

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ПОЛТАВСЬКА ПОЛІТЕХНІКА ІМЕНІ
ЮРІЯ КОНДРАТЮКА»
(повне найменування вищого навчального закладу)
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ І
РОБОТОТЕХНІКИ
(повна назва факультету)
КАФЕДРА КОМП'ЮТЕРНИХ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ І
СИСТЕМ
(повна назва кафедри)

Пояснювальна записка

до дипломного проекту (роботи)

магістра

(освітньо-кваліфікаційний рівень)

на тему:

«Проектування інтеграційної інформаційної системи для ТОВ «БВС
Ритейл».

Виконав: студент б курсу, групи 602-ТН
спеціальності 122 Комп'ютерні науки
(шифр і назва напрямку підготовки)

Чернуха Олексій Сергійович
(прізвище та ініціали)

Керівник к.т.н., доц. Головка Г.В
(прізвище та ініціали)

**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ПОЛТАВСЬКА ПОЛІТЕХНІКА ІМЕНІ
ЮРІЯ КОНДРАТЮКА»**

**НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ І
РОБОТОТЕХНІКИ**

**КАФЕДРА КОМП'ЮТЕРНИХ ТА ІНФОРМАЦІЙНИХ
ТЕХНОЛОГІЙ І СИСТЕМ**

КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА

спеціальність 122 «Комп'ютерні науки»
на тему:

**«ПРОЕКТУВАННЯ ІНТЕГРАЦІЙНОЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ ДЛЯ
ТОВ «БВС РИТЕЙЛ»**

Студента групи 602-ТН Чернухи О.С.

Керівник роботи
к.т.н., доц. Головка Г.В.

Завідувач кафедри
к.т.н., доц. Двірна О.А.

РЕФЕРАТ

Кваліфікаційна робота магістра: 87 сторінок, 40 малюнки, 9 таблиць, 1 додаток, 24 джерел.

Об'єкт дослідження: інтеграційна інформаційна система для ТОВ «БВС Ритейл» з використанням систем пошуку інформації, що стосуються діяльності АЗС.

Предмет досліджень: програмні засоби розробки інформаційної системи, діяльності АЗС.

Мета роботи: створення інформаційної системи, яка допоможе користувачеві ознайомитись з особливостями роботи та перевагами ТОВ «БВС Ритейл», знайти необхідний матеріал.

Методи: проектування та розроблення інформаційної системи.

Ключові слова: інформаційна система, пошук інформації, Yii Framework, CSS, APACHE PHP, JAVASCRIPT, MySQL, HTML.

ABSTRACT

Master's qualification work: 87 pages, 40 drawings, 9 tables, 1 appendix, 24 sources.

Object of research: integration information system for LLC "BVS Retail" using information search systems related to the activities of gas stations.

Subject of research: software tools for developing an information system, gas station activities.

Purpose of work: creation of an information system that will help the user to familiarize themselves with the features of the work and advantages of LLC "BVS Retail", find the necessary material.

Methods: design and development of an information system.

Keywords: information system, information search, Yii Framework, CSS, APACHE PHP, JAVASCRIPT, MySQL, HTML.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ.....	8
ВСТУП.....	9
РОЗДІЛ 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ	10
1.1 Постановка задачі.....	10
1.2 Актуальність роботи	11
1.3 Особливості інформаційних систем.....	11
1.4 Основні риси інформаційних систем	15
Основні рисами ІС є:.....	15
1.5 Управління відносинами з клієнтами.....	16
1.6 Бізнес-аналітика з використанням інформаційних технологій	18
Інформаційні технології стають невід'ємною частиною стратегічного зростання та розвитку сучасних підприємств, забезпечуючи їм конкурентну перевагу над ринком. [21].....	19
1.6 Інформаційний пошук.....	19
1.7 Огляд існуючих інформаційних систем пов'язаних з АЗС.....	21
РОЗДІЛ 2 ПРОЕКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ.....	24
2.1 Про необхідність інформаційних систем.....	24
2.2 Веб-сайта склад та визначення	24
2.3 Проектування інформаційної системи	27
2.4 Етапи розробки інформаційної системи	27
Вибір програмних засобів для розробки ІС.....	27
Yii framework.....	28
2.5 Інтерфейс інформаційної системи.....	29
Принципи проектування інтерфейсу користувача	29
Використання кольорів в графічному інтерфейсі.	31
2.6 Створення бази даних	33
2.7 Розробка адміністративної частини сайту	35
2.8 Засоби розробки	36

2.9.1 Мова розмітки гіпертекстових документів HTML.....	36
Мова розмітки гіпертекстових документів DHTML.....	37
Розширювана мова розмітки XML.....	37
Мова програмування PHP.....	38
Локальний веб-сервер.....	38
Локальний веб-сервер Apache.....	39
Діаграми варіантів використання.....	39
2.10 ER-модель бази даних.....	41
Пошук інформації.....	41
3.1 Розробка інформаційної системи.....	43
3.2 Захист інформації.....	46
Облікові записи.....	47
3.3 Демонстрація роботи адміністратора.....	48
3.4 Демонстрація сторінок інформаційної системи.....	50
РОЗДІЛ 4 ТЕСТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ.....	54
4.1 Тестування верстки.....	54
4.2 Функціональне тестування.....	56
ВИСНОВКИ.....	59
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	60
ДОДАТОК.....	63

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ

ІС – інформаційна система.

ОС – операційна система.

ПЗ – програмне забезпечення.

ІІ – інформаційний пошук.

ООП – об'єктно-орієнтоване програмування.

PHP – мова програмування для написання web-додатків, що виконуються на web-сервері.

HTML – стандартизована мова розмітки документів для перегляду вебсторінок у браузері..

XML – extensible markup language – розширювана мова розмітки.

JS — JavaScript, динамічна, об'єктно-орієнтована прототипна мова програмування.

CSS – cascading style sheets — каскадні таблиці стилів.

CMS – програмне забезпечення, призначене для конструювання і керування веб-сайтами.

Yii – високопродуктивний веб-фреймворк.

ВСТУП

Інформаційно-комунікаційні технології, що з'явилися у другій половині ХХ ст., суттєво змінили життя людства. Саме вони створили передумови формування інформаційного суспільства, в якому визначальну роль відіграють інформація та нові знання.

Перші ЕОМ були призначені лише для швидкої обробки числових даних. Згодом обчислювальна техніка стала широко використовуватися в наукових дослідженнях, виробництві, освіті, побуті тощо. У користувачів віддалених один від одного комп'ютерів з'явилася потреба у швидкому обміні даними. Для цього було запропоновано об'єднати комп'ютери в єдину систему і таким чином передавати дані від одного комп'ютера до іншого.

Останнім часом активно розвиваються нові технології на основі інформатики. Вони використовують важливість інформаційних систем. Взаємодія підприємства з постачальниками, споживачами та іншими організаціями здійснюється через матеріальні та інформаційні потоки.

Проте, існують виклики в подоланні місць перетину між інформаційними системами. Це стає проблемою, коли потоки перетинають межі компетенції та відповідальності різних підрозділів або організацій. Використання інформаційної техніки може значно покращити ефективність і забезпечити інтеграцію систем в межах підприємства.

Завданням інформаційної системи є збереження та підбір інформації, потрібної для ефективного управління, створення інформаційного та технічного середовища для управління діяльністю підприємства. Інформаційна система може існувати і без застосування комп'ютерної техніки – це питання економічної необхідності.

РОЗДІЛ 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ

1.1 Постановка задачі

Об'єкт дослідження: інтеграційна інформаційна система для ТОВ «БВС Ритейл» з використанням систем пошуку інформації, що стосуються діяльності АЗС.

Предмет досліджень: програмні засоби розробки інформаційної системи, діяльності АЗС.

Мета роботи: створенні інформаційної системи, яка допоможе користувачеві ознайомитись з особливостями роботи та перевагами ТОВ «БВС Ритейл», знайти необхідний матеріал.

В кваліфікаційній роботі необхідно запроектувати та впровадити інтеграційну інформаційну системи для ТОВ «БВС Ритейл», що надасть користувачеві можливість:

- переглянути інформацію;
- зареєструватися в якості користувача;
- знайти необхідні товари;
- зробити відповідне замовлення.

Завданням магістерської кваліфікаційної роботи є розробка клієнт – серверної структури, що забезпечить цілісність, доступність та спостережність інформації, а також створення зрозумілого та простого інтерфейсу.

Системою можуть користуватися всі клієнти, вони повинні мати доступ до перегляду інформації та створення замовлення. Адміністратор повинен мати можливість редагувати, переглядати, додавати та видаляти та розділи, а також створювати та керувати можливістю пошуку інформації.

1.2 Актуальність роботи

Використання різних інформаційних систем підвищує продуктивність й ефективність роботи, дозволяє по-іншому вирішувати багато нових та невідкладних завдань. Так, інформаційні системи дозволяють зберігати величезну кількість даних, аналізувати їх і на основі результату пропонувати найбільш ефективні рішення господарських задач.

При виборі інформаційної системи, необхідно пройти певні етапи:

- формування завдань і цілей, які необхідно досягти при запровадженні системи;
- визначення кошторису для виконання робіт;
- експрес-діагностика виробничих процесів;
- формування основних вимог для обраної систем;
- аналіз ринку і складання списку вимог;
- зустріч з розробником програм, обговорення з ними особливостей впровадження;
- попередній вибір рішення, обговорення з розробником програм деталей і етапів проекту та можливих ризиків;
- остаточний вибір інформаційної системи.

1.3 Особливості інформаційних систем

Інформаційна система (ІС) — система обробки інформації спільно з відповідними організаційними ресурсами (людськими, технічними, фінансовими тощо), яка забезпечує та розповсюджує інформацію (ISO/IEC 2382:2015) [1].

ІС призначена для своєчасного забезпечення належних людей належною інформацією, тобто для задоволення конкретних інформаційних потреб у рамках певної предметної галузі, при цьому результатом функціонування інформаційних

систем є інформаційна продукція - документи, інформаційні масиви, бази даних та інформаційні послуги [8]

Інформаційні системи – системи обробки даних про яку-небудь предметну область із засобами нагромадження, збереження, відновлення, пошуку і видачі даних. [1]

Поява інформаційних систем, основним призначенням яких є вирішення відзначеної проблеми, стало відповіддю комп'ютерної індустрії на вимоги дня. Звичайно, обсяги інформації, з якими доводиться мати справу інформаційним системам, досить великі, а сама інформація має досить складну структуру. Класичними прикладами інформаційних систем є банківські системи, системи резервування авіаційних або залізничних квитків, місць у готелях. [2]

У сучасному світі інформаційні технології (ІТ) відіграють ключову роль в оптимізації бізнес-процесів. Вони надають інструменти, які дозволяють компаніям реалізовувати стратегії, спрямовані на прискорення бізнес-процесів, зниження операційних витрат та покращення якості послуг та продуктів. Для прискорення операцій, зниження витрат та підвищення загальної ефективності. стала невід'ємною частиною кожного аспекту ведення бізнесу.

Інтеграція систем та автоматизація процесів за допомогою ІТ-інструментів надають компаніям низку значних переваг, робить бізнес більш гнучким, ефективним та конкурентоспроможним.

Централізоване управління бізнес-процесами: Системи ERP (Enterprise Resource Planning) дозволяють об'єднати різні аспекти бізнесу, включаючи фінанси, закупівлі, управління ланцюжками поставок, виробництво, продаж та управління персоналом в одній інтегрованій системі. Це забезпечує централізований доступ до всіх ключових даних та процесів.

Оптимізація бізнес-операцій: Автоматизація рутинних завдань знижує потребу в ручній обробці даних, що не лише прискорює виконання операцій, а й зменшує ймовірність помилок. Наприклад, автоматизація процесів обліку та інвентаризації покращує точність даних та прискорює їх обробку, що важливо для стратегічного планування та управління.

Поліпшення прийняття рішень: Інтегровані ІТ-системи надають повну та актуальну картину бізнесу, що допомагає керівництву компанії приймати обґрунтовані рішення. ERP-системи можуть надавати детальні звіти про продаж, запаси та фінансові показники, що дозволяє керівникам адаптувати бізнес до мінливих умов ринку та оптимізувати бізнес-стратегії.

Підвищення прозорості та контроль за процесами: Керівники отримують можливість спостерігати за операціями в реальному часі, що дозволяє своєчасно виявляти та вирішувати проблеми, а також вживати заходів для їх запобігання у майбутньому.

Зниження витрат: Автоматизація та інтеграція ІТ-систем призводять до зменшення часу на ручну роботу та підвищує ефективність процесів – дозволяють компаніям знизити витрати та підвищити прибутковість.

Гнучкість та масштабованість: Інтегровані ІТ-рішення забезпечують гнучкість, необхідну для адаптації до змін у бізнес-середовищі. Компанії можуть легко масштабувати свої операції, додавати нові функції або модулі до існуючих систем, що сприяє розширенню бізнесу та відкриває нові ринкові можливості.

Використання великих даних для прийняття рішень

За допомогою складних аналітичних інструментів та алгоритмів обробки даних компанії витягують цінну інформацію з величезних обсягів зібраних даних. Цей процес включає у собі як аналіз купівельного поведінки й переваг клієнтів, а й прогнозування майбутніх трендів ринку, виявлення неефективних операцій та оптимізацію процесів постачання. Особлива увага приділяється аналізу споживчих даних. Розуміння того, як клієнти взаємодіють із продуктами чи послугами, дозволяє бізнесу налаштовувати свої маркетингові та продажні стратегії для задоволення специфічних потреб цільової аудиторії.

Аналіз великих даних також допомагає оптимізувати внутрішні бізнес-процеси. Компанії можуть виявляти вузькі місця у своїх операційних процесах, покращувати управління запасами та підвищувати ефективність логістики. Не лише знижує витрати, а й покращує загальне якість обслуговування клієнтів.

1.4 Основні риси інформаційних систем

Основні рисами ІС є:

- наявність середовища, що забезпечує збереження і доступ до даних. Як уже зазначалося, будь-яка інформаційна система призначена для збору, збереження й обробки інформації. Тому в основі будь-якої інформаційної системи лежить середовище збереження й доступу до даних. В обчислювальних програмних системах наявність такого середовища не є обов'язковим. Основною вимогою до програми, що виконує розрахунки, є її швидкодія. Потрібно, щоб програма зробила досить точні результати за встановлений час. Зовнішня пам'ять, звичайно, використовується для періодичного збереження проміжних результатів обчислень, щоб у випадку збою комп'ютера можна було продовжити роботу програми від збереженої контрольної крапки;
- простий, зручний, легко освоюваний інтерфейс. Інтерфейс інформаційної системи повинен надати кінцевому користувачу всі необхідні для його роботи функції, але в той же час не дати йому можливість виконувати зайві дії. Зараз це графічний інтерфейс і в будь-якому випадку наявність розвинених інтерфейсних засобів є обов'язковим для будь-якої сучасної інформаційної системи;
- ефективний доступ до даних;
- надійність і тривалість збереження інформації. Найбільш істотною складовою є інформація, що довго накопичується і втрата якої непоправна;
- можливість збереження даних, що володіють різними структурами.
- збереження даних, що володіють різними структурами. Важко уявити собі більш-менш розвинену інформаційну систему, що працює з одним однорідним файлом даних. Вся накопичена раніше інформація повинна залишитися збереженою;

- введення, відновлення й видалення даних. Підтримка цих функцій істотно підвищує рівень вимог до інформаційних систем;
- можливість роботи із системою з кількох робочих місць. Якщо говорити про великі інформаційні системи, то їхня наявність припускає можливість роботи із системою з кількох робочих місць. [7]

1.5 Управління відносинами з клієнтами

Системи управління відносинами з клієнтами (CRM) є комплексними рішеннями, які дозволяють бізнесу глибше розуміти і ефективніше взаємодіяти з клієнтами. Ці системи інтегрують та аналізують дані з різних точок контакту з клієнтами, включаючи електронну пошту, соціальні мережі, дзвінки та особисті зустрічі. Такий підхід дозволяє компаніям отримувати цілісне уявлення про своїх клієнтів, що є критично важливим для розробки персоналізованих маркетингових стратегій та пропозицій. [7]

За допомогою CRM, таких як SAP або Oracle, компанії можуть відслідковувати історію взаємодій клієнтів, їх переваги та відгуки, що допомагає у створенні індивідуалізованих пропозицій та покращенні якості обслуговування. Наприклад, на основі даних CRM ритейлери можуть пропонувати персоналізовані знижки або рекомендації товарів, що підвищує лояльність клієнтів та сприяє збільшенню продажів. [7]

CRM-системи допомагають оптимізувати внутрішні процеси, пов'язані з обслуговуванням клієнтів, спрощують завдання для відділів продажу, маркетингу та підтримки, тим самим покращують комунікації між відділами та забезпечують доступ до актуальної інформації про клієнтів. [7]



Рисунок 1.2– CRM-системи

Нові можливості в просуванні та маркетингу

Сучасні IT-рішення дозволяють компаніям використовувати великі дані та аналітичні інструменти для розробки більш цілеспрямованих та ефективних маркетингових кампаній. Використання даних, отриманих з різних джерел, таких як соціальні мережі, веб-аналітика та CRM-системи, дає можливість глибше зрозуміти інтереси та потреби споживачів. [7]

Цифрові канали, включаючи соціальні мережі, мобільні програми та онлайн-платформи, пропонують нові шляхи для взаємодії зі споживачами. Ці платформи дозволяють компаніям вести діалог з клієнтами, отримувати миттєвий зворотний зв'язок та швидко адаптувати свої стратегії відповідно до змін у споживчій поведінці. [7]

Автоматизовані системи можуть виконувати ряд завдань, від сегментації аудиторії та персоналізації повідомлень до управління рекламними кампаніями та аналізу їхньої ефективності. Це дозволяє маркетологам більш ефективно розподіляти ресурси та збільшувати ROI своїх кампаній. Автоматизація процесів з використанням ІІ дозволяє маркетологам зосередитися на стратегічних аспектах, покращуючи ефективність маркетингових зусиль та підвищуючи залучення клієнтів. [7]

1.6 Бізнес-аналітика з використанням інформаційних технологій

Сучасні інформаційні технології надають бізнес-аналітикам потужні засоби для глибокого дослідження даних та виявлення ключових трендів. даних, а не інтуїції. Прикладом таких програм є Tableau, яка дозволяє бізнес-аналітикам візуалізувати дані та створювати інтерактивні звіти та дашборди, і Microsoft Power BI як засіб для створення динамічних звітів та інфографіки. Їх функціональність дозволяє проводити глибокий аналіз даних, інтегрувати дані з різних джерел, виявляти тренди та патерни, що допомагає у прийнятті обґрунтованих рішень.



Рисунок 1.3– Бізнес-аналітика з використанням інформаційних технологій

Інформаційні технології також дозволяють автоматизувати процеси збору та аналізу даних, що значно знижує ймовірність людських помилок та збільшує точність аналітичних висновків. Компанії можуть використовувати алгоритми машинного навчання для прогнозування майбутніх трендів на основі зібраних даних.

Бізнес-аналітика з використанням інформаційних технологій також сприяє оптимізації бізнес-процесів. Аналіз даних може виявити вузькі місця та неефективні операції, що дозволяє компаніям внести покращення та заощадити

ресурси. . Вона надає інструменти для візуалізації, аналізу та автоматизації бізнес-процесів, що сприяє підвищенню ефективності та скорочення витрат.

Інформаційні технології стають невід'ємною частиною стратегічного зростання та розвитку сучасних підприємств, забезпечуючи їм конкурентну перевагу над ринком. [21]

1.6 Інформаційний пошук

Інформаційний пошук - процес пошуку неструктурованої інформації, що задовольняє інформаційні потреби. [12]

Пошук інформації -- це процес виявлення у величезній кількості документів які присвячені певній темі, що задовольняє певній умові пошуку або містить в собі необхідні дані.

Пошук інформації складається:

- визначення інформаційної потреби та формулювання інформаційного запиту;
- визначення сукупності можливих власників інформаційних масивів;
- відбір інформації з виявлених інформаційних масивів;
- ознайомлення з отриманою інформацією та оцінка результатів пошуку. [8]

Завдання інформаційного пошуку

Центральне завдання ІІ – допомогти користувачеві задовольнити його інформаційну потребу. Так як описати інформаційні потреби користувача технічно непросто, вони формулюються як деякий запит, що є набором ключових слів, що характеризує те, що шукає користувач. [8]

Розробники алгоритму постаралися відтворити динамічну, асоціативну природу людського мислення. Вони виходили з того, що при прийнятті рішень люди, використовують для обробки інформації неієрархічні способи аналізу. Обдумуючи задачу, кожна людина йде до висновку своїм шляхом. Користувачам

надається можливість аналізувати дані відповідно до власних розумових процесів. [12]

Традиційні методи пошуку і фільтрації інформації були розроблені для бібліотечних баз даних, тобто для інформаційних ресурсів обмеженого обсягу і заздалегідь відомої структури. Створення глобальної мережі і вихід Web за рамки інтересів наукового співтовариства призвели до того, що число постачальників інформації стало стрімко рости, при тому, що опублікована ними інформація не мала однорідної структури. Наступний інформаційний вибух став викликом стандартним інформаційним технологіям. Нові масштаби з одного боку зробили аутсайдерами деякі раніше конкурентоспроможні інтелектуальні технології, а з іншого – стимулювали інтенсивні дослідження в області лінгвістичних та імовірнісних методів обробки текстової інформації та нових методів навігації в неоднорідному інформаційному морі. [8]

Асоціативний пошук ґрунтується на двох ключових принципах:

- всі дані зберігаються в пам'яті комп'ютера (ОЗУ);
- обчислення виконуються в реальному часі.

Ці архітектурні рішення підкріплюються двома важливими тенденціями в розвитку комп'ютерів. Перша – перехід від 32- до 64-розрядних обчислень, що призвело до експоненціального збільшенню розміру ОЗУ комп'ютера. Зміни привели до серйозного збільшення обчислювальної потужності додатків, орієнтованих на паралельні обчислення. Високопродуктивні сервери пропонуються менш ніж за 50 000 дол. Зовсім недавно для досягнення такого рівня обчислювальних можливостей були потрібні спеціально сконструйовані комп'ютери вартістю сотні тисяч і навіть мільйони доларів. [12]

Архітектура зберігання і обробки інформації в оперативній пам'яті (in-memory), дозволяє маніпулювати величезними обсягами даних, підтримуючи високий рівень інтерактивності. Стиснення даних при завантаженні в пам'ять дозволяє зберігати дані в ОЗУ ефективніше, ніж на диску, як це прийнято в традиційних реляційних базах даних. При передачі даних в пам'ять система буде карту посилань між елементами даних, щоб спростити візуалізацію їх зв'язків.

Завдяки здатності виконувати обчислення в реальному часі можна швидко розраховувати складні заходи і метрики. В новітніх системах закладена можливість розподіляти обчислення між всіма доступними ядрами процесорів і управляти робочим навантаженням одночасно для великої кількості користувачів. Крім того, платформа зберігає в кеш-пам'яті результати всіх запитів, тому типові обчислення виконуються мінімальна кількість разів. [12]

1.7 Огляд існуючих інформаційних систем пов'язаних з АЗС

Як правило подібні інформаційні системи створюють з гарним інтерфейсом. Головною задачею системи є - інформувати користувача та надати максимальну кількість відомостей про послугу. Інтерфейс має бути доступним та зрозумілим для будь-якого клієнта. Повинна бути проста навігація, щоб кожен зміг підібрати необхідні данні та замовити необхідний товар.

Приклади кількох сайтів, що спеціалізовані для надання послуг по пошуку інформації в області АЗС:

1 АЗС «ОККО» <https://www.okko.ua/>

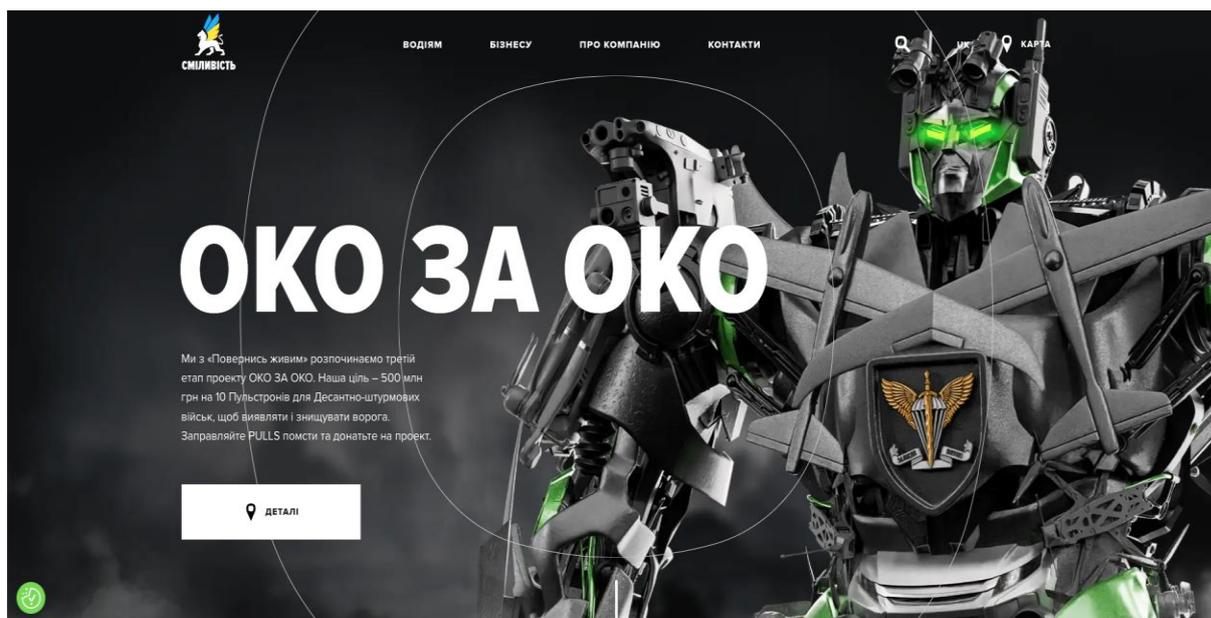


Рисунок 1.1– Сайт АЗС «ОККО»

Недоліки:

- трохи застарілий дизайн;
- темні кольори;
- не має анімацій;
- мало цінової інформації.

2 АЗС “WOG” <https://wog.ua/>

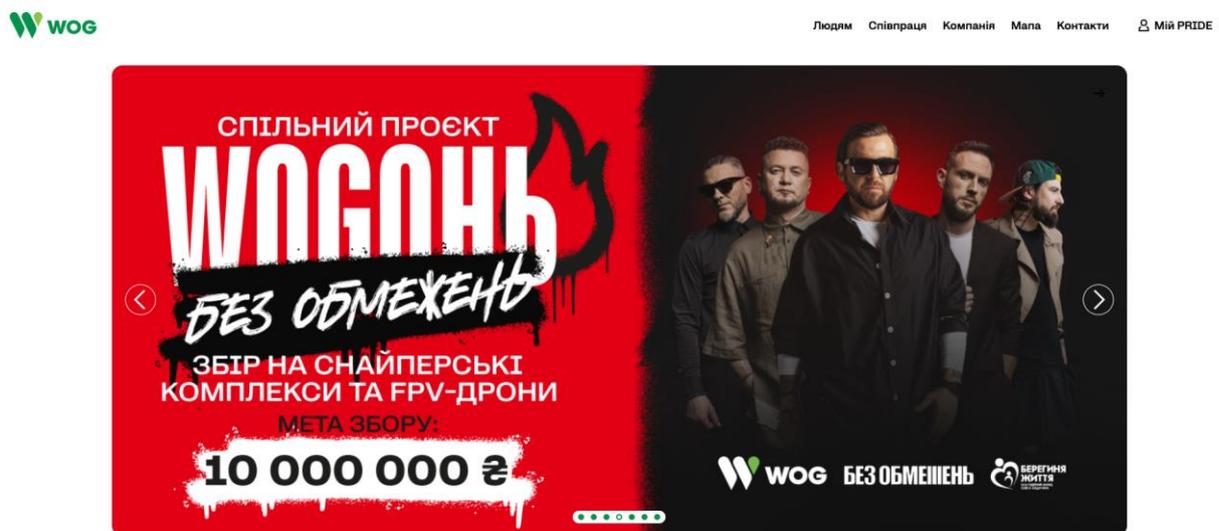


Рисунок 1.2 – Сайт АЗС “WOG”

Недоліки:

- щоб переглянути меню потрібно скачувати PDF що не зовсім зручно.

3 АЗС «Укрнафта» <https://azs.ukrnafta.com/>

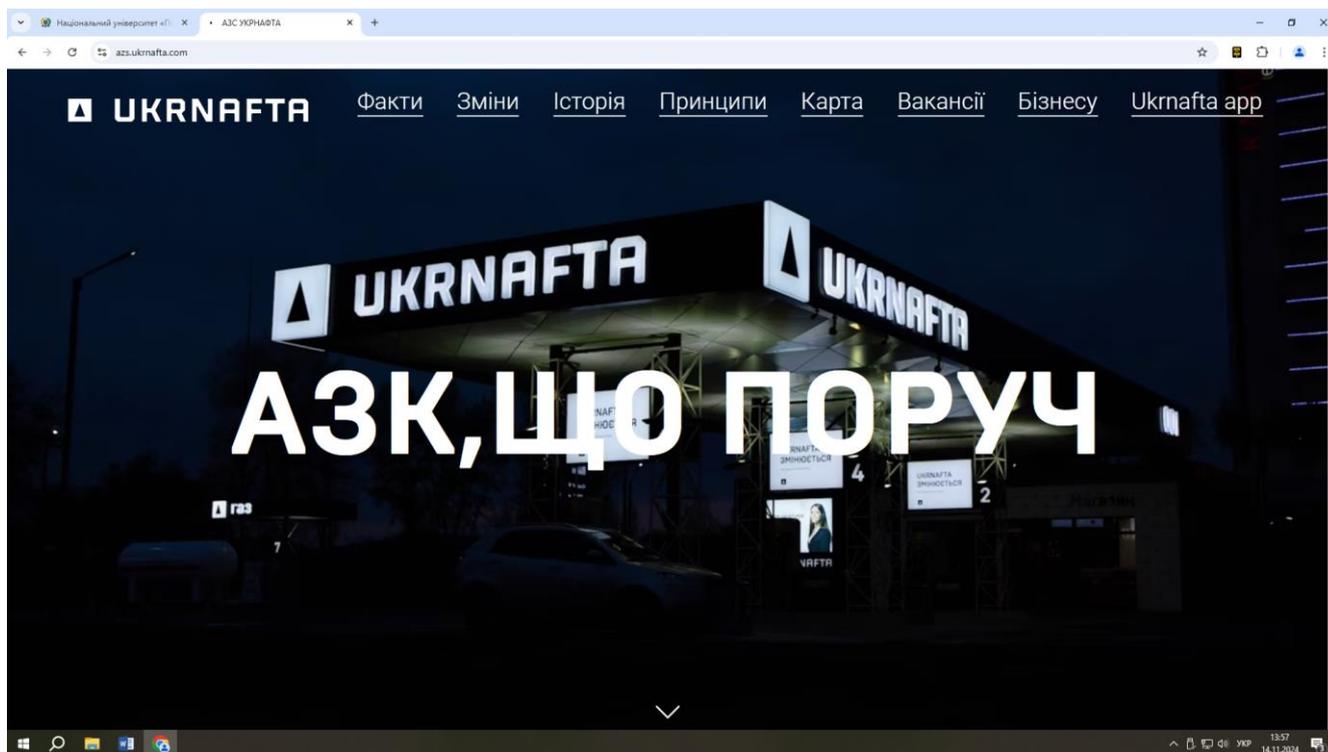


Рисунок 1.3 Сайт АЗС «Укрнафта»

Недоліки:

- досить похмурий вигляд.

РОЗДІЛ 2 ПРОЕКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ

2.1 Про необхідність інформаційних систем

Наразі кожна організація на сьогодні повинна мати власний веб-сайт – це необхідний чинник існування, що розширює рекламну діяльність і привертає додаткових клієнтів.

Проектування інформаційних систем складається з 5 пунктів:

1. Створення первинного технічного завдання на розробку ІС.
2. Розробка структурної схеми інформаційної системи.
3. Розробка веб-дизайну (графічних елементів ІС, стилів і елементів навігації).
4. Створення програмного коду і інших елементів інформаційної системи.
5. Тестування і розміщення в мережі Інтернет. [11]

2.2 Веб-сайта склад та визначення

Вебсайт або **сайт** (англ. *website*, від *web* (веб) і *site* (місце)) — сукупність вебсторінок та залежного вмісту, доступних у мережі Інтернет, які об'єднані як за змістом, так і за навігацією під єдиним доменним ім'ям. Фізично сайт може розміщуватися як на одному, так і на кількох серверах.

Сайтом також називають вузол мережі Інтернет, комп'ютер, за яким закріплена унікальна ІР-адреса, і взагалі будь-який об'єкт в Інтернеті, за яким закріплена адреса, що ідентифікує його в мережі (FTP-site, WWW-site тощо).

Набір зв'язаних між собою інформаційних онлайн-ресурсів, призначених для перегляду через комп'ютерну мережу за допомогою спеціальних програм — браузерів. Вебвузол може бути набором документів в електронному вигляді, онлайн-сервісом.

Сукупність веб-сторінок, тематично пов'язаних між собою і розроблених як єдине ціле, називають веб-сайтом. [11]



Рисунок 2.1 – Типи веб-сайтів

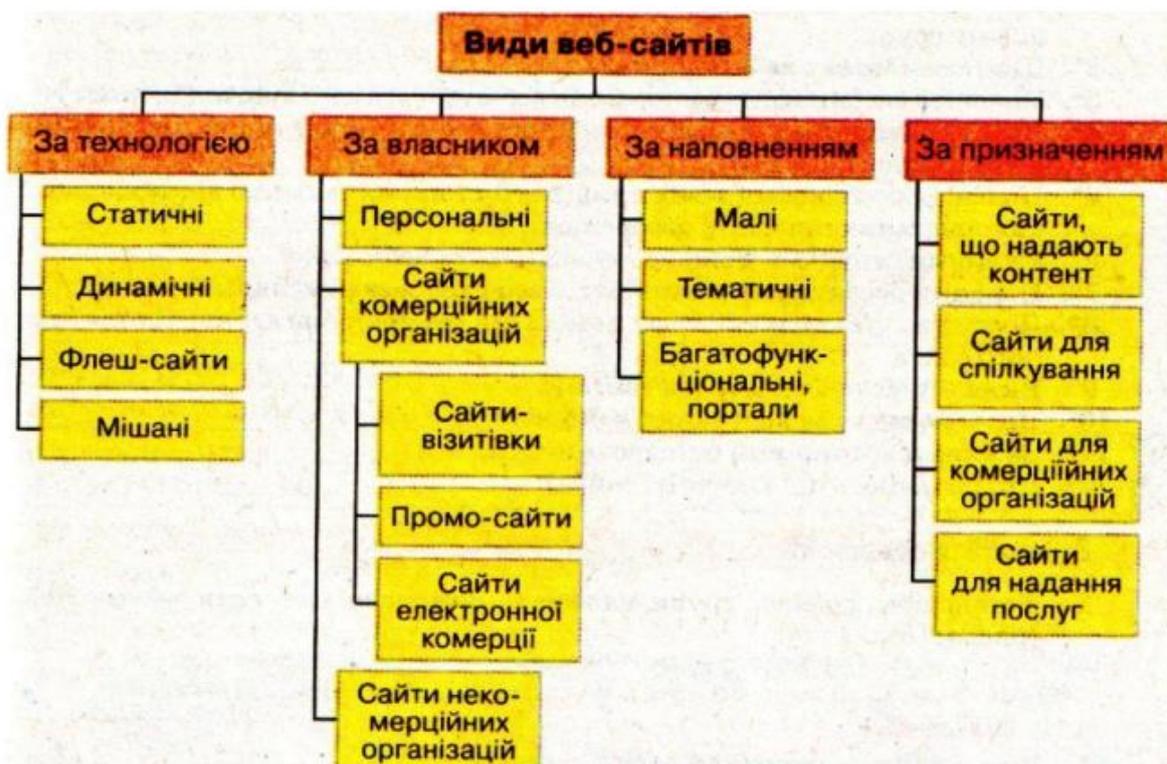


Рисунок 2.2 – Види веб-сайтів

Веб-сайти можна класифікувати за доступністю сервісів, фізичним розташуванням і призначенням.

За доступністю сервісів їх поділяють на:

- відкриті – усі сервіси повністю доступні для будь-яких відвідувачів і користувачів;
- напіввідкриті – для доступу потрібно зареєструватися (зазвичай безкоштовно);
- закриті – повністю закриті службові сайти організацій (наприклад, корпоративні сайти), особисті сайти приватних осіб. Такі сайти доступні для вузького кола людей. Доступ нових людей можливий через запрошення. [6]

Якщо сайт доступний користувачам з Інтернету, він вважається зовнішнім, натомість сайт, доступ до якого можуть здійснювати лише користувачі локальної мережі, є внутрішнім. Прикладами внутрішнього сайту можуть бути корпоративний сайт підприємства або сайт приватної особи в локальній мережі провайдера. За призначенням сайти поділяють на:

- бізнес-сайти – сайти, що містять інформацію про компанії та їхні послуги, здійснюють функцію електронної торгівлі;
- інформаційні сайти – призначені для інформування відвідувачів, поширення новин, тематичні сайти, енциклопедії, словники тощо;
- сайти соціальних мереж – інтерактивні багатокористувацькі веб-сайти, які наповнюються самими учасниками мережі. Сайт являє собою автоматизоване соціальне середовище, що дозволяє спілкуватися групі користувачів, об'єднаних спільним інтересом;
- веб-портали – універсальні сайти, через які можна вийти на інші ресурси Інтернету;
- сайти сервісів – сайти служб, які існують у мережі Інтернет, зокрема, сайти пошукових служб (Google, Bing), поштові сайти, веб-форуми, он-лайніві сховища даних (Skydrive), сайти служб онлайнного документообігу

(Google Docs), зберігання та обробки фотографій (Picnik, ImageShack, Panoramio, Photobucket), зберігання відео (You Tube). [6]

2.3 Проектування інформаційної системи

Етапи проектування ІС

1. Створення первинного технічного завдання на розробку ІС.
2. Розробка структурної схеми інформаційної системи.
3. Розробка веб-дизайну (графічних елементів ІС, стилів і елементів навігації).
4. Створення програмного коду і інших елементів інформаційної системи.
5. Тестування і розміщення в мережі Інтернет. [11]

2.4 Етапи розробки інформаційної системи

Вибір програмних засобів для розробки ІС. Для розробки інформаційної системи було вибрано CMS систему на базі фреймворку Yii, вона використовує базу даних MySQL для збереження інформації. Ця написана на мові PHP, а також є відносно простою в управлінні і має гнучку систему функціонування. [14]

Також було вибрано набір програм для створення сайту «Open Server», який містить у собі:

- Apache;
- PHP з підтримкою GD і MySQL;
- MySQL – з підтримкою транзакцій (mysqld-max);
- систему керування віртуальними хостами, засновану на шаблонах;
- CMS – WordPress,
- систему керування запуском і завершенням;
- Adminer – систему керування MySQL через Web-інтерфейс.

- ядро Perl без стандартних бібліотек (вони поставляються окремо);
- емулятор sendmail, підтримується робота разом з PHP і Perl.

За допомогою «Open Server» можна розмістити інформаційну систему на локальному комп'ютері без виходу в інтернет.

Також було використано:

- Visual Studio Code,
- Figma,
- Photoshop [14]

Yii framework. Yii (вимовляється як «ґеє» або [ji:]) – це високопродуктивний веб-фреймворк, написаний на PHP, реалізує парадигму модель-вид-контролер. Yii – скорочення від «Yes It Is!». [12]

Фреймворк PRADO був спробою перенести ASP.NET на платформу PHP, включаючи ViewState, PostBacks, Page_Load та OnClick. Якщо подивитися вихідний код, то можна побачити що деякі шматки коду просто скопійовані з ASP.NET (наприклад, поділ на Rare Fields та Occasional Fields в класі Control з метою оптимізації використання пам'яті, яка має сенс в .NET, але являє сумнівну цінність в PHP). [12]

PRADO успадкував від ASP.NET майже всі негативні сторони: повільно обробляв складні сторінки, мав круту криву навчання і був достатньо важкий у налаштуванні.

У певний момент автор (Qiang Xue) зрозумів, що PHP-фреймворк повинен бути побудований дещо по-іншому, і ось в жовтні 2008 року, після більш 10 місяців закритої розробки, вийшла перша альфа-версія. 3 грудня 2008 був випущений Yii 1.0.

Можливості фреймворку:

- висока продуктивність
- Паттерн Модель-вид-контролер
- інтерфейси DAO та Active Record для роботи з базами даних
- підтримка інтернаціоналізації

- кешування сторінок та окремих фрагментів
- перехоплення та обробка помилок
- введення та валідація веб-форм
- аутентифікація та авторизація
- використання AJAX та інтеграція з jQuery
- генерація базового PHP-коду для CRUD-операцій (скаффолдінг)
- підтримка тем оформлення для їх легкої зміни
- можливість підключення сторонніх бібліотек
- автоматизоване тестування
- підтримка REST
- підтримка баз даних MySQL, PostgreSQL, MSSQL, SQLite , Oracle. [14]

2.5 Інтерфейс інформаційної системи

Важливою частиною проектування ІС є розробка користувачого інтерфейсу. Від нього залежить:

- зручність користування системою;
- її популярність;
- вживаність.

Під час розробки графічного інтерфейсу створюваного ПЗ були використані такі програми:

1. Мови - PHP, JS, AJAX, HTML, MySQL
2. CMS – WordPress, Adminer
3. IDE – Visual Studio Code, Figma, Photoshop

Принципи проектування інтерфейсу користувача. Природний інтерфейс – такий інтерфейс, що не потребує від користувача істотно змінювати звичні для нього способи розв’язання задачі. Це, зокрема, означає, що повідомлення і

результати, що отримуються за допомогою програмного продукту, не повинні вимагати додаткових пояснень. [11]

Використання знайомих користувачу понять і образів (метафор) забезпечує інтуїтивно зрозумілий інтерфейс при виконанні завдань.

1. Узгодженість інтерфейсу означає використання однакових, або дуже схожих метафор, і способів взаємодії з користувачем, і порядку роботи в різних додатках.
2. Узгодженість надає можливість користувачам переносити наявні знання на нові завдання, освоювати нові аспекти швидше, і завдяки цьому зосередити увагу на задачу, що вирішується, а не витратити час на з'ясування відмінностей у використуванні тих або інших елементів управління, команд і т.д.
3. Узгодженість важлива для всіх аспектів інтерфейсу, включаючи імена команд, візуальне представлення інформації і поведінку інтерактивних елементів.
4. Узгодженість в межах робочого середовища. Підтримуючи узгодженість з інтерфейсом, що надається операційною системою (наприклад, ОС Windows), програмний продукт може «спиратися» на ті знання і навички користувача, які він одержав раніше при роботі з іншими ПП.
5. Узгодженість у використуванні метафор. Якщо поведінка деякого програмного об'єкту виходить за рамки того, що звичайно мається на увазі під відповідною йому метафорою, у користувача можуть виникнути труднощі при роботі з таким об'єктом. Наприклад, якщо для програмного об'єкту Корзина визначити операцію Запуск, то для з'ясування її значення користувачу, швидше за все, знадобиться стороння допомога. [11]

Інтерфейс повинен бути простим. При цьому мається на увазі не спрощення, а забезпечення легкості в його вивченні і в користуванні. Крім того, він повинен надавати доступ до всього переліку функціональних можливостей, передбачених програмним продуктом. Реалізація доступу до широких функціональних

можливостей і забезпечення простоти роботи суперечать один одному. Розробка ефективного інтерфейсу покликана збалансувати ці цілі.

Гнучкість інтерфейсу – це його здатність враховувати рівень підготовки і продуктивність праці користувача. Властивість гнучкості припускає можливість зміни структури діалогу та вхідних даних. Концепція гнучкого (адаптивного) інтерфейсу в даний час є однією з основних областей дослідження взаємодії людини і комп'ютера. Основна проблема полягає не в тому, як організувати зміни в діалозі, а в тому, які ознаки потрібно використовувати для визначення необхідності внесення змін і їх суті. [11]

Використання кольорів в графічному інтерфейсі. У всіх інтерактивних системах, незалежно від їх призначення, підтримуються кольорові екрани, тому в призначених для користувача інтерфейсах часто використовуються різні кольори. У деяких системах кольори застосовують в основному для виділення певних елементів (наприклад, в текстових редакторах для виділення фрагментів тексту); у інших системах (таких, як системи автоматичного проектування) кольори позначають різні рівні проектів. [16]

Правильне використання кольорів робить інтерфейс користувача зручнішим для розуміння і управління. Разом з тим використання кольорів може бути неправильним, внаслідок чого створюються інтерфейси, які візуально непривабливі і навіть провокують помилки. Основним принципом розробників інтерфейсів повинне бути обережне використання кольорів на екранах.

Основні принципи ефективного використання кольорів для інтерфейсу користувача:

1. Використовувати обмежену кількість кольорів. Для вікон не слід використовувати більше чотирьох або п'яти різних кольорів, в інтерфейсі системи не повинно бути більше семи кольорів.

2. Використовувати різні кольори для показу змін в стані системи. Якщо на екрані змінилися кольори, значить, відбулася якась подія. Виділення кольором

особливо важливе в складних екранах, в яких відображаються сотні різних об'єктів.

3. Фізіологічні особливості людського ока не дозволяють одночасно сфокусуватися на червоному і жовтому кольорах. Тому послідовність червоних і жовтих зображень викликає зорова напруга. Деякі комбінації кольорів також можуть візуально порушувати читання.

4. Фізіологічні особливості людського ока не дозволяють одночасно сфокусуватися на червоному і жовтому кольорах. Тому послідовність червоних і жовтих зображень викликає зорова напруга. Деякі комбінації кольорів також можуть візуально порушувати читання. [16]

Для даної інформаційної системи була підібрана та використана приємна кольорова гама (рис. 2,3).

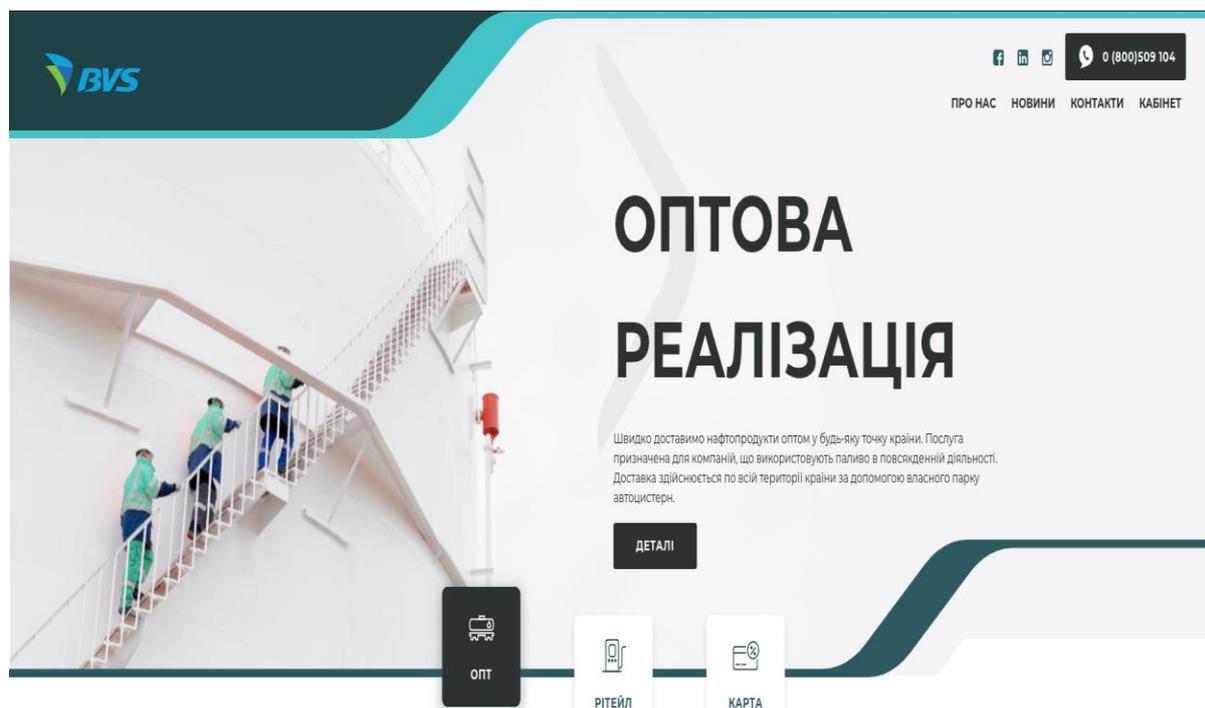


Рисунок 2.3 – Кольорова гама інформаційної системи

Прив'язка значення до певного кольору і використання великої кількості кольорів на екрані є помилкою. Використовувати кольори для представлення значення не слід по двох причинах. Близько 10% людей мають нечітке уявлення про кольори і тому можуть неправильно інтерпретувати значення. [15]

2.6 Створення бази даних

База даних (БД) – упорядкований набір логічно взаємопов'язаних даних, що використовується спільно, та призначений для задоволення інформаційних потреб користувачів. У технічному розумінні включно й система управління БД. [21]

Система управління базами даних (СУБД) – це комплекс програмних і мовних засобів, необхідних для створення баз даних, підтримання їх в актуальному стані та організації пошуку в них необхідної інформації. [21]

Головним завданням БД є гарантоване збереження значних обсягів інформації та надання доступу до неї користувачеві або ж прикладній програмі.

Таким чином БД складається з двох частин: збереженої інформації та системи управління нею. З метою забезпечення ефективності доступу записи даних організовують як множину фактів (елемент даних). [21]

Існує величезна кількість різновидів баз даних, що відрізняються за критеріями (наприклад, в Енциклопедії технологій баз даних визначаються понад 50 видів БД). [21]

Класифікація БД:

1. за моделлю даних:

- ієрархічні,
- мережеві,
- реляційні,
- об'єктні,
- об'єктно-орієнтовані,
- об'єктно-реляційні. [21]

2. За технологією фізичного зберігання:

- БД у вторинній пам'яті (традиційні);
- БД в оперативній пам'яті (in-memory databases);
- БД у третинній пам'яті (tertiary databases). [21]

3. За вмістом:

- географічні.
- історичні.
- наукові.
- мультимедійні. [21]

4. За ступенем розподіленості:

- централізовані (зосереджені);
- розподілені. [21]

Реляційна модель орієнтована на організацію даних у вигляді двовимірних таблиць. Кожна реляційна таблиця являє собою двовимірний масив і має наступні властивості:

- кожен елемент таблиці - один елемент даних;
- всі осередки в стовпчику таблиці однорідні, тобто всі елементи в стовпчику мають однаковий тип (числовий, символний тощо);
- кожен стовпчик має унікальне ім'я;
- однакові рядки в таблиці відсутні;
- порядок проходження рядків і стовпчиків може бути довільним. [21]

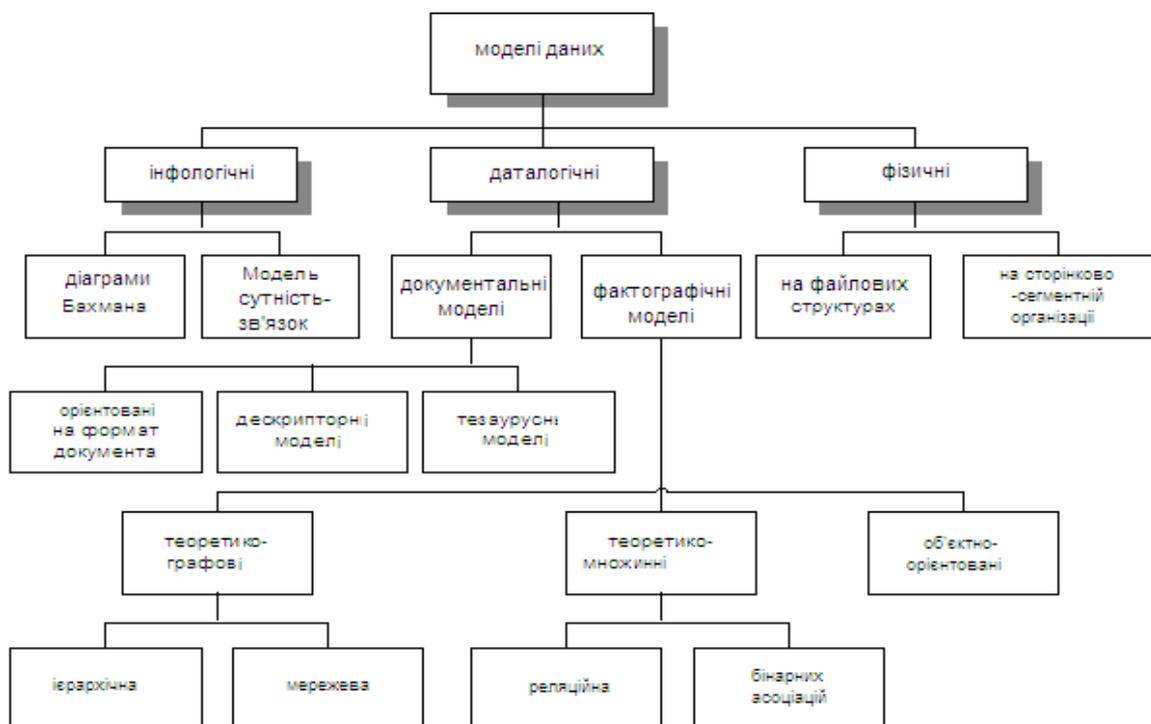


Рисунок 2.4 – Класифікація моделей даних.

2.7 Розробка адміністративної частини сайту

Адміністративна частина повинна включати: інформацію про нових користувачів, розділ налаштувань сайту, можливість адміністрування кафедри, можливість створення нових статтів, розділів та ключових слів для простішого пошуку новин. [21]

Адміністративна частина сайту представлена у вигляді системи управління вмістом.

Система управління вмістом (англ. Content management system, CMS) – комп'ютерна програма, використовувана для управління вмістом чого-небудь (звичайний цей вміст розглядається як неструктуровані дані наочного завдання в протилежність структурованим даним, СУБД, що зазвичай знаходяться під управлінням). Зазвичай такі системи використовуються для зберігання і публікації великої кількості документів, зображень, музики або відео.

Окремим випадком такого роду систем є системи управління сайтами. Подібні CMS дозволяє управляти текстовим і графічним наповненням веб-сайту, надаючи користувачеві зручні інструменти зберігання і публікації інформації.

Зараз існує безліч готових систем управління вмістом сайту, у тому числі і безкоштовних. Їх можна розділити на три типи, за способом роботи:

Генерація сторінок за запитом. Системи такого типу працюють на основі зв'язки «Модуль редагування > База даних > Модуль уявлення». Модуль уявлення генерує сторінку із вмістом при запиті на нього, на основі інформації з бази даних. Інформація в базі даних змінюється за допомогою модуля редагування. Сторінки наново створюються сервером при кожному запиті, а це створює навантаження на системні ресурси. Навантаження може бути багато разів понижена при використанні засобів кешування, які є в сучасних веб-серверах. [21]

Генерація сторінок при редагуванні. Системи цього типу програми для редагування сторінок, які при внесенні змін до змісту сайту створюють набір статичних сторінок. При такому способі жертвується інтерактивність між

відвідувачем і вмістом сайту. [21]

Змішаний тип. Як зрозуміло з назви, поєднує в собі переваги перших два. Може бути реалізований шляхом кешування – модуль уявлення генерує сторінку один раз, надалі вона в рази швидше підвантажується з кеша. Кеш може оновлюватися як автоматично, по закінченню деякого терміну часу або при внесенні змін до певних розділів сайту, так і вручну по команді адміністратора. Інший підхід – збереження певних інформаційних блоків на етапі редагування сайту і збірка сторінки з цих блоків при запиті відповідної сторінки користувачем.

2.8 Засоби розробки

2.9.1 Мова розмітки гіпертекстових документів HTML. Обмін інформацією в Інтернет здійснюється за допомогою протоколів прикладного рівня, що реалізують той або інший прикладний сервіс (пересилку файлів, гіпертекстової інформації, пошта і так далі). Одним з найбільш молодих і популярних сервісів Інтернет, розвиток якого і привело до сплеску популярності самої Інтернет, стала World Wide Web (WWW), заснована на протоколі HTTP (Hyper Text Transfer Protocol – протокол передачі гіпертекстовій інформації). Гіпертекстові документи, представлені в WWW, мають одну принципову відмінність від традиційних гіпертекстових документів - зв'язки, в них що використовуються, не обмежені одним документом, і більш того, не обмежені одним комп'ютером. Для підготовки гіпертекстових документів використовується мова HTML (Hyper Text Markup Language – мова розмітки гіпертекстових документів), що надає широкі можливості по форматуванню і структурній розмітці документів, організації зв'язків між різними документами, засоби включення графічної і мультимедійної інформації. HTML-документи є видимими за допомогою спеціальної програми - браузера. Найбільшого поширення в даний час набули браузери Mozilla Firefox і Internet Explorer компанії Microsoft (MSIE). Реалізації Mozilla Firefox доступні практично для всіх сучасних програмних і апаратних платформ, реалізації MSIE доступні для всіх Windows платформ,

Macintosh і деяких комерційних Unix-систем. [6]

Мова розмітки гіпертекстових документів DHTML. Динамічний HTML (Dynamic HTML, DHTML) не є якоюсь особливою мовою розмітки сторінок. Це всього лише термін, вживаний для позначень HTML-сторінок з динамічно змінним вмістом. [12]

Об'єктна модель документа робить всі елементи сторінки програмованими об'єктами. З її допомогою через мови сценаріїв можна дістати доступ і управляти всім, що є в документі. Кожен елемент HTML доступний як індивідуальний об'єкт, а це означає, що можна змінювати значення будь-якого параметра будь-якого тега HTML-сторінки, і, як наслідок, документ дійсно стає динамічним.

Будь-яка дія користувача (натискання кнопкою миші, переміщення миші у вікні браузера або натиснення клавіші клавіатури) об'єктною моделлю документа трактується як подія, яка може бути перехоплена і оброблена процедурою сценарію. [6]

Розширювана мова розмітки XML. На сьогоднішній день вже всім фахівцям в області веб-технологій стало очевидне, що існуючих стандартів передачі даних по Internet недостатньо. Формат HTML, ставши свого часу проривом в області відображення вмісту вузлів Internet, вже не задовольняє всім необхідним на даний момент вимогам. Він дозволяє описати те, яким чином повинні бути відображені дані на екрані кінцевого користувача, але не надає ніяких засобів для ефективного опису передаваних даних і управління ними. [12]

Крім того, каменем спотикання для багатьох компаній, що займаються розробкою програмного забезпечення, є необхідність сумісного використання різних компонентів, забезпечення їх взаємодії, можливості обміну даними між ними.

До недавнього часу не існувало стандарту, що надає засоби для інтелектуального пошуку інформації, обміну даними, адаптивної обробки отримуваних даних. [12]

Мова програмування PHP. PHP: Hypertext Preprocessor – скриптова мова програмування, створена для генерації HTML-сторінок на веб-сервері і роботи з базами даних. В даний час підтримується переважною більшістю провайдерів хостингу. Входить в LAMP – «стандартний» набір для створення веб-сайтів (Linux, Apache, MySQL, PHP (Python або Perl)). [12]

Може використовуватися для вирішення адміністративних завдань в операційних системах UNIX, GNU/Linux, Microsoft Windows, Mac OS X і AmigaOS. Проте в такій якості він не набув поширення, віддаючи пальму першості Perl, Python і VBScript.

Синтаксис PHP подібний синтаксису мови Сі. Деякі елементи, такі як асоціативні масиви і цикл `foreach`, запозичені з Perl.

Назва PHP – рекурсивна аббревіатура, що означає «PHP: Hypertext Preprocessor» (раніше акронім розшифровувався як «Personal Home Page Tools»). Спочатку PHP створювався як надбудова над Perl для полегшення розробки веб-сторінок. [12]

Локальний веб-сервер. Локальний сервер – це можливість перевірити роботу веб-проекту перед завантаженням його на «реальний» сервер.

Сучасний сайт являє собою не просто набір HTML-документів, але і включає в себе безліч технологій, в тому числі серверних, таких як: SSI (Server-Side Includes, включення на стороні сервера), PHP (PHP: Hypertext Preprocessor, PHP: препроцесор гіпертексту), бази даних та багато іншого. Для ознайомлення та вивчення цих технологій помилкою буде використовувати відвідуваний робочий сайт в Інтернеті, тому має сенс встановити необхідний комплект програм на локальний комп'ютер і тестувати всі на ньому. [24]

Найбільш популярною зв'язкою таких програм є веб-сервер Apache, мова програмування PHP, система управління базами даних MySQL, оболонка для адміністрування баз даних PhpMyAdmin, шаблонизатор Smarty. [15]

Зазначені програми працюють переважно під управлінням Linux подібних систем, але є і версії під Windows. Основний плюс цієї зв'язки програм в універсальності. Сайт, створений і перевірений на локальному комп'ютері під

Windows, легко може бути перенесений на сервер з FreeBSD. До того ж цей набір підтримує переважна більшість хостерів.

ХАМРР – найпопулярніше середовище розробки PHP. ХАМРР – це збірка веб-сервера, що містить Apache, MySQL, інтерпретатор скриптів PHP, мову програмування Perl та велику кількість додаткових бібліотек. Локальний сервер, який напханий найсучаснішими дистрибутивами. [15]

Локальний веб-сервер Apache. Web-сервер Apache є самостійним, некомерційним, вільно розповсюджуваним продуктом. Продукт підтримує безліч можливостей, багато з яких реалізовані як скомпільовані модулі, які розширюють основні функціональні можливості. Вони різняться від серверної підтримки мов програмування до схем аутентифікації. [14,24]

Найпоширеніший веб-сервер в світі – це Apache. За даними компанії Netcraft, загальне число веб-узлів, що працюють під його управлінням, до кінця 1998 р. досягло 2 млн. (55% загального числа вузлів) і постійно росте. Для порівняння: на долю серверів Microsoft доводиться 25%, Netscape – 7%. Будучи безкоштовною відкритою програмою, призначеною для безкоштовних же Unix-систем (FREEBSD, Linux і ін.), Apache по функціональних можливостях і надійності не поступається комерційним серверам, а широкі можливості конфігурації дозволяють побудувати його для роботи практично з будь-якою конкретною системою. Існують локалізації сервера для різних мов, у тому числі і для кирилиці. [24]

Історично склалося так, що кириличні тексти в Internet можуть бути представлені в різних кодуваннях, з яких найбільш поширені koі8-r (або просто koі8) і Windows-1251: з першою працюють більшість серверів і робочих станцій під управлінням Unix, друга є стандартною для всіх версій Windows. Оскільки кодування Windows-1251, природно, застосовується на переважній більшості клієнтських машин, частка тих, хто подорожує по російській частині WWW, використовуючи koі8, не перевищує зараз 5%.

Діаграми варіантів використання. Концептуальна модель виражається у вигляді діаграм варіантів використання (Use-case diagram). Цей тип діаграм

служить для проведення ітераційного циклу загальної постановки завдання разом із замовником. Варіант використання являє собою послідовність дій, виконуваних системою у відповідь на подію, ініційоване деяким зовнішнім об'єктом (діючою особою). Варіант використання описує типову взаємодію між користувачем і системою. [19]

На (рис. 2.5) зображені діаграми варіантів використання, простіше сказати, повноваження користувача та адміністратора.

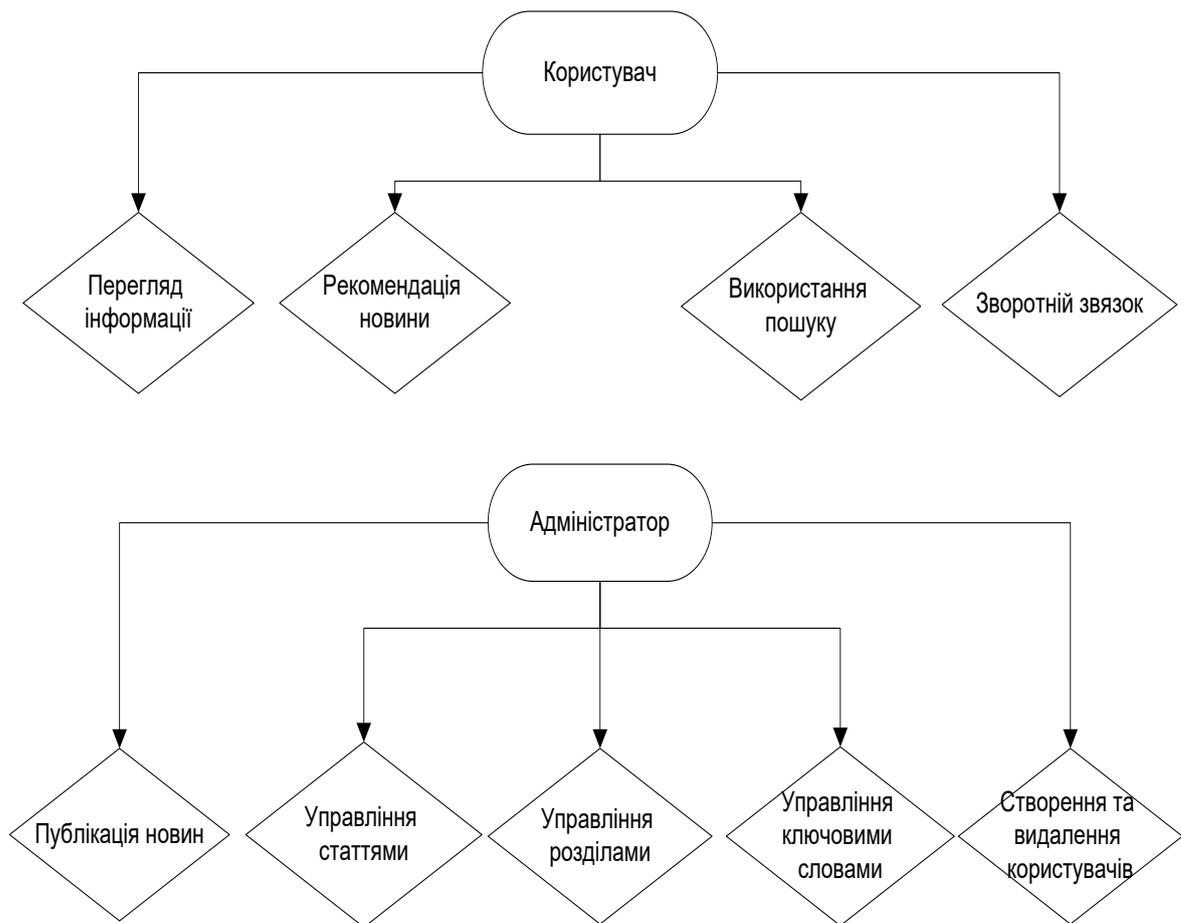


Рисунок 2.5 – Діаграми варіантів використання інформаційної системи

Такі види діаграм є основою для досягнення взаєморозуміння між програмістами-професіоналами, що розробляють проект, і замовниками проекту. [19]

2.10 ER-модель бази даних

ER-модель зручна при проектуванні інформаційних систем, баз даних, архітектури комп'ютерних та інших систем. За допомогою такої моделі виділяють найсуттєвіші елементи моделі і встановлюють зв'язки між ними. [21]

Модель «сутність-зв'язок» – модель даних, яка дозволяє описувати концептуальні схеми за допомогою узагальнених конструкцій-блоків. ER-модель – це мета-модель даних, тобто засіб опису моделей даних. [21]

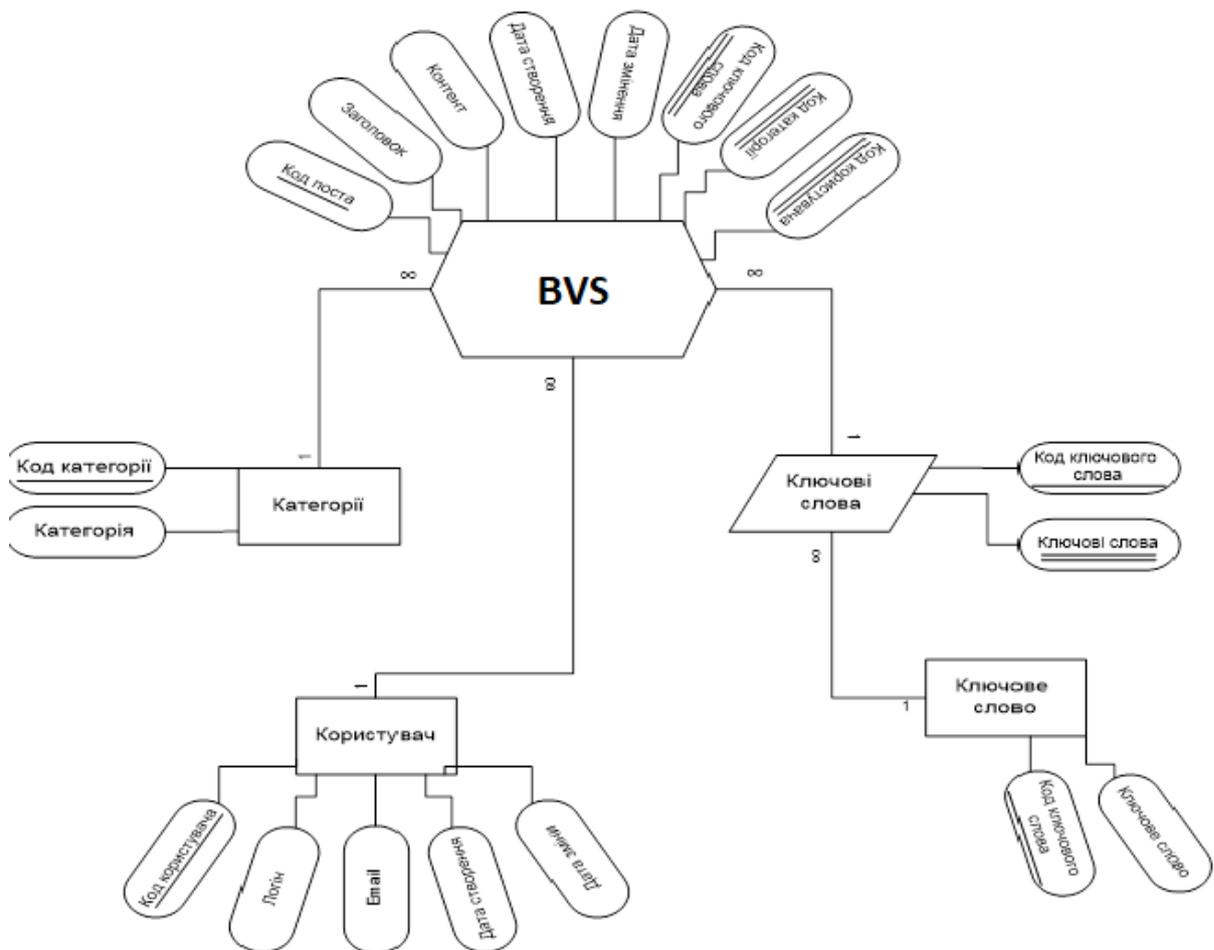


Рисунок 2.6 — Інфологічна модель

Пошук інформації

Запит - це формалізований спосіб вираження інформаційних потреб користувачем системи. Для вираження інформаційної потреби використовується мова пошукових запитів, синтаксис варіюється від системи до системи. Крім

спеціальної мови запитів, сучасні пошукові системи дозволяють вводити запит природною мовою.

Об'єкт запиту - це інформаційна сутність, яка зберігається в основі автоматизованої системи пошуку. Незважаючи на те, що найпоширенішим об'єктом запиту є текстовий документ, немає жодних принципових обмежень. Зокрема, можливий пошук зображень, музики та іншої мультимедіа інформації. Процес занесення об'єктів пошуку до ІС називається індексацією.[8]

Трійкове дерево пошуку — це тип префіксного дерева, де вузли розташовані таким чином, як в двійковому дереві пошуку, але мають щонайбільше троє дітей, замість двох (як у двійковому дереві пошуку). Як і інші префіксні дерева, трійкове дерево пошуку може бути використано як асоціативний масив з можливістю поступового пошуку рядків. Зазвичай трійкові дерева пошуку мають кращу просторову складність у порівнянні зі звичайними префіксними деревами, нехтуючи часовою складністю. Трійкові дерева пошуку переважно застосовують для написання алгоритмів перевірки орфографії та автодоповнення. [21]

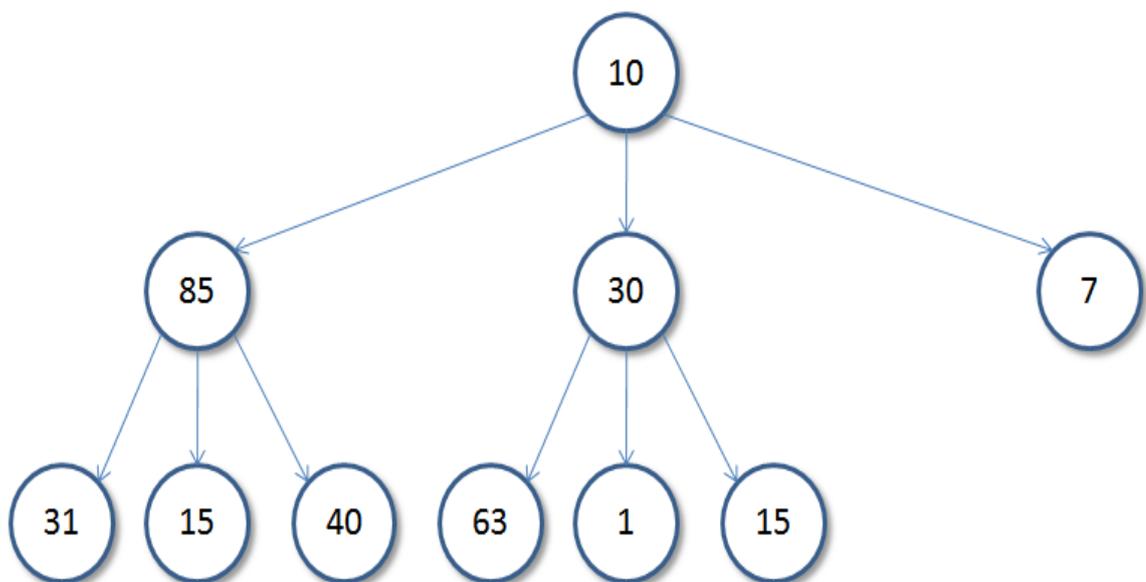


Рисунок 2.6 — Трійкове дерево пошуку

РОЗДІЛ 3 ПРАКТИЧНА ЧАСТИНА

3.1 Розробка інформаційної системи

Етапи проектування ІС

1. Створення первинного технічного завдання на розробку ІС.
2. Розробка структурної схеми інформаційної системи.
3. Розробка веб-дизайну (графічних елементів ІС, стилів і елементів навігації).
4. Створення програмного коду і інших елементів інформаційної системи.
5. Тестування і розміщення в мережі Інтернет. [11]

На другому етапі створюється структура інформаційної системи

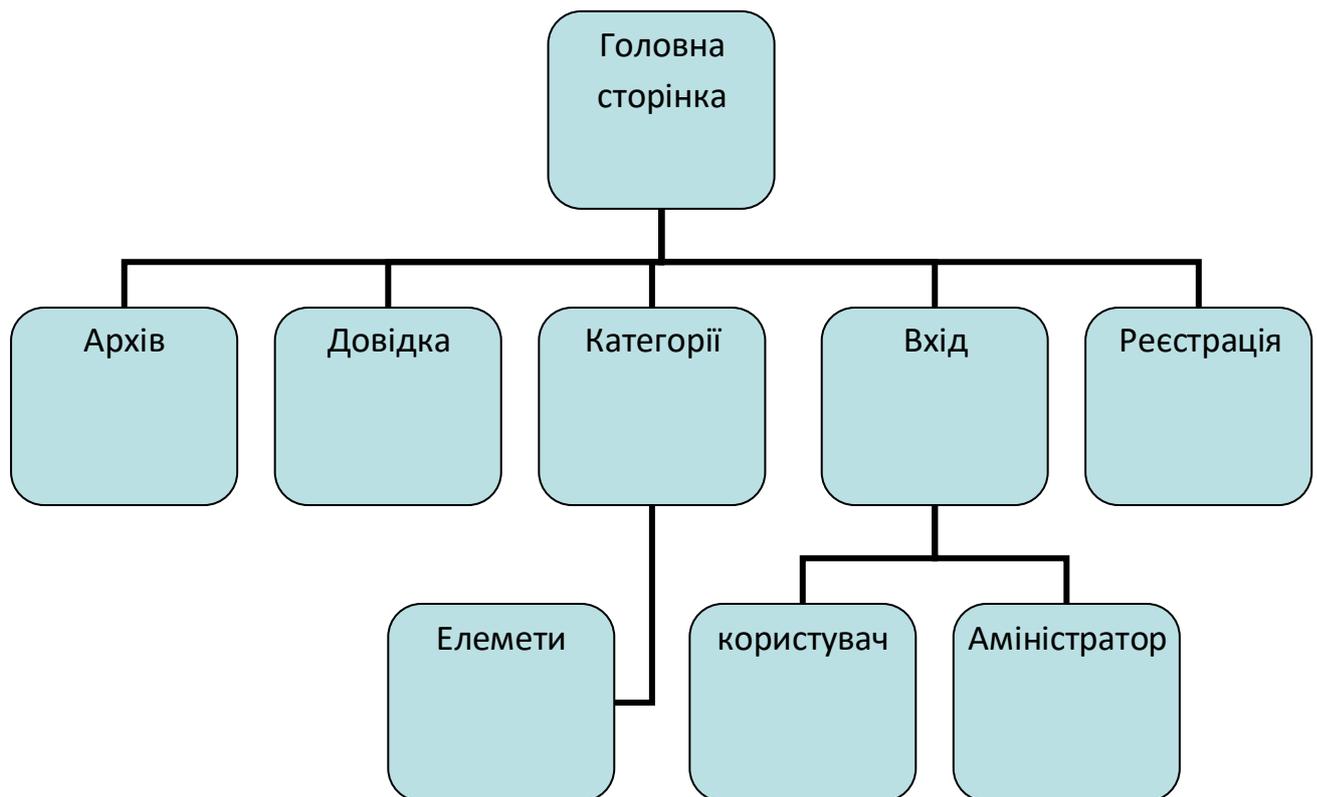


Рисунок 3.1 – Запропонована структура ІС

Потім визначатись, які сторінки потрібні на сайті, їхні назви та опис, про що буде написано чи зображено на цій сторінці. В цьому етапі потрібно чітко визначитися які задачі повинен виконувати сайт.

Технічне проектування ІС:

1-й етап – технічне проектування де ведеться пошук найбільш раціональних проектних рішень з усіх аспектів розробки, створюються і описуються всі компоненти системи;

2-й етап – проектування частини призначеної для звичайних користувачів, в процесі якої здійснюється розробка і доведення ІС, створення дизайну інформаційної системи, Рис. 3,2.

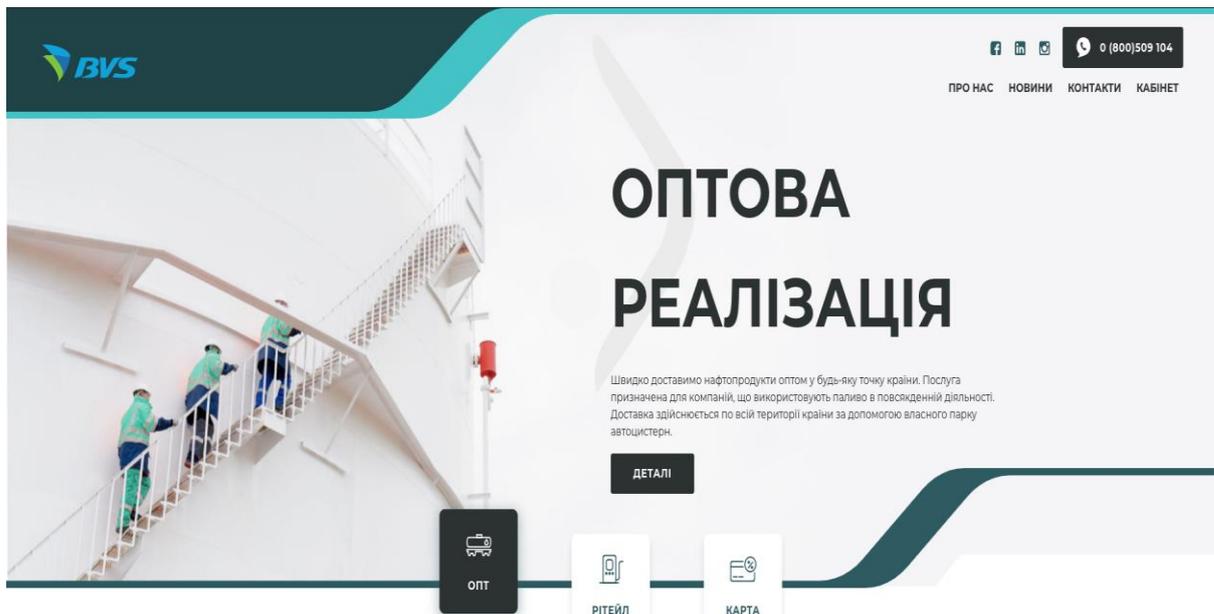


Рисунок 3.2– Дизайн інформаційної системи

Для розробки інформаційної системи використовуватимуться:

- PHP
- MySQL
- HTML
- CSS
- Javascript

Дизайн сайту. Web-дизайн ставить перед собою різні цілі:

- формування у користувача позитивного сприйняття образу об'єкта рекламного продукту

- простоту і чіткість структури сайту, інтуїтивно зрозумілого для користувача інтерфейсу
- зручність навігаційної системи.

Проектування та створення бази даних, та її структури. База даних являє собою певним чином структуровану сукупність даних, що зберігаються разом і обробляються відповідно до деякими правилами. [20]

Структура бази даних на рис. 3,4.

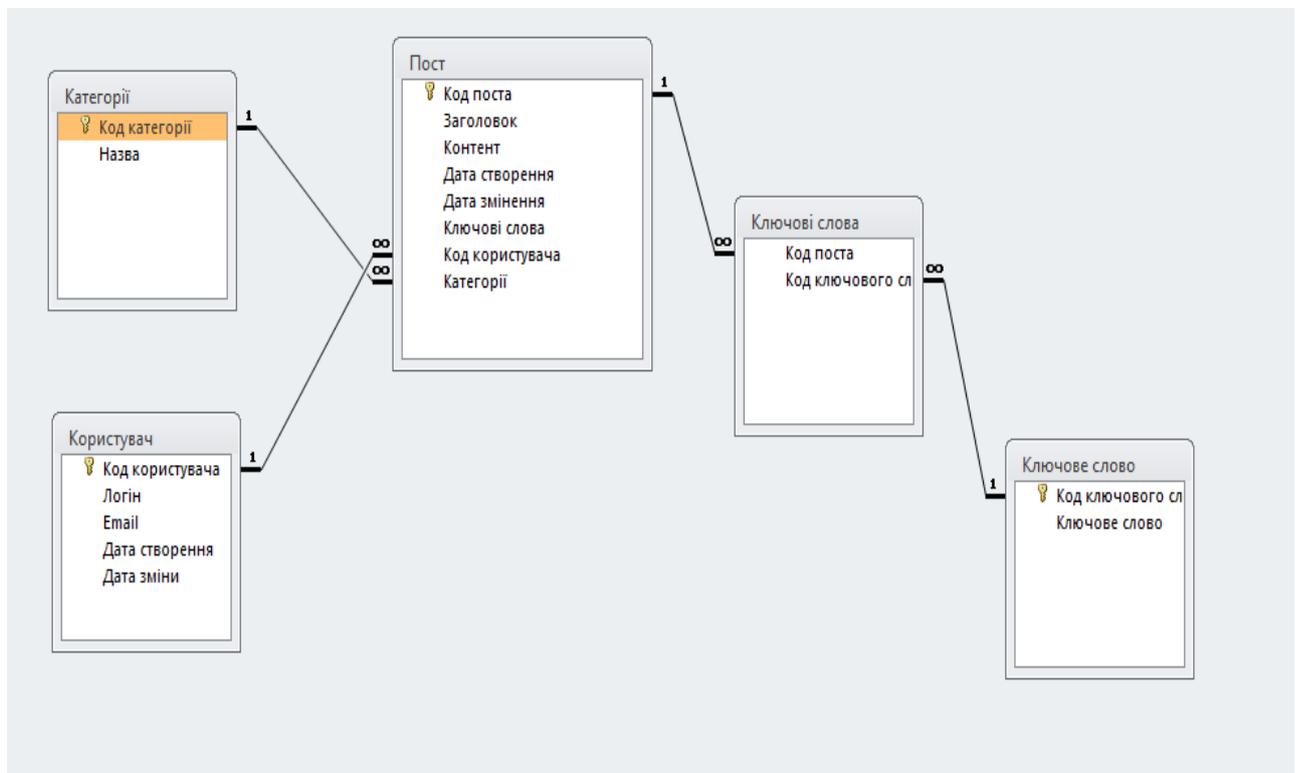


Рисунок 3.4. – Схема зв'язків в БД

Вигляд сторінки адмін-панелі нашої інформаційної системи зображена на, рис. 3,5.

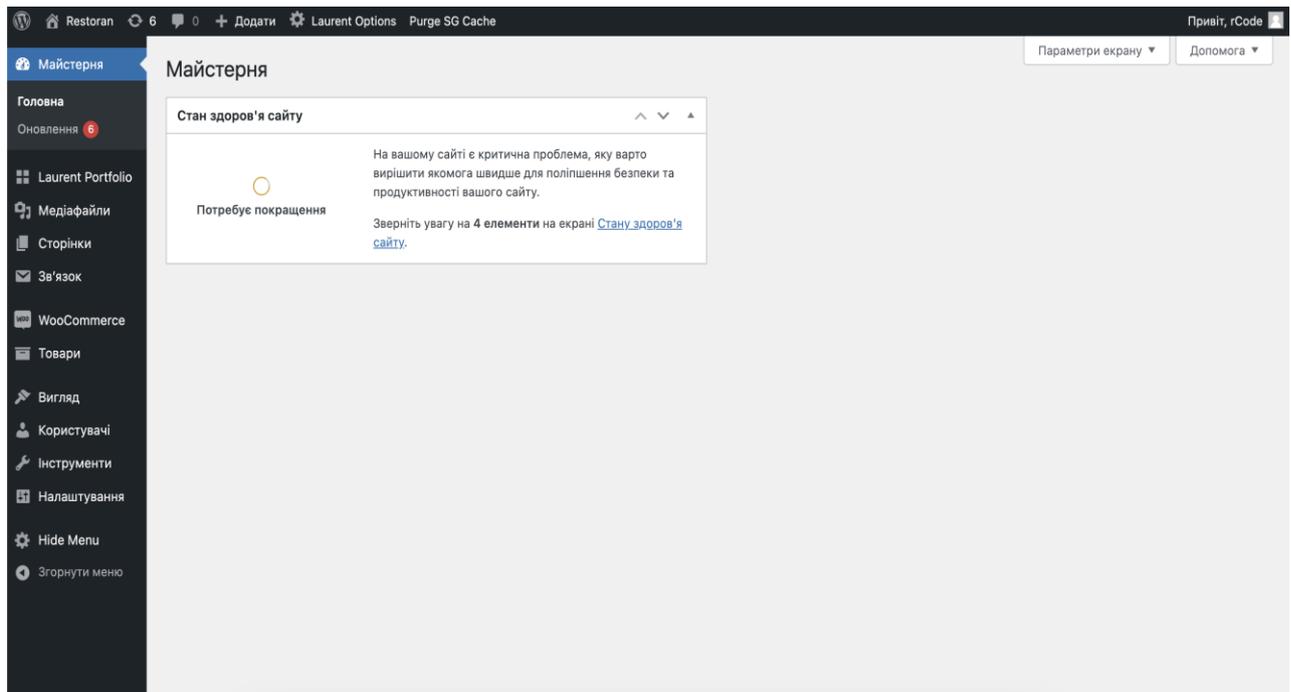


Рисунок 3.5 – Дизайн панелі адміністратора

Адмін-панель потрібна для того, щоб простіше було, в нашому випадку, редагувати, видаляти та додавати сторінки, розділи, користувачів, ключові слова.

Тестування інформаційної системи на локальному сервері.

Хостинг – послуга, що включає надання дискового простору, підключення до мережі та інших ресурсів для розміщення фізичної інформації на сервері, що постійно перебуває в мережі.

Для цього необхідно:

- підготовка до впровадження – установка і введення в експлуатацію технічних засобів, завантаження баз даних і дослідна експлуатація;
- проведення дослідних випробувань всіх компонентів системи перед встановленням ІС на хостинг.

3.2 Захист інформації

Захист інформації – це сукупність організаційно-технічних заходів і правових норм для попередження заподіяння збитку інтересам власника інформації. Тривалий час методи захисту інформації розроблялися тільки

державними органами, а їхнє впровадження розглядалося як виключне право тієї або іншої держави. Проте в останні роки з розвитком комерційної і підприємницької діяльності збільшилося число спроб несанкціонованого доступу до конфіденційної інформації, а проблеми захисту інформації виявилися в центрі уваги багатьох вчених і спеціалістів із різноманітних країн. [4]

Облікові записи. Обліковий запис – це запис, що містить відомості, необхідні для ідентифікації користувача при підключенні до системи, а також інформацію для авторизації і обліку. Це ім'я користувача та пароль (або інше аналогічне засіб аутентифікації – наприклад, біометричні характеристики).

Пароль або його аналог, як правило, зберігається в зашифрованому або хешірованому вигляді (з метою його безпеки). [5]

Для підвищення надійності можуть бути, поряд з паролем, передбачені альтернативні засоби аутентифікації – наприклад, спеціальний секретне питання (або декілька питань) такого змісту, що відповідь може бути відомий тільки користувачеві. Такі питання і відповіді також зберігаються в облікового запису.

Конкретні категорії даних, які можуть бути внесені в таку анкету, визначаються адміністраторами системи.

Обліковий запис може також містити одну або декілька фотографій або аватар користувача.

Обліковий запис користувача також може враховувати різні статистичні характеристики поведінки користувача в системі: давність останнього входу в систему, тривалість останнього перебування в системі, адреса використаного при підключенні комп'ютера, інтенсивність використання системи, сумарне і (або) питома кількість певних операцій, вироблених в системі, і так далі.

Ідентифікація суб'єкта доступу полягає в тім, що суб'єкт повідомляє операційній системі ідентифікуючу інформацію про себе (ім'я, обліковий номер і т.д.) і в такий спосіб ідентифікує себе. [5]

Аутентифікація суб'єкта доступу полягає в тім, що суб'єкт надає операційній системі крім ідентифікуючої інформації ще й аутентифікаційну

інформацію, що підтверджує, що він дійсно є тим суб'єктом доступу, до якого відноситься ідентифікуюча інформація. Нехай, наприклад, користувач, входячи в систему, ввів ім'я і пароль.

Авторизація – керування рівнями та засобами доступу до певного захищеного ресурсу, як в фізичному розумінні, так і в галузі цифрових технологій (наприклад, автоматизована система контролю доступу) та ресурсів системи залежно від ідентифікатора і пароля користувача або надання певних повноважень (особі, програмі) на виконання деяких дій у системі обробки даних. [5]

3.3 Демонстрація роботи адміністратора

Для даної роботи було розроблено асоціативну пошукову систему, яка працює за принципом асоціацій, якщо при введенні слова кафедра в рядок пошуку.

The screenshot displays an administrative interface for managing records. At the top, there are buttons for 'Додати новий запис' (Add new record) and 'Параметри екрану' (Screen parameters). Below this, a search bar is present with a 'Шукати записи' (Search records) button. The main area shows a list of records with the following columns: 'Заголовок' (Title), 'Автор' (Author), 'Категорії' (Categories), 'Позначки' (Notes), and 'Дата' (Date). Two records are listed:

Заголовок	Автор	Категорії	Позначки	Дата
Оновлена АЗС	L33T	Uncategorized	-	Опубліковано 03.11.2024 о 11:19
Акція вихідного дня на АЗС BVS!	L33T	Uncategorized	-	Опубліковано 03.11.2024 о 11:15

At the bottom, there are filter buttons for 'Групові дії' (Group actions) and 'Застосувати' (Apply), and a '2 елементи' (2 items) indicator.

Рисунок 3.6 – Адмін панель ІС

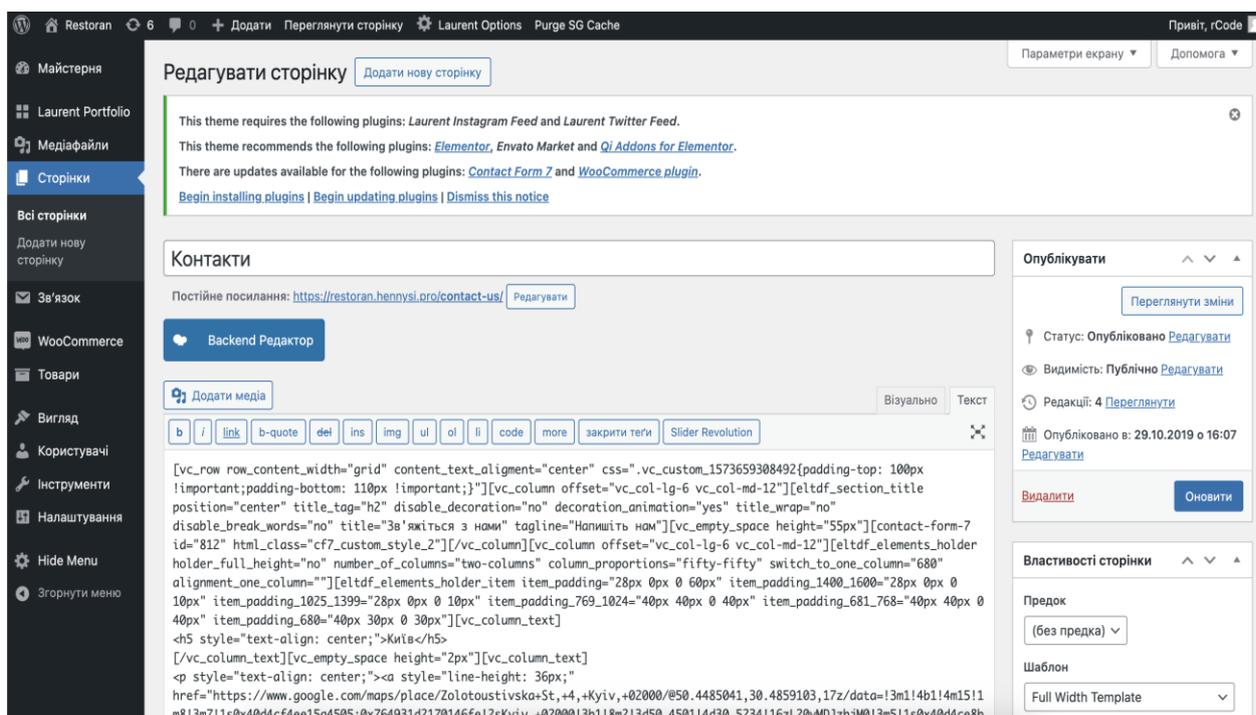


Рисунок 3.7 – Редагування існуючої сторінки

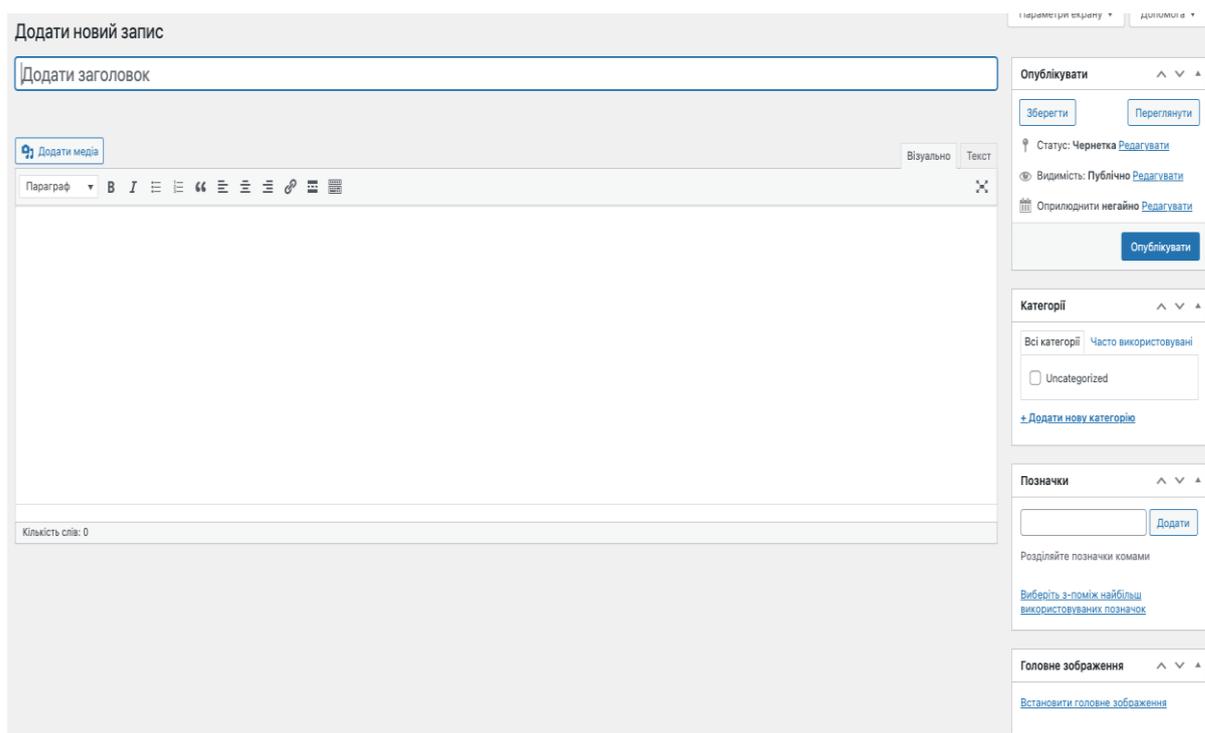


Рисунок 3.8 – Додавання та редагування новин

Користувачі [Додати нового користувача](#) Параметри екрану ▾ | Допомога ▾

Всі (5) | Адміністратор (1) | Учасник (4) [Пошук користувачів](#)

Групові дії ▾ [Застосувати](#) [Змінити роль на...](#) ▾ [Змінити](#) 5 елементів

<input type="checkbox"/>	Ім'я користувача ↕	Ім'я	Email ↕	Роль	Записи
<input type="checkbox"/>	L33T	–	–	Адміністратор	2
<input type="checkbox"/>	user_0502146157	–	0502146157@example.com	Учасник	0
<input type="checkbox"/>	user_0664667796	–	0664667796@example.com	Учасник	0
<input type="checkbox"/>	user_124124	–	124124@example.com	Учасник	0
<input type="checkbox"/>	user_25121122512	–	25121122512@example.com	Учасник	0

Групові дії ▾ [Застосувати](#) [Змінити роль на...](#) ▾ [Змінити](#) 5 елементів

Рисунок 3.9 – Додавання та редагування клієнтів

3.4 Демонстрація сторінок інформаційної системи

На головній сторінці ІС знаходиться меню які мають в собі підменю, Головна сторінка має вигляд: Рисунок 3.10

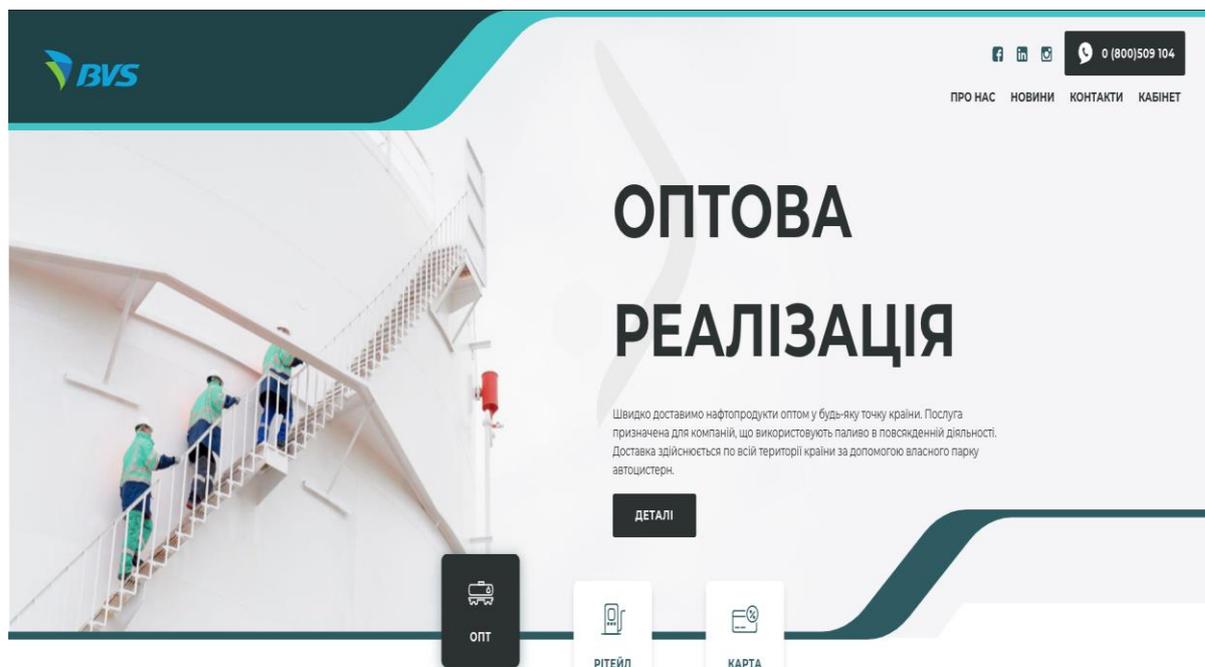


Рисунок 3.10 – Головна сторінка



Рисунок 3.11 – Підказки ІС закладу.

Головна > Бізнес-каталог > Інформація

БВС РИТЕЙЛ, ТОВ



Показати на карті

Відгуки про компанію «БВС РИТЕЙЛ, ТОВ». Поділіться і Ви своїми думками у коментарях!

Всі відгуки

Інформація Карта Відгуки

Контакти

Юридична адреса
36009, м. Полтава, вул. Зіньківська, 196

Телефони
[\(0532\) 629913](tel:0532629913)

E-mail
yurist@oil.pl.ua

Керівник, директор
Крачко Олександр Георгійович

Види діяльності КВЕД

- ✓ Роздрібна торгівля паливом
- ✓ Технічне обслуговування та ремонт автотранспортних засобів
- ✓ Компанії які вирощують зернові культури
- ✓ Виробництво продуктів нафтопереробки

Замовляйте пакет «Україна. VIP»
який збільшив дохід наших клієнтів у середньому на **50-100%** після старту.

Про компанію

Код ЄДРПОУ
44098532

Дата реєстрації
15.01.2021

Дата оновлення
09.01.2024

Виявили неточність або помилку?
[Відповісти на запитання](#)

Рисунок 3.12– Сторінка з контактами

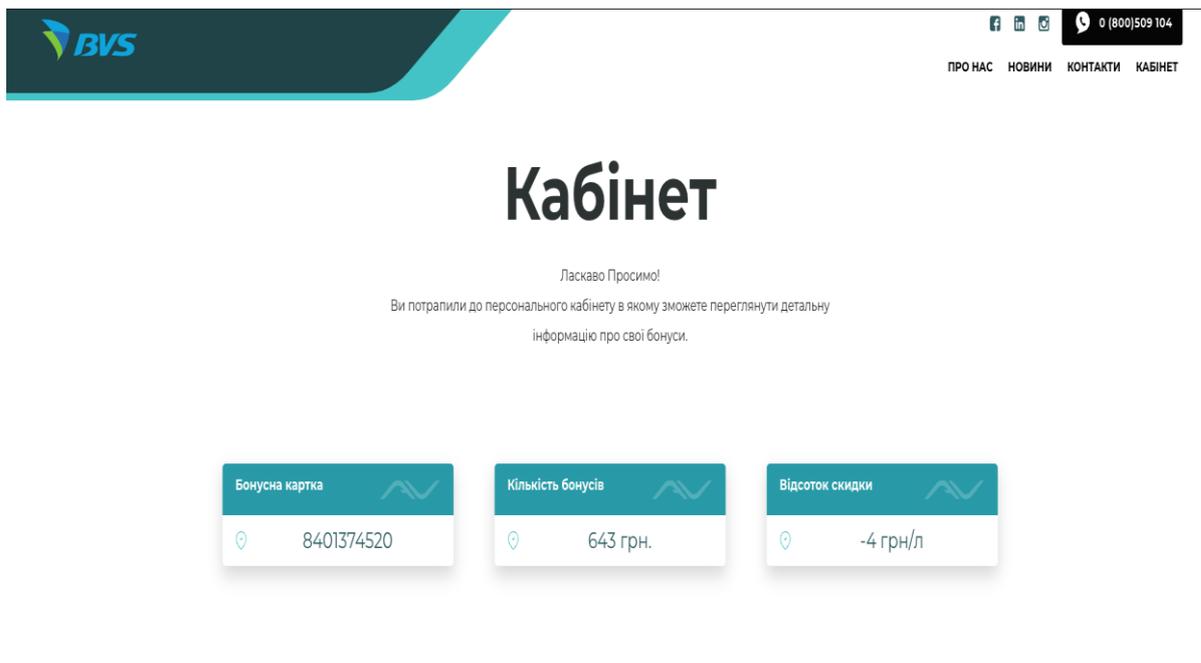


Рисунок 3.13 – Кабінет користувача

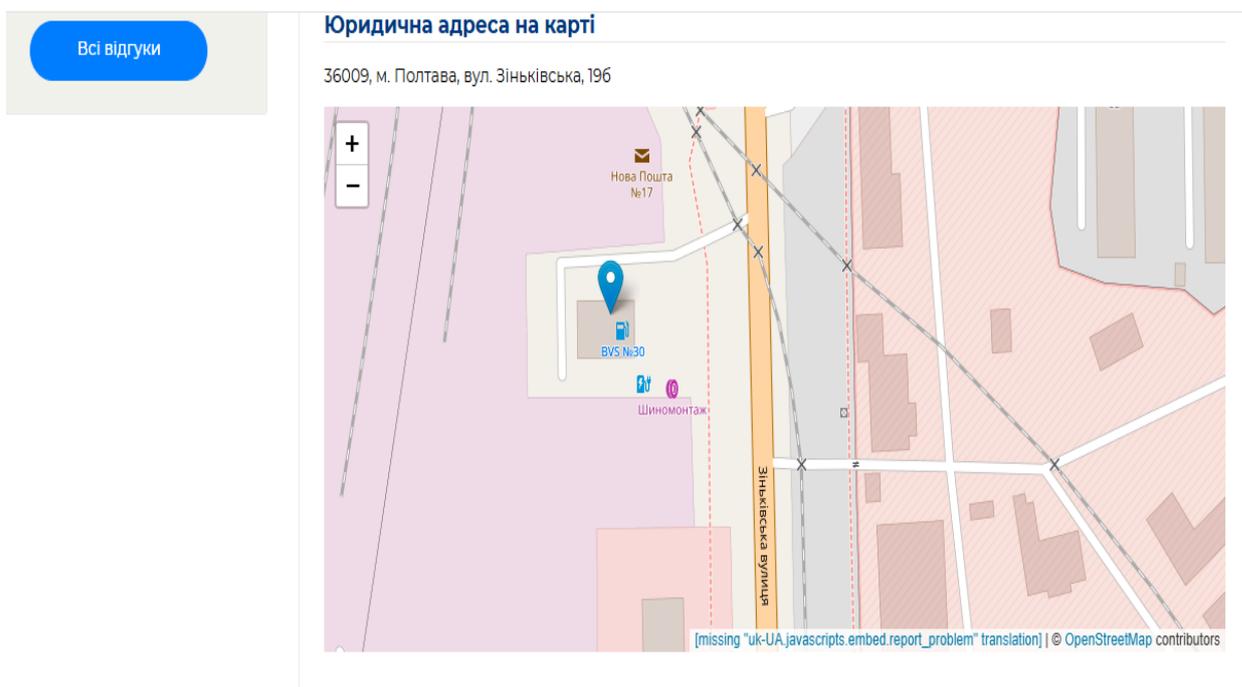


Рисунок 3.14– Карта

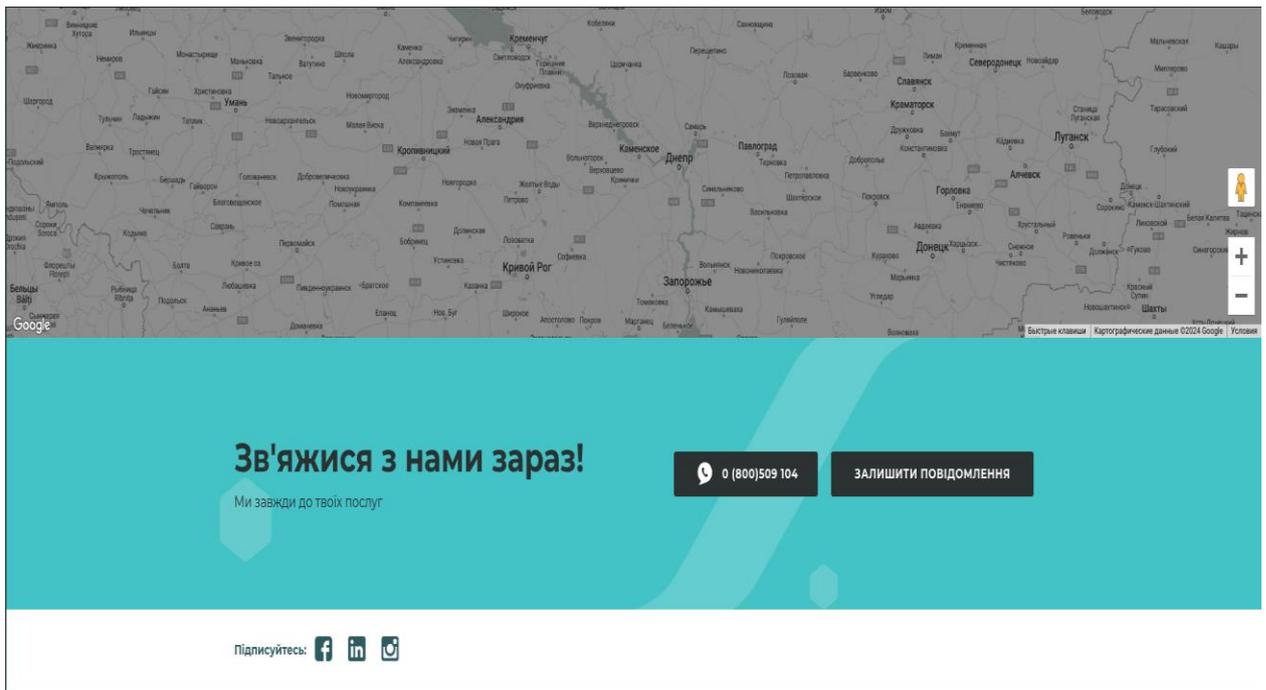


Рисунок 3.14 – Зворотній зв'язок

```

---function laurent_elated_custom_breadcrumbs() {
-----global $post;
-----
-----$pageid = laurent_elated_get_page_id();
-----$homeLink = esc_url( home_url( '/' ) );
-----$home = esc_html__( 'Головна', 'laurent' );
-----$delimiter = '<span class="eltdf-delimiter">'.
-----'<svg xmlns="http://www.w3.org/2000/svg" viewBox="0 0 8.3 8.5"
class="eltdf-breadcrumb-arrow">'.
-----'<polyline points="0.4 0.4 3.6 4.2 0.4 8.1" -/>'.
-----'<polyline points="4.5 0.4 7.7 4.2 4.5 8.1" -/>'.
-----'</svg></span>';
-----$before = '<span class="eltdf-current">';
-----$after = '</span>';

```

Рисунок 3.15 – Приклад коду

РОЗДІЛ 4 ТЕСТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ

4.1 Тестування верстки

Тестування Верстки – при перевірці верстки першим ділом тестувальник перевіряє розташування елементів, відповідність їх позицій наданими макетів, а так само перевіряє оптимізацію зображень і графіки. Далі здійснюється перевірка валідності коду. В процесі верстки важливо дотримувати правильну ієрархію об'єктів, і важливо упевнитися в її валідності за фактом завершення робіт. Браузери, незважаючи на явно неправильний код, в будь-якому випадку спробують відобразити веб-сторінку. Але оскільки не існує єдиного регламенту про те, як же повинен бути показаний «кривий» документ, кожен браузер намагається зробити це по-своєму. А це в свою чергу призводить до того, що один і той же документ може виглядати по-різному в різних браузерах. виправлення явних промахів і систематизація коду призводить, як правило, до стабільного результату. Завершивши перевірку на валідність, спеціаліст приступає до перевірки на кросбраузерність, тобто перевіряє працездатність сайту в різних браузерах, а так само при різних параметрах налаштування екрану. [20]

Навіщо перевіряти сайт на кросбраузерність? На сьогоднішній день існує ряд найбільш популярних веб-браузерів, таких як Google Chrome, Safari, Mozilla Firefox, Internet Explorer і Opera. Кожен з них дотримується загальних рекомендацій візуалізації розмітки сторінки, проте в той же час кожен обробляє код згідно з особливості власного движка. Ускладнюється все тим, що досить часто з'являються нові версії браузерів, і ресурс, який відмінно виглядає, наприклад, в ІЕ9, не обов'язково буде виглядати коректно в ІЕ7 або ІЕ8. Тому в процесі тестування враховується перелік браузерів, підтримка яких обумовлювалася з замовником на ранніх етапах обговорення проекту. Етап перевірки сайту на кросбраузерність при різних дозволах досить довгий час, але

результат того вартий – з вашим сайтом зможе ознайомитися будь-представник цільової аудиторії. [20]

Для мого тестування було вибрано такі браузери: Google chrome, Internet explorer, Mozilla firefox, Opera.

Такий вигляд має чек-лист для тестування вестки та кросбраузерності, та були виконані наступні перевірки. Таблиця 4.1

Таблиця 4.1 Лист перевірки для тесту вестки

	IE 11	Firefox	Gchrome	Opera
Наявність логотипу	Passed	Passed	Passed	Passed
Наявність футера	Passed	Passed	Passed	Passed
Наявність головного меню	Passed	Passed	Passed	Passed
Віображення шрифту тексту	Passed	Passed	Passed	Passed
Відображення кольорової гами	Passed	Passed	Passed	Passed
Коректне віображення кнопок, блоків меню і т.д.	Passed	Passed	Passed	Passed
Перевірка всіх пунктів користувачами різних виів	Passed	Passed	Passed	Passed
Коректність скрола	Passed	Passed	Passed	Passed
Коректність розміщення банерів	Passed	Passed	Passed	Passed
Наявність Favicon	Passed	Passed	Passed	Passed
Вісутність невірних посилань	Passed	Passed	Passed	Passed
Логотип є посиланням на головну сторінку	Passed	Passed	Passed	Passed
Коректне вірівнювання блоків	Passed	Passed	Passed	Passed
При наведенні, кнопки	Passed	Passed	Passed	Passed

змінюють вигляд				
Поля в формах мають підписи	Passed	Passed	Passed	Passed
Коректне змінювання текстових полів в формах	Passed	Passed	Passed	Passed
Наявність головного меню на всіх сторінках	Passed	Passed	Passed	Passed
Коректне відображення при зміні масштабу	Passed	Passed	Passed	Passed
Перевірка форми зворотнього зв'язку	Passed	Passed	Passed	Passed
Перевірка сітки вирівнювання	Passed	Passed	Passed	Passed
Перевірка працездатності пошуку	Passed	Passed	Passed	Passed

4.2 Функціональне тестування

Суть цього процесу полягає в перевірці всього описаного функціоналу:

- перевірки роботи всіх обов'язкових функцій сайту;
- тестування працездатності на сайті користувальницьких форм (наприклад, зворотний зв'язок, додавання коментаря);
- перевірки пошуку роботи (включаючи релевантність результатів);
- перевірки гіперпосилань, пошук неробочих посилань;
- перевірки завантаження файлів на сервер;
- перевірки працездатності лічильників, встановлених на сторінках сайту;
- перегляд на відповідність вмісту сторінок сайту вихідного вмісту, наданого замовником. [20]

Для функціонального тестування я, також, вибрав наступні браузері: Google chrome, Internet explorer, Mozila firefox, Opera.

Виконано такі перевірки, таблиця 4.2.

Таблиця 4.2 Перевірка функціонального тесту

	IE 11	Firefox	GChrome	Opera
Сторінки ІС				
Відсутність непрацюючих кнопок	Passed	Passed	Passed	Passed
Перехід по сторінкам сайту	Passed	Passed	Passed	Passed
Перехід по посиланням	Passed	Passed	Passed	Passed
Коректне керування картою на сторінці зворотнього зв'язку	Passed	Passed	Passed	Passed
Коректна робота пошуку	Passed	Passed	Passed	Passed
Пошук по ключовим словам	Passed	Passed	Passed	Passed
Можливість відправки повідомлення адміністратору	Passed	Passed	Passed	Passed
Валідація полів				
Валідація полів на сторінці входу	Passed	Passed	Passed	Passed
Валідація полів на сторінці зворотнього зв'язку	Passed	Passed	Passed	Passed
Перевірка амін-панелі				
Коректне редагування даних	Passed	Passed	Passed	Passed
Коректне додавання даних	Passed	Passed	Passed	Passed
Можливість добавки користувачів	Passed	Passed	Passed	Passed
Коректне додавання медіа даних	Passed	Passed	Passed	Passed
Відображення полів	Passed	Passed	Passed	Passed

Відповідно до результатів тестування інформаційна система привільно працює на всіх вибраних браузерях, та не має дефектів з візуальної та функціональної частин.

ВИСНОВКИ

В кваліфікаційній роботі було запроектовано та впроваджено інтеграційну інформаційну систему для ТОВ «БВС Ритейл».

Мета розробки даної кваліфікаційної роботи полягає у створенні інформаційної системи, яка допоможе користувачеві ознайомитись з особливостями роботи та перевагами ТОВ «БВС Ритейл», знайти необхідний матеріал.

Користувач матиме змогу:

- переглянути інформацію;
- зареєструватися в якості користувача;
- знайти необхідні товари;
- зробити відповідне замовлення.

Також:

- створено пошукову систему, яка працює за принципом асоціативності;
- розроблено клієнт – серверну структуру, яка забезпечить цілісність інформації, доступність для роботи клієнта у будь-який момент часу;
- створено зрозумілий та адекватний інтерфейс.
- протестовано сайт на функціональні баги, баги пов'язані з оформленням, тестування користувацького інтерфейсу.

Всі ці тести були проведені в різних інтернет браузерях.

Сайт виложеної на хостінг <https://bvs.maltabitcoinmining.com/>

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. ДСТУ 2392-94 Інформація та документація. Базові поняття
2. Закон України «Про захист інформації в інформаційно-телекомунікаційних системах» від 05.07.1994 № 80/94-ВР (Редакція від 19.04.2014). Архів оригіналу за 22 червня 2017. Процитовано 8 вересня 2017.
3. Положення про технічний захист інформації в Україні, затверджене Указом Президента України від 27.09.1999 № 1229/99. Архів оригіналу за 14 вересня 2017. Процитовано 8 вересня 2017.
4. Порядок взаємодії органів виконавчої влади з питань захисту державних інформаційних ресурсів в інформаційних та телекомунікаційних системах, затверджений постановою Кабінету Міністрів України від 16.11.2002 р. № 1772. Архів оригіналу за 28 серпня 2017. Процитовано 23 серпня 2017.
5. Карасик І. Програмні та апаратні засоби захисту інформації для персональних комп'ютерів // Комп'ютерПресс № 3, 1995
6. HTML та CSS бібліотека [Електронний ресурс]. – Режим доступу: <http://htmlbook.ru/>
7. Анісімов А.В. Інформаційні системи та бази даних: Навчальний посібник для студентів факультету комп'ютерних наук та кібернетики. / Анісімов А.В., Кулябко П.П. – Київ. – 2017. – 110 с.
8. Антоненко В. М. Сучасні інформаційні системи і технології: управління знаннями : навч. посібник / В. М. Антоненко, С. Д. Мамченко, Ю. В. Рогушина. – Ірпінь : Нац. університет ДПС України, 2016. – 212 с.
9. Воронін А. М. Інформаційні системи прийняття рішень: навчальний посібник. / Воронін А. М., Зіатдінов Ю. К., Климова А. С. – К. : НАУ-друк, 2009. – 136с.
10. Галузинський Г. П. Інформаційні системи у бізнесі. Практикум для індивідуальної роботи: навч.- метод. посіб. для самост. вивч. Дисципліни. / Галузинський Г. П., Денісова О. О., Писаревська Т. А. – К. : КНЕУ, 2008. – 524с.

11. <https://uk.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1%D1%81%D0%B0%D0%B9%D1%82XHTML> [Електронний ресурс] – режим доступу: <http://htmlbook.ru/xhtml>
12. https://uk.wikipedia.org/wiki/%D0%A2%D1%80%D1%96%D0%B9%D0%BA%D0%BE%D0%B2%D0%B5_%D0%B4%D0%B5%D1%80%D0%B5%D0%B2%D0%BE
13. ANGULAR [Електронний ресурс] – режим доступу: <https://metanit.com/web/angular/>
14. APACHE [Електронний ресурс] – режим доступу: <http://www.apache.ru/>
15. MYSQL [Електронний ресурс] – режим доступу: <https://www.mysql.com/>
16. Діаграма прецедентів [Електронний ресурс] – режим доступу: [http://www.planerka.info/item/Diagramma-precedentov-\(variantov-ispolzovaniya\)-UML](http://www.planerka.info/item/Diagramma-precedentov-(variantov-ispolzovaniya)-UML)
17. Діаграма діяльності [Електронний ресурс] – режим доступу: http://it-gost.ru/articles/view_articles/96
18. Контекстна діаграма [Електронний ресурс] – режим доступу: <http://techtrend.com.ua/index.php>
19. Діаграма декомпозиції [Електронний ресурс] – режим доступу: http://studopedia.com.ua/1_162873_diagrami-dekompozitsii.html
20. Розробка інформаційної системи [Електронний ресурс] – режим доступу: http://bukvar.su/informatika_programmirovanie
21. Аналіз баз даних [Електронний ресурс] – режим доступу: <https://www.osp.ru/os/2002/03/181272/>
22. Порівняльний аналіз СУБД [Електронний ресурс] – режим доступу: <https://sibac.info/studconf/tech/xxxvi/43413>
23. Порівняльний аналіз серверів [Електронний ресурс] – режим доступу: <http://www.it-ic.ru/sravnenie-veb-serverov-apache-iis>

24. Огляд web-серверів [Електронний ресурс] – режим доступу:
<http://info-comp.ru/sisadminst/221-popular-web-servers.html>


```

        <div class="hover-particle"></div>

        <div class="hover-particle"></div>

        <div class="hover-particle"></div>

    </button>

    </div>

</section>

</main>

<div class="footer footer__only--icon">

    <div class="wrapper">

        <div class="footer__inner-wrapper">

            <div class="socials"><span>Підписуйтесь:</span>

                <ul class="socials__list">

                    <li class="socials__item">

                        <a class="btn btn-ghost-primary social__link" href="#"
target="_blank">

                            <svg class="social__icon">

                                <use xlink:href="#ic-fb"></use>

                            </svg>

                        </a>

                    </li>

                    <li class="socials__item">

                        <a class="btn btn-ghost-primary social__link" href="#"
target="_blank">

                            <svg class="social__icon">

                                <use xlink:href="#ic-lin"></use>

                            </svg>

                        </a>

                    </li>

                    <li class="socials__item">

```

```

        <a class="btn btn-ghost-primary social__link" href="#"
target="_blank">

            <svg class="social__icon">

                <use xlink:href="#ic-inst"></use>

            </svg>

        </a>

    </li>

</ul>

</div>

</div>

</div>

</div>

</div>

<div class="modal avante-login" id="avante-login">

    <div class="modal__close">

        <svg class="icon stroke-black" width="32" height="32">

            <use xlink:href="#ic-x"></use>

        </svg>

    </div>

    <div class="modal__wrapper">

        <div class="modal-content">

            <form class="form" method="post">

                <div class="preloader">

                    <div class="preloader__img-container">

                        <div class="preloader__animate"></div>

                    </div>

                </div>

                <div class="form__group">

                    <label class="input__wrapper">

                        <input class="price-form__input" type="text" required=""
name="name" placeholder="Имя">

                    </label>

                    <div class="error_text">Це поле обов'язкове</div>

```

```

    </div>

    <div class="form__group">
        <label class="input__wrapper">
            <input class="price-form__input" type="number" required=""
name="phone" placeholder="Телефон">
        </label>
        <div class="error__text">Це поле обов'язкове</div>
    </div>

    <div class="form__group">
        <label class="input__wrapper">
            <input class="price-form__input" type="text" name="company"
placeholder="Компанія">
        </label>
    </div>

    <div class="form__group">
        <label class="input__wrapper">
            <input class="price-form__input" type="email" name="email"
required="" placeholder="Email">
        </label>
        <div class="error__text">Це поле обов'язкове</div>
    </div>

    <div class="form__group">
        <label class="input__wrapper">
            <input class="contact-info__input" type="text" name="message"
placeholder="Повідомлення">
        </label>
    </div>

    <button class="btn btn--primary contact-info__btn feedbackForm"
type="submit"><span>Відправити</span>
</button>

    <div class="form__success">
        <h1 class="title title--sm">Дякую! Ваше повідомлення відправлено</h1>
    </div>
</form>
</div>

```

```
</div>
</div>
<script src="<?php echo get_stylesheet_directory_uri() ?>/assets/js/jquery.js"></script>
<script>
    $.ajaxSetup({
        headers: {
            'X-CSRF-TOKEN': $('meta[name="csrf-token"]').attr('content')
        }
    });
    $(document).ajaxError(function(event, jqxhr, settings, thrownError) {
        if (jqxhr.status == 401 || jqxhr.status == 419) {
            if (confirm('errors.session_expired')) {
                location.reload();
            }
        }
    });
</script>
<script
src="https://maps.googleapis.com/maps/api/js?key=AIzaSyDBKD1hpdD_2D7gKBoGBwIzgHD3UVlivA">
</script>
<script src="<?php echo get_stylesheet_directory_uri() ?>/assets/js/common.js"></script>
<script src="<?php echo get_stylesheet_directory_uri() ?>/assets/js/default.js"></script>
<script src="<?php echo get_stylesheet_directory_uri() ?>/assets/js/index.js"></script>
<script>
    var map_4;
    function initMap_4() {
        map_4 = new google.maps.Map(document.getElementById('map_4'), {
            zoom: 7.6,
            styles: [{
                "featureType": "all",
                "elementType": "geometry.fill",
                "stylers": [{
```

```
        "weight": "2.00"
      }
    ], {
      "featureType": "all",
      "elementType": "geometry.stroke",
      "stylers": [{
        "color": "#9c9c9c"
      }]
    }, {
      "featureType": "all",
      "elementType": "labels.text",
      "stylers": [{
        "visibility": "on"
      }]
    }, {
      "featureType": "landscape",
      "elementType": "all",
      "stylers": [{
        "color": "#f2f2f2"
      }]
    }, {
      "featureType": "landscape",
      "elementType": "geometry.fill",
      "stylers": [{
        "color": "#ffffff"
      }]
    }, {
      "featureType": "landscape.man_made",
      "elementType": "geometry.fill",
      "stylers": [{
        "color": "#ffffff"
      }]
    }, {
```

```
    "featureType": "poi",
    "elementType": "all",
    "stylers": [{
      "visibility": "off"
    }]
  }, {
    "featureType": "road",
    "elementType": "all",
    "stylers": [{
      "saturation": -100
    }, {
      "lightness": 45
    }]
  }, {
    "featureType": "road",
    "elementType": "geometry.fill",
    "stylers": [{
      "color": "#eeeeee"
    }]
  }, {
    "featureType": "road",
    "elementType": "labels.text.fill",
    "stylers": [{
      "color": "#7b7b7b"
    }]
  }, {
    "featureType": "road",
    "elementType": "labels.text.stroke",
    "stylers": [{
      "color": "#ffffff"
    }]
  }, {
    "featureType": "road.highway",
```

```
    "elementType": "all",
    "stylers": [{
      "visibility": "simplified"
    }]
  }, {
    "featureType": "road.arterial",
    "elementType": "labels.icon",
    "stylers": [{
      "visibility": "off"
    }]
  }, {
    "featureType": "transit",
    "elementType": "all",
    "stylers": [{
      "visibility": "off"
    }]
  }, {
    "featureType": "water",
    "elementType": "all",
    "stylers": [{
      "color": "#46bcec"
    }], {
      "visibility": "on"
    }]
  }, {
    "featureType": "water",
    "elementType": "geometry.fill",
    "stylers": [{
      "color": "#c8d7d4"
    }]
  }, {
    "featureType": "water",
    "elementType": "labels.text.fill",
```

```
        "stylers": [{
            "color": "#070707"
        }]
    }, {
        "featureType": "water",
        "elementType": "labels.text.stroke",
        "stylers": [{
            "color": "#ffffff"
        }]
    }
  ],
  center: new google.maps.LatLng(49.590181986314896, 34.55265999103818),
  mapTypeId: 'roadmap'
});

var icons_4 = {
  0: {
    icon: 'https://avante.md/img/map-icons/station.svg'
  },
  1: {
    icon: 'https://avante.md/img/map-icons/office.svg'
  },
  2: {
    icon: 'https://avante.md/img/map-icons/office.svg'
  },
};

$.ajax({
  method: "POST",
  url: "https://avante.md/map-markers",
  dataType: 'json'
}).done(function(response) {

  for (var index in response) {
```

```

        var mystr = response[index].coordinates;

        var myarr = mystr.split(",");

        var lat = myarr[0];

        var lon = myarr[1];

        var marker_4 = new google.maps.Marker({

            position: new google.maps.LatLng(lat, lon),

            icon: icons_4[response[index].marker_type].icon,

            map: map_4,

            zIndex: (response[index].zIndex ? parseInt(response[index].zIndex) :
null),

            id: response[index].id

            //title: feature.title

        });

        google.maps.event.addListener(marker_4, 'click', function() {

            markerInfo(this.id)

        });

    }

});

}

google.maps.event.addDomListener(window, 'load', initMap_4);
</script>

<script>

$(document).ready(function() {

    (function($) {

        $.fn.serializefiles = function() {

            var obj = $(this);

            /* ADD FILE TO PARAM AJAX */

```

```
        var formData = new FormData();

        $.each($(obj).find("input[type='file']"), function(i, tag) {

            $.each($(tag)[0].files, function(i, file) {

                formData.append(tag.name, file);

            });

        });

        var params = $(obj).serializeArray();

        $.each(params, function(i, val) {

            formData.append(val.name, val.value);

        });

        return formData;

    });

})(jQuery);

});

</script>

<script>

function get_today_date() {

    var today = new Date();

    var dd = today.getDate();

    var mm = today.getMonth() + 1; //January is 0!

    var yyyy = today.getFullYear();

    var hh = today.getHours();

    var min = today.getMinutes();

    var ss = today.getSeconds();

    if (dd < 10) {

        dd = '0' + dd

    }

    if (mm < 10) {

        mm = '0' + mm

    }

}
```

```
        if (hh < 10) {
            hh = '0' + hh
        }

        if (min < 10) {
            min = '0' + min
        }

        if (ss < 10) {
            ss = '0' + ss
        }

        today = yyyy + '-' + mm + '-' + dd + ' ' + hh + ':' + min + ':' + ss;
        return today;
    }
</script>

<?php wp_footer(); ?>
</body>

</html>
<?php
include_once get_stylesheet_directory() . '/inc/theme_default.php';
require_once get_stylesheet_directory() . '/inc/class-l3-logger.php';

$L3_theme = new L3Theme();
$L3_logger = new L3Logger();

$L3_theme->add_cdn('https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.5.2/css/all.min.css');

$L3_theme->add_cdn('https://code.jquery.com/jquery-3.7.1.min.js');

$L3_theme->add_google_api('key');
$L3_theme->theme_version('1.0.0');
// $L3_theme->maintenance_mode_enable();
```

```
register_nav_menus([
    'header-menu' => 'Header Menu',
    'footer-menu' => 'Footer Menu'
]);

// add_image_size( 'name', width, height, array('center','center'));
add_image_size('full_hd', 1920, 0, array('center', 'center'));
add_image_size('large_high', 1024, 0, false);

// Custom Functions
add_action('wp_ajax_register', 'register');
add_action('wp_ajax_nopriv_register', 'register');
function register()
{
    $phone = isset($_POST['phone']) ? $_POST['phone'] : null;
    $password = 'password';
    // $password = wp_generate_password();
    $bonus_card = generate_unique_numeric_bonus_card_number();

    $username = 'user_' . $phone;
    $email = $phone . '@example.com';

    if (username_exists($username) || email_exists($email)) {
        return new WP_Error('user_exists', 'Користувач вже зареєстрован');
    }

    $user_id = wp_create_user($username, $password, $email);

    if (is_wp_error($user_id)) {
        return $user_id;
    }

    update_user_meta($user_id, 'phone number', $phone);
}
```

```
update_user_meta($user_id, 'bonus_card_number', $bonus_card);

wp_update_user([
    'ID' => $user_id,
    'role' => 'subscriber'
]);

wp_send_json(['success' => true, 'user' => $user_id]);
}

add_action('wp_ajax_login', 'login');
add_action('wp_ajax_nopriv_login', 'login');

function login()
{
    $phone = isset($_POST['phone']) ? $_POST['phone'] : null;
    $password = isset($_POST['password']) ? $_POST['password'] : null;

    if (!$phone || !$password) {
        wp_send_json(['success' => false, 'message' => 'Заповніть всі поля.']);
    }

    $users = get_users([
        'meta_key' => 'phone_number',
        'meta_value' => $phone,
        'number' => 1,
        'count_total' => false
    ]);

    if (empty($users) || !isset($users[0])) {
        wp_send_json(['success' => false, 'message' => 'Користувач не знайдено.']);
    }

    $user = $users[0];
```

```
if (!wp_check_password($password, $user->user_pass, $user->ID)) {
    wp_send_json(['success' => false, 'message' => 'Невірний пароль.']);
}

wp_set_current_user($user->ID);
wp_set_auth_cookie($user->ID);

wp_send_json(['success' => true, 'user_id' => $user->ID, 'message' => 'Вхід
виконано.']);
}

function generate_unique_numeric_bonus_card_number($length = 10)
{
    $unique = false;
    $bonus_card_number = '';

    while (!$unique) {
        $bonus_card_number = '';

        for ($i = 0; $i < $length; $i++) {
            $bonus_card_number .= mt_rand(0, 9);
        }

        $existing_user = get_users([
            'meta_key' => 'bonus_card_number',
            'meta_value' => $bonus_card_number,
            'number' => 1,
            'count_total' => false
        ]);

        if (empty($existing_user)) {
            $unique = true;
        }
    }
}
```

```
}

return $bonus_card_number;
}

$( document ).ready(function() {

    var timer;

    var duration = 300;

    $(' .feedbackForm' ).on('click', function(e) {

        e.preventDefault();

        var form = $(this).closest(' .form' );

        var result = form[0].validate()

        if(timer) clearTimeout(timer);

        timer = setTimeout(function () {

            if (result.isValid == true) {

                result.showPreloader();

                $.ajax({

                    method: "POST",

                    url: '/forms/feedback',

                    data: form.serialize(),

                    dataType :'json',

                }).done(function(response) {

                    if(response['errors'] == true){

                        form.find(' .preloader' ).removeClass('preloader--shown');

                        alert('Internal Error!');

                    }else{

                        result.successForm()

                    }

                })

            }

        }, duration);

    });

});
```

```
        )) ;  
    }  
    }, duration);  
  
});  
  
$('.registerForm').on('click', function(e) {  
    e.preventDefault();  
    var form = $(this).closest('.form');  
    var result = form[0].validate()  
  
    if(timer) clearTimeout(timer);  
  
    timer = setTimeout(function () {  
        if (result.isValid == true) {  
  
            result.showPreloader();  
  
            $.ajax({  
                method: "POST",  
                url: '/wp-admin/admin-ajax.php',  
                data: form.serialize(),  
                dataType : 'json',  
            }).done(function(response) {  
                if(response['errors'] == true){  
                    form.find('.preloader').removeClass('preloader--shown');  
                    alert('Internal Error!');  
                }else{  
                    result.successForm()  
                }  
            });  
        }  
    }, duration);  
});
```

```
});

$('.loginForm').on('click', function(e) {

    e.preventDefault();

    var form = $(this).closest('.form');

    var result = form[0].validate()

    if(timer) clearTimeout(timer);

    timer = setTimeout(function () {

        if (result.isValid == true) {

            $.ajax({

                method: "POST",

                url: '/wp-admin/admin-ajax.php',

                data: form.serialize(),

                dataType : 'json',

            }).done(function(response) {

                if(response['errors'] == true){

                    form.find('.preloader').removeClass('preloader--shown');

                    alert('Internal Error!');

                }else{

                    window.location.reload();

                }

            });

        }, duration);

    });

$('.whatPrice').on('click', function(e) {

    e.preventDefault();

    var form = $(this).closest('.form');
```

```
var result = form[0].validate()

if(timer) clearTimeout(timer);

timer = setTimeout(function () {
    if (result.isValid == true) {
        result.showPreloader();

        $.ajax({
            method: "POST",
            url: '/forms/what-price',
            data: form.serialize(),
            dataType : 'json',
        }).done(function(response) {
            if(response['errors'] == true){
                form.find('.preloader').removeClass('preloader--shown');
                alert('Internal Error!');
            }else{
                result.successForm()
            }
        });
    }
}, duration);

});

$('.whantCard').on('click', function(e){
    e.preventDefault();
    var form = $(this).closest('.form');
    var result = form[0].validate()

    if(timer) clearTimeout(timer);

    timer = setTimeout(function () {
```

```
    if (result.isValid == true) {  
        result.showPreloader();  
        $.ajax({  
            method: "POST",  
            url: '/forms/whant-card',  
            data: form.serialize(),  
            dataType : 'json',  
        }).done(function(response) {  
            if(response['errors'] == true){  
                form.find('.preloader').removeClass('preloader--shown');  
                alert('Internal Error!');  
            }else{  
                result.successForm()  
            }  
        });  
    }  
}, duration);  
});  
  
$('.moreNews').on( 'click', function(e) {  
    e.preventDefault();  
    var page = $('.infinite-pagination').data('next-page');  
    if(page !== null) {  
        clearTimeout( $.data( this, "scrollCheck" ) );  
        $.get(page, function(data){  
            $('.appendNewsHere').append(data.news);  
            $('.infinite-pagination').data('next-page', data.newsNextPageUrl);  
            if(data.newsNextPageUrl == null){  
                $('.moreNews').remove();  
            }  
        });  
    }  
});
```

```

    }

    });

}

});

});

var panorama;

function stationInfo(station, stationPrices, stationGallery){

    var name = station.data('name');

    var streetCoordinates = station.data('streetviews');

    var streetViewsTab = '';

    var prices = station.data('price');

    var gallery = station.data('gallery');

    stationPrices.find('h1.title span').html(name);

    var pricesBody = '';

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

        pricesBody += '<div class="peco-prices__row-container">\n' +

            '                <div class="peco-prices__row">\n' +

            '                    <div class="peco-prices__col col-fuel"

style="color:'+ prices[i].title_color +'!important">'+ (prices[i].title !== null ?

prices[i].title : '') +'</div>\n' +

            '

                    <div class="peco-prices__col col-fuel-type"

style="color:'+ prices[i].subtitle_color +'!important">'+ (prices[i].subtitle !== null ?

prices[i].subtitle : '') +'</div>\n' +

            '

                    <div class="peco-prices__col col-price '+

(prices[i].price == null ? 'in-shadow' : '') +'">'+ (prices[i].price !== null ?

prices[i].price : '0.00') +'</div>\n' +

            '

                    </div>\n' +

            '                </div>';

    }

```

```

var galleryItems = '';

var galleryModal = '';

// Clear section street view

$('.street-view #street-view').html('');

$('.street-view').addClass('hidden');

// If exist street views, render the tabs and render street view from first tab
if(streetCoordinates !== '' && streetCoordinates.length){

    $('.street-view').removeClass('hidden');

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

        streetViewsTab += '<div class="products-list__tab-unit renderStreetView '+ (i ==
0 ? 'js-tab-active' : '') +' " data-coordinates="'+ streetCoordinates[i].coordinates +' "
data-heading="'+ streetCoordinates[i].heading_street +' ">\n' +

            '                <svg class="icon" width="10" height="20">\n' +

            '                <use xlink:href="#ic-place"></use>\n' +

            '                </svg>\n' +

            '                <h4 class="products-list__tab-unit-
heading">'+streetCoordinates[i].title+'</h4>\n' +

            '                </div>';

    }

    $('.street-view .products-list__tab-units--street').html(streetViewsTab);

    // Render first street view

    var latLong = streetCoordinates[0].coordinates.split(', ');

    var streetHeading = parseInt(streetCoordinates[0].heading_street);

    renderStreetView(latLong, streetHeading);

    $('.gallery').css('display', 'none');

} else {

    if(gallery.length){

        $('.gallery').css('display', 'block');
    }
}

```

```

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

            galleryItems += '<li class="gallery__item"></li>';

            galleryModal += '<div class="swiper-slide">\n' +

                '            <picture></picture>\n' +

                '            </div>';

        }

    }

}

stationPrices.find('.peco-prices__table').html(pricesBody);
stationGallery.find('.gallery__list').html(galleryItems);
stationGallery.find('.swiper-wrapper').html(galleryModal);
window.App.reinitSwiper('#gallery-swiper');
}

$(document).on('click', '.renderStreetView', function(){

    var item = $(this);

    var latLong = item.data('coordinates').split(', ');
    var streetHeading = parseInt(item.data('heading'));

    renderStreetView(latLong, streetHeading);
});

function renderStreetView(latLong, streetHeading){

    panorama = new google.maps.StreetViewPanorama(

        document.getElementById('street-view'),

        {

            position: new google.maps.LatLng(latLong[0],latLong[1]),

            pov: {heading: streetHeading, pitch: 0},

            zoom: 0

        })

}
}

```

```

function markerInfo(id) {

    var mapLocation = $('#map__locations');
    var mapInteractive = $('#map-interactive');

    $.ajax({

        method: "POST",

        url: '/map-marker/' + id,

        dataType : 'json',

        beforeSend: function () {

            mapLocation.addClass('js-opened-locations');
            mapInteractive.addClass('js-opened-locations');
            mapLocation.find('.preloader').addClass('preloader--shown');

        }

    }).done(function(response) {

        mapLocation.find('.locations__list').html(response.data);
        mapLocation.find('.preloader').removeClass('preloader--shown');
        if($('#locations-sweeper').length){

            window.App.reinitSwiper('#locations-sweeper');

        }

    });

}

<?php

/**
 * Template Name: Blog Page
 */

get_header(); ?>

<section class="newsfeed-main">

    <div class="wrapper-small newsfeed-main__info">

        <h1 class="title--lg newsfeed__heading">ИЮ HOBOFO?</h1>

        <div class="text--main">

```

```

        <p>Ласкаво Просимо!<br>Будь у курсі останніх новин та спеціальних пропозицій,
які доступні тобі.</p>

    </div>

</div>

<?php
$postsArgs = array(
    'post_type' => 'post',
    'posts_per_page' => -1,
);
$posts = new WP_Query($postsArgs); ?>
<?php if ($posts->have_posts()): ?>
    <div class="promotion__news appendNewsHere infinite-pagination">
        <?php while ($posts->have_posts()): $posts->the_post(); ?>
            <div class="promotion__news-big-item">
                <div class="promotion__news-item">
                    <time class="news__date" datetime="2021-06-30 08:52:28"><?php echo
get_the_date('d M, Y') ?></time>
                    <div class="news__wrapper">
                        <a class="news__preview" href="<?php echo get_the_permalink()
?>">
                            <h2 class="preview__heading"><?php echo get_the_title();
?></h2>
                        </a>
                        <div class="preview__btns">
                            <a class="btn btn-ghost-primary preview__details"
href="<?php echo get_the_permalink() ?>"><span>Детальніше</span>
                            </a>
                        </div>
                    </div>
                    <a class="news-item__img-link" href="<?php echo get_the_permalink()
?>" aria-label="Изменения в программе скидок">
                        <div class="news-item__img">
                            <picture>
                                <?php echo get_the_post_thumbnail(null, 'large') ?>

```

```
        </picture>
    </div>
</a>
</div>
</div>
<?php endwhile; ?>
<?php wp_reset_postdata(); ?>
</div>
<?php endif; ?>
</section>
<?php get_footer(); ?>
```