "організаційні правила", які диктують стандарти даних. Як наслідок, усуваються витрати на кодування і супровід перевірок в численних додатках бази даних.

Програмне забезпечення ORACLE дозволяє різним типам комп'ютерів і операційних систем спільно використовувати інформацію за допомогою мереж. [17]

Таблиця 3.2 – Порівняльна характеристика СУБД (п'ятибальна система)

Критерії	MySQL	ORACLE
Кількість ф-ій	3	5
Ціна	5	3
Гнучкість	5	4
Надійність	4	5
Технічна підтримка	4	3
Сумарний коефіцієнт	21	20

3.3 Вибір середовища розробки

Aptana Studio - кроссплатформенне, вільне, інтегроване середовище розробки додатків (IDE) с відкритим вихідним кодом для створення динамічних веб-додатків. Включає в себе такі функції, як автодоповнення конструкцій на льоту для коду JavaScript, HTML, і CSS, виділення кольором JavaScript, HTML і CSS коду, попередження про помилки і можливість для налаштування і розширення призначеного для користувача інтерфейсу. За допомогою додаткових плагінів Aptana Studio також підтримує розробку для наступних мов і платформ: Ruby on Rails, PHP, Python, Adobe Air.

Остання версія Aptana Studio включає в себе сервер Jaxer, що дозволяє виконувати JavaScript на стороні сервера. А також інтегрована з сервісом Aptana Cloud, який пропонує розгортання і хостинг веб-додатків на серверах cloud computing компанії Aptana з можливістю масштабованості ресурсів на

вимогу. Даний хостинг підтримує наступні платформи: PHP 5, Jaxer, Ruby on Rails і Java, найближчим часом очікується підтримка веб-додатків на Python.

PhpDesigner - потужна програма для створення, редагування, налагодження, аналізу і публікації веб-сторінок і додатків на мові PHP, яка підійде як початківцям програмістам, так і професійним розробникам. Програма примітна тим, що крім PHP в ній реалізована підтримка HTML, MySQL, XML, CSS, JavaScript, VBScript, JAVA, C, Python і Ruby. Завдяки засобам для автоматичного підсвічування коду, phpDesigner також підійде WAMP / LAMP- і AJAX-розробникам. Крім цього програма містить ємні бібліотеки, що містять більше 3000 функцій, доступ до яких легко здійснюється в процесі програмування. [18]

Основні можливості phpDesigner:

- Підсвічування синтаксису для багатьох мов;
- Перевірка синтаксису для PHP, HTML і CSS;
- Автоматичне завершення коду для PHP, HTML, CSS і JavaScript;
- Підтримка таких бібліотек JavaScript, як JQuery, Ext JS, YUI, Dojo, MooTools і Prototype;
- Вбудована система підказок для PHP і JavaScript;
- Інспектор коду для HTML;
- Налагодження і профілювання PHP скриптів за допомогою Xdebug;
- Підтримка FTP, SFTP і TortoiseSVN;
- Інтуїтивна навігація по об'єктах вашого коду;
- Наявність вбудованих помічників.

Sublime Text 3 - це гарний крос-платформний редактор. Він швидкий і багатий функціоналом, для практично кожної мови програмування. Підтримує декілька виділень, згортання коду, макроси, проекти та інше. Також можливо повноекранне редагування, яке виглядає чудово на великих моніторах. Запускається на Linux, Windows і OSX. Цей редактор надається з необмеженим тестовим періодом. [12]

Концептуальні переваги редактора:

- Швидка навігація (Goto Anything)

- Множинні виділення (Multiple Selections)
- Розподілене редагування (Split Editing)
- Перемикання між проектами на льоту (Instant Project Switch).
- Збірка програм (Build System)
- можливість розширення функціональності через Plugin API
- Крос- платфоменость.

Особлива цінність програми - в можливості підключення плагінів і пакетів. Через модуль управління пакетами і плагінами Package Control можна знайти і додати (а також без проблем видалити, оновити, активувати або деактивувати) безліч необхідного в роботі - і перевірку орфографії, і розширення спектра даних, що відображаються, і елементи, що дозволяють більше "заточити" програму під розробку, і підгонку середовища під звичні стилі роботи (наприклад, робота з Emacs) і під звичні користувачу набори гарячих клавіш, додавання фрагментів для роботи з HTML5 або грамотної роботи з HTML-тегами.

Таблиця 3.3 – Порівняльна характеристика середовищ розробки (п'ятибальна система)

Критерії	Aptana Studio	PhpDesigner	Sublime Text 3
Зручність використання	3	4	5
Потужність	5	5	5
Розширюваність	5	5	5
Естетика	3	4	5
Швидкість	2	4	5
Кросплатформеність	5	5	5
Сумарний коефіцієнт	23	27	30

3.4 Розробка структури веб-ресурсу

Створення сайту починається з розробки його логічної структури. Структура сайту – це організація даних, ієрархія матеріалів, необхідна для представлення їх інтернет-аудиторії. Процес створення структури сайту можна розділити на два етапи:

1. структуризація інформації;
2. візуальне представлення структури.

Структуризація інформації заключається в необхідності класифікації і групування розрізнених матеріалів, об'єднанні їх в категорії або рубрики, присвоєнні їм зрозумілих для користувачів назв. При цьому слід враховувати логіку більшості користувачів. [23]

Візуальне представлення структури сайту дозволяє розмістити елементи структури відносно один одного таким чином, щоб користувач з першого погляду міг зорієнтуватись на сайті і зрозуміти, де знайти потрібну йому інформацію. При цьому слід враховувати правило Міллера, згідно якого кількість категорій сайту повинно дорівнювати числу $7+2$. Це засновано на властивості короткочасної пам'яті людини: одночасно людина може тримати в голові 5-9 слів (назви категорій сайту). [21]

Односторінковий сайт побудований в одному вікні і не включає в себе посилання на інші сторінки. Принцип такого сайту заснований на прокручуванні, а не натисканні і переході. Хоча натискання ніхто не відміняв: воно скролл призведе на місце на сторінці з відповідним контентом. [22]Цей досить простий і цікавий спосіб піднесення інформації, більш чіткий і легкий для розуміння. Варто відзначити, що односторінкові відвідувач вивчає більш уважно, переглядаючи всі розділи.

3.5 Проектування бази даних

База даних для веб-додатка - це місце на веб-сервері, де буде зберігається контент нашого веб-ресурсу.

База даних потрібна веб – додатку для зберігання і зміни даних в автоматизованому режимі. [23]

За базу даних у відповідає адміністратор. Адміністратор використовує базу даних для інформації яка часто змінюється. Даними веб – додатку є в основному текстові матеріали. База складається з таблиць, в якій розміщені записи. Інформація в базі зберігається дуже різноманітна: різні тексти, числові дані, файли і т.п. Адміністратор з використанням бази даних вирішує такі завдання як пошук по веб - додатку, розбиття на форми, авторизація користувачів, приймає листи від користувачів. З використанням бази даних адміністратор можемо вести статистику веб - додатка, дивитися хто на нас підписується, переглядати замовлення і листи, переглядати відгуки. Проаналізувавши предметну область нашого веб – додатка, була розроблена база даних. База даних (табл.4.1) включає в себе такі таблиці: zakaz, have_a_question, subscription, reviews.

На підставі даних веб - додатка були відібрані такі об'єкти, які наведені нижче.

Таблиця 4.1 – «База даних»

№	Найменування об'єктів	Первинний ключ
1	Zakaz	ID
2	Have_a_question	ID
3	Subscription	ID
4	Reviews	ID

На підставі обраної предметної області та виділених об'єктів вибираємо атрибути, які необхідно для формування вихідного документу. Для кожного об'єкта наводимо перелік атрибутів, та опис.

Таблиця «Zakaz» містить такі атрибути: id, your_email , other_details , comment (табл. 4.2).

Таблиця 4.2 – «Zakaz»

Таблиця «Have_a_question» містить такі атрибути: id, email , message (табл. 4.3).

Атрибут	Тип	Опис
Атрибут	Тип	Опис
id	INT (10)	Ідентифікатор поста
email	VARCHAR (255)	Пошта клієнта
message	VARCHAR (255)	Повідомлення з питанням
comment	TEXT	Коментар користувача

Таблиця 4.3 – «Have_a_question»

Таблиця «Subscription» містить такі атрибути: id, email , name, user_id (табл. 4.4).

Таблиця 4.4 – «Subscription»

Атрибут	Тип	Опис
id	INT (10)	Ідентифікатор поста
email	VARCHAR (255)	Пошта підписника
user_id	VARCHAR (255)	ID підписника

Таблиця «Reviews» містить такі атрибути: id, email , name, review (табл. 4.5).

Таблиця 4.5 –« Reviews»

Атрибут	Тип	Опис
id	INT (10)	ID користувача, який написав коментар
email	VARCHAR (255)	Пошта замовника
name	TEXT	Ім'я користувача
review	TEXT	Відгук

3.6 Фізична структура

Для роботи сайту використовується наступний перелік файлів:

- app.php - збереження даних про структуру сайту в файл;
- index.html - відображення клієнтської частини сайту;
- order.php - форма калькулятора;
- send.php – форма для відправки замовлення;
- send-form.php - форма для зв'язку з адміністратором;
- stay-update.php - форма для підписки на сайт.

Також використовувалися фото матеріали, що розташовані в папці img, всі javascript файли знаходяться в папці js, таблиці стилів в папці css.

3.7 Реалізація основних функцій

Функція реалізовує форму для калькулятора.

```
$success_url = './success';  
$server = $_SERVER['HTTP_HOST'];  
$radio = $_POST['radio'];  
$categoryName = $_POST['categoryName'];  
$pagesNum = $_POST['pagesNum'];  
$approxSum = $_POST['approx-sum'];  
$userEmail = $_POST['user-email'];  
  
$mail_header = "MIME-Version: 1.0\r\n";  
$mail_header.= "Content-type: text/html; charset=UTF-8\r\n";  
$mail_header.= "From: Your Website <informer@$server>\r\n";  
$mail_header.= "Reply-to: Reply to Name <reply@$server>\r\n";
```

```

$to = "getitwrittenio@gmail.com";
$subject = "Request from Your Website: $server";

$message = "
    <strong>Write or Proofread book:</strong> $radio
    <br><strong>Category Name:</strong> $categoryName
    <br><strong>Pages Num:</strong> $pagesNum
    <br><strong>Approximate price:</strong> $approxSum
    <br><strong>User's email:</strong> $userEmail

";
mail($to,$subject,$message,$mail_header);

if ($_POST) {
    echo "Sent"; die;
}

```

Функція реалізовує форму підтвердження відправки замовлення.

```

<title>Wealth Fields</title>
<style>
    body {
        color: #313E47;
        font-family: Arial;
        font-size: 15px;
        line-height: 1;
        text-align: center;
        /* background: url(body.jpg) repeat; */
        padding-top: 20px;
    }
    h2 {
        opacity: 0;
    }
</style>
<script type="text/javascript">
    alert('Thanks for your message! We will contact you soon');
    setTimeout('location.replace("index.html")', 1000);
</script>
</head>
<body>
    <h2>THANKS FOR YOUR MESSAGE!</h2>
</body>

```

Функція реалізовує форму для зв'язку з адміністратором.

```

$success_url = './success';
$server = $_SERVER['HTTP_HOST'];
$email = $_POST['form-email'];

```

```

$textMessage = $_POST['form-message'];
$mail_header = "MIME-Version: 1.0\r\n";
$mail_header.= "Content-type: text/html; charset=UTF-8\r\n";
$mail_header.= "From: Your Website <informer@$server>\r\n";
$mail_header.= "Reply-to: Reply to Name <reply@$server>\r\n";
$to = "getitwrittenio@gmail.com";
$subject = "Request from Contact Form: $server";
$message = "
    <strong>E-mail:</strong> $email
    <br><strong>Message:</strong> $textMessage
";
mail($to,$subject,$message,$mail_header);
if ($_POST) {
    echo "Sent"; die;
}

```

Функція реалізовує форму для підписки на сайт.

```

$success_url = './success';
$server = $_SERVER['HTTP_HOST'];
$subscribeEmail = $_POST['subscribe-email'];

$mail_header = "MIME-Version: 1.0\r\n";
$mail_header.= "Content-type: text/html; charset=UTF-8\r\n";
$mail_header.= "From: Your Website <informer@$server>\r\n";
$mail_header.= "Reply-to: Reply to Name <reply@$server>\r\n";

$to = "getitwrittenio@gmail.com";
$subject = "Request for Subscription: $server";

$message = "
<strong>E-mail, that wants to subscribe:</strong> $subscribeEmail;
";
mail($to,$subject,$message,$mail_header);
if ($_POST) {
    echo "Sent"; die;
}

```

Функція реалізовує форму для відгуків.

```

<div class="col-lg-8 offset-lg-2 col-md-8 offset-md-2 send-message-
block">
<h3>Share with us your opinion</h3>
<div>
<input name="form-email" type="email" placeholder="Your email"
required>
</div>

```

```

    <div>
      <input name="form-email" type="name" placeholder="Your name"
required>
    </div>
    <div>
      <textarea name="form-message" rows="6" placeholder="Share with us
your opinion..." required></textarea>
    </div>
    <div class="d-flex justify-content-end">
      <button class="btn" type="submit">Send Feedback</button>
    </div>
  </div>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
</div>

```

Код відповідає за створення калькулятора, для того щоб розрахувати ціну за послуги. Для цього потрібно вибрати тип обслуговування, вибрати категорію, вказати кількість сторінок.

```

function hideBlock() {
  let hiddenBlock = document.getElementById('calculator-form'),
    showThisBlock = document.getElementById('get-user-info-form');
  hiddenBlock.style.display = "none";
  showThisBlock.style.display = "block";
}

function showThisBlock() {
  let hiddenBlock = document.getElementById('calculator-form'),
    showThisBlock = document.getElementById('get-user-info-form');
  showThisBlock.style.display = "none";
  hiddenBlock.style.display = "block";
}

function makeNewOrder() {
  let sentRequest = document.getElementById("sent-request-form"),
    hiddenBlock = document.getElementById('calculator-form');
  calculator.reset();
  woDiscountPrice.innerHTML = "$0";
  sentRequest.style.display = "none";
  hiddenBlock.style.display = "block";
}

//Mobile Navigation
let mobileNav = document.getElementById('mobileNav'),
  closeMobileNav = document.getElementsByClassName('closeNav');

for (i = 0; i < closeMobileNav.length; i++ ) {

```



```

        closeMobileNav[i].addEventListener('click', closeIt);
        function closeIt() {
            mobileNav.style.width = "0%";
        }
    }
    function openNav() {
        mobileNav.style.width = "100% ";
    }
    function closeNav() {
        mobileNav.style.width = "0% ";
    }
}

```

Код відповідає за створення знижки. При замовленні дається двадцять відсотків.

```

function getDiscount() {
    calculator.reset();
    woDiscountPrice.innerHTML = "$0";
    woDiscountPriceSecondary.innerHTML = "$0";
    discountBlock.style.opacity = "1";
    discountBlock.style.height = "100% ";
    discountBlockSecondary.style.opacity = "1";
    discountBlockSecondary.style.height = "100% ";
    resultBlock.value = Math.floor(resultBlock.value * 0.8);
    resultDisplaySecondary.value = resultDisplaySecondary.value * 0.8;
}

```

```

$(function () {

    $("#calculator").submit(function(event) {
        event.preventDefault();

        let ajaxRequest,
            values = $(this).serialize();

        ajaxRequest= $.ajax({
            url: "order.php",
            type: "post",
            data: values
        });

        ajaxRequest.done(function (response, textStatus, jqXHR){
            let sentRequest = document.getElementById("sent-request-form"),
                showThisBlock = document.getElementById('get-user-info-
form');

            showThisBlock.style.display = "none";
            sentRequest.style.display = "block";

```

```

    });

    ajaxRequest.fail(function (){
        alert('Failed! Try again later!');
    });
});

$("#contact-form").submit(function(event) {
    event.preventDefault();

    let ajaxRequest,
        values = $(this).serialize();

    ajaxRequest= $.ajax({
        url: "send-form.php",
        type: "post",
        data: values
    });

    ajaxRequest.done(function (response, textStatus, jqXHR){
        $('body, html').animate({ scrollTop: 0}, 1500);
        $("#message-contact-
form").fadeIn('slow').delay(5000).fadeOut('slow');
        $("#contact-form")[0].reset();
    });

    ajaxRequest.fail(function (){
        alert('Failed! Try again later!');
    });
});

$("#stay-updated").submit(function(event) {
    event.preventDefault();

    let ajaxRequest,
        values = $(this).serialize();

    ajaxRequest= $.ajax({
        url: "stay-updated.php",
        type: "post",
        data: values
    });

    ajaxRequest.done(function (response, textStatus, jqXHR){
        $('body, html').animate({ scrollTop: 0}, 1500);
        $("#message-
subscription").fadeIn('slow').delay(5000).fadeOut('slow');

```

```

        $("#stay-updated")[0].reset();
    });

    ajaxRequest.fail(function (){
        alert('Failed! Try again later!');
    });
});

//Smooth slide to menu link
$('.go').click(function() {
    let go = $(this).attr('href');
    if ($(go).length != 0) {
        $('html,body').animate({ scrollTop: $(go).offset().top }, 1000);
    }
    return false;
});

//Remove blur effect after clicking on buttons, links and etc.
$('button', 'a', 'input', 'img').click(function() {
    this.blur();
});

//Validation for calculator first screen
$(validateInput).on('click', function() {
    if (typedNumberOfPages.value == "" ||
bookCategory[selectedCategory.value] == "") {
        alert('Please, select a category name and book length value!');
    } else {
        hideBlock();
    }
})

//Don't allow to press 0 as first symbol in forms
$('input').keypress(function(e){
    if (this.value.length == 0 && e.which == 48 ){
        return false;
    }
});

// Slider section reviews
$('.testimonials-slider').slick({
    slidesToShow: 3,
    slidesToScroll: 1,
    prevArrow: '<div class="talking__slider__arrows left-arrow"></div>',
    nextArrow: '<div class="talking__slider__arrows right-
arrow"></div>',
    dots: false,

```

```
        responsive:[
    {
      breakpoint: 1199,
      settings: {
        slidesToShow: 2,
        slidesToScroll: 1,
        arrows: false
      }
    },
    {
      breakpoint: 767,
      settings: {
        slidesToShow: 1,
        slidesToScroll: 1,
        arrows: false,
        centerMode: false
      }
    }
  ]
}
```

3.8 Опис режимів роботи

Графічний дизайн можна визначити як створення візуальних зображень за допомогою графічних елементів, з метою передачі певних ідей і послань. Поняття Графічний дизайн використовується як для позначення самого процесу дизайну, так і його результатів.

Графічний дизайн є першорядним у створенні оригінал-макетів для поліграфії і фірмового стилю. [25]

Розробка привабливих рекламних матеріалів створюється на трьох складових:

- створення композиції
- вибір шрифтів
- колірне рішення

Правильно скомпонувати і стилістично правильні рішення, облік сприйняття людиною колірної гами, здатні надати рекламних матеріалів досконалий і притягальний вид.

Сайт повинен мати інтерфейс, що дозволяє відвідувачеві легко орієнтуватися в його ресурсах. Візуальне виділення інформаційних блоків, контрастні посилання, інтуїтивно зрозуміле розташування керуючих елементів.

Для проектування сайту компанії необхідно детально вивчити діяльність даної компанії і виявити ті інформаційні ресурси, які можуть бути опубліковані в інтернет.

Веб-додаток розроблявся з урахуванням переваг і недоліків проаналізованих вище сайтів.

Основна функція нашого головного розділу веб-додатка - це привернення уваги цільової аудиторії і конвертації користувачів в реальних клієнтів. Це наша можливість сказати потенційному клієнту хто ми, що ми пропонуємо, чим ми кращі за своїх конкурентів і навіщо відвідувачеві взагалі зв'язуватися з вами.

Правила яких дотримались при створенні головного розділу це: користувач відразу розуміє на якій веб-додаток він потрапив; за допомогою навігації і активних кнопок дає можливість користувачеві швидко орієнтуватися і пересуватися по сайту в пошуку цікавить його інформацією, це робить сайт більш простим і зрозумілим для користувача; чим він займається і що пропонує; які переваги має наш веб-додаток; веб-додаток має підказки, щодо матеріалу, наявність посилань на подібні теми.

Використовували ключові слова в описі веб-додатка, ресурс буде більш авторитетним і завоює вищий рейтинг в пошукових системах, якщо використовувати більше посилань з ключовими фразами. Наш заголовок скаже користувачеві веб-додатка куди він потрапив, чим займається компанія і дасть уявлення про нас. Уже виходячи з заголовка, користувач може зрозуміти чи цікавий йому веб-додаток і діяльність компанії в загальному.

Головний розділ оформлений так, щоб користувачі змогли розібратися в навігації сайту, ближче захотіли ознайомитися з ресурсом і захотів перейти на наступні розділи. Матеріал на першому розділі побудований таким чином, що зацікавлює користувачів і потім приманює вчинити будь-яку дію, наприклад, розрахувати суму за написання книги.

Головний розділ веб-додатку (рис. 3.1) - на якому відображається логотип сайту, опис чим саме ми займаємося, зверху кнопки для переміщення по сайту, прайс-калькулятор для розрахування суми. Прайс-

калькулятор дозволяє розрахувати ціну за написання книги, або її редагування. Обираємо тип послуги, категорію, і вводимо кількість сторінок.

The screenshot shows the website **getitwritten.io** with a navigation menu: "Що Ми Робимо", "Як Це Працює", "Наша Статистика", "Відгуки", and "Зв'яжіться З Нами". A prominent orange button says "Зробити замовлення".

Що ми робимо

Наша головна мета - допомогти особі зосередитися на своєму бізнесі та позбутися рутинної роботи з написанням книг.

Ми - команда професійних авторів книжок з більш ніж 5-річним досвідом роботи. Тому ми знаємо, як зробити книгу, яку ми пишемо, відмінною від ваших конкурентів, і мати можливість отримати найвищий рейтинг і найкращі відгуки.

Просто скажіть нам, яка книга вам потрібна, і ми зробимо все інше!

Розрахуйте ціну

Вам зараз не потрібно платити нічого

Тип послуги: **Напишіть книгу** / Корируйте книгу

Виберіть категорію: Почніть вводити назву категорії

Довжина книги: 200 сторінок 0 слова

Орієнтовна ціна: **\$0**

[Продовжити](#)

Рисунок 3.1 — Головний розділ веб-додатка

Форма замовлення - це ідеальний варіант прийому замовлень, для невеликих компаній. Як показує практика, якщо користувач зіткнеться з якоюсь проблемою при оформленні замовлення, він найчастіше залишає ресурс, побоюючись зіткнутися з цією проблемою знову. З цього форма замовлення містить кілька простих етапів і заповнення не займає особливих зусиль, нам гарантовані стабільні продажі.

Форма приємна і легка в використанні: немає реєстрації, прості поля заповнення. Коротка форма найбільш ефективна в комунікації з користувачами. Всі замовлення, отримані з форми, зберігаються у нас в системі, а також розсилаються на зазначені вами email адреси, і можуть бути експортовані з системи в будь-який момент. Замовлення не буде відправлене якщо не має підключення до інтернета. Не заповнено поле с поштою, а тільки заповнено поле з замовленням. Веб-додаток не співпрацює з тими клієнтами хто зареєстрований на пошті mail.ru. Форма для замовлення (рис. 3.2) вводимо в поля пошту, телефон для зв'язку. І пишемо деталі на

← **back to price calculation**
You do not need to pay anything right now

Your email

Phone

Other details (optional)

Approximate price:
\$ 1632

Send Request

замовлення.

Рисунок 3.2 - Форма для замовлення

Використовувати адресу електронної пошти клієнта для підтвердження замовлення дає максимально «чисту» базу.

Після натискання кнопки «Відправити замовлення» з'явиться повідомлення про успішну завершену процедуру оформлення замовлення. Замовник бачить форму з повідомленням «Ваше замовлення надіслано! Незабаром ми зв'яжемося з вами електронною поштою!»

Велику частину інформації ми отримуємо візуально. Тому не дивно, що наша приязнь або неприязнь до сайту може ґрунтуватися всього-лише на якомусь зображенні.

Зображення на веб - додатку викликає інтерес і занурюють клієнта в контекст. Разом з текстом фотографія працює на головну мету - донести до аудиторії повідомлення про сервіс. Зображення привертають додаткових клієнтів. Вони впливають на швидкість завантаження сторінок і поведінкові чинники. При якісній оптимізації вони можуть поліпшити рейтинг вашого сайту в пошукових системах. Користувачі часто не читають і не зупиняють свою увагу на тексті, зазвичай вони «сканують» сторінки. Найчастіше звертає увагу на графічні елементи. Тому крім наявності зображень на сайті, важливо, щоб зображення швидко завантажувалися, були привабливими, інформативними і якісними.

На цю форму (рис. 3.3) можна потрапити натиснувши на кнопку або змахнувши мишкою вниз.

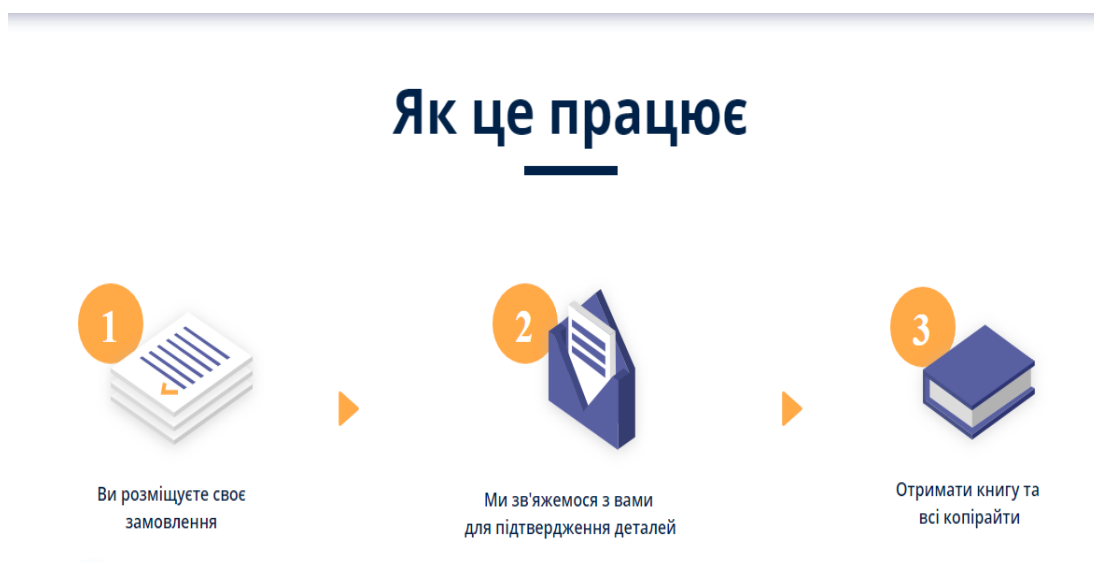


Рисунок 3.3 - Форма «Як працює веб - додаток»

В даний час практика надання знижок використовується великими і середніми компаніями, організаціями малого бізнесу та індивідуальними підприємцями. Знижка - добровільне, одностороннє зниження вартості товару. Застосування знижки забезпечує позитивний ефект, який виражається як в збільшенні клієнтської бази, так і в досягненні позитивних економічних

показників, допомагає переманити частину клієнтів собі у конкуруючих фірм, добитися великих обсягів продажів книжок, підтримувати лояльність постійних клієнтів. Розмір наданої знижки враховує як інтереси замовника, так і наші інтереси. Крім того, що знижка сприяє економії коштів, вона також впливає на ставлення замовника до нашої діяльності в цілому. Замовник без сумнівів вірить, що пропонована знижка є справедливим зниженням ціни. Тобто ми можемо просто збільшити ціну на 20%, а потім скинути ці ж 20%, і вказати на веб-додатку стару і нову ціни.

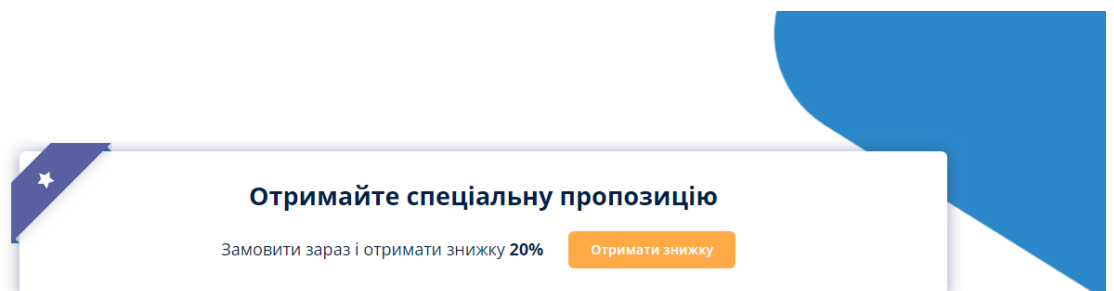
З метою стимулювання замовлень, ми додали знижку двадцять відсотків. На цю форму можна потрапити змахнувши мишкою вниз (рис. 3.4).

Кожен може отримати знижку, потрібно просто натиснути кнопку на формі, і ви отримаєте знижку двадцять відсотків.

Рисунок 3.4 - Форма «Знижка»

Щоб завоювати довіру клієнтів, потрібно створити міцний зв'язок між нами і цільовою аудиторією. Для багатьох користувачів якість сервісу одна з найважливіших цінностей, яку ми можемо запропонувати.

Актуальна і точна статистика відвідувань веб-додатка - найважливіша



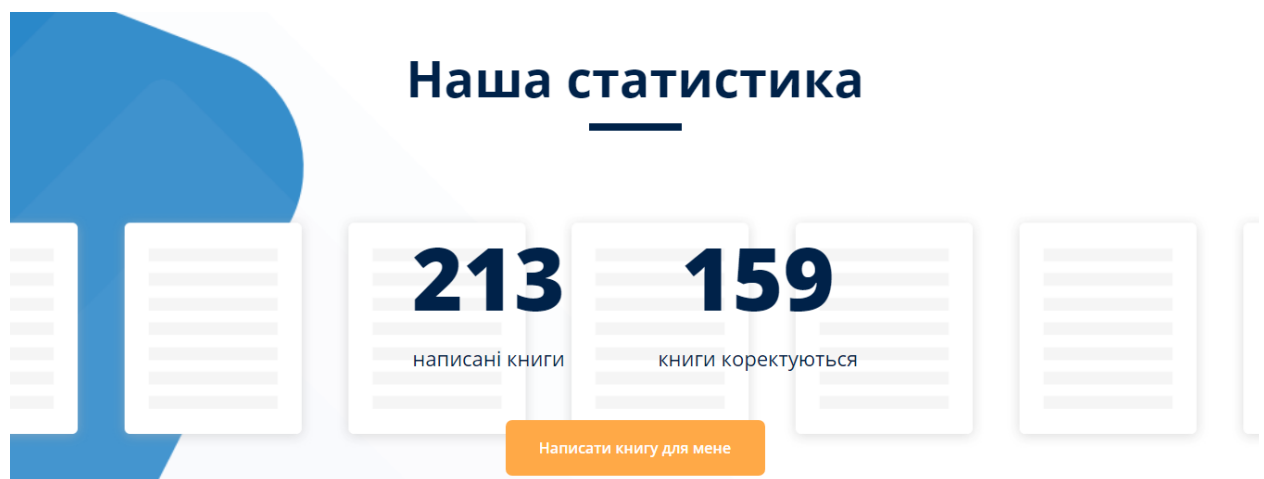
інформація для онлайн-бізнесу. Завдяки статистиці, збільшується рейтинг і відвідуваність сайту. Статистика дозволяє визначити аудиторію відвідувачів, з її допомогою легше розвивати свій веб-додаток. Важливо викликати у відвідувача відчуття, що перед ним надійна фірма, яка якісно і відповідально

виконує свою роботу. Компанія, якій можна довіряти свої гроші. Разом з текстом фотографія працює на головну мету - донести до аудиторії повідомлення про веб-додаток. Тому вся інформація і відомості виставлені.

На цю форму (рис. 3.5) можна потрапити натиснувши на кнопку або змахнувши мишкою вниз. Статистика оновлюється кожен раз після зробленої роботи. Показано скільки вже написано книг нами, і змінено. Знизу кнопка, де при натисканні можна зразу перейти до оформлення замовлення.

Рисунок 3.5 - Форма «Наша статистика»

Чужу думку про товар або послугу. В наш час використовується як інструмент маркетингу. Однак, соціальний доказ - не єдина причина ефективності рекомендацій. Принаймні, вона не повинна бути єдиною. Недостатньо просто сказати, що книга хороша. Важливо щоб було вказано на деталі і переваги на роботу з нами, вказати якомога більше плюсів нашого веб-додатка. Відгуки допомагають потенційному клієнту позбутися від

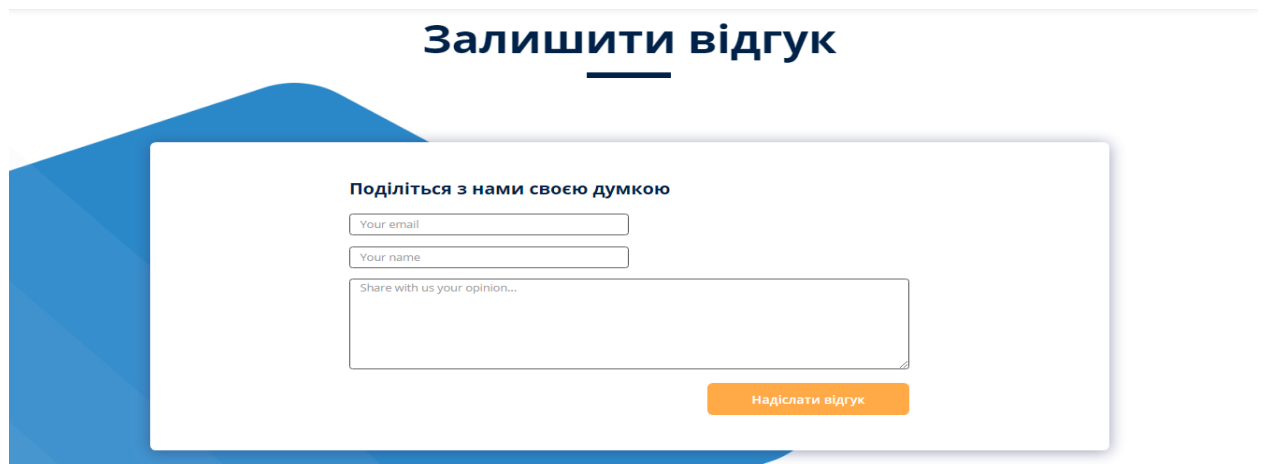


сумнівів щодо нашої роботи або допомогти в остаточному виборі у кого замовити книгу з декількох схожих варіантів.

Негативні відгуки теж потрібні. Всі відгуки є цінними для бізнесу, і поєднання позитивних відгуків разом з негативними дозволяє збільшувати рівень довіри з боку людини, який їх читає. Негативні коментарі будуть помічені. Причому робиться це все максимально швидко і якісно. Обробка негативу йде нам на користь, і ми не тільки перетворимо незадоволеного

клієнта в вдячного, але і підвищимо лояльність з боку інших користувачів, які переконуються, що наша компанія дійсно намагається допомогти своїм клієнтам і орієнтована на задоволення їх потреб. Це можливість виставити процес вирішення проблеми клієнта на загальний огляд.

Форма де можна залишити відгук (рис. 3.6), вводимо пошту, своє ім'я, і сам відгук, натискаємо кнопку «Надіслати відгук»



Залишити відгук

Поділіться з нами своєю думкою

Your email

Your name

Share with us your opinion...

Надіслати відгук

Рисунок 3.6 – Форма «Залишити відгук»

Залишаючи чесні коментарі, наші замовники допомагають іншим користувачам зрозуміти особливості нашого веб-додатка. Коментар послужить нам гарною рекомендацією або, навпаки, застережуть від помилок в роботі. Після того як користувач залишив відгук його не можна змінити або видалити.

Подивитись на відгуки можна натиснувши на кнопку або змахнувши мишкою вниз. Перейшовши на цей розділ можна побачити самі відгуки, кнопкою можна їх переглядати (рис. 3.7).



Рисунок 3.7 – Форма «Відгуки»

На цю форму (рис. 3.8) можна потрапити натиснувши на кнопку або змахнувши мишкою вниз. Можна задавати нам питання які цікавлять, якщо користувач не хоче писати його до відгуків щоб його всі бачили.

Є питання?

Рисунок 3.8 – Форма «Для зв'язку з нами»

Підписатися на наш веб-додаток можна натиснувши на кнопку (рис.3.9) «Підписатися» зверху з'являється push-повідомлення, підтвердження підписки. Та також можна переглянути наші соц мережі в facebook, instagram, twitter та також підписатися на них, для слідкування за нашими новинами.

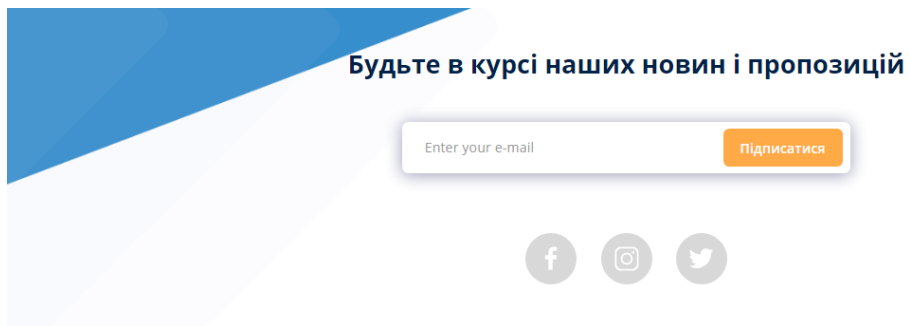


Рисунок 3.9 – Форма «Підписка»

Метою випускної кваліфікаційної роботи бакалавра була розробка веб-додаток для написання видавничої продукції.

У рамках даної роботи були розглянуті актуальні сайти з написання книжок в інтернеті.

В роботі було досліджено та проаналізовано найбільш важливу і актуальну інформацію щодо розробки веб-ресурсів а саме основні принципи створення веб-ресурсів, їх структуру і функціональність, взаємодію основних компонентів. Також були розглянуті новітні та найбільш перспективні вебтехнології, які з успіхом вже використовуються користувачами по всьому світі.

Після проведення аналізу було вирішено розробити веб-ресурс за допомогою веб-технологій PHP – для розробки серверної частини сайту, HTML 5 – в якості базової розмітки та Bootstrap 3 – для оформлення вебресурсу і його адаптивності на різних пристроях.

Проаналізувавши найпопулярніші СКБД для розробки бази даних було обрано СУБД MySQL, оскільки вона безкоштовна, відносно швидка і проста у використанні.

В якості середовища розробки було обрано Sublime Text 3, оскільки він швидкий і простий у використанні а також підтримує велику кількість плагінів , які значно спрощують та поліпшують процес розробки.

Описано основні режими роботи з веб - додатком «Getitwritten.io».

В процесі розробки веб-ресурсу та оформлення бакалаврської роботи на практиці були закріплені теоретичні знання, вдосконалені навички програмування, проведено оформлення технічної документації в текстових та графічних редакторах.

На підставі виконаної роботи, розробки веб-додатка для написання видавничої продукції можна зробити висновок про те, що поставлені до даного проекту вимоги виконані.

СПИСОК ЛІТЕРАТУРИ

1. Все для создания сайтов [Електронний ресурс] // Режим доступа: <http://siteprog.ru/> - Назва з екрану.
2. Діаграма прецедентів [Електронний ресурс] // Режим доступа: <https://uk.wikipedia.org/wiki/> - Назва з екрану.
3. Створення таблиць в HTML — код таблиці, ширина, висота [Електронний ресурс] // Режим доступа: <http://ab-w.net/HTML/table.php> - Назва з екрану.
4. Средства создания сайтов [Електронний ресурс] // Режим доступа: <http://www.in-internet.narod.ru/teor/sreda.html> - Назва з екрану.
5. UML — діаграма варіантів використання. [Електронний ресурс] // Режим доступа: <https://habrahabr.ru-/post/47940/> - Назва з екрану.
6. MySQL [Електронний ресурс] // Режим доступа: <http://www.webmasters.by> - Назва з екрану.
7. MySQL 4 - Строковые функции [Електронний ресурс] // Режим доступа: <http://www.codenet.ru/db/mysql/mystring4> - Назва з екрану.
8. Microsoft SQL Server [Електронний ресурс] // Режим доступа: http://ru.wikipedia.org/wiki/Microsoft_SQL_Server - Назва з екрану.
9. Web-Розробка [Електронний ресурс] // Режим доступа: <http://fcit.tneu.org/web-rozrobka/> - Назва з екрану.
10. CMS List. Огляд cms. Системи управління сайтом. [Електронний ресурс] // Режим доступа: <http://www.cmslist.ru> - Назва з екрану.
11. Oracle Database [Електронний ресурс] // Режим доступа: https://uk.wikipedia.org/wiki/Oracle_DB - Назва з екрану
12. Sublime Text 3 [Електронний ресурс] // Режим доступа: <http://eah.me/sublime-text/> – Назва з екрану.
13. HTML 5 [Електронний ресурс] // Режим доступа: <https://ru.wikipedia.org/wiki/HTML5> - Назва з екрану

14. Java Script [Электронный ресурс] // Режим доступа: <https://ru.wikipedia.org/wiki/javascript> - Назва з екрану.
15. Oracle Database [Электронный ресурс] // Режим доступа: https://uk.wikipedia.org/wiki/Oracle_DB - Назва з екрану.
16. Aptana Studio [Электронный ресурс] // Режим доступа: <http://www.aptana.com/> – Назва з екрану.
17. [Гутманс](#) Е. PHP 5. Професійне програмування : [Гутманс](#) Е. — М: 2006. — 1418 с.
18. Давид Скляр. PHP. Рецепты программирования. 3-е изд.: Скляр Д. — М: 2017. — 369 с.
19. Дронов В. PHP, MySQL, HTML5 и CSS 3. Разработка современных динамических Web-сайтов: Дронов В. — М: 2016. — 688 с.
20. Кирилов, В. В. Введення в реляційні бази даних : В. В. Кирилов. — СПб. : BHV-СПб, 2012. — 464 с
21. Рудікова, Л. В. Бази даних. Розробка додатків : Л. В. Рудікова. — СПб. : BHV-СПб, 2018. — 496 с.
22. Стив Суэринг, Тим Конверс, Джойс Парк. PHP и MySQL. Библия программиста, 2-е издание = PHP 6 and MySQL 6 Bible. — М.: «Диалектика», 2010. — 912 с.
23. Чарльз У.С. Стильный сайт с помощью css «ИТ Пресс», 2016. — 267с.
24. West M. HTML5 Foundations «Packt», 2013. — 386р.
25. PHP, MySQL и Dreamweaver MX 2004. Разработка интерактивных Web-сайтов. — СПб.: БХВ-Петербург, 2005. — 448 с
26. Романюк О.Н. Веб-дизайн і комп'ютерна графіка. — Вінниця: ВНТУ, 2007. — 147с.
27. PHP 5 / Д. В. Котеров, А. Ф. Костарев. — 2-е изд., перераб. и доп. — СПб.: БХВ-Петербург, 2008; Зельдман Д.

Исходный html код.

```

<!DOCTYPE html>
<html lang="en">

<head>
<meta charset="utf-8">

<title>Get It Written</title>
<meta name="description" content="Our main goal is to help individual concentrate on their business and
git rid of book writing routine. We are a team of professional book writers with over 5 years of
experience in. That's why we know how to make a book we write different from your competitors' and to
be able to get the highest rating and best feedbacks. Just tell us what kind of book you need and we will
do the rest!">
<meta name="keywords" content="write a book, proofread a book, proofread, write, amazon publishers,
book writing, book proofreading, services for writing a book, services for proofreading a book">

<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1">

<!-- Template Basic Images Start -->
<link rel="shortcut icon" href="img/icons/favicon.ico" type="image/x-icon">
<link rel="icon" href="img/icons/favicon.ico" type="image/x-icon">
<!-- Template Basic Images End -->

<!-- Custom Browsers Color Start -->
<meta name="theme-color" content="#ffa947">
<!-- Custom Browsers Color End -->

<link rel="stylesheet" href="css/main.min.css">
<link rel="stylesheet" href="libs/slick-carousel/slick/slick.css">
<link rel="stylesheet" href="libs/slick-carousel/slick/slick-theme.css">

<!-- Google Tag Manager -->
<script>
(function(w,d,s,l,i){w[l]=w[l]||[];w[l].push({'gtm.start': new Date().getTime(),event:'gtm.js'});var
f=d.getElementsByTagName(s)[0],
j=d.createElement(s),dl=l!='dataLayer'?'&l='+l:'';j.async=true;j.src=
'https://www.googletagmanager.com/gtm.js?id='+i+dl;f.parentNode.insertBefore(j,f);
})(window,document,'script','dataLayer','GTM-PLT24KK');
</script>
<!-- End Google Tag Manager -->
<!-- Google Tag Manager (noscript) -->
<noscript>
<iframe src="https://www.googletagmanager.com/ns.html?id=GTM-PLT24KK" height="0" width="0"
style="display:none;visibility:hidden"></iframe>
</noscript>
<!-- End Google Tag Manager (noscript) -->

</head>

<body>

<div id="message-contact-form" class="col-lg-4 offset-4 notification-block">
<p>Message was successfully sent</p>
</div>

```

```
<div id="message-subscription" class="col-lg-4 offset-4 notification-block">
<p>You were successfully subscribed</p>
</div>
```

```
<header id="header" class="header">
<div class="container">
<div class="row">
<!--LOGO BLOCK-->
<div class="col-3 d-flex align-items-end">
<a class="go" href="#what-we-do">
<?xml version="1.0" encoding="UTF-8"?>
<svg width="225px" height="48px" viewBox="0 0 225 48" version="1.1"
xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink">
<title>getitwritten</title>
<desc>Created with Sketch.</desc>
<defs></defs>
<g id="Artboard-2" stroke="none" stroke-width="1" fill="none" fill-rule="evenodd">
<g id="Group-5" transform="translate(8.000000, 8.000000)">
<g id="Group-55" stroke-linecap="round" stroke-width="9">
<path d="M29.8998888,32 C29.8998888,23.163444 23.2065707,16 14.9499444,16 C6.69331809,16
0,23.163444 0,32" id="Oval-8" stroke="#FF9B29" opacity="0.701785714"
transform="translate(14.949944, 24.000000) scale(1, -1) translate(-14.949944, -24.000000) "></path>
<path d="M24.240267,3.46809 C21.688855,1.29703018 18.4592926,0 14.9467348,0 C6.69188111,0
0,7.163444 0,16" id="Oval-8" stroke="#404A9C" opacity="0.701785714"></path>
<path d="M15.6974416,16.1051075 L29.8515811,16.1051075" id="Line-5" stroke="#8F55FF"
opacity="0.699955"></path>
</g>
<text id="getitwritten.io" font-family="SFProDisplay-Bold, SF Pro Display" font-size="25" font-
weight="bold" fill="#4A4A4A">
<tspan x="49" y="25">getitwritten.io</tspan>
</text>
</g>
</g>
</svg>
</a>
</div>
```

```
<div class="col-9 d-lg-none d-flex justify-content-end align-items-center">

<div id="mobileNav" class="sidenav">
<a href="#" class="closeNav closebtn" onclick="closeNav()">&times;</a>
<a class="go closeNav" href="#calculator">Place Order</a>
<a class="go closeNav" href="#what-we-do">What we do</a>
<a class="go closeNav" href="#how-it-works">How it works</a>
<a class="go closeNav" href="#statistics">Our statistics</a>
<a class="go closeNav" href="#testimonials">Testimonials</a>
<a class="go closeNav" href="#contact">Contacts</a>
</div>
</div>
```

```
<!--NAVIGATION BLOCK-->
<div class="col-lg-7 menu-desktop d-flex align-items-center">
<ul>
<li>
<a class="go" href="#what-we-do">What we do</a>
</li>
<li>
<a class="go" href="#how-it-works">How it works</a>
</li>
<li>
```

```
<a class="go" href="#statistics">Our statistics</a>
</li>
<li>
<a class="go" href="#testimonials">Testimonials</a>
</li>
<li>
<a class="go" href="#contact">Contact us</a>
</li>
</ul>
</div>
```

```
<!--Button block-->
<div class="col-2 cta-btn-desktop d-flex justify-content-end align-items-center">
<a class="go" href="#what-we-do">
<button>Place Order</button>
</a>
</div>
</div>
</div>
</div>
</div>
```

```
<section id="what-we-do">
<div class="container">
```

```
<div class="row info-row">
<!--Headline with description-->
<div class="col-lg-6 col-md-12 d-flex flex-column justify-content-center">
<div class="headline">
<h1>What We Do</h1>
<hr>
</div>
</div>
```

```
<p class="heading-description">Our main goal is to <b>help individual concentrate on their business and
git rid of book writing routine.</b></p>
<p class="heading-description"><b>We are a team of professional book writers</b> with over 5 years of
experience in. That's why we know how to make a book we write different from your competitors' and to
be able to get <b>the highest rating and best feedbacks.</b></p>
<p class="heading-description">Just tell us what kind of book you need and we will do the rest!</p>
</div>
```

```
<div id="anchor-btn" class="col-lg-6 offset-lg-0 col-md-10 offset-md-1 d-flex justify-content-end">
<form id="calculator" action="" method="POST">
```

```
<!--Calculator-->
<div id="calculator-form" class="calculator-form">
<h2>Calculate price</h2>
<p>You do not need to pay anything right now</p>
```

```
<!--Service type-->
<div class="row calculate-block">
<div class="col-lg-4 d-flex align-items-center">
<label for="">Service type</label>
</div>
<div class="col-lg-8 d-flex">
<label class="container-radio">
<input type="radio" checked="checked" name="radio" value="write" onclick="calculateSumm()">
<span class="checkmark">Write a book</span>
</label>
<label class="container-radio proofread-box">
```

```
<input type="radio" name="radio" value="proofread" onclick="calculateSumm()">
<span class="checkmark">Proofread a book</span>
</label>
</div>
</div>
```

```
<!--Category type-->
<div class="row calculate-block">
<div class="col-lg-4 d-flex align-items-center">
<label for="">Select Category</label>
</div>
<div class="col-lg-8 d-flex">
<select class="choose-category" id="choose-category" name="categoryName"
onchange="calculateSumm()">
<option value="" default>Start typing category name</option>
<option value="arts">Arts & Photography</option>
<option value="biography">Biographies & Memoirs</option>
<option value="business">Business & Money</option>
<option value="calendars">Calendars</option>
<option value="children">Children's Books</option>
<option value="christian">Christian Books & Bibles</option>
<option value="comics">Comics & Graphic Novels</option>
<option value="computers">Computers & Technology</option>
<option value="cookbooks">Cookbooks, Food & Wine</option>
<option value="crafts">Crafts, Hobbies & Home</option>
<option value="education">Education & Teaching</option>
<option value="engineering">Engineering & Transportation</option>
<option value="health">Health, Fitness & Dieting</option>
<option value="history">History</option>
<option value="humor">Humor & Entertainment</option>
<option value="law">Law</option>
<option value="lesbian">Lesbian, Gay & Bisexual Books</option>
<option value="literature">Literature & Fiction</option>
<option value="medical">Medical Books</option>
<option value="mystery">Mystery, Thriller & Suspense</option>
<option value="parenting">Parenting & Relationships</option>
<option value="politics">Politics & Social Sciences</option>
<option value="reference">Reference</option>
<option value="religion">Religion & Spirituality</option>
<option value="romance">Romance</option>
<option value="science">Science & Math</option>
<option value="science-fiction">Science Fiction & Fantasy</option>
<option value="self-help">Self-Help</option>
<option value="sports">Sports & Outdoors</option>
<option value="teen">Teen & Young Adult</option>
<option value="test">Test Preparation</option>
<option value="travel">Travel</option>
</select>
</div>
</div>
```

```
<!--Book length type-->
<div class="row calculate-block">
<div class="col-lg-4 d-flex align-items-center">
<label for="">Book length</label>
</div>
<div class="col-lg-8 d-flex">
<input id="pagesNum" type="text" maxlength="4" name="pagesNum" placeholder="200"
onkeypress="if ( isNaN( String.fromCharCode(event.keyCode) )) return false;"
oninput="calculateSumm()" required>
```

```
<span class="additional-info">pages</span>
<input id="wordsNumb" name="wordsNum" type="text" placeholder="" oninput="calculateSumm()"
readonly value="0">
<span class="additional-info">words</span>
</div>
</div>
```

```
<div class="row calculate-block">
<div class="col-lg-7 d-flex flex-column price-block">
<label>Approximate price:</label>
<div class="approx-price-block">
```

```
<span class="approx-price__usd">$</span>
<input name="approx-sum" id="resultPrice" class="approx-price" type="text" value="0" readonly>
<div id="discount-block" class="d-flex discount-block">
<p>You've got a</p>
<span id="price-wo-discount" class="price-wo-discount">$0</span>
<div class="d-flex justify-content-center discount-info">
-20%
</div>
<p>discount!</p>
</div>
</div>
</div>
<div class="col-lg-5 d-flex align-items-center">
<a id="validate-input" class="btn link-btn">
Continue
</a>
</div>
</div>
</div>
```

```
<!--Form after calculator-->
<div id="get-user-info-form" class="calculator-form">
<a class="back-link" onclick="showThisBlock()">
<h2>

back to price calculation
</h2>
</a>
```

```
<p>You do not need to pay anything right now</p>
```

```
<div class="row calculate-block">
<div class="col-lg-4 d-flex align-items-center">
<label for="">Your email</label>
</div>
<div class="col-lg-8 d-flex">
<input name="user-email" class="user-info" type="email" placeholder="email@email.com" required>
</div>
</div>
```

```
<div class="row calculate-block">
<div class="col-lg-4 d-flex align-items-center">
<label for="">Phone</label>
</div>
<div class="col-lg-8 d-flex">
<input name="user-phone" class="user-info" type="phone" placeholder="(099) 000-00-00" required>
```

```
</div>
</div>
```

```
<div class="row calculate-block">
<div class="col-lg-4 d-flex align-items-center">
<label for="">Other details (optional)</label>
</div>
<div class="col-lg-8 d-flex">
<textarea class="user-info" name="" id="" cols="5" rows="5" placeholder="You can add any
information you think can be valuable"></textarea>
</div>
</div>
```

```
<div class="row calculate-block">
<div class="col-lg-7 d-flex flex-column price-block">
<label>Approximate price:</label>
<div class="approx-price-block">
<span class="approx-price__usd">${</span>
<input id="getResultPriceValue" maxlength="5" class="approx-price" type="text" value="0" readonly>
<div id="discount-block-secondary" class="d-flex discount-block">
<p>You've got a</p>
<span id="price-wo-discount-secondary" class="price-wo-discount">${</span>
<div class="d-flex justify-content-center discount-info">
-20%
</div>
<p>discount!</p>
</div>
</div>
</div>
<div class="col-lg-5 d-flex align-items-center">
<button class="btn" type="submit">Send Request</button>
</div>
</div>
```

```
</div>
```

```
<!--Send request-->
<div id="sent-request-form" class="calculator-form hide">
```

```

```

```
<div class="row">
<div class="col-lg-12 sent-info">
<p><b>Your order was sent!</b></p>
<p>We will contact you through email shortly!</p>
</div>
</div>
```

```
<div class="row calculate-block">
<div class="col-lg-6 offset-lg-6 d-flex align-items-center">
<a class="btn link-btn" href="#" onclick="makeNewOrder()">
New Order Request
</a>
</div>
</div>
```

```
</div>
</form>
</div>
```

```
</div>
</div>
</section>
```

```
<section id="how-it-works">
<div class="container">
<div class="d-flex flex-column">
<div class="headline">
<h1>How It Works</h1>
<hr>
</div>
```

```
<!--Images with text block here-->
<div class="row d-flex justify-content-between">
```

```
<div class="col-lg-3 col-md-12 d-flex flex-column align-items-center justify-content-center">

<p>You place your order</p>
</div>
```

```
<div class="divider-block col-lg-1 col-md-12 d-flex justify-content-center">

</div>
```

```
<div class="col-lg-4 col-md-12 d-flex flex-column align-items-center justify-content-center">

<p>We will contact you<br> to confirm details</p>
</div>
```

```
<div class="divider-block col-lg-1 col-md-12 d-flex justify-content-center">

</div>
```

```
<div class="col-lg-3 col-md-12 d-flex flex-column align-items-center justify-content-center">

<p>Get your book and<br> all copywrites</p>
</div>
</div>
</div>
</div>
</section>
```

```
<section id="statistics-testimonials-offer">
```

```
<!--Here are stats block-->
<div id="statistics" >
<div class="container">
<div class="d-flex flex-column">
<div class="headline">
<h1>Our Statistics</h1>
<hr>
</div>
```

```
<!--Stats block here-->
<div class="row d-flex justify-content-center">
<div class="statistics-block d-flex flex-column align-items-center">
<h2 class="numscroller" data-min="1" data-max="213" data-delay="5" data-increment="4">213</h2>
<p>books written</p>
</div>
<div class="statistics-block d-flex flex-column align-items-center">
```

```
<h2 class="numscroller" data-min="1" data-max="159" data-delay="5" data-increment="4">159</h2>
<p>books proofread</p>
</div>
</div>
<div class="col-lg-12 d-flex justify-content-center">
<a class="go" href="#what-we-do">
<button class="btn">Write a Book For Me</button>
</a>
</div>
</div>
</div>
</div>
```

```
<!--Here are all testimonials-->
<div id="testimonials">
<div class="container">
<div class="row">
<div class="headline">
<h1>Testimonials</h1>
<hr>
</div>
```

```
<!--Slider starts here-->
<div class="col-lg-12 testimonials-slider">
<div class="col-lg-4 testimonials-block">
<h4>Oscar Holmberg</h4>
<span>Client's feedback: 5.0 </span>
<p>“Thanks for the book you’ve written for me! All the deadlines were met, and communication was
never a problem.</p>
<p>Will contact again no doubts.”</p>
</div>
<div class="col-lg-4 testimonials-block">
<h4>Aaron King</h4>
<span>Client's feedback: 5.0 </span>
<p>“Get It Written really delivers — both in terms of text quality but also as reliable company who will
get the job done.</p>
<p>This was my second time working with guys and they delivered the finished text ahead of schedule
and exceed my expectations. Highly recommend!”</p>
</div>
<div class="col-lg-4 testimonials-block">
<h4>Alan Park</h4>
<span>Client's feedback: 4.9 </span>
<p>“Get it written is great. Super responsive, quick turnaround, and good quality of work.”</p>
</div>
<div class="col-lg-4 testimonials-block">
<h4>Henrik F</h4>
<span>Client's feedback: 4.9 </span>
<p>“They did a very good job in converting my scribbled notes into proper text. Will contact again when
I have more books to write.”</p>
</div>
<div class="col-lg-4 testimonials-block">
<h4>Andonis Mitidis</h4>
<span>Client's feedback: 5.0 </span>
```


<p>“GetItWritten is the best service we've ever encountered before. They have exceeded our expectation and amazed by the final result. Managers are always ready to help available and try hard to meet client's satisfaction.”</p>

</div>

<div class="col-lg-4 testimonials-block">

<h4>Jack Robson</h4>

Client's feedback: 5.0

<p>“Guys were extremely helpful in strategically touching up our book. They was able to provide insightful ideas for proofreading and offer advice on areas we didn't even think of. Extremely talented, very responsive - high quality work in a short amount of time.”</p>

</div>

<div class="col-lg-4 testimonials-block">

<h4>Ernesto Galardo</h4>

Client's feedback: 5.0

<p>“Great service to work with! You are efficient and know what you are doing. Already preparing for my next project with GetItWritten!”</p>

</div>

<div class="col-lg-4 testimonials-block">

<h4>Adam Dorfman</h4>

Client's feedback: 4.9

<p>“This is a great service if you want to write a book and have a limited budget and time.”</p>

</div>

<div class="col-lg-4 testimonials-block">

<h4>Christopher B</h4>

Client's feedback: 5.0

<p>“They are simple the best! Will collaborate with this company again and again!”</p>

</div>

</div>

</div>

</div>

</div>

<!--Here is discount block-->

<div id="special-offer">

<div class="container">

<div class="row">

<div class="col-lg-12">

<div class="box get-offer-block">

<div class="ribbon ribbon-top-right">

</div>

<h3>Get a Special Offer</h3>

<div class="row justify-content-center">

<div class="col-lg-7 col-md-7 col-sm-12 d-flex align-items-center justify-content-xl-end justify-content-lg-end justify-content-md-center justify-content-sm-center justify-content-center">

<p>Order now and get 20% discount</p>

</div>

<div class="col-lg-5 col-md-5 col-md-12 d-flex justify-content-lg-start justify-content-md-center justify-content-center">


```
<button class="btn" onclick="getDiscount()">Get Discount</button>
</a>
</div>
```

```
</div>
</div>
</div>
</div>
</div>
</div>
```

```
</section>
```

```
<section id="contact">
<div class="container">
```

```
<!--Send message form-->
<div class="row">
```

```
<div class="headline">
<h1>Have a Question?</h1>
<hr>
</div>
```

```
<div class="col-lg-12">
<form id="contact-form" action="" method="POST">
<div class="row form-block">
```

```
<!--Contact info block-->
<div class="col-lg-6 col-md-6 contact-info-block">
<h3>Contact us</h3>
<p><a href="mailto:getitwrittenio@gmail.com">getitwrittenio@gmail.com</a></p>
</div>
```

```
<!--Send message block-->
<div class="col-lg-6 col-md-6 send-message-block">
<h3>Send us a Message</h3>
<div>
<input name="form-name" type="name" placeholder="Your name" required>
</div>
<div>
<input name="form-email" type="email" placeholder="Your email" required>
</div>
<div>
<textarea name="form-message" rows="6" placeholder="Start typing your message..."
required></textarea>
</div>
```

```
<div class="d-flex justify-content-end">
<button class="btn" type="submit">Send Message</button>
</div>
```

```
</div>
```

```
</div>
</form>
</div>
```

```
</div>
```

```
<!--Send message form-->  
<div class="row">
```

```
<div class="headline headline-feedback">  
<h1>Leave a Feedback</h1>  
<hr>  
</div>
```

```
<div class="col-lg-12">  
<form id="feedback-form" action="" method="POST">  
<div class="row form-block">
```

```
<!--Send message block-->  
<div class="col-lg-8 offset-lg-2 col-md-8 offset-md-2 send-message-block">  
<h3>Share with us your opinion</h3>  
<div>  
<input name="form-email" type="email" placeholder="Your email" required>  
</div>  
<div>  
<input name="form-email" type="name" placeholder="Your name" required>  
</div>  
<div>  
<textarea name="form-message" rows="6" placeholder="Share with us your opinion..."  
required></textarea>  
</div>  
<div class="d-flex justify-content-end">  
<button class="btn" type="submit">Send Feedback</button>  
</div>  
</div>
```

```
</div>  
</form>  
</div>
```

```
</div>
```

```
<!--Subscribe for news block-->  
<div class="row">  
<div class="col-lg-12 stay-updated-block">  
<h3>Stay updated to our news and offers</h3>
```

```
<form id="stay-updated" enctype="multipart/form-data" method="post">  
<div class="col-lg-6 col-md-8 offset-md-2 offset-lg-3 d-flex justify-content-center">  
<input type="email" name="subscribe-email" placeholder="Enter your e-mail" required>  
<div>  
<a id="go-to-message-subscription" href="#message-subscription">  
<button class="btn" type="submit">Subscribe</button>  
</a>  
</div>  
</div>  
</form>
```

```
</div>  
</div>
```

```
<!--Social Media icons-->
```

```
<div class="row">
<div class="col-lg-12 social-media-block d-flex justify-content-center">
```

```
<!--Facebook link-->
```

```
<a target="_blank" href="https://www.facebook.com/Getitwrittenio-325991454765364">
<svg width="60px" height="60px" viewBox="0 0 60 60" version="1.1"
xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink">
<g id="Page-1" stroke="none" stroke-width="1" fill="none" fill-rule="evenodd">
<g>
<circle class="circle-to-change" fill="#D8D8D8" cx="30" cy="30" r="30"></circle>
<path d="M27.0127277,22.6243484 C27.1399371,20.5392152 27.6031408,19.0610953
28.4023387,18.1899887 C28.8626129,17.6883009 29.4412977,16.9468344 30.4821308,16.530034
C31.2485502,16.2231222 31.6501544,16.1098461 33.4208596,16.0215546 C34.6013297,15.9626937
35.783745,16.0243849 36.9681055,16.2066283 L36.9681055,20.5400718 C35.4422358,20.4683609
34.4287128,20.4683609 33.9275365,20.5400718 C33.1757721,20.6476382 32.733476,20.8372564
32.4040413,21.2612568 C32.1844181,21.5439238 32.0539803,22.1255931 32.0127277,23.0062647
L32.0127277,25.8808045 L36.9080102,25.8808045 L36.1666845,30.7063916 L32.0127277,30.7063916
L32.0127277,43.0062647 L27.0127277,43.0062647 L27.0127277,30.7063916 L23,30.7063916
L23,25.874901 L27.0127277,25.874901 L27.0127277,22.6243484 Z" id="Rectangle-45"
fill="#FFFFFF"></path>
</g>
</g>
</svg>
</a>
```

```
<!--Instagram link-->
```

```
<a target="_blank" href="https://www.instagram.com/getitwritten.io/">
<svg width="60px" height="60px" viewBox="0 0 60 60" version="1.1"
xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink">
<g id="Page-1" stroke="none" stroke-width="1" fill="none" fill-rule="evenodd">
<g>
<circle class="circle-to-change" fill="#D8D8D8" cx="30" cy="30" r="30"></circle>
<g id="Group-51" transform="translate(16.000000, 16.000000)" fill="#FFFFFF">
<path d="M7,0 L21,0 C24.8659932,-7.10171439e-16 28,3.13400675 28,7 L28,21 C28,24.8659932
24.8659932,28 21,28 L7,28 C3.13400675,28 4.73447626e-16,24.8659932 0,21 L0,7 C-4.73447626e-
16,3.13400675 3.13400675,7.10171439e-16 7,0 Z M8,2 C4.6862915,2 2,4.6862915 2,8 L2,20
C2,23.3137085 4.6862915,26 8,26 L20,26 C23.3137085,26 26,23.3137085 26,20 L26,8 C26,4.6862915
23.3137085,2 20,2 L8,2 Z" id="Combined-Shape"></path>
<path d="M14,21 C10.1340068,21 7,17.8659932 7,14 C7,10.1340068 10.1340068,7 14,7 C17.8659932,7
21,10.1340068 21,14 C21,17.8659932 17.8659932,21 14,21 Z M14,19 C16.7614237,19 19,16.7614237
19,14 C19,11.2385763 16.7614237,9 14,9 C11.2385763,9 9,11.2385763 9,14 C9,16.7614237
11.2385763,19 14,19 Z" id="Combined-Shape"></path>
<circle fill="#FFFFFF" id="Oval-6-Copy" cx="21.5" cy="6.5" r="1.5"></circle>
</g>
</g>
</g>
</svg>
</a>
```

```
<!--Twitter link-->
```

```
<a target="_blank" href="https://twitter.com/GetitwrittenI">
<svg width="60px" height="60px" viewBox="0 0 60 60" version="1.1"
xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink">
<g id="Page-1" stroke="none" stroke-width="1" fill="none" fill-rule="evenodd">
<g>
<circle class="circle-to-change" fill="#D8D8D8" cx="30" cy="30" r="30"></circle>
<path d="M15,38.3377866 C21.4396662,42.1048576 27.5354832,42.1048576 33.2874509,38.3377866
C36.3301764,36.3450489 38.0683591,34.079111 39.1982688,32.0151978 C41.2723309,28.2266792
41.0139779,24.9077282 41.0139779,23.7170757 C41.0139779,23.6432042 41.6325895,23.3262454
42.4390946,22.529459 C42.7466901,22.2255701 43.2682763,21.6119798 44.003853,20.6886882
```

L40.6580589,21.5884155 C41.0880916,21.3206631 41.5511164,20.9057412 42.0471334,20.3436495
C42.3130353,20.0423267 42.6948236,19.3922068 43.1924981,18.3932899 C40.8466513,19.4796126
39.5952129,19.9481739 39.4381828,19.7989736 C38.2399671,18.6605027 36.4599271,18
34.7140922,18 C31.6765261,18 28.9030789,20.798496 28.9030789,23.8360621
C28.9030789,24.0370455 28.9649191,24.4609217 29.0885996,25.1076905 C26.7494128,25.0057366
24.5280152,24.4372313 22.4244069,23.4021745 C20.3207985,22.3671177 18.4513944,20.9498763
16.8161945,19.1504505 C15.9116568,20.5934366 15.736173,22.0729223 16.2897431,23.5889076
C16.8433132,25.1048928 17.6055975,26.1540454 18.5765959,26.7363651 C17.9500432,26.6865247
17.4887579,26.6203378 17.19274,26.5378042 C16.8967221,26.4552707 16.4907104,26.2796442
15.9747048,26.0109248 C15.9286734,27.5865846 16.3939026,28.8851982 17.3703923,29.9067656
C18.3468821,30.928333 19.4452941,31.5573429 20.6656285,31.7937954 C20.0761111,31.941397
19.6372439,32.0151978 19.3490269,32.0151978 C19.06081,32.0151978 18.6160001,31.981619
18.0145973,31.9144615 C18.713892,33.3496876 19.339643,34.2649521 19.8918502,34.660255
C21.7304705,35.9764493 23.5560046,35.9764493 23.5560046,35.9764493 C23.5560046,35.9764493
20.9554705,37.8272662 18.5765959,38.3377866 C17.8237872,38.4993438 16.6315885,38.4993438
15,38.3377866 Z" id="Oval-7" fill="#FFFFFF"></path>

</g>

</g>

</svg>

</div>

</div>

</div>

</section>

</body>

<script src="js/scripts.min.js"></script>

<script src="libs/slick-carousel/slick/slick.min.js"></script>

<script src="libs/numscroller-1.0.js"></script>

</body>

</html>