

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ХАРЧОВИХ ТЕХНОЛОГІЙ

Інститут (факультет) Автоматизації і комп'ютерних систем
 Кафедра Інформаційних технологій, штучного інтелекту і кібербезпеки
 Освітній ступінь бакалавр
 Спеціальність 122 «Комп'ютерні науки»
 Освітньо-професійна програма Комп'ютерні науки

ЗАТВЕРДЖУЮ

Завідувач кафедри
Інформаційних технологій, штучного
інтелекту і кібербезпеки
Сергій ГРИБКОВ

“ 15 ” квітня 2024 року

З А В Д А Н Н Я

НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧА

Борисова Дениса Сергійовича

(прізвище, ім'я, по батькові)

1. Тема роботи Розроблення веб-сайту ТОВ “Укравіт”
 керівник роботи Литвинов Валерій Андроникович, професор, доктор
технічних наук

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

затверджені наказом закладу вищої освіти від 15 квітня 2024 року № 279-кв

2. Строк подання здобувачем роботи 03.06.2024 р.

3. Вихідні дані до роботи

Структура підприємства, асортимент підприємства

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити) 1 Системний аналіз об'єкта автоматизації, 2 Технічне завдання, 3 Розробка комплексу задач автоматизації, 4 Охорона праці, Висновки, Список використаних джерел, Додатки

5. Перелік графічного матеріалу

Організаційна структура підприємства, схема моделей бази даних, діаграма бази даних, форми інформаційної системи

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1	Литвинов Валерій Андроникович	15.03.24	30.05.24

7. Дата видачі завдання 15 квітня 2023 року

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів виконання кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Дослідження предметної області	22.04.24-24.04.24	Виконано
2	Розробка функціональної моделі	25.04.24-26.04.24	Виконано
3	Розробка концептуальної моделі	27.04.24-28.04.24	Виконано
4	Розробка технічного завдання	29.04.24-01.04.24	Виконано
5	Визначення вимог до функцій системи	02.05.24-03.05.24	Виконано
6	Реалізація задач автоматизації системи	04.05.24-18.05.24	Виконано
7	Дослідження питань з охорони праці	19.05.24-20.05.24	Виконано
8	Оформлення пояснювальної записки	21.05.24-26.05.24	Виконано
9	Створення презентації	01.06.24-03.06.24	Виконано

Здобувач

_____ (підпис)

_____ (прізвище та ініціали)

Керівник роботи

_____ (підпис)

_____ (прізвище та ініціали)

АНОТАЦІЯ

Борисов Д.С. Розроблення веб-сайту ТОВ «Укравіт».

Дипломний проект на здобуття ступеню спеціаліста за напрямом підготовки «Комп'ютерні науки».

Дана робота складається з 93 сторінки, 22 таблиці, 39 рисунків, 3 додатків та 22 літературних джерел.

В даній роботі проаналізовано діяльність ТОВ «Укравіт» і розроблено інтернет-магазин з використанням структурного підходу до розробки.

Робота містить опис основних етапів проектування та розробки інтернет-магазину. Головним засобом для розробки та функціонування системи обрано серверну мову програмування PHP, Turbo CMS, базу даних MySQL та клієнтську мову програмування JavaScript.

Ключові слова: веб-сайт, Укравіт, Ukravit, Мікродобрива, Агрохімікати, ЗЗР.

ABSTRACTS

Borisov D.S. Development of the website of "Ukravit" LLC.

Diploma project for obtaining a specialist degree in the field of "Computer Science".

This work consists of 93 pages, 22 tables, 39 figures, 3 appendices and 22 references.

This paper analyzes the activities of Ukravit LLC and develops an online store using a structural development approach.

The paper describes the main stages of designing and developing an online store. The main tools for the development and operation of the system are the server-side programming language PHP, Turbo CMS, MySQL database and the client-side programming language JavaScript.

Keywords: website. Ukravit, Ukravit, Microfertilizers, Agrochemicals, Plant protection products.

ЗМІСТ

ВСТУП.....	7
РОЗДІЛ 1 СИСТЕМНИЙ АНАЛІЗ ТОВ «Укравіт»	8
1.1 Історія компанії ТОВ «Укравіт»	8
1.2 Аналіз нинішнього стану комп'ютеризації.....	11
1.3 Антикорупційна діяльність	11
1.4 Структура підприємства «Укравіт».....	13
1.5 Декомпозиція роботи відділу збуту ТОВ «Укравіт»	14
1.6 Аналіз існуючого сайту Ukravit	19
1.7 Аналіз аналогів сайтів для реалізації сайту ТОВ “Укравіт”	22
1.8 Обґрунтування доцільності проектування й розроблення сайту підтримки діяльності компанії	29
1.9 Концептуальна модель системи.....	29
1.10 Постановка задачі	32
1.11 Розрахунок економічного ефекту від впровадження сайту для компанії Ukravit.....	34
1.12 Висновок до розділу 1	36
РОЗДІЛ 2 ТЕХНІЧНЕ ЗАВДАННЯ НА ПРОЕКТУВАННЯ	36
2.1 Загальні положення.	36
2.2 Призначення і цілі створення системи.....	36
2.3 Характеристика об'єкта автоматизації.	37
2.4 Вимоги до системи	39
2.5 Склад і зміст робіт по створенню сайту.	50
2.6 Порядок контролю і приймання системи.	50
2.7 Вимоги до складу і змісту робіт із підготовки до введення системи в дію	51
2.8. Вимоги до документації.....	51
РОЗДІЛ 3 Опис комплексу задач автоматизації.....	53
3.1. Інформаційне забезпечення системи.....	53
3.2. Алгоритмізація та реалізація комплексу задач автоматизації.....	61
3.3. Інструкція користувача	78
3.4. Технічне та системне забезпечення розробки	84
РОЗДІЛ 4 ОХОРОНА ПРАЦІ	94
ВИСНОВКИ.....	96
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ.....	97

ВСТУП

У сучасному світі інформаційних технологій стає все більш очевидним, що використання веб-технологій є ключем до успіху для багатьох компаній. З розвитком інтернету і поширенням комп'ютерів в усіх сферах діяльності, веб-сайти стають важливим інструментом для просування бізнесу, залучення клієнтів та забезпечення зручного доступу до інформації.

Компанія "Укравіт", яка спеціалізується на виробництві вітчизняних вітамінних комплексів, не може залишатися осторонь від цього тренду. Із зростанням конкуренції на ринку зростає питання сучасного веб ресурсу, ефективна система в інтернеті може стати ключовим фактором успіху компанії.

Ця кваліфікаційна робота присвячена розробці веб-сайту для компанії "Укравіт", який буде не лише інформаційним ресурсом для клієнтів, але й інструментом управління внутрішніми процесами виробничого відділу. Мета - створити веб-платформу, яка сприятиме покращенню взаємодії з клієнтами, оптимізації виробничих процесів та підвищенню ефективності компанії "Укравіт" на ринку вітамінних комплексів

1. СИСТЕМНИЙ АНАЛІЗ ТОВ «Укравіт»

1.1 Історія компанії ТОВ «Укравіт»

Ukravit — українська компанія, заснована 1999 року, найбільший український виробник засобів захисту рослин та мікродобрив.

Діяльність

Основними напрямками діяльності компанії є виробництво та продаж засобів захисту рослин (ЗЗР), добрив з мікроелементами, рідких комплексних добрив та насіння, які застосовуються в ресурсозберігаючому промисловому агровиробництві.

Засоби захисту рослин та добрива виробляються на підприємстві у м. Черкаси і доставляються клієнтам у тарі власного виробництва, спеціально розробленій для ЗЗР. Компанія також контролює збір цієї тари для подальшої утилізації.

Препаратами Ukravit в Україні обробляється понад 10 млн га сільгоспугідь. Продукція компанії експортується до 4 країн світу (Грузія, Вірменія, Молдова, Узбекистан). Поставки до Ефіопії у 2024 призупинені через складну логістику.

Виробничі потужності розташовані в м. Черкаси на площі 8,5 га, обсяг виробництва становить понад 12 тис. тонн на рік, площі складських приміщень – 30 тис. м². Станом на 2024 компанія виробляє 178 препаратів для захисту та живлення в сільському господарстві, портфель препаратів щороку розширюється.

Підприємства Ukravit сертифіковано за ISO 9001:2015 (система управління якістю), ISO 14001:2015 (система екологічного управління) та ISO 45001:2018 (система управління охороною здоров'я та безпекою праці).

Ukravit має власну мережу із 20 регіональних представництв та 40 складів по всій країні. Ці представництва займаються реалізацією продукції та надають експертний супровід, розробляють індивідуальні рішення для кожного

агровиробника.

До складу компанії входить Ukravit Institute. Це сучасний R&D-центр, який поєднує навчально-наукову роботу, фундаментальні галузеві дослідження та агробіологічний сервіс для аграріїв.

Ukravit Institute має лабораторний комплекс, що включає 2,5 тис. м² лабораторій, фітотронних кімнат, кліматичних камер та теплиць.

Виконує дослідження ґрунту, насінневого матеріалу, агрохімікатів, добрив, води, рослинного матеріалу, якісних/кількісних показників у готовій с/г продукції. Дає змогу визначати залишкові кількості пестицидів та агрохімікатів.

Компанія випускає продукцію під власними торговельними марками, що об'єднані під брендами Ukravit, Авангард, Аптека садівника, Захисник.

Компанія надає підтримку українським аграріям через партнерські фінансові програми Ukravit Finance із банками (Укргазбанк, Ощадбанк, ОТП Банк, Райффайзен Банк, Креді Агріколь Банк та ін.) та співпрацює за програмою фінансування з USAID АГРО.

В рамках програми Ukravit Finance компанія забезпечує фінансування для придбання засобів захисту рослин, мікродобрив, насіння, дослідження ґрунтів, води, насіння та інших виробничих ресурсів в Ukravit Institute, проведення діагностики посівів, дослідження якості врожаю у лабораторних умовах.

Історія

Ukravit було засновано у 1999 році.

Компанія розпочала свою діяльність із дистрибуції, а за 3 роки започаткувала власне виробництво засобів для знищення побутових комах та гризунів, поступово посіла провідні позиції у забезпеченні засобами дезінфекції та дератизації спеціалізованих служб. Також розпочала виробництво засобів захисту рослин для приватного сектора.

У 2004 — отримала державну реєстрацію та у 2005 розпочала виробництво препаратів із захисту рослин для агросектору у м. Черкаси.

У 2008 році Ukravit розширив асортимент деззасобів та засобів захисту рослин для промислового та приватного секторів – під ТМ «Аптека садівника».

Згодом до складу компанії Ukravit увійшла «Фабрика агрохімікатів» (нині Завод Ukravit). Компанія розпочала виробництво засобів захисту рослин для агросектора.

У 2010 році компанія розпочала виробництво добрив з мікроелементами та стимуляторів росту рослин під ТМ Авангард.

Наступного року Ukravit вийшов на міжнародний ринок та розпочав експорт препаратів до Молдови та Грузії. Отримав сертифікати ISO 9001

У 2016—2018 роках було розпочато будівництво нових об'єктів, включаючи Науково-дослідний центр «Інституті здоров'я рослин» (згодом — Ukravit Institute), логістичний центр та нові виробничі корпуси.

У 2018 – підтвердив та отримав європейську сертифікацію за ISO 9001:2015, ISO 14001:2015 та ISO 45001:2018.

У 2019 році було створено агроіндустріальний парк Ukravit Science Park, який об'єднав усі активи компанії.

У 2019 році компанія перейшла на прямі продажі для аграріїв, відкривши 20 регіональних торговельних представництв. Упровадження напрямків мінеральних добрив та дистрибуції насіння.

Станом на 2024 рік у 20 обласних центрах діють представництва Ukravit, крім того, компанія має 40 складів по всій країні, що дає можливість швидко постачати необхідні засоби захисту рослин аграріям протягом. У кожному регіональному представництві діє агрономічна служба, що надає аграріям консультації щодо застосування засобів захисту рослин та досягнення запланованої врожайності.

У 2024 розпочато виробництво рідких комплексних добрив. Це стало ще одним напрямком діяльності для комплексного забезпечення аграріїв основними матеріально-технічними ресурсами.

У 2023 році компанія випустила сонцезахисний крем для аграріїв

1.2 Аналіз нинішнього стану комп'ютеризації

Аналіз поточного стану комп'ютеризації Ukravit в контексті розробки

автоматизованого веб-сайту розкриває деякі важливі недоліки та виклики, з якими стикається підрозділ, що відповідає за цю сферу діяльності.

Наразі, Ukravit використовує певне програмне забезпечення для реалізації певних функцій, проте ці інструменти не забезпечують достатньої функціональності для повноцінної розробки та управління веб-сайтом. Наявність обмежень у функціональності та можливостях наявного програмного забезпечення ускладнює роботу з розробкою та підтримкою веб-сайту.

Зокрема, деякі процеси, такі як управління контентом, взаємодія з клієнтами та аналітика веб-сторінки, можуть бути недостатньо ефективними або не повністю автоматизованими через обмежені можливості наявного програмного забезпечення. Це може призвести до затримок у впровадженні нових функцій, нестабільності роботи сайту та неефективного взаємодії з клієнтами.

Крім того, важливо враховувати, що швидкість розвитку технологій веб-розробки вимагає постійного оновлення інструментів та підходів. У цьому контексті, використання застарілих або неповноцінних інструментів може стати серйозним обмеженням для подальшого розвитку та конкурентоспроможності компанії.

Отже, нинішній стан комп'ютеризації Ukravit у контексті розробки автоматизованого веб-сайту є незадовільним через недоліки у використанні програмного забезпечення, неефективність деяких процесів та недостатню автоматизацію ключових функцій для розробки та управління веб-сайтом.

1.3 Антикорупційна діяльність

Акціонерне товариство "Укравіт" веде активну антикорупційну діяльність, спрямовану на забезпечення прозорості, чесності та відкритості у всіх сферах своєї діяльності. Для досягнення цієї мети, компанія реалізує широкий спектр заходів:

1. Внутрішні контрольні процедури: "Укравіт" встановлює та підтримує ефективну систему внутрішнього контролю, яка включає в себе розробку та впровадження правил, політик та процедур, спрямованих на запобігання

корупції та недобросовісній діяльності.

2. Тренінги та освітня робота: "Укравіт" проводить тренінги та семінари для своїх співробітників з питань антикорупційної політики, етики та чесної практики у бізнесі. Це допомагає підвищити свідомість персоналу та збільшити відповідальність за їхні дії.
3. Запобігання конфлікту інтересів: Компанія встановлює та дотримується політики, яка забороняє конфлікт інтересів серед своїх працівників та управлінського складу. Це включає обмеження участі в підприємницькій діяльності, яка може призвести до конфлікту інтересів з обов'язками на посаді.
4. Відкритість та звітність: Компанія практикує прозору звітність перед зацікавленими сторонами, включаючи інвесторів, клієнтів, урядові органи та громадськість. Це допомагає підтримувати високий рівень довіри та відкритості у взаємодії зі своїми стейкхолдерами.
5. Взаємодія з органами влади та громадськими організаціями: "Укравіт" активно співпрацює з органами влади та громадськими організаціями з метою виявлення та запобігання випадків корупції, а також сприяння у вирішенні антикорупційних проблем на рівні галузі.

Ці заходи дозволяють "Укравіт" забезпечувати високі стандарти етики та чесності у своїй діяльності, підтримуючи стійкий розвиток компанії та сприяючи загальному розвитку суспільства.

1.4 Структура підприємства «Укравіт»

З ростом виробництва та розвитком компанії "Укравіт" виникає потреба в чітко структурованій організаційній системі, що забезпечує ефективне управління всіма аспектами діяльності. Для цього встановлюється наступна структура підприємства, що складається з ряду служб, підрозділів та відділів, які взаємодіють для досягнення стратегічних цілей та ефективного функціонування компанії.

Структура підприємства "Укравіт" організована у наступний спосіб:

1. Вище керівництво (Топ-менеджмент):

- Генеральний директор
- Заступник генерального директора з виробництва
- Фінансовий директор
- Директор з розвитку бізнесу

2. Адміністративний та організаційний блок:

- Юридичний відділ
- Відділ кадрів та управління персоналом
- Відділ інформаційних технологій
- Відділ охорони праці та безпеки

3. Виробничий блок:

- Відділ розробки нових продуктів
- Відділ контролю якості продукції
- Виробничі цехи:
 - Цех з виробництва добрив
 - Цех з виробництва захисних засобів
 - Цех з виробництва обробних засобів

4. Дистрибуційний блок:

- Відділ логістики та транспорту
- Відділ збуту
- Відділ клієнтських відносин

5. Дослідницько-розвідувальний блок:

- Науково-дослідний відділ
- Відділ клінічних досліджень та тестування продукції

6. Агрономічний блок:

- Відділ агрономії та агротехнологій
- Відділ консультацій з використання агрохімікатів



Рисунок 1.1 – Структура підприємства Ukravit

Ця структура дозволяє ефективно керувати всіма аспектами діяльності компанії "Укравіт", включаючи виробництво, дистрибуцію, дослідження та розвиток продуктів, а також забезпечення відповідності нормативним вимогам і стандартам якості.

1.5 Декомпозиція роботи відділу збуту ТОВ «Укравіт»

Для виконання поставленого індивідуального завдання аналіз виробничої діяльності за допомогою сучасних CASE-засобів ми можемо скористатися програмним забезпеченням AllFusion Process Modeler, також відомим як BPwin. Це програмне забезпечення, розроблене компанією LogicWorks, призначене для створення та підтримки систем контролю інформації. Однією з важливих функцій BPwin є оптимізація бізнес-процесів. З його допомогою можна чітко відстежувати, хто що робить, коли це відбувається, які дані отримує та передає, а також якими документами керується. Таким чином, застосовуючи цей програмний продукт, ми можемо створити верхній рівень роботи відділу збуту ТОВ "Укравіт". (рис. 1.2).

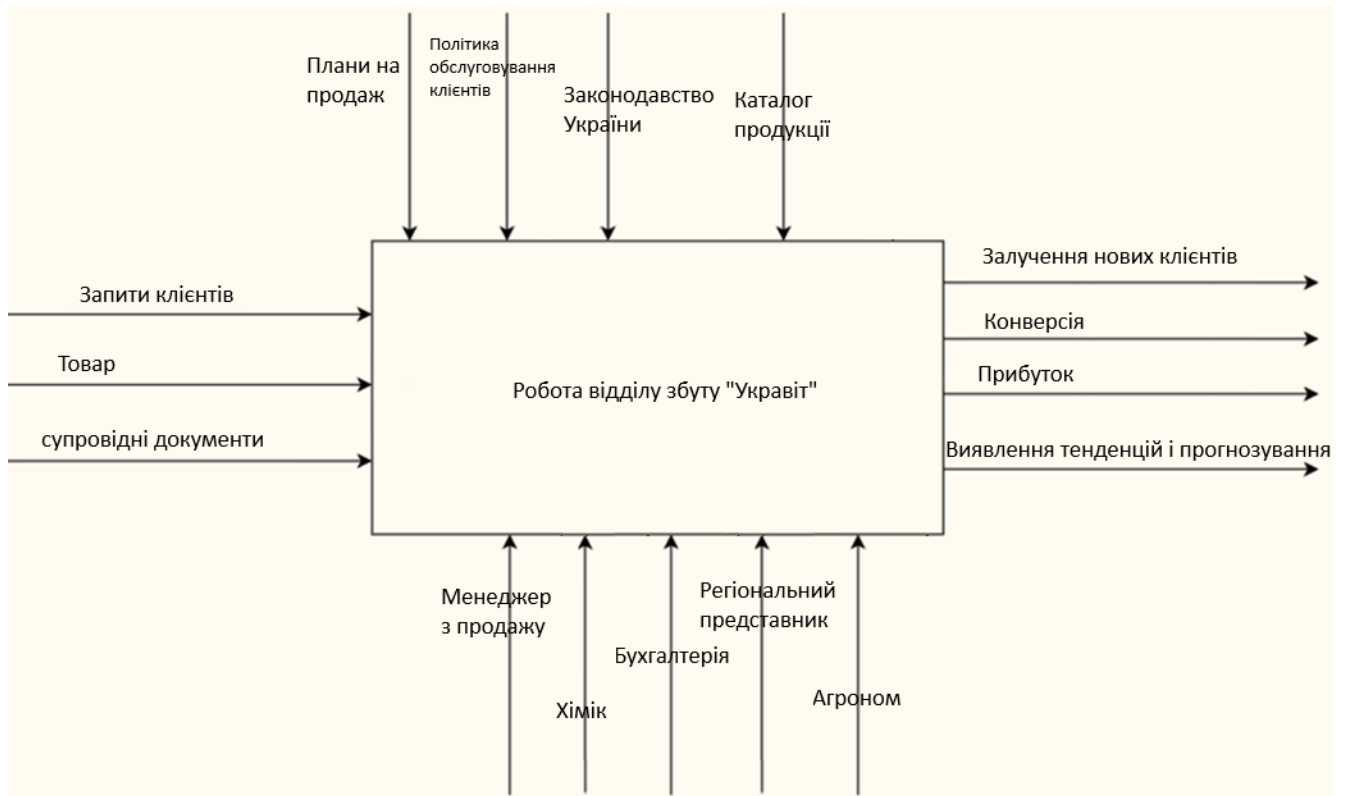
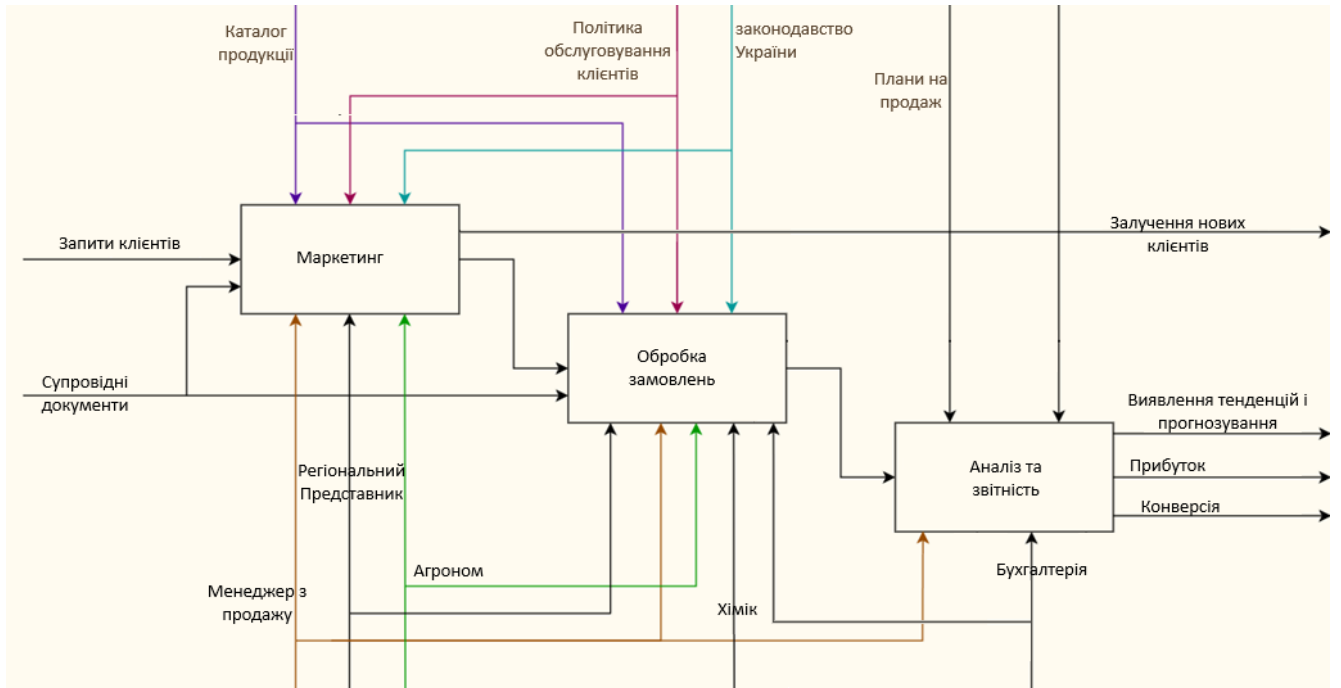


Рисунок 1.2 – Контекстна діаграма роботи відділу збуту

Оглядаючи складну структуру відділу на даному підприємстві, можна помітити значну кількість працівників у нижній частині діаграми, які виконують різноманітні завдання за різними рівнями складності. Однак, наразі ми не можемо конкретизувати, які саме завдання вони виконують. Тому нам необхідно провести декомпозицію на першому рівні. Діаграму можна переглянути на рисунку 1.3.



Рисунк 1.3 – Декомпозиція 1-го рівня організації роботи відділу збуту

На даній діаграмі відображені такі загальні процеси, а саме:

1. Маркетинг;
2. Обробка замовлень;
3. Аналіз та звітність.

Отже, тепер у нас є загальне уявлення про роботу відділу на підприємстві. Проте, зрозуміло, що дана схема все ще не розкриває конкретних завдань, які стоять за зазначеними на рисунку 1.3 процесами. Тому логічним кроком буде перехід до декомпозиції другого рівня.

Почнемо з огляду першого такого процесу - Маркетингу. Його підпроцеси відображено на рисунку 1.4.

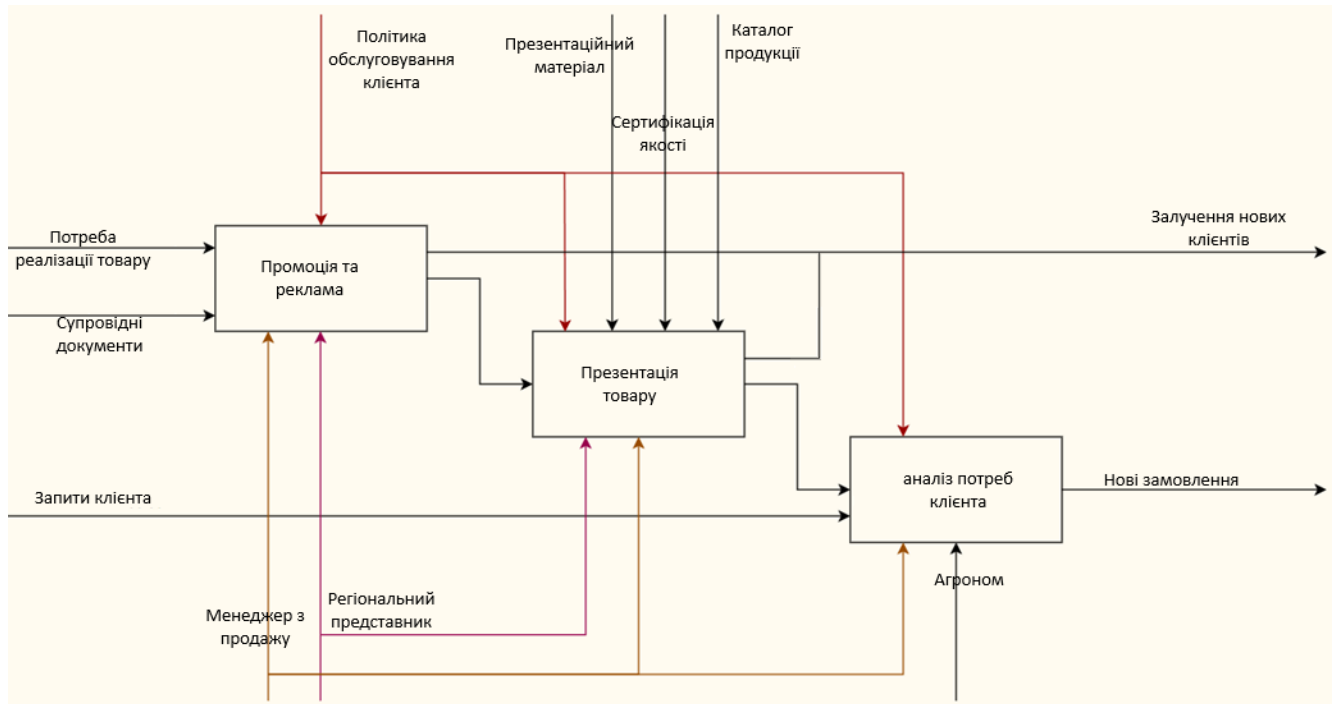


Рисунок 1.4 – Декомпозиція 2-го рівня процесу «Маркетинг»

На цій діаграмі відображені такі підпроцеси більшого процесу, а саме:

1. Промоція та реклама;
2. Презентація товару;
3. Аналіз потреб клієнта.

Як бачимо, на цьому етапі декомпозиції з'являється вже конкретні документи, що зберігаються, обробляються чи передаються іншим відділам підприємства. Зрозуміло, що представляти їх на вищих рівнях декомпозиції не є доцільним, оскільки тут вже йде специфіка кожного окремого процесу. Саме для цього й покликані діаграми нижчих рівнів – дати детальну картину роботи того чи іншого процесу роботи.

Ще один процес, який потребує детального огляду – це проведення співбесіди з потенційним кандидатом. Декомпозицію даного процесу можна побачити на рисунку 1.5.

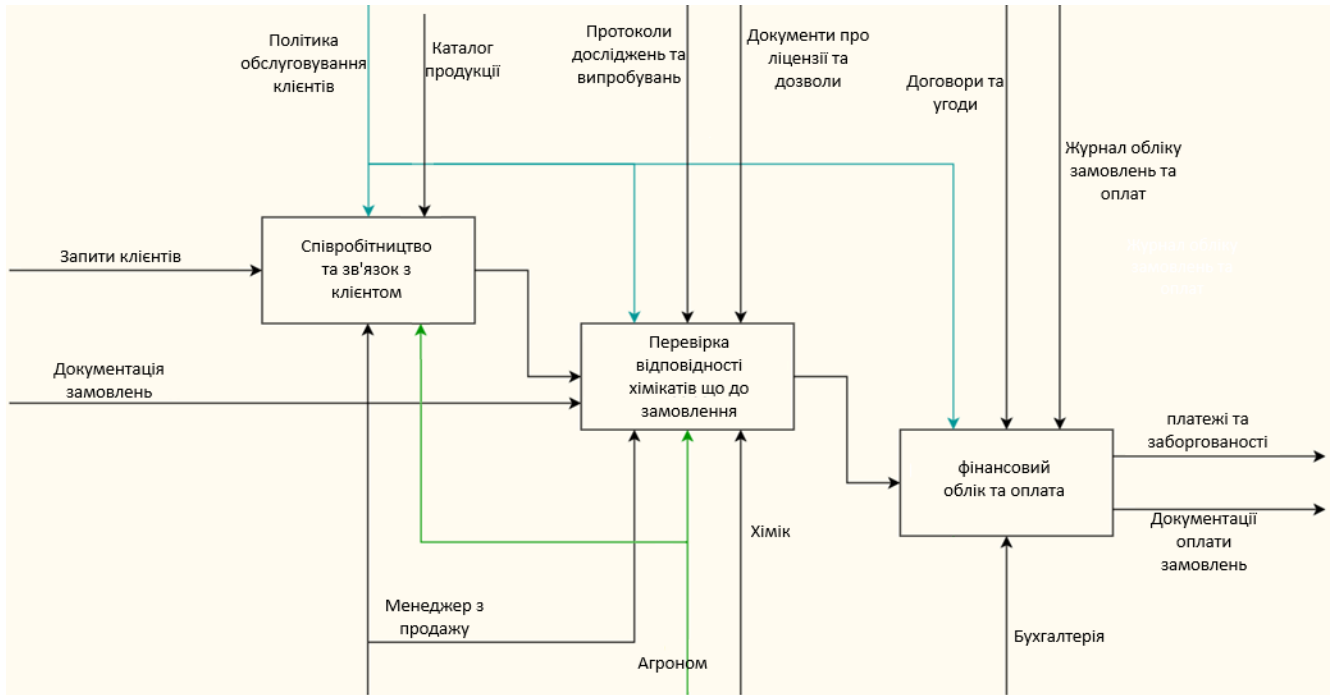


Рисунок 1.5 – Декомпозиція 2-го рівня процесу «обробка замовлень»

На даній діаграмі відображені такі загальні процеси, а саме:

1. Співробітництво та зв'язок з клієнтом;
2. Перевірка відповідності хімікатів що до замовлення;
3. Фінансовий облік та оплата.

Як бачимо, Менеджер з продажу виявився ключовою фігурою на різних рівнях декомпозиції бізнес-процесів на підприємстві. Він бере участь у всіх етапах від пошуку клієнтів до виконання замовлень та забезпечення високого рівня задоволеності клієнтів.

1. Пошук потенційних клієнтів:

- Менеджер з продажу активно взаємодіє з потенційними клієнтами, представляючи продукти та послуги підприємства.

2. Обробка замовлень:

- Він відповідає за отримання та обробку замовлень від клієнтів, включаючи узгодження умов оплати та доставки.

3. Підтримка клієнтів:

- Менеджер з продажу забезпечує підтримку та консультує клієнтів щодо продуктів та послуг, відповідає на їх запитання та вирішує

можливі проблеми.

4. Аналіз та звітність:

- Він забезпечує збір та аналіз даних щодо продажів, виставлення рахунків та звітності про результативність роботи відділу продажів.

5. Розвиток стратегій продажу:

- Менеджер з продажу бере участь у розробці стратегій та тактик продажу продукції підприємства, враховуючи потреби та побажання клієнтів.

Загалом, роль менеджера з продажу в системі декомпозиції бізнес-процесів є незамінною для ефективної роботи підприємства, оскільки він відіграє важливу роль у взаємодії з клієнтами та забезпечує досягнення цілей збуту та прибутковості.

1.6 Аналіз існуючого сайту Ukravit

В ході проектування кваліфікаційної роботи було проведено детальний аналіз існуючого сайту компанії Ukravit. На рисунку 1.6 зображено фрагмент головної сторінки нинішнього сайту компанії.

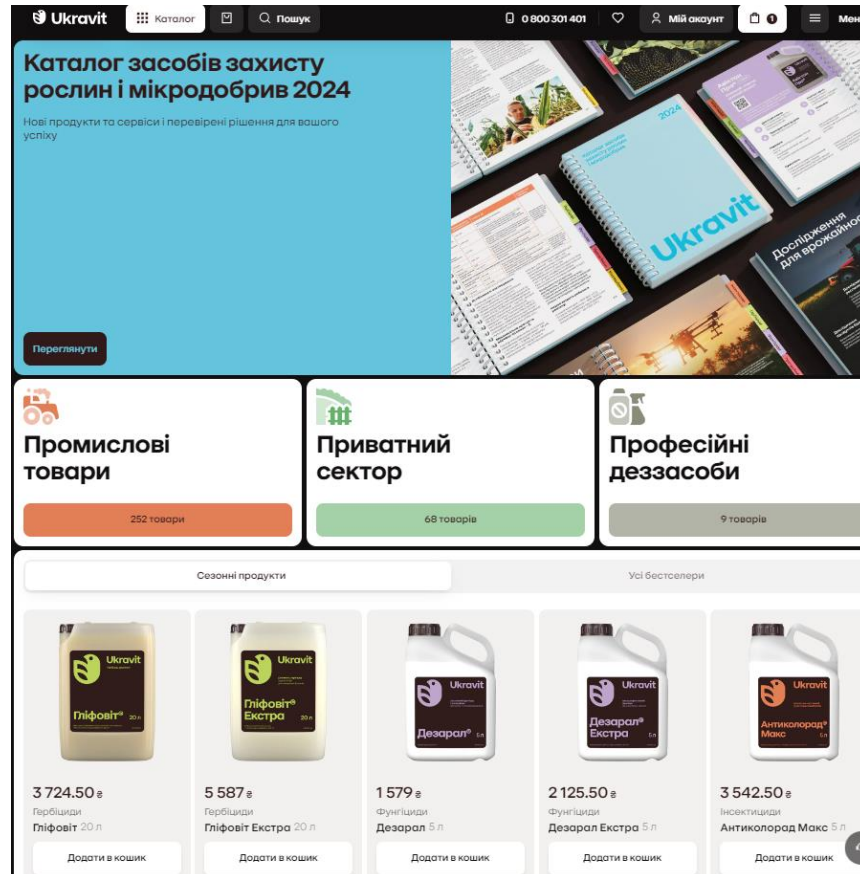


Рисунок 1.6 – Існуючий сайт компанії Ukravit

Метою аналізу було визначити основні недоліки поточного сайту, які заважають ефективному управлінню та використанню ресурсу. Результати аналізу виявили низку критичних проблем, які потребують вирішення в процесі розробки нового веб-сайту. Основні недоліки поточного сайту включають наступне:

Недоліки Існуючого Сайту

Складність редагування контенту

Ручне редагування коду: Основним недоліком існуючого сайту є необхідність змінювати контент вручну в коді. Це вимагає значних витрат часу і наявності спеціальних знань у програмуванні. Навіть для внесення незначних змін, таких як корекція тексту або заміна зображення, необхідно втручатися в HTML, CSS або JavaScript коди сайту. Це не лише ускладнює процес редагування, але й підвищує ризик виникнення помилок, які можуть вплинути на функціональність всього сайту.

Відсутність централізованої системи управління:

Розрізненість даних: Всі дані сайту зберігаються в різних файлах або базах даних без централізованого управління. Це призводить до труднощів в організації і доступі до даних. Адміністраторам доводиться витратити більше часу на пошук та редагування необхідної інформації, що негативно впливає на продуктивність.

Обмежені можливості адміністрування

Відсутність ролей і прав доступу: На існуючому сайті відсутня можливість призначення різних ролей для користувачів з різними рівнями доступу. Це створює труднощі в адмініструванні сайту, особливо коли потрібно делегувати різні завдання різним співробітникам. Відсутність розподілу прав доступу також підвищує ризик несанкціонованих змін або доступу до конфіденційної інформації.

Відсутність інтеграції з сучасними інструментами

Інтеграція з аналітичними інструментами: Сайт не підтримує інтеграцію з сучасними аналітичними інструментами, такими як Google Analytics, що ускладнює відстеження статистики відвідувань та аналіз поведінки користувачів. Відсутність цих можливостей обмежує здатність компанії проводити ефективний маркетинговий аналіз і приймати обґрунтовані рішення на основі даних.

Переваги Нової Системи Управління Контентом (CMS)

Централізоване управління контентом.

Панель адміністратора: Нова CMS надає зручну панель адміністратора, яка дозволяє керувати всім контентом сайту з одного місця. Це значно спрощує процес оновлення інформації та знижує ризик виникнення помилок.

Гнучкість і простота редагування.

Візуальний редактор: Завдяки вбудованому візуальному редактору (WYSIWYG), редагування текстів, зображень та інших елементів стає інтуїтивно зрозумілим і не вимагає знань програмування. Адміністратори можуть швидко вносити зміни безпосередньо через інтерфейс.

Ролі та права доступу.

Розподіл обов'язків: Нова CMS дозволяє створювати різні ролі для користувачів з різними рівнями доступу (адміністратор, редактор, автор тощо). Це покращує безпеку та спрощує процес управління сайтом, оскільки кожен користувач може виконувати тільки ті дії, які дозволені його роллю.

Управління товарами та контентом

Редагування товарів та блогів: Новий сайт надає зручні інструменти для додавання, видалення та редагування товарів, блогів і статей. Це дозволяє оперативно оновлювати асортимент продукції та інформаційні матеріали.

Інтеграція з аналітичними інструментами

Аналітика і звітність: Нова CMS підтримує інтеграцію з Google Analytics та іншими аналітичними інструментами, що дозволяє легко відстежувати статистику відвідувань, аналізувати поведінку користувачів і оптимізувати маркетингові стратегії.

Підтримка мультимовності.

Багатомовний контент: Нова система дозволяє легко додавати та керувати контентом на різних мовах, що важливо для міжнародного бізнесу і підвищує доступність сайту для користувачів з різних країн.

Висновок.

Перехід на нову систему управління контентом (CMS) дозволить компанії Ukravit значно спростити процес управління сайтом, підвищити його ефективність і забезпечити кращу взаємодію з користувачами. Завдяки новій CMS, редагування контенту стане швидким та інтуїтивно зрозумілим, а адміністратори отримають інструменти для ефективного керування всім сайтом з однієї панелі.

1.7 Аналіз аналогів сайтів для реалізації сайту ТОВ “Укравіт”

1.7.1 Функціональність сайту та його характеристика в сфері діяльності “Укравіт”

Сайт компанії "Укравіт" має стати важливим інструментом для залучення нових клієнтів, підтримки існуючих клієнтів та представлення продуктів і послуг компанії. Ось характеристики та функціональні можливості сайту:

1. Каталог продукції: Сайт містить детальний каталог продукції компанії "Укравіт", де клієнти можуть ознайомитися з асортиментом товарів та послуг, які надаються.

2. Опис продукції: Кожен товар чи послугу супроводжує докладний опис, технічні характеристики, фотографії та, можливо, відео-огляди, що дозволяє клієнтам зробити обізнаний вибір.

3. Онлайн-консультант: На сайті може бути встановлений онлайн-чат або система зворотного зв'язку, щоб клієнти могли отримати консультацію в режимі реального часу від представників компанії.

4. Форма замовлення: Клієнти можуть легко зробити замовлення через сайт, заповнивши спеціальну форму замовлення і вказавши всю необхідну інформацію.

5. Новини та акції: Сайт може містити розділ з новинами та акціями компанії, де клієнти можуть дізнатися про останні новини, акції та спеціальні пропозиції.

6. Контактна інформація: На сайті розміщена контактна інформація компанії, включаючи адресу, телефони та електронну пошту, щоб клієнти могли зв'язатися з представниками компанії для отримання додаткової інформації чи консультації.

7. Мультимовність: Якщо компанія працює в міжнародному масштабі або має аудиторію з різних країн, сайт може підтримувати кілька мов, щоб забезпечити комфортну взаємодію з клієнтами з різних культур і мовних груп.

Ці функції і характеристики дозволять сайту компанії "Укравіт" ефективно

представляти її продукцію та послуги, залучати нових клієнтів та підтримувати довгострокові відносини зі своєю аудиторією.

1.7.2 Сайт підприємства «Альфа Сمارт Агро»

Сайт компанії "Альфа Смарт Агро", вас вітає відкритий портал до світу сучасних технологій та інновацій у сфері сільського господарства. Сайт відображається в привітному для користувача форматі, де кожен відвідувач може знайти необхідну інформацію та ресурси для покращення своєї сільськогосподарської діяльності.

На головній сторінці сайту вас вітає загальна інформація про компанію та її місію, яка полягає в сприянні впровадженню новітніх технологій у агропромисловому секторі. Тут також представлений перелік основних розділів сайту для зручності навігації.

Один із розділів присвячений продукції та послугам компанії. В цьому розділі можна знайти детальні описи сільськогосподарського обладнання, яке пропонується до придбання, а також послуги консультування з питань вибору та використання цих засобів.

Додатково, на сайті є розділ, присвячений технологіям та інноваціям. Тут зібрана інформація про новітні досягнення у сільському господарстві, які можуть бути корисні для аграріїв. Від дронів для моніторингу полів до систем штучного інтелекту для аналізу ґрунту - усе це представлено на сайті.

Окрім цього, на сайті доступні новини та статті зі світу агропромислової галузі, які допоможуть користувачам бути в курсі останніх тенденцій та подій.

В цілому, сайт компанії "Альфа Смарт Агро" створений для того, щоб надати користувачам доступ до цінної інформації та ресурсів, які допоможуть їм вдосконалити свою сільськогосподарську діяльність.

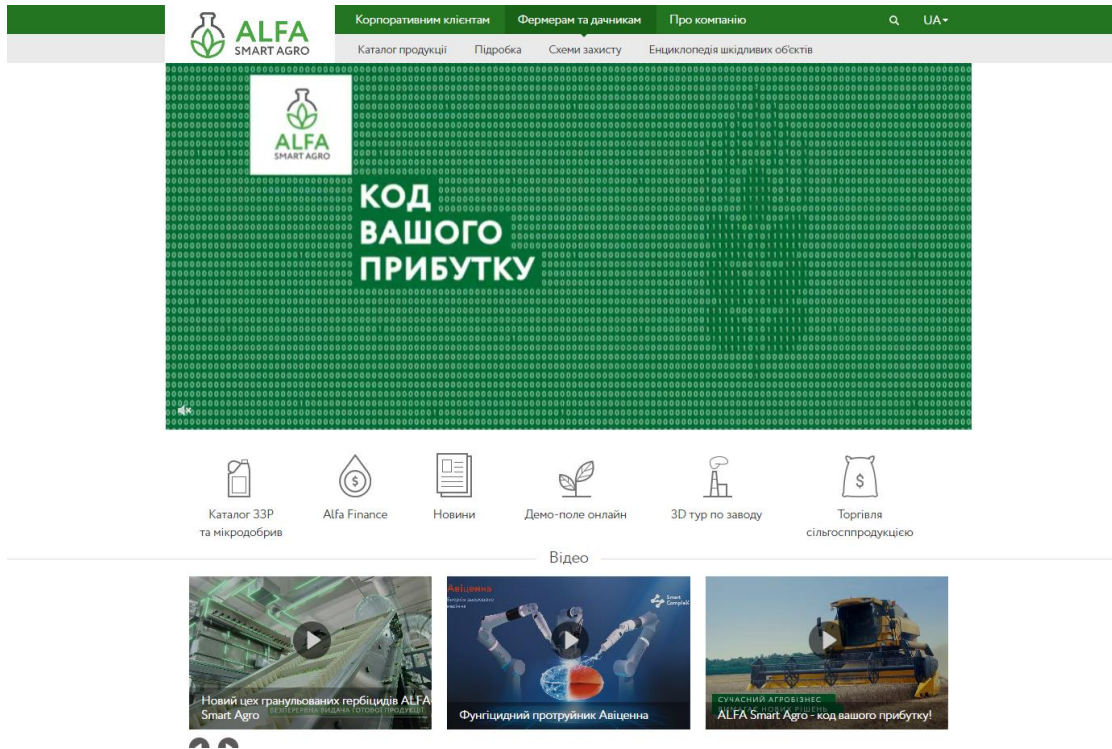


Рисунок 1.7 – Головна сторінка сайту «Альфа Смарт Агро»

1.7.3 Сайт підприємства «ЛНЗ Дефенда»

Коли ви відвідуєте сайт компанії "ЛНЗ Дефенда", вас вітає відкритий портал до світу сучасних технологій та інновацій у сфері захисту і безпеки. Сайт відображається в привітному для користувача форматі, де кожен відвідувач може знайти необхідну інформацію та ресурси для покращення своїх потреб у сфері захисту.

На головній сторінці сайту представлена загальна інформація про компанію "ЛНЗ Дефенда" та її місію, яка полягає в постачанні інноваційних засобів захисту та безпеки для різних сфер діяльності. Тут також доступний перелік основних розділів сайту для зручності навігації.

Один із розділів присвячений продукції та послугам компанії. Тут відображені детальні описи захисного та безпечного обладнання, яке пропонується до придбання, а також послуги консультування з питань вибору та використання цих засобів.

Додатково, на сайті є розділ, присвячений технологіям та інноваціям у сфері безпеки. Тут зібрана інформація про новітні технологічні рішення, що

використовуються для підвищення рівня захисту у різних сферах, від промисловості до побутового використання.

Окрім цього, на сайті доступні новини та статті зі світу безпеки та захисту, що дозволяє користувачам бути в курсі останніх тенденцій та інновацій у цій галузі. В цілому, сайт компанії "ЛНЗ Дефенда" створений для того, щоб надати користувачам доступ до цінної інформації та ресурсів, які допоможуть їм покращити свої системи захисту та безпеки.



Рисунок 1.8 – Головна сторінка сайту «ЛНЗ Дефенда»

1.7.4 Сайт підприємства «Агрорось»

При відвідуванні сайту Спідприємства "Агрорось" вас вітає віртуальний портал у світ сільськогосподарського розвитку та продуктивності. Наш веб-ресурс створений з метою забезпечити фермерів та агрономів усіма необхідними ресурсами та інформацією для вдосконалення їх сільськогосподарської діяльності.

На головній сторінці сайту вас вітаємо із загальною інформацією про компанію "Агрорось" та її головні цілі. Тут також представлено широкий перелік доступних розділів для зручної навігації.

Один із розділів присвячений продукції та послугам. Ви можете ознайомитися з

нашим асортиментом сільськогосподарського обладнання, розсадою та добривами, а також отримати консультацію від наших фахівців щодо їх використання та підтримки.

Також на сайті є розділ, присвячений сільськогосподарським технологіям та інноваціям. Тут ви знайдете інформацію про новітні методи обробки ґрунту, системи поливу та агротехнічні рішення, які допоможуть вам оптимізувати виробничі процеси.

Додатково, на сайті доступні останні новини та статті зі світу сільськогосподарського господарства, які допоможуть вам бути в курсі останніх тенденцій та інновацій у галузі.

Загалом, сайт Спідприємства "Агрорось" призначений для забезпечення користувачів цінною інформацією та ресурсами, необхідними для успішної сільськогосподарської діяльності.

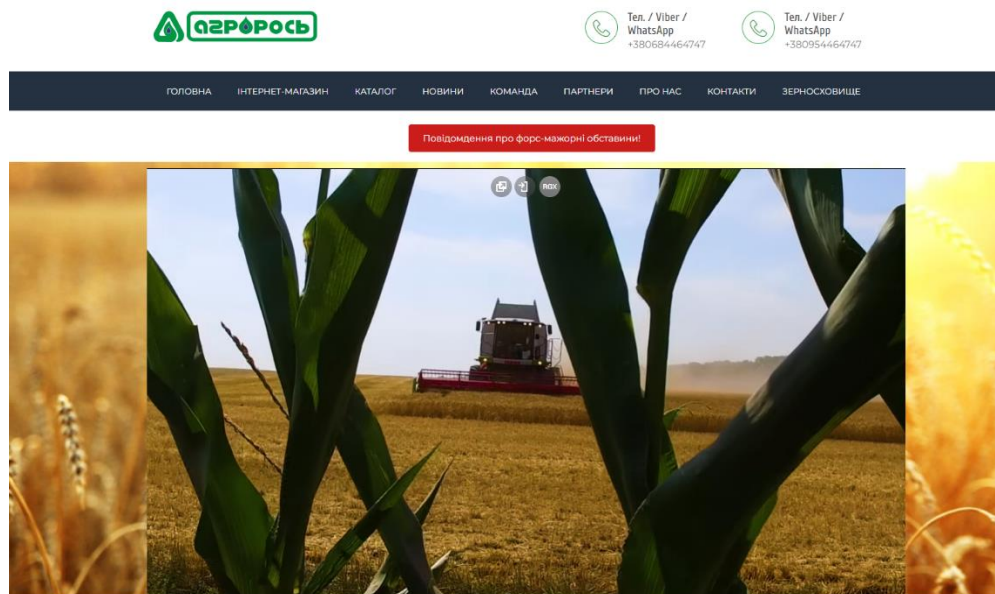


Рисунок 1.9 - Головна сторінка сайту «Агрорось»

1.7.5 Зведена характеристика сайтів підприємств

Отже, після докладного аналізу трьох сайтів, ми можемо зробити висновки, що відображаються у зведеній таблиці 1.1. Створення такої таблиці дозволить нам систематизувати та узагальнити отриману інформацію для більшої наглядності та зрозуміння.

Табл. 1.1. Порівняння сайтів

Параметр	Агрорось	Альфа Смарт Агро	ЛНЗ Дефенда
Дизайн	Традиційний, з підкресленням фаховості	Мінімалістичний, з акцентом на продукцію	Класичний, з акцентом на професіоналізм
Інформаційний наповнення	Повне, з можливістю додаткового уточнення	Середнє, більше акценту на продукцію та послуги	Основне, інформація мінімально необхідна
Навігація	Покращена, з хорошою структурою	Проста та зрозуміла	Стандартна, деякі сторінки можуть бути складними
Функціонал	Базовий, без додаткових опцій	Середній, з можливістю онлайн замовлень	Мінімальний, основні функції присутні
Присутність блогу	Відсутній	Відсутній	Відсутній
Онлайн замовлення	Немає	Присутнє	Немає
Форма зворотнього зв'язку	Присутня	Присутня	Присутня
Пошукова система	Присутня	Відсутня	Відсутня

Ця таблиця надає більш детальне порівняння веб-сайтів цих компаній за різними параметрами, включаючи функціональність, а також переваги та недоліки для кожного сайту.

Після детального порівняння веб-сайтів компаній Агрорось, Альфа Смарт Агро та ЛНЗ Дефенда можна зробити наступні висновки:

Веб-сайт Агрорось вражає своєю традиційною естетикою та повнотою інформації, проте він не має ряду сучасних функцій, таких як онлайн замовлення та блог. Альфа Смарт Агро, хоча й має мінімалістичний дизайн, але компенсує це зручною навігацією та присутністю онлайн замовлень і форм зворотного зв'язку. З іншого боку, веб-сайт ЛНЗ Дефенда пропонує класичний дизайн із загальною інформацією, але недостатньою функціональністю.

Отже, кожен веб-сайт має свої переваги та недоліки, і вибір найбільш підходящого залежить від потреб користувача. Наприклад, якщо важлива повна інформація, то Агрорось може бути кращим варіантом. У той же час, для тих, хто шукає зручність у використанні, Альфа Смарт Агро може бути більш

привабливим варіантом завдяки своїм онлайн можливостям.

1.8 Обґрунтування доцільності проектування й розроблення сайту підтримки діяльності компанії

У пункті 1.5 вказано, що менеджер з продажу виявився ключовою фігурою на різних рівнях декомпозиції бізнес-процесів на підприємстві. Він бере участь у всіх етапах від пошуку клієнтів до виконання замовлень та забезпечення високого рівня задоволеності клієнтів, тому заради спрощення роботи ключового персоналу та збільшення його функціоналу буде виконана робота.

На основі вищесказаного, розробка та проектування сайту для компанії є цілком обґрунтованою.

1.9 Концептуальна модель системи

Розробка веб-сайту для компанії "Укравіт" ґрунтується на комплексному підході до вивчення її потреб, цілей та специфіки діяльності. Концептуальна модель цього проекту базується на відомостях, зібраних під час аналізу вимог клієнта, а також на кращих практиках у галузі розробки веб-додатків.

Основними компонентами концептуальної моделі є:

1. Аналіз вимог користувача: Проведений детальний аналіз потреб та очікувань користувачів дозволив визначити ключові функціональні вимоги до веб-сайту. Це включає в себе можливість швидкого доступу до інформації про продукцію компанії, зручну систему замовлення та оплати, а також інтерактивні можливості зворотнього зв'язку з клієнтами.
2. Дизайн та користувацький досвід: Основна увага в концептуальній моделі приділяється створенню естетично привабливого та інтуїтивно зрозумілого дизайну. Планується розробка зручного інтерфейсу

користувача, який би максимально спрощував процес навігації по сайту та забезпечував приємний користувацький досвід.

3. Технічна архітектура: Основою технічної архітектури веб-сайту буде використання сучасних веб-технологій та платформ. Планується розробка сайту з використанням відкритих стандартів, що дозволить забезпечити його сумісність з різноманітними пристроями та браузерами.
4. Безпека та захист інформації: Концептуальна модель передбачає використання сучасних засобів захисту даних та інформаційної безпеки. Планується застосування шифрування та заходів захисту від несанкціонованого доступу до конфіденційної інформації клієнтів та компанії.

Концептуальна модель визначає загальну стратегію розробки веб-сайту для компанії "Укравіт" та встановлює основні принципи, які будуть використовуватися на кожному етапі проекту.

1.10 Постановка задачі

1.10.1. Призначення та цілі створення системи

Користувачами системи, що розробляється, виступають **адміністратор, менеджер, клієнт**.

Головною задачею системи є підвищення ефективності та полегшення роботи персоналу, полегшення роботи зі звітністю та обліком. У функціонал покладена зручність у використанні.

Завдяки даній системі відпаде необхідність тримати при собі у менеджера десятки каталогів та сотні документів ціноутворення підв'язаних під валюту .

1.10.2.1. Вимоги до апаратного забезпечення

Розробка веб-сайту для компанії "Укравіт" передбачає певні вимоги до апаратного забезпечення для забезпечення оптимальної продуктивності та надійності функціонування. Основними вимогами є:

1. Серверна інфраструктура: Для забезпечення роботи веб-сайту потрібен серверний комп'ютер або хостинговий сервіс. Рекомендується використання серверів з достатньою кількістю оперативної пам'яті та потужними процесорами для швидкого відгуку на запити користувачів.

2. Мережеве з'єднання: Стабільне та швидке інтернет-з'єднання є важливим аспектом забезпечення доступності веб-сайту для користувачів. Рекомендується використання високошвидкісного Інтернет-з'єднання для максимальної продуктивності та зниження часу завантаження сторінок.

3. Клієнтські пристрої: Веб-сайт повинен бути оптимізований для перегляду на різних клієнтських пристроях, таких як комп'ютери, планшети та смартфони. Це передбачає розробку адаптивного дизайну, який забезпечить коректне відображення контенту на різних розмірах екранів.

4. Засоби зберігання даних: Для зберігання даних веб-сайту, таких як інформація про клієнтів, замовлення та інші дані, потрібні надійні засоби зберігання, такі як серверні бази даних або хмарні сховища.

5. Засоби безпеки: Для захисту веб-сайту від вразливостей та атак з боку зловмисників необхідно використовувати засоби захисту, такі як файрволи, антивірусне програмне забезпечення та системи виявлення вторгнень.

Враховуючи ці вимоги до апаратного забезпечення, розроблений веб-сайт для компанії "Укравіт" буде забезпечувати надійну та продуктивну роботу для користувачів.

1.10.2.2. Вимоги до програмного забезпечення

Розробка веб-сайту для компанії "Укравіт" передбачає використання програмного забезпечення, яке забезпечить ефективне функціонування та надійність веб-сайту. Для досягнення цих цілей необхідно вибрати та налагодити такі програмні компоненти, як веб-сервер, система управління контентом (CMS), база даних та засоби розробки.

Веб-сервер є основним компонентом, який обробляє запити користувачів та відповідає на них, надаючи доступ до веб-сторінок та інших ресурсів. Вибір веб-сервера буде залежати від потреб проекту та специфікацій середовища

розгортання.

Система управління контентом (CMS) дозволяє зручно керувати контентом веб-сайту, включаючи додавання, редагування та видалення інформації. Рекомендується використання відомих та надійних CMS, які надають розширені можливості з управління контентом та розвитку.

База даних використовується для зберігання інформації, яка використовується веб-сайтом, такої як дані користувачів, продуктів та замовлень. Вибір системи управління базами даних (СУБД) буде залежати від потреб проекту та особливостей даних.

Засоби розробки використовуються для написання, відлагодження та тестування коду веб-сайту. Вибір інструментів розробки також буде залежати від ваших уподобань та досвіду роботи з ними.

1.11 Розрахунок економічного ефекту від впровадження сайту для компанії Ukravit

В даному розділі буде проведено розрахунок економічного ефекту від впровадження сайту для компанії Ukravit. Ми врахуємо вартість усіх витрат, пов'язаних з розробкою і підтримкою сайту, а також оцінюємо економічний ефект від його використання.

Таблиця 1.2. Розрахунок вартості розробки сайту

№ п/п	Назва витрати	Один. вимір.	Кількість	Ціна, грн.	Вартість, грн.
1	Розробка дизайну сайту	Шт.	1	25000	25000
2	Програмування і тестування	Шт.	1	30000	30000
3	Хостинг і домен (на рік)	Шт.	1	5000	5000
4	SSL сертифікат (на рік)	Шт.	1	1500	1500
Всього					61500

Таблиця 1.3. Розрахунок вартості підтримки сайту

№ п/п	Назва витрати	Один. вимір.	Кількість	Ціна, грн.	Вартість, грн.
1	Підтримка і оновлення контенту	Міс.	12	2000	24000
2	Технічна підтримка	Міс.	12	1000	12000
Всього					36000

Впровадження сайту може призвести до наступних економічних ефектів:

Збільшення кількості клієнтів завдяки онлайн-замовленням.

Зменшення витрат на маркетинг завдяки ефективній онлайн-рекламі.

Підвищення ефективності роботи персоналу завдяки автоматизації обробки замовлень.

Збільшення кількості клієнтів

До впровадження сайту компанія обслуговувала 100 клієнтів на місяць. Після впровадження сайту кількість клієнтів зросте приблизно до 150 на місяць. Середній дохід з одного клієнта становить 20000 грн.

Дохід до впровадження сайту: 100 клієнтів * 20000 грн = 2 000 000 грн

Дохід після впровадження сайту: 150 клієнтів * 20000 грн = 3 000 000 грн

Збільшення доходу: 3 000 000 грн - 2 000 000 грн = 1 000 000 грн на місяць

Зменшення витрат на маркетинг

Раніше на маркетинг витрачалось 5000 грн на місяць. Завдяки сайту витрати знизились до 3000 грн на місяць.

Економія на маркетингу: 5000 грн - 3000 грн = 2000 грн на місяць

Підвищення ефективності роботи персоналу

Завдяки автоматизації обробки замовлень, час, витрачений на обробку одного замовлення, зменшився з 10 хвилин до 5 хвилин. Це дозволяє обслуговувати більше клієнтів за той же час.

Висновок

Загальна сума витрат на розробку і підтримку сайту:

Розробка сайту: 61500 грн

Підтримка сайту за рік: 36000 грн

Загальна сума витрат за рік: 97500 грн

Загальна сума економічного ефекту за рік:

Збільшення доходу: 1 000 000 грн * 12 = 12 000 000 грн

Економія на маркетингу: 2000 грн * 12 = 24000 грн

Загальний економічний ефект: 12 000 000 грн + 240 000 грн = 12240000 грн

Таким чином, впровадження сайту для компанії Ukravit дозволить отримати чистий економічний ефект у розмірі:

Чистий економічний ефект: 12240000 грн - 97500 грн = 12142500 грн на рік

Впровадження сайту є економічно вигідним і сприяє значному збільшенню доходів компанії та ефективності роботи.

1.12 Висновок до розділу 1

У результаті аналізу загальної характеристики ТОВ «Укравіт», включаючи структуру підприємства та нинішній рівень комп'ютеризації, виникла необхідність у створенні нового, сучасного та функціонального веб-ресурсу для Ukravit, з використанням передових інструментів розробки.

Спостереження за конкурентами, аналіз їхніх веб-ресурсів, а також порівняльний огляд аналогів сайтів для реалізації проекту, підтвердили важливість створення нового сайту для Ukravit. Існуючий сайт не відповідає сучасним вимогам та не задовольняє потреб клієнтів, тому його заміна на новий є критичною для подальшого успішного розвитку компанії.

Розрахунок економічного ефекту показав, що впровадження нового сайту призведе до значного зростання доходу компанії та ефективнішого використання ресурсів. Такий крок є стратегічно важливим для Ukravit, оскільки це дозволить підвищити конкурентоспроможність компанії та зміцнити її лідерські позиції на ринку.

Отже, проектування та розроблення нового сайту підтримки діяльності компанії Ukravit є не лише логічним кроком у відповідь на потреби сучасного ринку, а й стратегічно важливим рішенням для подальшого успішного розвитку бізнесу.

2. ТЕХНІЧНЕ ЗАВДАННЯ НА ПРОЕКТУВАННЯ

2.1. Загальні положення.

Найменування системи: «веб сайт для компанії “Укравіт”»

Результати робіт зі створення системи оформлюються згідно з вимогами ДСТУ на відповідні етапи розробки. Порядок оформлення і передачі результатів у даному випадку визначається змістом і календарним планом виконання розробки.

У випадку необхідності на наступних стадіях робіт по створенню системи окремі положення можуть уточнюватися і розвиватися.

2.2. Призначення і цілі створення системи

Розробка веб сайту для компанії "Ukravit" має на меті досягнення кількох ключових цілей, які сприятимуть розвитку бізнесу, покращенню взаємодії з клієнтами та підвищенню рівня обслуговування. Основні цілі створення веб сайту включають наступне:

1. Підвищення впізнаваності бренду: Веб сайт служить важливим інструментом для просування бренду "Ukravit" на ринку. Завдяки професійному дизайну та зручній навігації сайт буде привертати увагу потенційних клієнтів та партнерів, тим самим зміцнюючи імідж компанії.

2. Продаж продукції: Однією з головних функцій веб сайту є створення зручного інтернет-магазину, де користувачі зможуть ознайомитися з асортиментом товарів компанії, прочитати їх описи, переглянути фотографії та зробити замовлення онлайн. Це дозволить розширити ринок збуту і збільшити обсяги продажів.

3. Інформаційна підтримка клієнтів: Веб сайт буде включати розділ з блогом та статтями, де будуть публікуватися корисні матеріали, поради, новини компанії та інша актуальна інформація. Це допоможе клієнтам отримувати

необхідну інформацію щодо продуктів, їх використання та догляду, а також бути в курсі останніх новин та акцій.

4. Поліпшення комунікації з клієнтами: За допомогою веб сайту клієнти зможуть легко зв'язатися з представниками компанії, отримати консультації, залишити відгуки та пропозиції. Наявність контактної форми, онлайн-чату та інших засобів зв'язку підвищить рівень обслуговування та задоволення клієнтів.

5. Оптимізація бізнес-процесів: Інтеграція веб сайту з внутрішніми системами управління компанії, такими як CRM та ERP, дозволить автоматизувати багато бізнес-процесів, що призведе до зменшення витрат часу і ресурсів на обробку замовлень, ведення обліку та інших операцій.

6. Залучення нових клієнтів та партнерів: Завдяки добре оптимізованому веб сайту, який буде відповідати вимогам пошукових систем, компанія "Ukravit" зможе залучати більше нових клієнтів та партнерів. SEO-оптимізація і контент-маркетинг сприятимуть підвищенню видимості сайту в Інтернеті.

7. Аналіз та покращення маркетингових стратегій: Веб сайт надасть можливість збору та аналізу даних про поведінку користувачів, їхні уподобання та потреби. Це дозволить більш ефективно планувати маркетингові кампанії, адаптувати пропозиції під вимоги ринку та підвищувати конверсію.

Загалом, створення веб сайту для компанії "Ukravit" є стратегічно важливим кроком, що сприятиме підвищенню конкурентоспроможності, покращенню якості обслуговування та збільшенню доходів компанії.

2.3. Характеристика об'єкта автоматизації.

Об'єкт автоматизації - це компанія "Ukravit", яка є однією з провідних компаній на ринку засобів захисту рослин та мікродобрив в Україні. Основною метою автоматизації є створення ефективного, інтерактивного та

функціонального веб сайту, який включатиме інтернет-магазин, блог та розділи зі статтями. Для успішної реалізації проекту необхідно врахувати специфіку діяльності компанії, її продукцію та потреби цільової аудиторії.

Основні компоненти об'єкта автоматизації:

1. Продукція:

- Засоби захисту рослин.
- Мікродобрива.
- Інші агрохімікати.

2. Цільова аудиторія:

- Фермери та агрономи.
- Великі агрохолдинги.
- Роздрібні покупці.
- Партнери та дистриб'ютори.

3. Основні функціональні вимоги до веб сайту:

• Каталог продукції: Детальний опис товарів, категорії, можливість фільтрації та пошуку.

• Інтернет-магазин: Кошик покупця, можливість замовлення товарів онлайн, різні способи оплати та доставки.

• Блог: Регулярні публікації на тему агротехнологій, новин компанії, порад щодо використання продукції.

• Статті та навчальні матеріали: Глибокі аналітичні статті, наукові матеріали, керівництва та рекомендації.

• Зворотний зв'язок: Форма для зворотного зв'язку, онлайн-чат, контактна інформація.

• Особистий кабінет: Можливість реєстрації та авторизації користувачів, збереження історії замовлень, персональні налаштування.

4. Технічні вимоги:

• Система управління контентом (CMS): Для зручного управління контентом сайту.

• Безпека: Захист персональних даних користувачів, захист від кібератак.

- Сумісність: Підтримка різних браузерів та мобільних пристроїв.
- Швидкість завантаження: Оптимізація сайту для швидкого завантаження сторінок.
- SEO-оптимізація: Адаптація контенту та структури сайту під вимоги пошукових систем для покращення видимості в Інтернеті.

2.4. Вимоги до системи

2.4.1. Вимоги до системи в цілому

2.4.1.1. Вимоги до структури і функціонування системи

Система повинна складатися з двох підсистем: підсистема магазину (фронтенд), яка доступна для всіх користувачів. та підсистема адміністрування сайту (бекенд), яка доступна тільки для адміністратора і надає можливості редагування всієї інформації на сайті.

Вся система повинна бути побудована за рахунок модульної системи проектування для забезпечення гнучкості при модернізації системи в майбутньому.

Інтерфейси підсистеми повинні буди розроблені з врахуванням ергономічних і естетичних вимог. Інтерфейс системи повинен складатися з блоків, які можна переміщувати.

2.4.1.2. Вимоги до чисельності і кваліфікації персоналу

2.4.1.2.1. Персонал, який використовує сайт повинен дотримуватися наступних вимог:

- пройти навчання і отримати навички роботи на ПК;
- дотримуватись технологічних інструкцій при роботі з системою;
- дотримуватись умов експлуатації ПК у відповідності з інструкціями по експлуатації;
- дотримуватись правил зберігання інформації і організації дублів;
- дотримуватись техніки безпеки при роботі на ПК.

2.4.1.3. Показники призначення

2.4.1.3.1. У відповідності з п. 2.2.1.1. показники призначення повинні характеризувати ступінь та якість автоматизації планової, інформаційно-облікової й управлінської діяльності персоналу підприємства. Перелік та допустимі значення показників, при яких зберігається цільове призначення системи, повинні бути визначені на стадії техноробочого проекту.

2.4.1.4. Вимоги до надійності

2.4.1.4.1 Система є багатофункціональною. Оцінка надійності виконується по кожній функції окремо. Основними показниками надійності являються:

L_i – ймовірність безвідмовного виконання задачі в заданий термін;

K_G – коефіцієнт готовності;

T_B – середній час відновлення;

$L_i \geq 0,95$, $K_G \geq 0,95$, $T_B \leq 1$ год.

2.4.1.4.3. Комплекс технічних засобів повинен передбачати: блокове компонування апаратури і простоту заміни зіпсованих блоків.

Для забезпечення надійності програмного та інформаційного забезпечення необхідно передбачити:

- використання об'єктно-орієнтованого програмування;
- використання системи контролю вхідної інформації з видачею користувачу повідомлень про виявлені помилки;
- застосування програм коригування при виявленні помилок в базі даних;
- створення резервних копій бази даних.

2.4.1.5. Вимоги безпеки

2.4.1.5.1. Технічні засоби при налагодженні, обслуговуванні та ремонті повинні відповідати:

- загальним вимогам безпеки за ГОСТ 12.2.003-74, ГОСТ 12.3.002-75, ГОСТ 25861-83;
- в розділі електробезпеки – ГОСТ 12.1.019-79, ГОСТ 12.1.030-81, ГОСТ 12.1.038-82, ГОСТ 12.2.007-75;
- в розділі пожежобезпеки – ГОСТ 12.1.004-91, СН –01-87.

2.4.1.5.2. Всі елементи системи, що знаходяться під напругою, повинні мати захист від випадкового доторкання, а самі технічні засоби мати захисне заземлення у відповідності з ГОСТ 12.1.030-81.

2.4.1.5.3. При встановленні, ремонті, технічному обслуговуванні та експлуатації комплексу повинні дотримуватись "Правила влаштування електрообладнання".

2.4.1.5.4. Обслуговуючий персонал повинен мати посвідчення на право експлуатації комплексу і зобов'язаний пройти інструктаж з техніки безпеки при роботі з електроустаткуваннями, включеними в мережу 220V через розподільчий трансформатор.

2.4.1.5.5. Технічні засоби повинні бути встановлені так, щоб забезпечувалася їх безпечна експлуатація та технічне обслуговування.

2.4.1.5.6. Рівні шуму та звукового навантаження в місці розташування персоналу не повинні перевищувати значень, встановлених ГОСТ 12.1.003-83.

2.4.1.5.7. На робочому місці користувача системи повинні бути обмежені небезпечні і шкідливі виробничі фактори за ГОСТ 12.0.003-74, в тому числі рівні електромагнітних, електростатичних полів відповідно за ГОСТ 12.1.006-84, 12.1.045-84.

2.4.1.5.8. Приміщення для системи бути обладнане контурами – шинами захисного заземлення.

2.4.1.6. Вимоги по ергономіці і технічній естетиці

2.4.1.6.1. Загальні ергономічні вимоги, що регламентують організацію робочого місця, взаємне розташування засобів відображення інформації, засобів зв'язку в межах робочого місця повинні відповідати ГОСТ 22269 –76.

Освітленість робочого місця повинна відповідати СН 512-78.

Засоби відображення повинні розміщуватися таким чином, щоб кут спостереження екрану складав не більше ніж 45 градусів, мінімальна відстань спостереження екрану - 0,3 м, рекомендована - 0,5 м.

Побудова програмного забезпечення повинна враховувати втомлюваність користувачів при роботі з дисплеєм.

2.4.1.7. Вимоги транспортабельності до сайту не висуваються, тому що він не є рухомою АС.

2.4.1.8. Вимоги до експлуатації, технічного обслуговування, ремонту і зберігання компонентів системи

2.4.1.8.1. Види обслуговування системи визначаються у відповідності з ГОСТ 18322-78. Загальні вимоги по експлуатації, технічному обслуговуванні і ремонту повинні відповідати ГОСТ 21552-84 та ГОСТ 20397-82.

2.4.1.8.2. Для розміщення технічних засобів підсистеми необхідні площі, визначені в СН 512-78. При цьому повинні дотримуватися вимоги, зазначені в експлуатаційній документації.

Напруга живлення технічних засобів системи 220/380 В змінного струму, частотою (50 +/- 1) Гц. Допустиме відхилення напруги від плюс 10 до мінус 15 %, допустима тривалість перерв в живленні не повинна перевищувати 0,001 с.

2.4.1.8.3. Кількість, кваліфікація і режими роботи обслуговуючого персоналу повинні відповідати рекомендаціям, зазначеним в технічних умовах і інструкціях з експлуатації окремих технічних засобів.

2.4.1.8.4. Склад, розміщення і умови зберігання компонентів та елементів технічних засобів системи визначається рекомендаціями, зазначеними в експлуатаційній документації на ці елементи.

2.4.1.8.5. Регламент обслуговування повинен відповідати їх рівню і умовам роботи, щоб у випадку відмови системи забезпечити роботу в аварійному режимі.

2.4.1.9. Вимоги до захисту інформації від несанкціонованого доступу

2.4.1.9.2. Кожен сеанс роботи з адміністративною частиною системи повинен розпочинатися з введення індивідуального паролю.

2.4.1.9.3. Система парольного захисту повинна мати власні засоби періодичної зміни паролів або використовувати стандартні засоби середовища розробки.

2.4.1.10. Вимоги до збереження інформації при аваріях

2.4.1.10.1. Повинні бути розроблені засоби збереження бази даних в архів після проведення її коригування і завантаження її з архіву у випадку її руйнування.

2.4.1.10.2. Архів і база даних повинні знаходитись на різних машинних носіях чи пристроях.

2.4.1.11. Вимоги до захисту від впливу зовнішніх діянь

2.4.1.11.1. Електрична складова електромагнітного поля в приміщеннях не повинна перевищувати $0,3 \text{ В/м}^2$ в діапазоні частот від 0,15 до 300 МГц у відповідності з ГОСТ 16325-88. Для захисту від впливу електромагнітних полів та індустриальних завод необхідно передбачити різноманітні екрани та фільтри у відповідності з ГОСТ 23450-79.

2.4.1.11.2. Засоби, які виключають вплив шкідливих факторів на функціонування комплексу технічних засобів, повинні бути запроектовані у відповідності з СН 512-78. Обчислювальні засоби за стійкістю до зовнішніх впливів повинні відповідати ГОСТ 21552-84.

2.4.1.12. Вимоги до патентної чистоти

2.4.1.12.1. Технічні рішення при проектуванні сайту і алгоритми управління повинні бути патентно чистими у відношенні країн СНД.

2.4.1.13. Вимоги по стандартизації і уніфікації

2.4.1.13.1. В системі кодування інформації необхідно проводити за світовим класифікатором і стандартом

2.4.2. Вимоги до функцій

2.4.2.1. Перелік функцій з зазначенням вхідної та вихідної інформації представлено в таблиці 2.1.

Таблиця 2.1 Функції системи

Назва функції	Вхідна інформація	Вихідна інформація
Перегляд каталогу товарів	Ідентифікатор категорії	Список товарів вибраної категорії
Перегляд товару	Ідентифікатор товару	Інформація про товар
Фільтрація товарів	Властивості товару	Список товарів
Додавання товару в кошик замовлень	Ідентифікатор варіанту товару	Запис у сесію коду товару
Перегляд списку доданих у кошик товарів	Адреса кошику	Список товарів, доданих у кошик
Оформлення замовлення	Інформація про замовника, список замовлених товарів	Сформоване замовлення
Додавання коментарів до продукту	Коментар від користувача	Сформований коментар
Реєстрація на сайті	Інформація про користувача	Сформований акаунт
Ведення каталогу товарів	Перелік товарів магазину	Каталог товарів
Ведення списку замовлень	Перелік замовлень користувачів	Список замовлень
Ведення списку клієнтів	Перелік клієнтів	Список клієнтів
Ведення коментарів до товару	Перелік коментарів до товару	Список коментарів до товару
Ведення інформаційних сторінок	Інформація для сторінки	Список інформаційних сторінок

2.4.3. Вимоги до видів забезпечення

2.4.3.1. Вимоги до математичного забезпечення

2.4.3.1.1. Математичне забезпечення повинно мати достатню гнучкість і універсальність, що полягає у наступному:

- побудоване математичне забезпечення виконується за модульним принципом;
- кожний програмний модуль повинен бути відкритим;

- групування по модулях повинно забезпечити максимальну незалежність модуля.

2.4.3.1.2. Математичні методи та реалізуючі їх алгоритми повинні допускати можливість удосконалення на всіх стадіях створення та експлуатації системи.

2.4.3.2. Вимоги до інформаційного забезпечення

2.4.3.2.1. Інформаційне забезпечення системи повинно містити дані, достатні для виконання всіх функцій системи. Інформаційне забезпечення повинно гарантувати раціональну організацію зберігання інформації і доступу до неї.

Заповнення інформаційної бази покладається на замовника за методиками і формами, розробленими виконавцем системи.

Склад, структура і способи організації даних в системі можуть уточнюватись на етапі технічного проектування.

2.4.3.2.2. Необхідно передбачити захист даних від руйнувань при аваріях і порушеннях в енергоживленні системи – використання дубля бази даних.

2.4.3.3. Вимоги до лінгвістичного забезпечення

2.4.3.3.1. Для розробки програмного забезпечення (програмних засобів, що реалізують виконання функцій та забезпечують сервіс користувача) повинні використовуватися мови об'єктно-орієнтованого програмування, які забезпечують створення інтерфейсу користувача, а також мова обраної СКБД для здійснення доступу та маніпулювання даними.

2.4.3.3.2. Організація діалогу користувача з системою повинна будуватися на наборах меню і підказок, орієнтованих на виконання користувачем функції. Запити користувача до системи повинні вводитися переважно на природній мові.

2.4.3.4. Вимоги до програмного забезпечення (далі ПЗ)

2.4.3.4.1. Загальносистемне ПЗ повинно забезпечувати надійне і якісне виконання функціональних задач, покладених на сайт. До загальносистемного ПЗ належить:

- операційна система (далі ОС);

- система керування базою даних (далі СКБД).

2.4.3.4.2. Загальні вимоги до системного ПЗ можна сформулювати таким чином:

- повне задоволення потреб функціональних задач, покладених на систему;
- мінімальні вимоги до ресурсів технічних засобів;
- максимальне використання можливостей технічних засобів;
- максимальна швидкодія.

2.4.3.4.3. Вимоги до ОС:

•максимальне використання можливостей електронної обчислюваної машини (далі ЕОМ);

•максимальна швидкодія при забезпечені управління зовнішніми пристроями.

2.4.3.4.4. Вимоги до СКБД:

- максимальне задоволення потреб функціональних задач;
- ефективне управління базою даних потрібного об'єму і структури;
- надійність;
- швидкість виконання запитів користувачів.

2.4.3.4.5. Програмні засоби введення і виведення даних і ведення діалогу повинні забезпечувати:

•супроводження введення даних контролем і сигналізацією оператора про наявність помилок по реквізітам з можливістю їх виправлення під час введення даних;

•керований ПК діалог при введенні даних;

•виведення даних у відповідному вигляді (формі документа) за запитом користувача.

2.4.3.4.6. При розробці спеціального ПЗ необхідно виконувати наступні вимоги:

•розроблені програми повинні бути сумісними між собою і з загальносистемним ПЗ;

- ПЗ повинно бути виконане на засобах об'єктно-орієнтованого проектування;
- Забезпечити інтерфейс користувача у відповідності з стандартами User interface (далі – UI);
- повинна бути передбачена можливість розширення складу задач у відповідності з новими потребами;
- ПЗ не повинно залежати від типу зовнішніх пристроїв (принтерів, дисків, сканерів і т.д.);
- діалог з користувачем повинен проводитись за допомогою клавіатури або миші з поясненням виконуваних дій і можливістю отримання підказки.

2.4.3.5. Вимоги до технічного забезпечення

2.4.3.5.1. Технічні засоби системи повинні забезпечувати виконання функцій, перерахованих вище.

2.4.3.5.2. Засоби обчислювальної техніки і апаратура зв'язку повинні забезпечувати обмін інформації в об'ємах, приведених в п. 2.4.3.2. (вимоги до інформаційного забезпечення).

2.4.3.6. Вимоги до метрологічного забезпечення

2.4.3.6.1. Система не має вимірювальних каналів, вимірювального обладнання і приладів, тому вимоги до даного виду забезпечення не висуваються.

2.4.3.7. Вимоги до організаційного забезпечення

2.4.3.7.1. Організаційне забезпечення системи розробляється у відповідності з вимогами державного стандарту до автоматизованої системи.

2.4.3.7.2. При впровадженні системи не передбачається збільшення штатної чисельності підприємства.

Територіальне розміщення робочого місця, на якому буде встановлено систему, визначається замовником.

2.4.3.7.3. До організації функціонування системи пред'являються наступні вимоги:

- наказом директора визначається список співробітників, які матимуть доступ до системи (адміністративної частини);

- контроль і прийняття рішень при аварійних ситуаціях роботи системи здійснює відповідальний за систему.

Розробка веб-сайту для компанії "Ukravit" вимагає ретельного визначення вимог до системи, які забезпечать його ефективну роботу, безпеку та задоволення потреб користувачів. Ці вимоги можна поділити на функціональні та нефункціональні.

Функціональні вимоги

1. Управління контентом:

- Система управління контентом (CMS) повинна забезпечувати зручне додавання, редагування та видалення контенту (текстів, зображень, відео).

- Можливість створення та редагування блогу та статей.

- Управління каталогом продукції: додавання нових продуктів, редагування описів, завантаження зображень.

2. Інтернет-магазин:

- Каталог продукції з фільтрами для пошуку за категоріями, ціною, брендом тощо.

- Кошик покупця з можливістю редагування замовлень.

- Підтримка різних методів оплати (банківські картки, електронні платежі).

- Опції доставки: вибір способу доставки, обчислення вартості доставки..

3. Користувацький кабінет:

- Реєстрація та авторизація користувачів.

- Перегляд статусу поточних замовлень.

- Можливість редагування персональних даних та налаштувань.

4. Комунікація з користувачами:

- Форми зворотного зв'язку та контактна інформація.

- Онлайн-чат для консультацій у реальному часі.

- Підписка на новини та розсилки.

5. SEO та аналітика:

- Інтеграція з інструментами веб-аналітики (наприклад, Google Analytics).
- Підтримка SEO-оптимізації: метатеги, друзі для пошукових систем URL,

карта сайту.

Нефункціональні вимоги

6. Продуктивність:

- Швидке завантаження сторінок (до 2 секунд).
- Висока продуктивність навіть при великій кількості одночасних

користувачів.

7. Масштабованість:

• Можливість легко додавати нові функціональні можливості та розширювати контент.

• Підтримка великої кількості товарів та користувачів без зниження продуктивності.

8. Безпека:

- SSL шифрування для всіх сторінок..

9. Доступність:

• Підтримка роботи на різних пристроях (десктопи, планшети, смартфони).
 • Адаптивний дизайн, який забезпечує зручність використання на різних екранах.

10. Надійність:

- Висока надійність та мінімальний час простою.
- Регулярне резервне копіювання даних.
- Система моніторингу для своєчасного виявлення та усунення проблем.

11. Зручність використання (Usability):

• Інтуїтивно зрозумілий інтерфейс для користувачів різного рівня технічної підготовки.

- Зручна навігація та пошук по сайту.

- Відгуки та оцінки товарів від користувачів.

12. Технічні вимоги

- Серверна частина:

Використання надійного веб-сервера .

База даних (MySQL).

Серверні мови програмування (PHP).

Дотримання цих вимог забезпечить створення сучасного, функціонального та безпечного веб-сайту, який відповідатиме потребам компанії "Ukravit" та її клієнтів.

2.5. Склад і зміст робіт по створенню сайту.

Стадії створення системи і терміни виконання робіт наведені в таблиці 3.

Таблиця 2.2. Найменування робіт при створенні системи

№ п/п	Найменування робіт	Строки виконання робіт
1	Передпроектне дослідження об'єкта автоматизації	20.09.2020
2	Технічне завдання	15.10.2020
3	Технічний проект	20.11.2020
4	Оформлення документації	10.12.2020

2.6. Порядок контролю і приймання системи.

Система впроваджується на діючому підприємстві. Перед введенням в експлуатацію сайт має пройти приймальні випробування згідно з ДСТУ 3974-2000. Випробування для визначення працездатності та прийняття рішення про можливість введення сайту в дослідну експлуатацію проводяться розробниками разом із замовником. Програму випробувань складає розробник і затверджує замовник.

Здача в дослідну експлуатацію здійснюється на основі технічного завдання та інструкції користувача. За результатами дослідної експлуатації формується список доробок та рекомендовані строки їх виконання. Введення в експлуатацію сайту оформлюється актом здачі-прийому.

2.7. Вимоги до складу і змісту робіт із підготовки до введення системи в дію.

Для введення системи в дію замовник виконує ряд робіт з підготовки об'єкту:

- проводить укомплектування технічних засобів;
- організує навчання користувачів роботі на ПК і вивчення інструкції з експлуатації;
- проводить дослідну експлуатацію і вводить систему в дію.

2.8. Вимоги до документації.

На систему розробляється комплекс документації у складі: технічне завдання та технічний проект.

Документація на систему розробляється у відповідності з вимогами Державних стандартів серії 19 «Єдина система програмної документації» та серії 24 «Єдина система стандартів автоматизованих систем управління».

При розробленні технічного завдання на систему використано наступні документи:

- ДСТУ 3008-2015. Документація. Звіти у сфері науки і техніки. Структура та правила оформлювання;
- ДСТУ 3973–2000 Система розроблення та поставлення продукції на виробництво;
- ДСТУ Б В.2.5–82:2016 Електробезпека в будівлях і спорудах. Вимоги до захисних заходів від ураження електричним струмом.

3. Опис комплексу задач автоматизації

3.1. Інформаційне забезпечення системи.

Логічна модель бази даних відображає структуру інформації, що зберігається, та зв'язки між різними об'єктами бази даних. Тоді як фізична модель бази даних відображає конкретну реалізацію логічної моделі на рівні СУБД. Для проекту створення сайту для компанії "Укравіт" використовується система керування контентом TurboShop CMS.

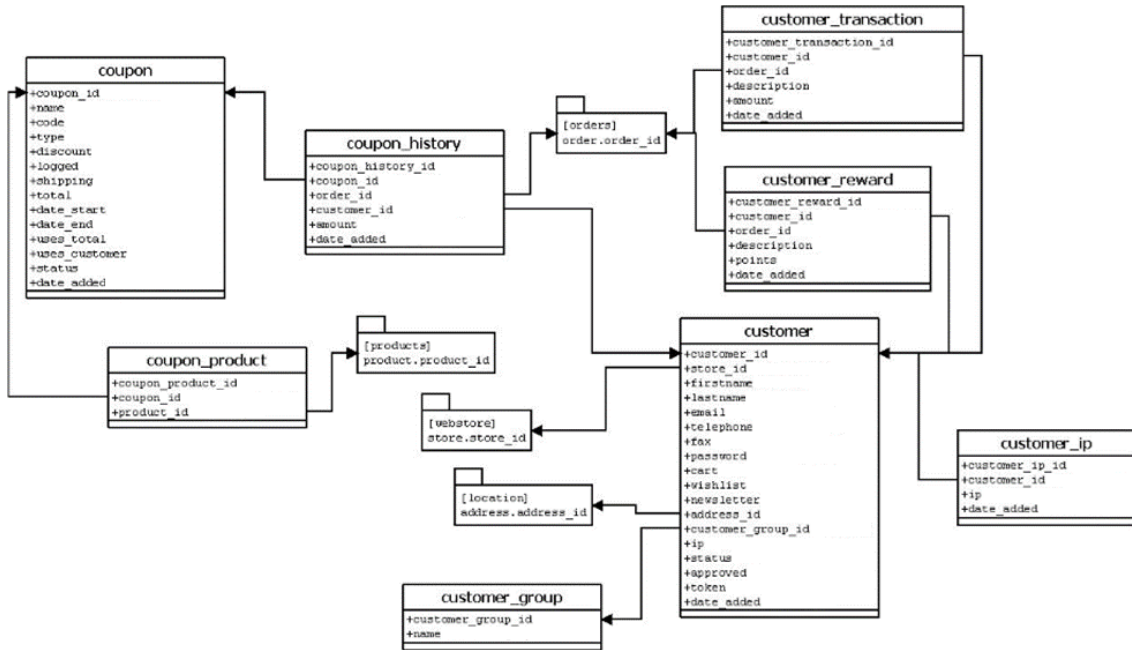


Рисунок 3.1 Структура логічної моделі даних клієнтів

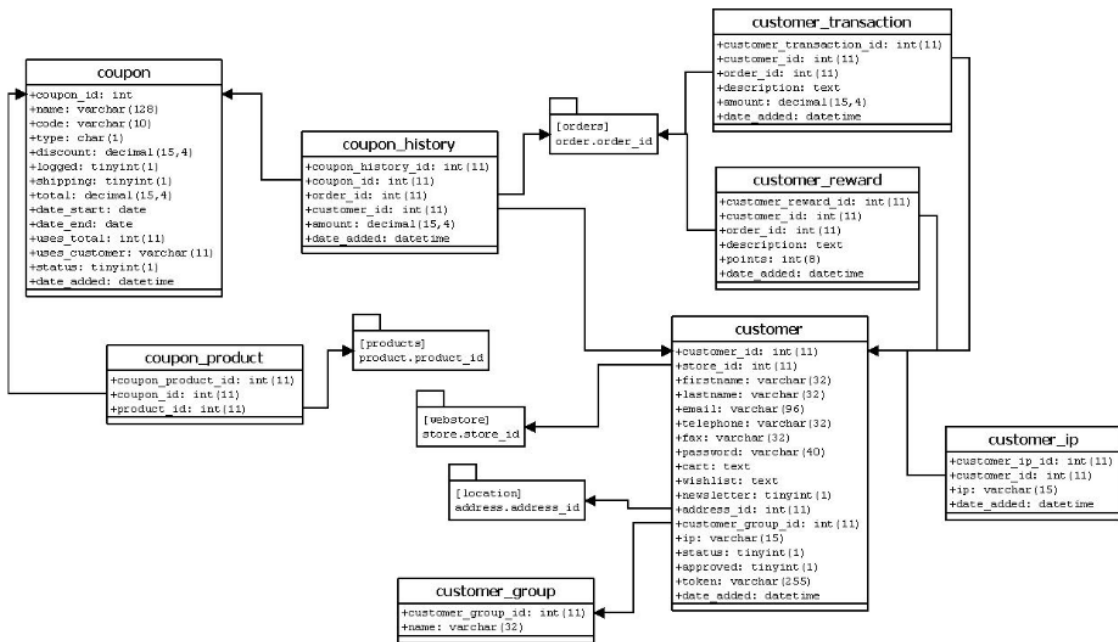


Рисунок 3.2 Структура фізичної моделі даних клієнтів

Для наглядного прикладу буде розібрана таблиці `t_article`, `t_categories` та `t_users`:

Таблиця `t_article` служить для зберігання статей.

Поля:

- `id`: Унікальний ідентифікатор статті.
- `category_id`: Ідентифікатор категорії, до якої належить стаття.
- `name`: Назва статті.
- `url`: URL статті.
- `meta_title`: Метатег заголовку для SEO.
- `meta_keywords`: Метатег ключових слів для SEO.
- `meta_description`: Метатег опису для SEO.
- `annotation`: Короткий опис статті.
- `text`: Повний текст статті.
- `visible`: Видимість статті (0 або 1).
- `date`: Дата створення статті.
- `last_modified`: Дата останньої модифікації.
- `position`: Позиція статті в списку.
- `image`: Зображення, пов'язане зі статтею.
- `views`: Кількість переглядів статті.
- `rate`: Рейтинг статті.
- `author`: Автор статті.

Таблиця `t_categories` служить для зберігання категорій статей.

Поля:

- `id`: Унікальний ідентифікатор категорії.
- `name`: Назва категорії.
- `url`: URL категорії.
- `meta_title`: Метатег заголовку для SEO.
- `meta_keywords`: Метатег ключових слів для SEO.
- `meta_description`: Метатег опису для SEO.

- position: Позиція категорії в списку.

Таблиця t_users служить для зберігання користувачів..

Поля:

- id: Унікальний ідентифікатор користувача.
- username: Ім'я користувача.
- password: Хеш паролю.
- email: Електронна адреса.
- created_at: Дата створення облікового запису.
- last_login: Дата останнього входу.

Конкретна реалізація логічної моделі на рівні СУБД. В нашому випадку це MySQL. Ось приклад фізичної моделі для таблиці t_articles:

```
CREATE TABLE `t_articles` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `category_id` int(11) NOT NULL DEFAULT '0',
  `name` varchar(500) COLLATE utf8mb4_unicode_ci NOT NULL,
  `url` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL,
  `meta_title` varchar(500) COLLATE utf8mb4_unicode_ci NOT NULL,
  `meta_keywords` varchar(500) COLLATE utf8mb4_unicode_ci NOT NULL,
  `meta_description` varchar(500) COLLATE utf8mb4_unicode_ci NOT NULL,
  `annotation` text COLLATE utf8mb4_unicode_ci NOT NULL,
  `text` longtext COLLATE utf8mb4_unicode_ci NOT NULL,
  `visible` tinyint(1) NOT NULL DEFAULT '0',
  `date` timestamp NULL DEFAULT NULL,
  `last_modified` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP
ON UPDATE CURRENT_TIMESTAMP,
  `position` int(11) NOT NULL DEFAULT '0',
  `image` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL,
  `views` int(10) unsigned NOT NULL DEFAULT '0',
  `rate` int(11) NOT NULL DEFAULT '0',
  `author` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL,
```

```
PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4  
COLLATE=utf8mb4_unicode_ci;
```

Структура бази даних визначає, як дані організовані та зв'язані між собою. Для цього прикладу структура бази даних складається з трьох основних таблиць: `t_articles`, `t_categories`, `t_users`. Кожна таблиця має свої специфічні поля та первинні ключі, що забезпечують унікальність записів.

1. `t_articles`

Зберігає дані про статті, включаючи їх вміст, метаінформацію для SEO, та інші атрибути, такі як видимість, кількість переглядів, рейтинг та авторство.

2. `t_categories`

Містить інформацію про категорії, до яких належать статті, що допомагає структурувати контент на сайті.

3. `t_users`

Містить дані про користувачів сайту, включаючи їх імена, паролі, електронні адреси, та іншу інформацію, що стосується їх облікових записів.

Основні таблиці бази даних:

Таблиця 3.1 *t_articles*

Назва	Тип	Значення
id	int(11)	Унікальний ідентифікатор
category_id	int(11)	Ідентифікатор категорії
name	varchar(500)	Назва статті
url	varchar(255)	URL статті
meta_title	varchar(500)	Метатег заголовку для SEO
meta_keywords	varchar(500)	Метатег ключових слів для SEO
meta_description	varchar(500)	Метатег опису для SEO
annotation	text	Короткий опис статті
text	longtext	Повний текст статті
visible	tinyint(1)	Видимість статті
date	timestamp	Дата створення статті
last_modified	timestamp	Дата останньої модифікації
position	int(11)	Позиція статті в списку
image	varchar(255)	Зображення статті
views	int(10)	Кількість переглядів статті
rate	int(11)	Рейтинг статті
author	varchar(255)	Автор статті

Таблиця 3.2 *t_categories*

Назва	Тип	Значення
id	int(11)	Унікальний ідентифікатор
name	varchar(255)	Назва категорії
url	varchar(255)	URL категорії
meta_title	varchar(500)	Метатег заголовку для SEO
meta_keywords	varchar(500)	Метатег ключових слів для SEO
meta_description	varchar(500)	Метатег опису для SEO
position	int(11)	Позиція категорії в списку

Таблиця 3.3 *t_users*

Назва	Тип	Значення
id	int(11)	Унікальний ідентифікатор
username	varchar(255)	Ім'я користувача
password	varchar(255)	Хеш паролю
email	varchar(255)	Електронна адреса
created_at	timestamp	Дата створення облікового запису
last_login	timestamp	Дата останнього входу

Таблиця 3.4 *t_articles_categories*

Назва	Тип	Значення
id	int(11)	Унікальний ідентифікатор
parent_id	int(11)	Ідентифікатор батьківської категорії
name	varchar(255)	Назва категорії
name_h1	varchar(255)	Назва заголовку H1
meta_title	varchar(255)	Метатег заголовку для SEO
meta_keywords	varchar(255)	Метатег ключових слів для SEO
meta_description	varchar(255)	Метатег опису для SEO
description	text	Опис категорії
url	varchar(255)	URL категорії
image	varchar(255)	Зображення категорії

Таблиця 3.5 *t_banners*

Назва	Тип	Значення
id	int(11)	Унікальний ідентифікатор
group_id	varchar(32)	Ідентифікатор групи
name	varchar(255)	Назва банера
position	int(11)	Позиція банера в списку
visible	tinyint(1)	Видимість банера
show_all_pages	tinyint(1)	Показувати на всіх сторінках
categories	varchar(200)	Категорії банера
pages	varchar(200)	Сторінки банера
brands	varchar(200)	Бренди банера
articles_categories	varchar(200)	Категорії статей банера

Таблиця 3.6 *t_banners_images*

Назва	Тип	Значення
id	int(11)	Унікальний ідентифікатор
banner_id	int(11)	Ідентифікатор банера
name	varchar(255)	Назва зображення
alt	varchar(255)	Альтернативний текст зображення
title	varchar(255)	Заголовок зображення
description	text	Опис зображення
url	varchar(255)	URL зображення
color	varchar(7)	Колір зображення
style	varchar(255)	Стиль зображення
code	varchar(255)	Код зображення
button	varchar(255)	Кнопка зображення

Таблиця 3.7 *t_callbacks*

Назва	Тип	Значення
id	int(11)	Унікальний ідентифікатор
name	varchar(255)	Ім'я клієнта
phone	varchar(255)	Номер телефону
date	datetime	Дата запиту зворотного дзвінка
status	tinyint(1)	Статус обробки

Таблиця 3.8 *t_comments*

Назва	Тип	Значення
id	int(11)	Унікальний ідентифікатор
name	varchar(255)	Ім'я користувача
text	text	Текст коментаря
article_id	int(11)	Ідентифікатор статті
date	datetime	Дата коментаря
approved	tinyint(1)	Схвалення коментаря

Таблиця 3.9 *t_currencies*

Назва	Тип	Значення
id	int(11)	Унікальний ідентифікатор
name	varchar(255)	Назва валюти
code	varchar(255)	Код валюти
rate	decimal(15,8)	Курс валюти
position	int(11)	Позиція валюти в списку
enabled	tinyint(1)	Активність валюти

Таблиця 3.10 *t_delivery*

Назва	Тип	Значення
id	int(11)	Унікальний ідентифікатор
name	varchar(255)	Назва способу доставки
price	decimal(10,2)	Вартість доставки
free_from	decimal(10,2)	Безкоштовна доставка від
description	text	Опис доставки
enabled	tinyint(1)	Активність способу доставки

Таблиця 3.11 *t_orders*

Назва	Тип	Значення
id	int(11)	Унікальний ідентифікатор
user_id	int(11)	Ідентифікатор користувача
delivery_id	int(11)	Ідентифікатор доставки
payment_id	int(11)	Ідентифікатор оплати
date	datetime	Дата замовлення
status_id	int(11)	Ідентифікатор статусу замовлення
total_price	decimal(10,2)	Загальна вартість замовлення

Продовження таблиці 3.11 *t_orders*

note	text	Примітка до замовлення
ip	varchar(255)	IP-адреса замовника

Таблиця 3.12 *t_callbacks*

Назва	Тип	Значення
id	int(11)	Унікальний ідентифікатор
name	varchar(255)	Ім'я клієнта
phone	varchar(255)	Номер телефону
date	datetime	Дата запиту зворотного дзвінка
status	tinyint(1)	Статус обробки

Таблиця 3.13 *t_comments*

Назва	Тип	Значення
id	int(11)	Унікальний ідентифікатор
name	varchar(255)	Ім'я користувача
text	text	Текст коментаря
article_id	int(11)	Ідентифікатор статті
date	datetime	Дата коментаря
approved	tinyint(1)	Схвалення коментаря

Таблиця 3.14 *t_currencies*

Назва	Тип	Значення
id	int(11)	Унікальний ідентифікатор
name	varchar(255)	Назва валюти
code	varchar(255)	Код валюти
rate	decimal(15,8)	Курс валюти
position	int(11)	Позиція валюти в списку
enabled	tinyint(1)	Активність валюти

Таблиця 3.15 *t_delivery*

Назва	Тип	Значення
id	int(11)	Унікальний ідентифікатор
name	varchar(255)	Назва способу доставки
price	decimal(10,2)	Вартість доставки
free_from	decimal(10,2)	Безкоштовна доставка від
description	text	Опис доставки
enabled	tinyint(1)	Активність способу доставки

Таблиця 3.16 *t_orders*

Назва	Тип	Значення
id	int(11)	Унікальний ідентифікатор
user_id	int(11)	Ідентифікатор користувача

delivery_id	int(11)	Ідентифікатор доставки
payment_id	int(11)	Ідентифікатор оплати
date	datetime	Дата замовлення
status_id	int(11)	Ідентифікатор статусу замовлення
total_price	decimal(10,2)	Загальна вартість замовлення
note	text	Примітка до замовлення
ip	varchar(255)	IP-адреса замовника

Ці таблиці містять інформацію про статичні сторінки, роботи з банерами та їх зображеннями, способи оплати, продукти, налаштування системи та групи користувачів, зворотні дзвінки, коментарі, валюти, способи доставки та замовлення, що є важливими для функціонування сайту компанії "Укравіт", повні фото таблиць зображені в додатку.

Приклад таблиці `t_options` в середовищі phpMyAdmin зображено на рисунку 3.3.

✓ Показано рядки 0 - 4 (всього 5, Запит виконувався 0.0004 секунди.)

```
SELECT * FROM `t_options`
```

Профілювання [[Порядкове редагування](#)] [[Редагувати](#)] [[Тлумачити SQL](#)] [[Створити PHP код](#)] [[Оновити](#)]

Показати все | Число рядків: 25 | Фільтрувати рядки: | Сортувати за ключем: Жодного

+ Параметри

	id	feature_id	value	translit	position
<input type="checkbox"/> Редагувати Копіювати Видалити	149	22	Бродіфакум, 0.25%	brodifakum025	149
<input type="checkbox"/> Редагувати Копіювати Видалити	150	23	Розчин	rozchin	150
<input type="checkbox"/> Редагувати Копіювати Видалити	151	22	Циперметрин, 25%	tsipermetrin25	151
<input type="checkbox"/> Редагувати Копіювати Видалити	152	23	Концентрат емульсії	konsentratemulsii	152
<input type="checkbox"/> Редагувати Копіювати Видалити	153	22	Хлорпіріфос, 50%, циперметрин, 5%	hlorpirifos50tsipermetrin5	153

[Перевірити все](#) | Вибрані: [Редагувати](#) [Копіювати](#) [Видалити](#) [Експорт](#)

Показати все | Число рядків: 25 | Фільтрувати рядки: | Сортувати за ключем: Жодного

Рисунок 3.3 – таблиця `t_options`

3.2 Алгоритмізація та реалізація комплексу задач автоматизації.

В ході виконання задачі було створено назву функціональностей сайту, найголовнішими з них є:

- реалізація кошику
- додавання товару
- Додавання категорії товару

- Реалізація відгуків
- підтримка блогу на сайті
- реалізація функціоналу профілю користувача

далі детальніше описано кожну з функціональностей з демонстрацією отриманих результатів реалізації кожної з них.

реалізація кошику

Був створений кошик товарів для оформлення замовлення на сайті .
Приклади реалізації Рисунок 3.4 – 3.5

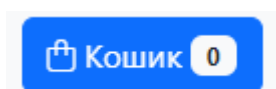


Рисунок 3.4 – кнопка кошику в шапці сайту

[Головна](#) / [Кошик](#)

Кошик (2)

Мінімальна сума замовлення 8 000 грн.

[Очистити кошик](#)

	Назва	Ціна	Кількість	Сума	
	Ципервіт	600.00 ₴	1 шт	600.00 ₴	
	Дезарал 5 літрів	1 600.00 ₴	1 шт	1 600.00 ₴	

Введіть код купона... [Застосувати купон](#)

Активувати Windows
Сума: 2 200.00 ₴
Доставка: 800.00 ₴
Разом: 3 000.00 ₴
Перейдіть до розділу "Налаштування Windows".

Рисунок 3.5 – приклад роботи кошику

Функціональності класу Cart в PHP забезпечують управління кошиком покупок на веб-сайті. Ось детальний опис кожної з них:

1. getCart()

Ця функція отримує поточний стан кошика користувача. Створює об'єкт `cart`, який містить інформацію про покупки, загальну суму кошика, ціну, кількість продуктів, знижки та вагу. Дії які виконуються:

- Перевіряє, чи є щось в сесії користувача.
- Отримує варіанти продуктів за ID з сесії.
- Формує масив товарів та їх кількість.
- Отримує інформацію про продукти та їх зображення.
- Розраховує загальну суму кошика, знижки та вагу.
- Якщо користувач авторизований, застосовує персональну знижку.
- Якщо є купон, перевіряє його та застосовує знижку за купоном.
- Повертає об'єкт `cart`.

`addItem($variantId, $amount = 1)`

Ця функція додає продукт до кошика. Додає вказаний продукт до кошика або збільшує його кількість. Дії які виконуються:

- Перевіряє, чи продукт вже є в кошику, та додає нову кількість.
- Отримує інформацію про варіант продукту за ID.
- Перевіряє наявність продукту на складі.
- Додає продукт до кошика з максимально допустимою кількістю (не більше наявності на складі).

`updateItem($variantId, $amount = 1)`

Ця функція оновлює кількість продукту в кошику. Дії які виконуються:

- Перевіряє та встановлює мінімальну кількість продукту (не менше 1).
- Отримує інформацію про варіант продукту за ID.
- Перевіряє наявність продукту на складі.
- Оновлює кількість продукту в кошику з максимально допустимою кількістю (не більше наявності на складі).

`deleteItem($variantId)`

Ця функція видаляє продукт з кошика. Дії які виконуються:

- Видаляє продукт з сесії користувача за вказаним ID.

`emptyCart()`

Ця функція очищає кошик. Очищає весь кошик та видаляє купон. Дії які виконуються:

- Видаляє кошик та купон з сесії користувача.

`applyCoupon($couponCode)`

Ця функція застосовує купон до кошика.Дії які виконуються:

Отримує інформацію про купон за кодом.

Перевіряє дійсність купона.

Якщо купон дійсний, зберігає його в сесії, інакше видаляє купон з сесії.

Загальна логіка роботи



Ці функції дозволяють користувачам додавати, оновлювати, видаляти продукти з кошика, очищати кошик та застосовувати купони на знижку. Дані про кошик зберігаються в сесії, що дозволяє зберігати стан кошика між різними запитами користувача.

Повний код функціональності класу `Cart` знаходиться в додатку `C`

Був створений скрипт який перевіряє, чи досягнуто мінімальної суми замовлення в кошику, і відповідно показує або приховує повідомлення про мінімальну суму замовлення. Скрипт зображено на рисунку 3.6 - 3.7 Якщо сума менше ніж потрібно, на рисунках рисунку 3.8 – 3.9 Якщо умова схвалена.

Кошик (2)

Мінімальна сума замовлення 8 000 грн.

Назва	Ціна	Кількість	Сума
 Ципервіт	600.00 ₴	1 шт	600.00 ₴
 Дезарал 5 літрів	1 600.00 ₴	1 шт	1 600.00 ₴

Введіть код купона...

Активувати купон
Сума: 2 200.00 ₴
Доставка: 800.00 ₴
Разом: 3 000.00 ₴

Рисунок 3.6 – Повідомлення про мінімальну суму

Капча*

32 + ? = 33



Введіть капчу

Мінімальна сума замовлення 8 000 грн.

Рисунок 3.7 – Повідомлення про мінімальну суму замість кнопки оформлення

Кошик (15)

[Очистити кошик](#)

Назва	Ціна	Кількість	Сума
 Ципервіт	600.00 ₴	5 шт	3 000.00 ₴
 Дезарал 5 літрів	1 600.00 ₴	10 шт	16 000.00 ₴

Введіть код купона... [Застосувати купон](#)

Сума: 19 000.00 ₴
Доставка: безкоштовно
Разом: 19 000.00 ₴

Рисунок 3.8 – Кошик без повідомлення про мінімальну суму

Капча*

5 + ? = 12

Введіть капчу

[Оформити замовлення >](#)

Рисунок 3.9 – Кнопка “Оформити замовлення” замість повідомлення про мінімальну суму

Ось детальний опис роботи скрипту:

Функція stringSumToNumber

Ця функція приймає рядок, який може містити текст, цифри та різні символи (крапки, коми, пробіли), і повертає число, що відповідає цьому рядку, очищене від зайвих символів.

Опис: Функція перетворює рядок на масив символів, проходить по кожному символу та фільтрує його, залишаючи тільки цифри і одну крапку (десятковий роздільник).

Дії:

- Розбиває рядок на масив символів.
- Створює новий масив для коректних символів.
- Проходить по кожному символу масиву:
- Пропускає пробіли.
- Обробляє коми та крапки: додає крапку тільки, якщо її ще немає в масиві.
- Перетворює символ на число, і якщо це не NaN, додає до нового масиву.
- Об'єднує новий масив у рядок та перетворює його на число.
- Повертає отримане число.
- Перевірка мінімальної суми замовлення

Опис: Скрипт отримує поточну суму замовлення та мінімальну дозволена суму замовлення. Якщо поточна сума замовлення дорівнює або перевищує мінімальну суму, скрипт приховує повідомлення про недостатню суму і активує кнопку для підтвердження замовлення.

Дії:

- Отримує всі елементи з класом `.min-sum-msg` для повідомлень про мінімальну суму замовлення.
- Отримує кнопку підтвердження замовлення за ID `#btn-submit`.
- Отримує загальну суму кошика з серверної частини (значення `$cart->total_cart`).
- Викликає функцію `stringSumToNumber` для перетворення мінімальної суми замовлення з рядка в число (значення `{$lang->min_order_sum}`).
- Перевіряє, чи поточна сума замовлення більша або дорівнює мінімальній сумі замовлення.

Якщо умова виконується:

Для кожного повідомлення про мінімальну суму (`.min-sum-msg`) видаляє клас `active`, що приховує повідомлення.

Видаляє клас `active` з кнопки підтвердження замовлення, що робить її активною для натискання.

Повний код скрипту знаходиться в додатку С

додавання товару

Додавання товару виконане через автоматизовану адмін панель яка створює нові записи в базі даних і додає їх на сайт продемонстровану на рисунку 3.10 .

Рисунок 3.10 – Фрагмент сторінки додавання товару

Функція `addProduct` приймає об'єкт продукту, обробляє його дані, генерує унікальний URL, додає опис продукту різними мовами та вставляє дані продукту до бази даних.

2. Перетворення продукту в масив

```
$product = (array) $product;
```

Ця строка коду перетворює об'єкт `$product` на асоціативний масив, щоб полегшити роботу з його властивостями.

Генерація URL продукту

```
if (empty($product['url'])) { $product['url'] = preg_replace("/[\s]+/ui", '-', $product['name']); $product['url'] = strtolower(preg_replace("/[^0-9a-zA-Z-]+/ui", "", $product['url'])); }
```

Якщо URL продукту не задано, генерується URL на основі імені продукту:

Замінюються всі пробіли на дефіси.

Видаляються всі символи, які не є літерами, цифрами або дефісами.

Приводиться до нижнього регістру.

Перевірка унікальності URL

```
while ($this->getProduct((string)$product['url'])) { if (preg_match('/(.+)_([0-9]+)$/', $product['url'], $parts)) { $product['url'] = $parts[1] . '_' . ($parts[2] + 1); } else { $product['url'] = $product['url'] . '_2'; } }
```

Цей блок коду перевіряє, чи існує продукт з таким же URL:

Якщо URL вже існує, додає суфікс (наприклад, `_2`, `_3` і т.д.) до кінця URL, щоб зробити його унікальним.

Цикл продовжується, поки не буде знайдено унікальний URL.

Повернення до об'єкта продукту

```
$product = (object) $product;
```

Після обробки URL, масив знову перетворюється на об'єкт.

Обробка багатомовного опису продукту

```
$result = $this->languages->getDescription($product, 'product'); if (!empty($result->data)) { $product = $result->data; }
```

Цей блок коду отримує опис продукту різними мовами, якщо вони є:

Викликається метод `getDescription` для отримання багатомовного опису.

Якщо є дані опису, вони додаються до об'єкта продукту.

Вставка продукту в базу даних

```
if ($this->db->query("INSERT INTO __products SET ?", $product)) { $id = $this->db->insertId(); $this->db->query("UPDATE __products SET last_modified=NOW(), position=id WHERE id=?", $id); if (!empty($result->description)) { $this->languages->actionDescription($id, $result->description, 'product'); } return $id; } else { return false; }
```

Цей блок коду вставляє продукт в базу даних:

Вставка продукту в таблицю `__products`.

Отримання ID щойно вставленого продукту.

Оновлення полів `last_modified` і `position` для нового продукту.

Якщо є багатомовний опис, він додається за допомогою методу `actionDescription`.

Повертається ID нового продукту або `false`, якщо вставка не вдалася.

Повний фрагмент коду оновлення та додавання товару знаходиться в

додатку С.

Відгуки

Відгуки реалізовано через четоди які будуть описані далі. Приклад відгуків наведено на рисунку 3.11 .

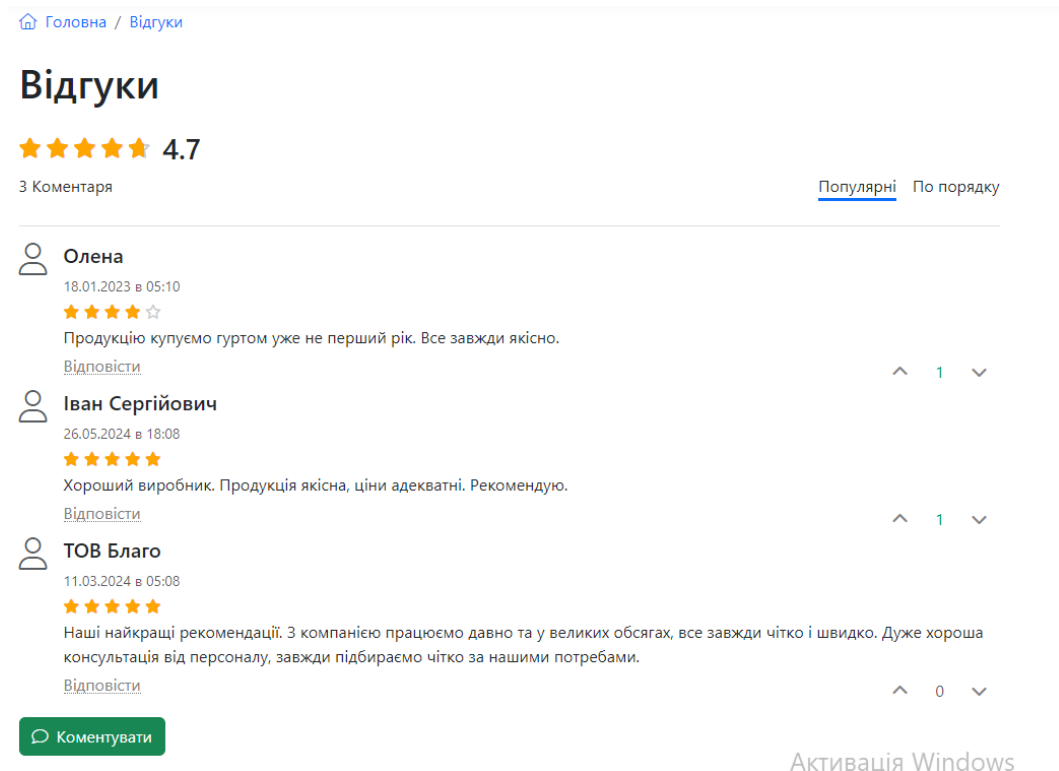


Рисунок 3.11 – Приклад відгуків на сторінці ‘Відгуки’

Метод `addComment`. Цей метод додає новий відгук до бази даних.

Вставка відгуку в базу даних:

Формується SQL-запит для вставки відгуку в таблицю `__comments` з поточною датою.

Виконується запит, якщо він не вдається, повертається `false`.

Отримання ID вставленого відгуку:

Після успішної вставки отримується ID новоствореного відгуку.

Перетворення відгуку в масив:

Відгук перетворюється з об'єкта в асоціативний масив для подальшої обробки.

Оновлення дати останньої зміни для пов'язаного об'єкта:

Якщо відгук схвалений (`approved == 1`) і має `object_id`, оновлюється дата

останньої зміни для пов'язаного об'єкта (блог, продукт, стаття або відгук).

Це робиться для відстеження активності вігуків на сайті.

Повернення ID відгуку:

Повертається ID новоствореного відгуку.

Метод `updateComment`. Цей метод оновлює існуючий відгук у базі даних.

Формування SQL-запиту для оновлення відгуку:

Використовується метод `placeholder` для підготовки SQL-запиту, який оновлює відгук з заданим ID.

Запит оновлює відповідні поля в таблиці `__comments`.

Виконання запиту:

Виконується SQL-запит для оновлення відгуку.

Повернення ID відгуку:

Повертається ID оновленого відгуку.

Метод `deleteComment`. Цей метод видаляє відгук з бази даних.

Перевірка наявності ID відгуку:

Якщо ID не пусте, виконується наступний код.

Отримання даних про відгук:

Виконується SQL-запит для отримання `object_id`, `type` та `approved` відгуку з заданим ID.

Оновлення дати останньої зміни для пов'язаного об'єкта (якщо відгук схвалений):

Якщо відгук схвалений (`approved == 1`), оновлюється дата останньої зміни для пов'язаного об'єкта (блог, продукт, стаття).

Формування SQL-запиту для видалення відгуку:

Використовується метод `placeholder` для підготовки SQL-запиту, який видаляє відгук з таблиці `__comments`.

Виконання запиту:

Виконується SQL-запит для видалення відгуку.

Ці методи забезпечують основні CRUD-операції (створення, читання, оновлення, видалення) для роботи з відгуками в системі. Вони також

підтримують оновлення дати останньої зміни для пов'язаних об'єктів, що дозволяє відстежувати активність на сайті.

Повний код цих методів знаходиться в додатку С

Блог

Був реалізований код який створений для роботи з постами в блозі. Один метод додає новий пост, а інший оновлює існуючий. Приклад поста в блозі рисунок 3.12 .

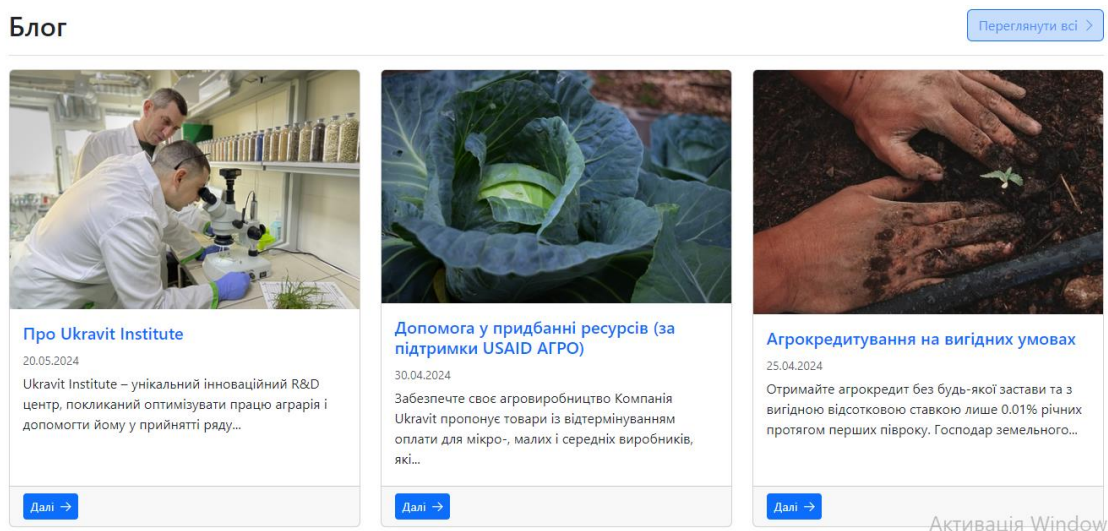


Рисунок 3.12 – Приклад постів блогу на головній сторінці сайту

Розглянемо кожен метод детально:

Метод `addPost`. Цей метод додає новий пост в базу даних.

Опис:

Перевірка наявності дати:

Якщо дата не задана, створюється строка `$dateQuery`, яка додає поточну дату і час до запиту.

Якщо дата задана, строка `$dateQuery` залишається пустою.

Перетворення поста в об'єкт:

Пост перетворюється в об'єкт для спрощення роботи з його властивостями.

Отримання багатомовного опису поста:

Викликається метод `getDescription` для отримання опису поста різними мовами.

Якщо дані опису не пусті, вони додаються до об'єкта поста.

Оновлення налаштування `lastModifyPosts`:

Зберігається поточна дата і час в налаштуваннях, щоб відстежувати останню модифікацію постів.

Формування та виконання SQL-запиту для вставки поста:

Формується SQL-запит для вставки поста в таблицю `__blog`, включаючи дату останньої зміни та (якщо необхідно) поточну дату.

Виконується запит, якщо він не вдається, повертається `false`.

Отримання ID вставленого поста:

Після успішної вставки отримується ID новоствореного поста.

Додавання багатомовного опису поста:

Якщо опис поста не пустий, він додається до бази даних за допомогою методу `actionDescription`.

Повернення ID поста:

Повертається ID новоствореного поста.

Метод `updatePost`

Цей метод оновлює існуючий пост в базі даних.

Опис:

Перетворення поста в об'єкт:

Пост перетворюється в об'єкт для спрощення роботи з його властивостями.

Отримання багатомовного опису поста:

Викликається метод `getDescription` для отримання опису поста різними мовами.

Якщо дані опису не пусті, вони додаються до об'єкта поста.

Оновлення налаштування `lastModifyPosts`:

Зберігається поточна дата і час в налаштуваннях, щоб відстежувати останню модифікацію постів.

Формування та виконання SQL-запиту для оновлення поста:

Формується SQL-запит для оновлення поста в таблиці `__blog`, включаючи дату останньої зміни.

Запит виконується для поста з відповідним ID.

Додавання багатомовного опису поста:

Якщо опис поста не пустий, він оновлюється в базі даних за допомогою методу `actionDescription`.

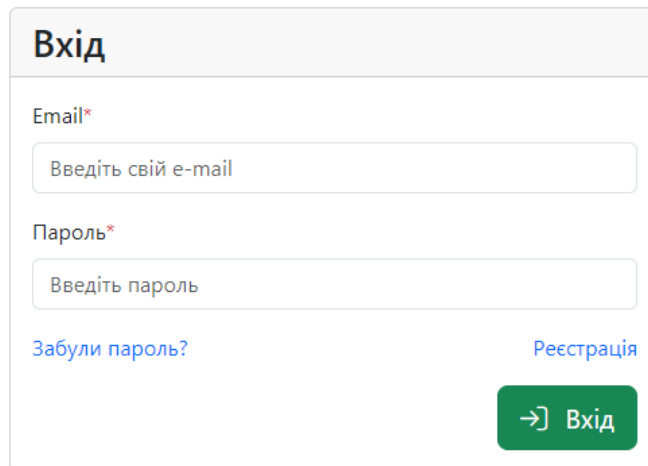
Повернення ID поста:

Повертається ID оновленого поста.

Повний код методів знаходиться в додатку С.

Профіль користувача

Був реалізований код який створений для роботи функціоналу користувачів для додавання, оновлення та видалення користувачів у системі. Приклад профілю рисунок 3.13 – 3.14 .



Вхід

Email*

Введіть свій e-mail

Пароль*

Введіть пароль

[Забули пароль?](#) [Реєстрація](#)

→ Вхід

Рисунок 3.13 – Вхід до профілю

example name

Ім'я*

Email*

Телефон

Адреса

[Змінити пароль](#)

Ваші змовлення

27.05.2024 [Заказ №15](#) чекає обробки

Рисунок 3.14 – Сторінка профілю користувача

Давайте розглянемо кожен метод детальніше:

Метод addUser

Цей метод відповідає за додавання нового користувача до бази даних.

Опис:

Хешування паролю: Якщо користувач вказав пароль, він хешується за допомогою функції md5 для забезпечення безпеки даних.

Перевірка унікальності email: Виконується SQL-запит для перевірки, чи вже існує користувач з такою ж електронною адресою в базі даних. Якщо користувач з таким email вже існує, метод повертає значення false.

Вставка користувача в базу даних: Виконується SQL-запит для вставки даних користувача в таблицю __users бази даних. Після успішного додавання користувача, метод повертає ID нового користувача.

Метод updateUser

Цей метод призначений для оновлення існуючого користувача в базі даних.

Опис:

Хешування паролю (за наявності): Якщо користувач вказав новий пароль,

він хешується за допомогою функції md5 для збереження безпеки даних.

Оновлення користувача в базі даних: Виконується SQL-запит для оновлення даних користувача в таблиці `__users` за його унікальним ідентифікатором (ID). Після успішного оновлення, метод повертає ID оновленого користувача.

Метод `deleteUser`

Цей метод відповідає за видалення користувача з бази даних.

Опис:

Видалення користувача з бази даних: Виконується SQL-запит для видалення користувача з таблиці `__users` за його унікальним ідентифікатором (ID). Якщо операція видалення пройшла успішно, метод повертає значення `true`, інакше - `false`.

Ці методи управління користувачами забезпечують надійну функціональність для додавання, оновлення та видалення користувачів у системі електронної комерції. Вони гарантують безпеку та ефективність управління користувачами платформи.

Повний код для методів додавання користувача знаходиться в додатку С:

Додавання категорії товару

У моїй дипломній роботі я досліджую процеси додавання та оновлення категорій у системі електронної комерції. Ці процеси є важливими для підтримки структурованого та організованого інтерфейсу для користувачів платформи.

Додавання Категорії

Метод `addCategory` відповідає за додавання нових категорій у систему. При додаванні категорії, система перевіряє унікальність URL категорії, генерує його, якщо він не був заданий, та забезпечує належне збереження мовних описів категорії. Після успішного додавання категорії в базу даних, метод повертає її унікальний ідентифікатор.

Оновлення Категорії

Метод `updateCategory`, зворотно, відповідає за оновлення існуючих категорій. Він приймає унікальний ідентифікатор категорії та нові дані для оновлення. Метод перевіряє оновлені дані, забезпечує їх належне збереження у базі даних та оновлення мовних описів, якщо вони були змінені. Після успішного оновлення, метод повертає ідентифікатор оновленої категорії.

Ці процеси є критичними для забезпечення структурованості та доступності товарів на платформі електронної комерції, а також для покращення користувацького досвіду покупців.

Додавання категорії товару виконане через автоматизовану адмін панель яка створює нові записи в базі даних і додає їх на сайт .Створення категорій продемонстровано на рисунку 3.15 – 1.16 .

Додавання категорії

Назва

Заголовок H1

URL Коренева категорія

Активність

Рекомендована

Зображення

Піктограма

Код іконки

Мета-дані

Meta-title (0)

Meta-description (0)

Meta-keywords

Рисунок 3.15 – Приклад додавання категорій на сайт для товарів

Промислові товари ^	3
Гербіциди	1
Фунгіциди	1
Ад'юванти	1
Приватний сектор ^	1
Родентициди	1
Професійні деззасоби	3
☰ Усі товари	

Рисунок 3.16 – Приклад категорій товару на сайті

Повний код реалізації додавання категорій знаходиться в додатку С :

3.3. Інструкція користувача

Для входу на сайт необхідно увійти в систему, набравши в браузері адресу ukraivit.pp.ua можна як з приставкою http:// все одно переадресує на https з ssl протоколом . Після цього буде відображена головна сторінка сайту , зображено на рисунку 3.17

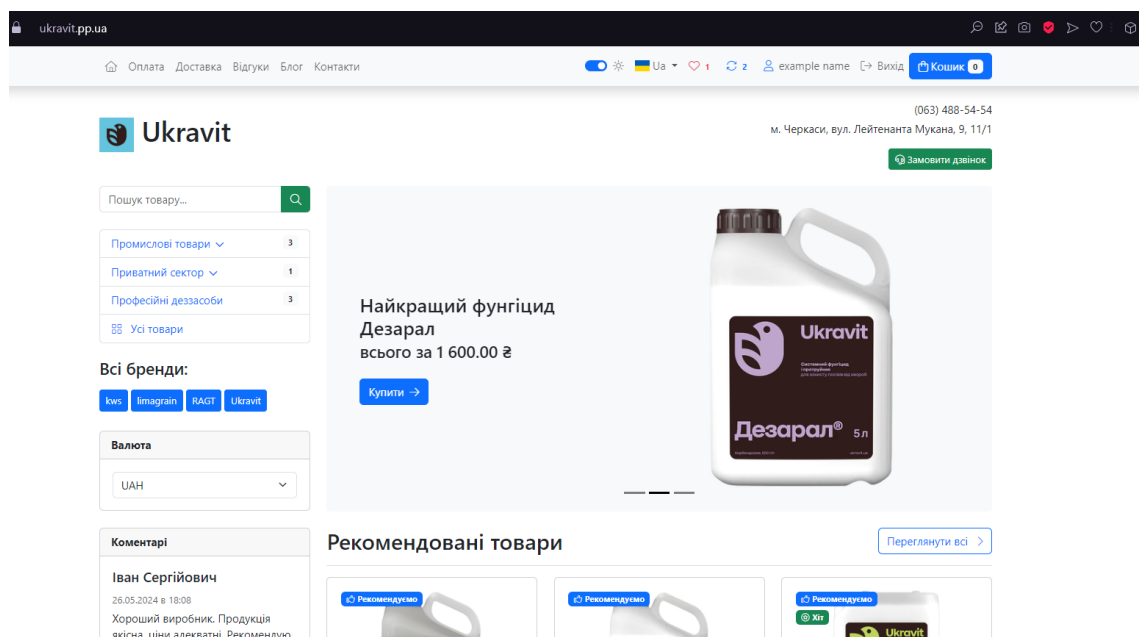



Рисунок 3.17 – Головна сторінка


Тепер є варіанти вибору функціоналу, на сайті реалізовано вибір товару по категоріям та підкатегоріям (попередньо рисунок 3.16). На головній сторінці реалізовано вибірку товару по рекомендаціям та хіту продажу, Рисунок 3.18

Рекомендовані товари


Переглянути всі >



Тандем
★★★★☆ 4.5 0
600.00 ₴
В наявності
Вибрати




Дезарал
★★★★★ 5.0 0
1 600.00 ₴
В наявності
Купити




Варяг
★★★★★ 5.0 0
7 200.00 ₴
В наявності
Купити

Хіти продажів

Переглянути всі >



Антищур
★★★★☆ 4.2 0
20.00 ₴
В наявності
Вибрати




Варяг
★★★★★ 5.0 0
7 200.00 ₴
В наявності
Купити

Рисунок 3.18 – Рекомендація та хіти продажів


Також на головній реалізовано функціонал блогу та статті від компанії, зображення Рисунок 3.19

Блог


Переглянути всі >



Про Ukravit Institute
20.05.2024
Ukravit Institute – унікальний інноваційний R&D центр, покликаний оптимізувати працю аграрія і допомогти йому у прийнятті раду...



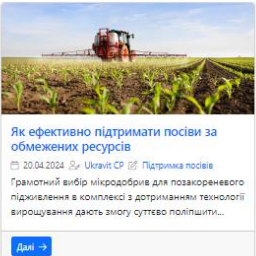
Допомога у придбанні ресурсів (за підтримки USAID APRO)
30.04.2024
Забезпечте своє агровиробництво Компанія Ukravit пропонує товари із відтермінуванням оплати для мікро-, малих і середніх виробників, які...



Агрокредитування на вигідних умовах
25.04.2024
Отримайте агрокредит без будь-якої застави та з вигідною відсотковою ставкою лише 0.01% річних протягом перших піаросу. Господар земельного...

Статті

Переглянути всі >



Як ефективно підтримати посіви за обмежених ресурсів
20.04.2024 | Ukravit SP | Підтримка посівів
Грамотний вибір мікродобрив для позакореневого підживлення в комплексі з дотриманням технології вирощування дають змогу суттєво поліпшити...

Рисунок 3.19 – Блог та Статті

При натисненні на Аккаунт в шапці сторінки (рисунок 3.20) вас переадресує на сторінку авторизації на сайті (Рисунок 1.21)



Рисунок 3.20 –v Аккаунт в шапці профілю

[Головна](#) / [Вхід](#)

Вхід

 A screenshot of a login form titled 'Вхід'. It has two input fields: 'Email*' with the placeholder 'Введіть свій e-mail' and 'Пароль*' with the placeholder 'Введіть пароль'. Below the password field are two links: 'Забули пароль?' on the left and 'Реєстрація' on the right. At the bottom right is a green button with a right arrow and the text 'Вхід'.

Рисунок 3.21 – сторінка авторизації

На сторінці входу можна перейти на сторінку реєстрації або відновити пароль. Відновлення паролю працює і продемонстровано нижче.

При натисканні “Забули пароль?”, вас перенаправить на сторінку Нагадування пароля і після введення коректної пошти яка була зареєстрована на сайті, вам прийде лист на вашу пошту з можливістю зміни паролю. Приклад зображено на рисунках 3.22 – 3.24

[Головна](#) / [Нагадування пароля](#)

Нагадування пароля

 A screenshot of a password reminder form titled 'Нагадування пароля'. It has one input field for 'Email' containing 'mindfreakwh1skey@gmail.com'. Below the field is the text 'Введіть email, який ви вказали при реєстрації'. At the bottom right is a green button with an envelope icon and the text 'Надіслати'.

Рисунок 3.22 – Нагадування пароля

[Головна](#) / [Нагадування пароля](#)

Вам надіслано листа

mindfreakwh1skey@gmail.com - відправлено лист для відновлення пароля.

Рисунок 3.23 – Відправлення листа з відновленням

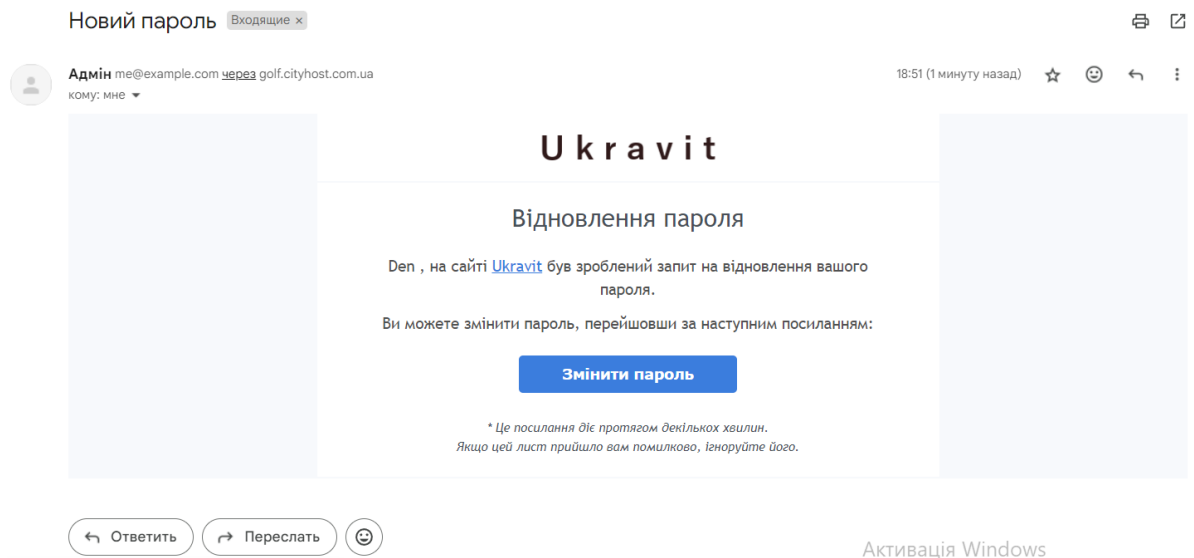


Рисунок 3.24 – лист з зміною пароля

Після натиснення кнопки **Змінити пароль**, вас автоматично індефікує на сайті та перенаправить на сторінку з можливістю зміни вашої інформації профілю, приклад на рисунку 3.25 .

[Головна](#) / Den

Den

Ім'я*

Email*

Телефон


Адреса

[Змінити пароль](#)

Рисунок 3.25 – Зміна інформації профілю користувача

Після вибору категорії товару є можливість вибрати товар або його варіацію, наприклад 5 чи 20 літрів, приклад зображено на рисунку 3.26.

Рекомендуємо



Тандем

★★★★☆ 4.5 (0 голосів) Ukravit

Прилипач для підвищення ефективності засобів захисту рослин та агрохімікатів

Артикул: 000002

🟢 В наявності

2 200.00 ₴

20 літрів ▾

5 літрів

20 літрів

🛒 Купити

🚚 Швидке замовлення

Рисунок 3.26 – вибір варіації товару

Після вибору товару при натисненні на кнопку ‘купити’ позиція додається в корзину після чого можливо маніпулювати з товаром: видаляти, вибирати кількість застосовувати купон або оформити замовлення вибравши спосіб доставки та оплати, заповнивши анкету замовлення можливо його оформити, якщо сума замовлення більше ніж зазначена мінімальна, в даному випадку більше ніж 2 000 грн. Приклад наведено на рисунках 3.27 – 3.28.

Адреса отримувача

ПІБ*

Email*

Телефон

Адреса доставки

Коментар

Капча*

[Оформити замовлення >](#)

Рисунок 3.27 – анкета оформлення замовлення

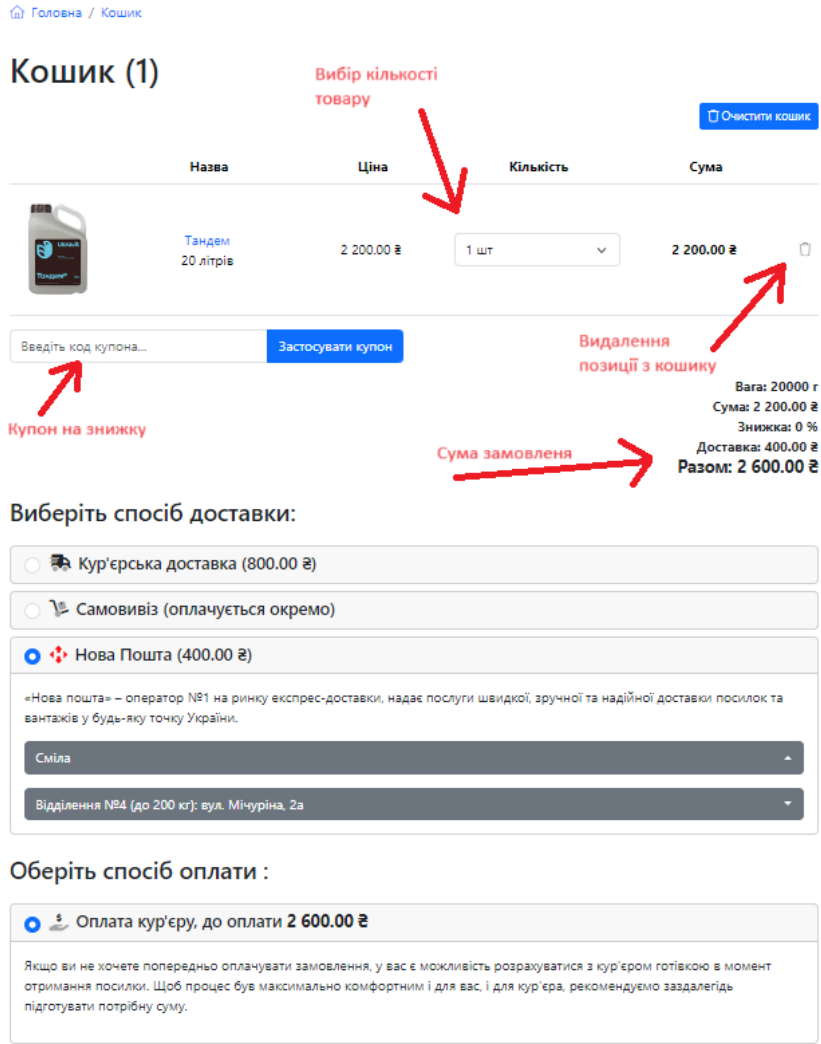


Рисунок 3.28 – Кошик товарів

Реалізовано сторінку з порівняннями товарів, приклад на рисунку 3.26.

При додаванні до порівняння товари з’являються на сторінці ‘Порівняння’, рисунок 3.29 .

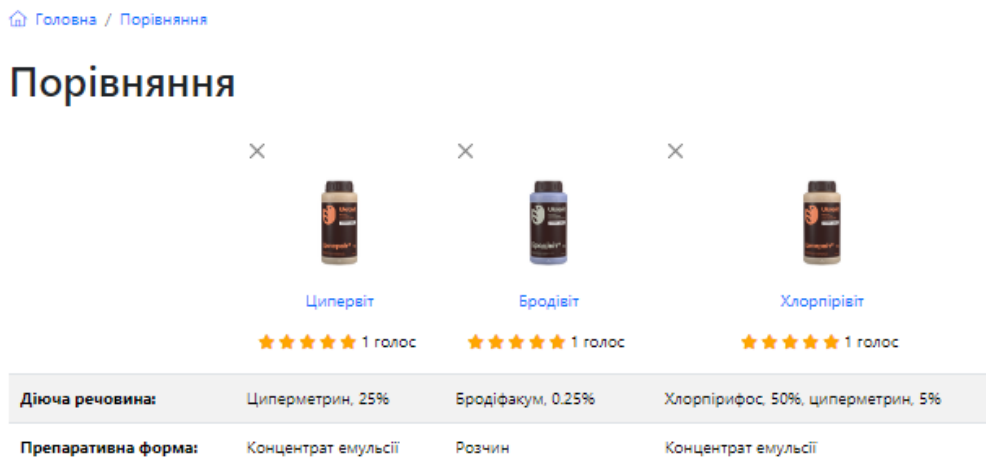


Рисунок 3.29 – сторінка порівняння товарів

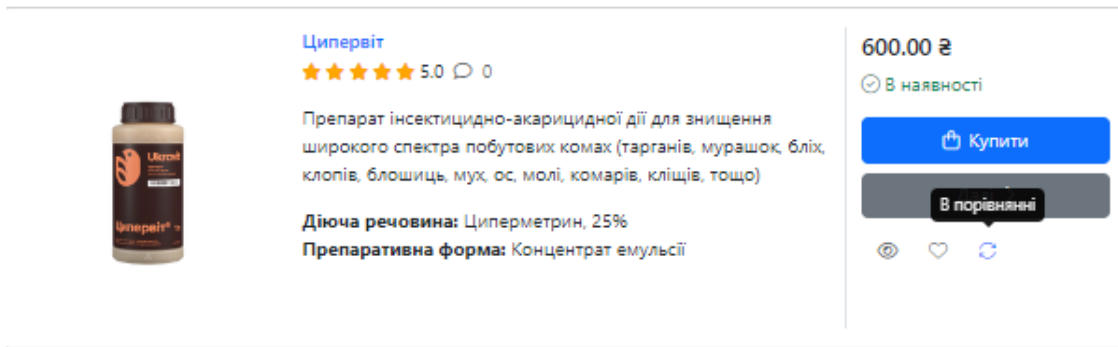


Рисунок 3.30 – Додавання товару до порівняння

3.4. Технічне та системне забезпечення розробки

3.4.1. Обґрунтування вибору технічних засобів

В цьому розділі розглянуто обґрунтування вибору технічних засобів, які використовувалися у процесі розробки проекту. Обрані технології та інструменти сприяли ефективній інтеграції та розвитку проекту, забезпечуючи високу продуктивність та зручність для розробників та кінцевих користувачів.

1. FileZilla

FileZilla була обрана для роботи з файлами на сервері через його зручний інтерфейс та надійну роботу з FTP-підключенням. Завдяки FileZilla розробники мали зручний доступ до файлової структури сервера, що сприяло швидкій і зручній роботі з файлами проекту.

2. PHP

Мова програмування PHP була обрана як основна мова для розробки серверної частини проекту. PHP відома своєю простотою в освоєнні, широким спектром функціональності та великою спільнотою розробників. Це дозволило забезпечити ефективну реалізацію функціональності, високу швидкодію та надійність серверної частини проекту.

3. JavaScript

JavaScript використовувався для реалізації клієнтської логіки та інтерактивності веб-сторінок. Це дозволило створити динамічні та інтерактивні веб-додатки, які покращили користувацький досвід та забезпечили більшу

функціональність.

4. CSS

Мова CSS була використана для стилізації та оформлення веб-сторінок. Завдяки CSS було можливо надати веб-додаткам привабливий та сучасний дизайн, а також забезпечити їхню адаптацію під різні пристрої та екрани.

5. Обробник шаблонів Smarty

Smarty – це компілюючий обробник шаблонів для PHP. Говорячи чіткіше, він надає один із інструментів, які дозволяє домогтися відділення прикладної логіки та даних від уявлення. Це дуже зручно в ситуаціях, коли програміст і верстальник шаблону – різні люди. Наприклад, скажімо, ви створюєте сторінку, яка показує газетну статтю. Назва статті, автор та сама стаття - елементи, які не містять жодної інформації про те, як вони будуть представлені. Їх передають до Smarty із програми. Потім верстальник шаблону редагує шаблони і використовує комбінацію тегів HTML та тегів шаблону, щоб відформатувати представлення цих змінних, що містять елементи типу таблиць HTML, фонових кольорів, розмірів шрифту, стилів, SVG і т.д.). Якимось програміст захоче змінити спосіб зберігання статті, тобто внести зміни до логіки програми. Ця зміна не викликає змін у шаблонах. Зміст все ще передаватиметься в шаблон таким же способом. Аналогічно, якщо верстальник захоче повністю перепроєктувати шаблони, це не вимагатиме жодних змін у прикладній логіці. Таким чином, програміст може вносити зміни до прикладної логіки без необхідності зміни шаблонів, а дизайнер шаблонів може вносити зміни до шаблонів без шкоди для прикладної логіки. Одне з призначень Smarty – це відокремлення логіки програми від уявлення. Звичайно, шаблони можуть містити в собі логіку, але лише за умови, що ця логіка необхідна для правильного представлення даних. Такі завдання, як підключення інших шаблонів, забарвлення рядків, що чергуються в таблиці, приведення букв до верхнього регістру, циклічний прохід по масиву для його відображення і т.д. - Все це приклади логіки уявлення. Тим не менш, не слід вважати, що Smarty змушує вас розділяти прикладну логіку та логіку уявлення. Smarty не бачить різниці між цими речами, тому переносити прикладну логіку в шаблони ви

можете на свій страх і ризик. Якщо ви вважаєте, що у шаблоні взагалі має бути логіки, ви можете обмежитися використанням чистого тексту і змінних. Одна з унікальних можливостей Smarty – компілювання шаблонів. Це означає, що Smarty читає файли шаблонів та створює PHP-код на їх основі. Код створюється один раз і потім лише виконується. Тому немає потреби у повільній обробці файл шаблону для кожного запиту. Кожен шаблон може користуватися всіма перевагами таких компіляторів PHP та кешируючих рішень, як eAccelerator, ionCube, mmCache, Zend Accelerator та інших.

6. Хост на сайті <https://cp.cityhost.ua>

Хостування проекту було здійснено на платформі хостинг-провайдера CityHost.ua. Його було обрано через його надійність, швидкість роботи та зручний інтерфейс управління хостингом через контрольну панель. Це дозволило забезпечити стабільну роботу проекту та надійне зберігання даних користувачів.

7. Реєстрація домену на сайті <https://nic.ua>

Для проекту був зареєстрований домен на сайті <https://nic.ua>. Реєстрація домену була проведена через платформу NIC.ua через їхню зручну систему реєстрації та управління доменами. Обрання домену здійснювалося з урахуванням відповідності доменного імені бренду та легкості запам'ятовування для користувачів. Був обраний домен urkavit.pp.ua pp як приставка яка означає People to People що в перекладі: Людьми для людей.

8. Adobe Photoshop

Графічний редактор, розроблений та поширюваний компанією Adobe Systems. Цей продукт є лідером ринку серед комерційних засобів редагування растрових зображень і є найвідомішим продуктом компанії Adobe [20]. Часто цю програму називають просто Photoshop. Наразі Photoshop доступний на платформах Mac OS та Microsoft Windows. Раніші версії редактора також були доступні для SGI IRIX, але офіційну підтримку було припинено з третьої версії продукту. Версія CS 2 може працювати під Linux за допомогою альтернативи Windows API – Wine 0.9.54 та вище.

Зараз Photoshop широко використовується для створення дизайну сайтів, оскільки він є лідером у сфері обробки растрової графіки. У Photoshop є спеціальні режими збереження графіки для Web, що дозволяють значно зменшити розмір графічних файлів, що є важливим фактором для швидкодії сайту. Сайт складається з великої кількості графічних елементів, які з'єднані між собою за допомогою HTML.

Сучасний Photoshop також підтримує роботу з векторною графікою, хоча він не є лідером у цій сфері. Використання Photoshop для створення сайту забезпечує якісні та компактні графічні елементи.

9. Turbo CMS

система управління контентом (CMS), яка включає в себе адмін-панель для зручного управління вмістом сайту. Використання Turbo CMS забезпечує легкість і ефективність в управлінні контентом, дозволяючи адміністраторам сайту швидко додавати, редагувати і видаляти інформацію без необхідності втручання в код.

10. Vs Code

Для розробки проекту був використаний редактор коду Visual Studio Code (VS Code), який підтримує безліч мов програмування, включаючи PHP, JavaScript та CSS. Він забезпечує зручність розробки завдяки інтуїтивно зрозумілому інтерфейсу, розширенням та плагінам, вбудованому терміналу, інструментам для налагодження коду та підтримці систем контролю версій, таких як Git. Це дозволить в подальшому підвищити ефективність роботи над проектом та забезпечити високу якість коду.

3.4.2 Розрахунок та визначення топології комп'ютерної мережі

Розрахунок та визначення топології комп'ютерної мережі є одним із важливих етапів у процесі проектування та розгортання інформаційних систем. Топологія визначає фізичне та логічне розташування пристроїв у мережі, їхні

зв'язки та взаємодію. Від правильного вибору топології залежить ефективність, надійність та масштабованість мережі.

Однією з найпоширеніших топологій є зіркова топологія. При цьому всі пристрої підключені до центрального комутатора або концентратора. Цей центральний вузол є точкою збору та розподілу даних і забезпечує простоту в управлінні та моніторингу мережі. Кожен пристрій має окреме з'єднання з центральним комутатором, що забезпечує високу надійність: відмова одного пристрою не впливає на роботу інших. Однак, центральний комутатор може стати єдиною точкою відмови, що потребує додаткових заходів забезпечення надійності.

Іншою популярною топологією є шинна топологія. У цьому випадку всі пристрої підключені до спільної шини чи кабелю, через який передаються дані. Шинна топологія характеризується меншою кількістю кабелів та простотою в реалізації для невеликих мереж. Однак, зі збільшенням кількості пристроїв продуктивність може знизитися, а виявлення та усунення несправностей може стати складнішим.

У випадку нашого проекту було прийнято рішення використовувати зіркову топологію через її простоту в управлінні та високу надійність. Центральний комутатор забезпечує зручну точку керування мережею, а окреме з'єднання з кожним пристроєм гарантує незалежність роботи вузлів мережі. Такий підхід підходить для наших потреб і дозволить ефективно функціонувати інформаційній системі, яка буде розгортана.

3.4.3 Обґрунтування вибору ОС та протоколу обміну даними

У процесі проектування інформаційної системи важливим етапом є вибір операційної системи (ОС) та протоколу обміну даними. Ці компоненти впливають на продуктивність, безпеку та сумісність системи. Розглянемо детальніше вибір ОС та протоколу обміну даними для даного проекту.

Вибір операційної системи:

Для даного проекту було обрано операційну систему Windows 11. Вибір Windows 11 обумовлений наступними причинами:

Інтуїтивно зрозумілий інтерфейс:

Windows 11 пропонує сучасний, зручний та інтуїтивно зрозумілий інтерфейс, що полегшує взаємодію користувачів з системою.

Нова панель завдань, вдосконалене меню "Пуск" та візуальні ефекти сприяють покращенню користувацького досвіду.

Сумісність:

Windows 11 забезпечує високу сумісність з широким спектром програмного забезпечення та апаратного забезпечення, що є критично важливим для безперебійної роботи системи.

Підтримка різноманітних драйверів та периферійних пристроїв гарантує стабільну роботу.

Безпека:

Windows 11 включає вдосконалені функції безпеки, такі як Windows Hello, TPM (Trusted Platform Module) 2.0 та Virtualization-based Security (VBS), що забезпечують захист даних та знижують ризики несанкціонованого доступу.

Регулярні оновлення безпеки та патчі від Microsoft забезпечують захист від нових загроз та вразливостей.

Продуктивність:

Оптимізації в Windows 11 покращують продуктивність системи, включаючи швидший запуск додатків та ефективніше використання ресурсів.

Підтримка новітніх технологій апаратного забезпечення, таких як DirectStorage та Auto HDR, забезпечує високу продуктивність для сучасних задач.

Можливості для розробників:

Windows 11 надає розширені можливості для розробників, включаючи Windows Subsystem for Linux (WSL) 2, який дозволяє запускати Linux додатки безпосередньо на Windows, що розширює функціональність та гнучкість

системи.

Інтеграція з популярними інструментами розробки, такими як Visual Studio Code, спрощує процес розробки та налагодження програмного забезпечення.

Вибір протоколу обміну даними

Для забезпечення надійного та безпечного обміну даними було обрано протокол HTTPS (HyperText Transfer Protocol Secure). Основні причини вибору HTTPS:

Безпека:

HTTPS забезпечує шифрування даних, що передаються між клієнтом та сервером, використовуючи SSL/TLS (Secure Sockets Layer/Transport Layer Security). Це захищає дані від перехоплення та маніпуляцій.

Ауθενфікація сервера за допомогою сертифікатів SSL/TLS, що гарантує клієнтам справжність сервера.

Конфіденційність:

Шифрування даних забезпечує конфіденційність інформації, що передається, запобігаючи її несанкціонованому доступу.

Цілісність даних:

Протокол HTTPS забезпечує цілісність даних, що передаються, завдяки використанню криптографічних хеш-функцій. Це гарантує, що дані не були змінені під час передачі.

Сумісність:

HTTPS підтримується всіма сучасними веб-браузерами та клієнтами, що робить його універсальним протоколом для обміну даними в Інтернеті.

Підтримка різноманітних криптографічних алгоритмів дозволяє налаштовувати рівень безпеки відповідно до потреб проекту.

Підвищення довіри користувачів:

Використання HTTPS підвищує довіру користувачів до веб-сайту або веб-додатку, оскільки вони бачать, що з'єднання захищене.

Таким чином, вибір операційної системи Windows 11 та протоколу обміну даними HTTPS забезпечує високу безпеку, надійність та продуктивність

системи, що є критично важливим для успішної реалізації проекту.

3.4.4. Розробка і обґрунтування стратегії адміністрування системи

У рамках цього проекту використовувались різноманітні технічні засоби для забезпечення ефективного управління та адміністрування системи. Першочерговою задачею було забезпечити надійність та зручність управління файлами на сервері. Для цього використовувався FileZilla, що є потужним FTP-клієнтом з інтуїтивно зрозумілим інтерфейсом.

Окрім цього, для керування контентом та іншими аспектами веб-сайту використовувалася адміністративна панель Turbo CMS. Її інтерфейс надавав можливості зручного та ефективного керування різноманітними аспектами веб-проекту, включаючи редагування контенту, керування користувачами та налаштування параметрів сайту.

Окрім цього, для розміщення сайту використовувався хостинг від <https://cp.cityhost.ua>. Цей хостинг надавав повний доступ до налаштувань сервера, що дозволило гнучко налаштовувати сервер під потреби проекту. Крім того, він забезпечував високу надійність та безпеку, що є важливими аспектами для успішної експлуатації веб-проекту.

Загалом, використання цих технічних засобів дозволило ефективно керувати системою, забезпечити безпеку та надійність її роботи, а також зручно розміщувати та адмініструвати веб-сайт.

3.4.5 Заходи захисту від несанкціонованого доступу до системи

У зв'язку з постійними загрозами несанкціонованого доступу до системи, було розроблено та впроваджено комплекс заходів захисту для забезпечення безпеки та конфіденційності даних. Основними заходами захисту від

несанкціонованого доступу до системи є:

Захист паролем: Усі адміністративні облікові записи та доступ до системи забезпечені надійними паролями. Паролі складаються з комбінації різних символів, включаючи цифри, великі та малі літери, а також спеціальні символи. Крім того, паролі регулярно оновлюються та не зберігаються у відкритому вигляді.

Багаторівнева аутентифікація: Для підвищення рівня безпеки використовується багаторівнева аутентифікація. Після введення паролю користувач повинен пройти додаткову перевірку, таку як введення одноразового коду, відправленого на зареєстрований телефон або електронну пошту.

Шифрування даних: Усі дані, які передаються між користувачем та сервером, шифруються за допомогою надійних алгоритмів шифрування. Це дозволяє захистити конфіденційні дані від перехоплення або несанкціонованого доступу.

Моніторинг активності: Система постійно моніторить активність користувачів та виявляє будь-які підозрілі або нестандартні дії. У разі виявлення підозрілої активності автоматично вводяться заходи для блокування доступу та запобігання можливих загроз.

Загальна мета цих заходів захисту полягає в забезпеченні безпеки та надійності системи, а також у запобіганні можливих атак та несанкціонованого доступу до даних.

4. Охорона праці

Охорона праці є одним із важливих аспектів у діяльності будь-якого підприємства, включаючи ТОВ "Укравіт". Забезпечення безпечних умов праці та дотримання вимог законодавства з охорони праці має велике значення як для самого персоналу компанії, так і для здоров'я та безпеки клієнтів та партнерів.

Оцінка ризиків: Першим кроком у забезпеченні безпеки праці є проведення оцінки ризиків. Це включає визначення потенційних небезпек та вжиття заходів для їх запобігання.

Навчання та підготовка персоналу: Усі працівники повинні бути наділені необхідними знаннями та навичками щодо безпеки праці. Це включає ознайомлення з правилами та процедурами охорони праці, а також навчання використанню необхідного обладнання та заходів індивідуального захисту.

Створення безпечних робочих умов: Підприємство зобов'язане створити безпечне робоче середовище для всіх працівників. Це може включати правильне облаштування робочих місць, належне освітлення, а також забезпечення вентиляції та ергономічного обладнання.

Проведення регулярних перевірок та аудитів: Для забезпечення відповідності стандартам охорони праці необхідно регулярно проводити перевірки та аудити. Це допомагає вчасно виявляти та усувати будь-які потенційні загрози та порушення.

Постійне вдосконалення системи управління безпекою праці: Охорона праці є динамічним процесом, тому важливо постійно вдосконалювати систему управління безпекою праці на підприємстві. Це включає аналіз результатів аудитів, впровадження новітніх технологій та нормативів, а також врахування кращих практик у галузі.

Запобігання та реагування на небезпеку: Важливо мати систему запобігання та реагування на небезпеку, що може виникнути в ході роботи. Це включає надання персоналу доступу до необхідних засобів пожежогасіння, наділення персоналу навичками першої допомоги та розробку плану евакуації в разі необхідності.

Забезпечення ефективної системи охорони праці на підприємстві є ключовим елементом у створенні безпечного та продуктивного робочого середовища для всіх працівників

ВИСНОВКИ

В даному дипломному проєкті проведено аналіз ТОВ «Укравіт», в результаті якого створена функціональна модель процесу функціонування інтернет-магазину, виявлені задачі автоматизації та складено технічне завдання на розроблення сайту.

На основі серверної мови програмування PHP, Turbo CMS, бази даних MySQL та клієнтської мови програмування JavaScript розроблено інтернет-магазин для ТОВ «Укравіт».

При розробці сайту велика увага виділялась швидкодії сайту ,функціональності та атоматизації , що дало можливість створити повноцінний сайт.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Контакти «Укравіт». – Дані. – Режим доступу: <https://www.ua-region.com.ua/30632100>
2. Актуальна інформація по підприємству. – Дані. – Режим доступу: <https://www.ukravit.ua/about/>
3. Перелік промислових товарів. – Режим доступу: <https://www.ukravit.ua/prom/>
4. Звіт компанії за 2019-2022 рік. - Електронний варіант документу. - Режим доступу: <https://www.ukravit.ua/zvitnist/>
5. Екологічна політика підприємства, безпечне зберігання ЗЗР та МКД. – Режим доступу: <https://www.ukravit.ua/utilizaciya-tari/>
6. Аналіз вакансій компанії – Режим доступу: <https://robota.ua/company3844948>
7. Політика конфіденційності – Режим доступу: <https://www.ukravit.ua/privacy/>
8. AllFushion Data Modeler. – Дані. – Режим доступу: <https://hi-news.pp.ua/kompyuteri/16669-programa-bpwin-ce-scho-take.html>
9. Сайт компанії ЛНЗ Дефенда. – Дані. – Режим доступу: https://defenda.com.ua/?utm_source=google&utm_medium=cpc&utm_campaign=ROMAN_PMax&gad_source=1&gclid=CjwKCAiArfauBhApEiwAeoB7qCKBbOCu72t0aiFhcZY4fv23KIRoRv4yVHtUs5Ov8y1ImK_xMb-8shoC0TYQAvD_BwE
10. Сайт компанії Альфа Смарт Агро. – Дані. – Режим доступу: <https://alfasmartagro.com>
11. Сайт компанії Агрорось. – Дані. – Режим доступу: <https://agroros.com.ua>
12. Smith, J. (2020). "The Importance of Website Design in Business Success." – Режим доступу: <https://www.businessnewsdaily.com/11012-website-design.html>
13. Ivanov, A. (2019). "Effective Website Navigation: What to Consider." –

Режим доступу: <https://www.smashingmagazine.com/2019/01/effective-website-navigation/>

14. Pew Research Center. (2022). "Internet/Broadband Fact Sheet." – Режим доступу: <https://www.pewresearch.org/internet/fact-sheet/internet-broadband/>

15. "Why User Reviews Are Important for E-commerce Success." (2021). BigCommerce. – Режим доступу: <https://www.bigcommerce.com/blog/ecommerce-reviews/#why-user-reviews-are-important>

16. "How to Improve Website Loading Speed." (2020). – Режим доступу: <https://developers.google.com/web/fundamentals/performance/get-started>

17. Brown, K. (2021). "Top Website Design Trends for 2021." – Режим доступу: <https://blog.adobe.com/en/publish/2021/02/10/top-website-design-trends-for-2021.html>

18. Вишняков, О. (2022). "Основи проектування користувацьких інтерфейсів." Ukrainian Design School. – Режим доступу: <https://uds.design/courses/uid>

19. The Nielsen Norman Group. (2021). "10 Usability Heuristics for User Interface Design." – Режим доступу: <https://www.nngroup.com/articles/ten-usability-heuristics/>

20. W3Schools. (2023). "HTML Introduction." – Режим доступу: https://www.w3schools.com/html/html_intro.asp

21. Smith, J. (2021). "The Impact of Mobile Responsiveness on Website Success." – Режим доступу: <https://www.mobilewebweekly.com/importance-of-mobile-responsiveness>

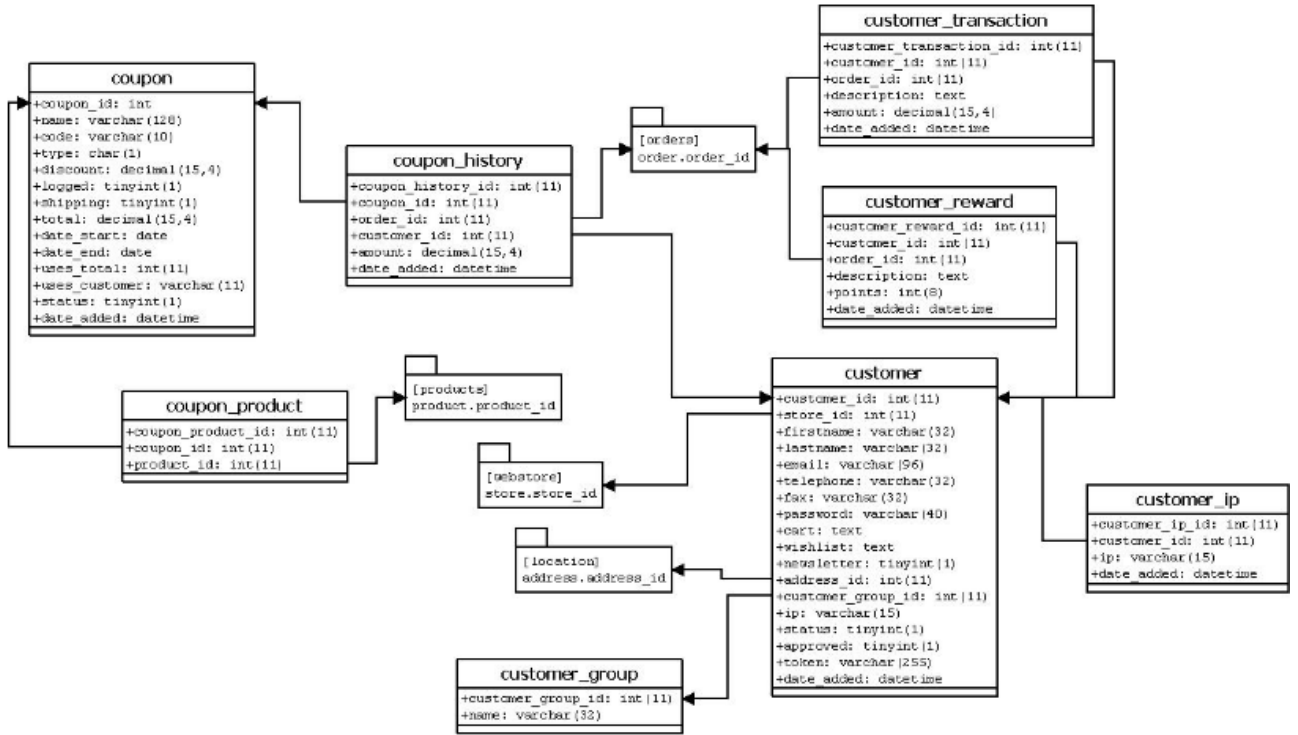


Рисунок А.2 — Модель користувачів бази даних

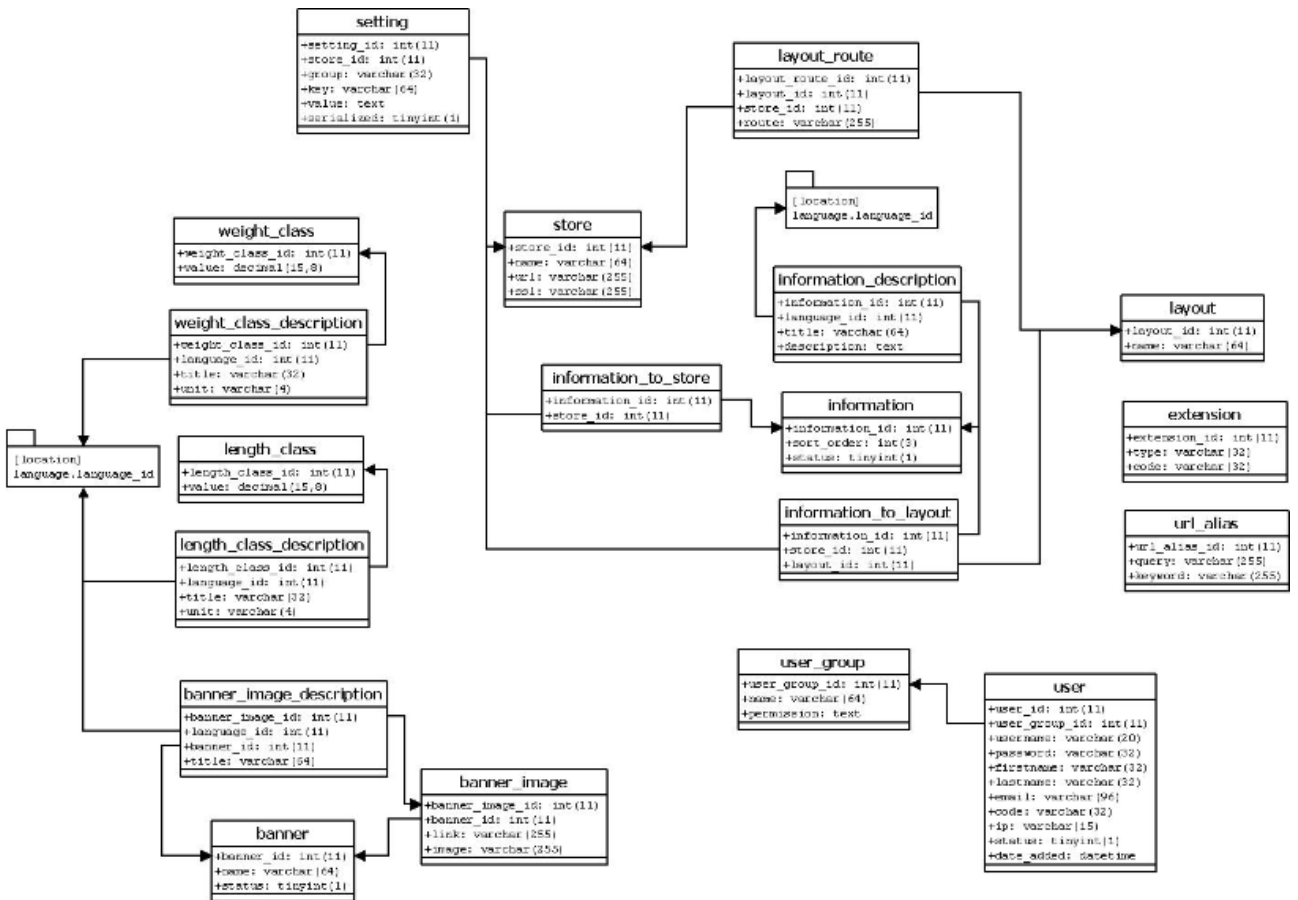
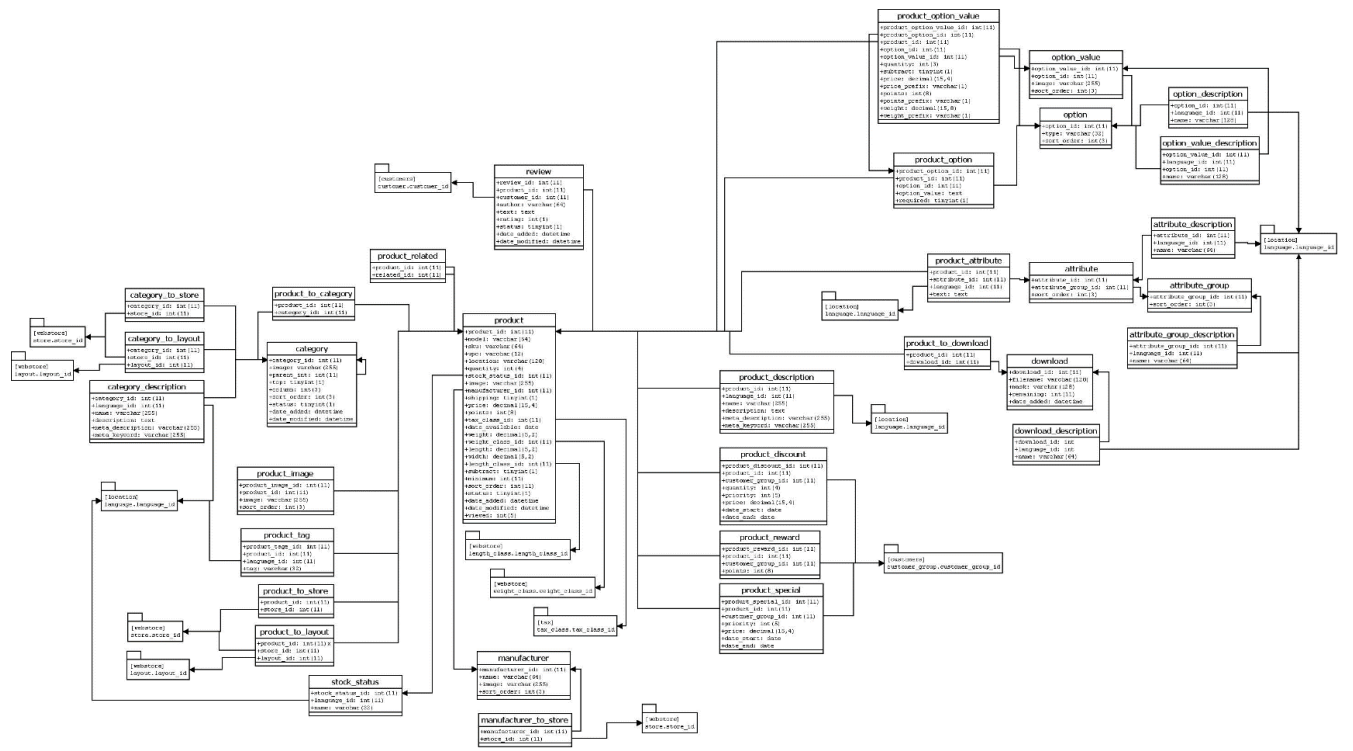


Рисунок А.3 — Модель системи бази даних

Рисунок А.4 — Модель товарів бази даних



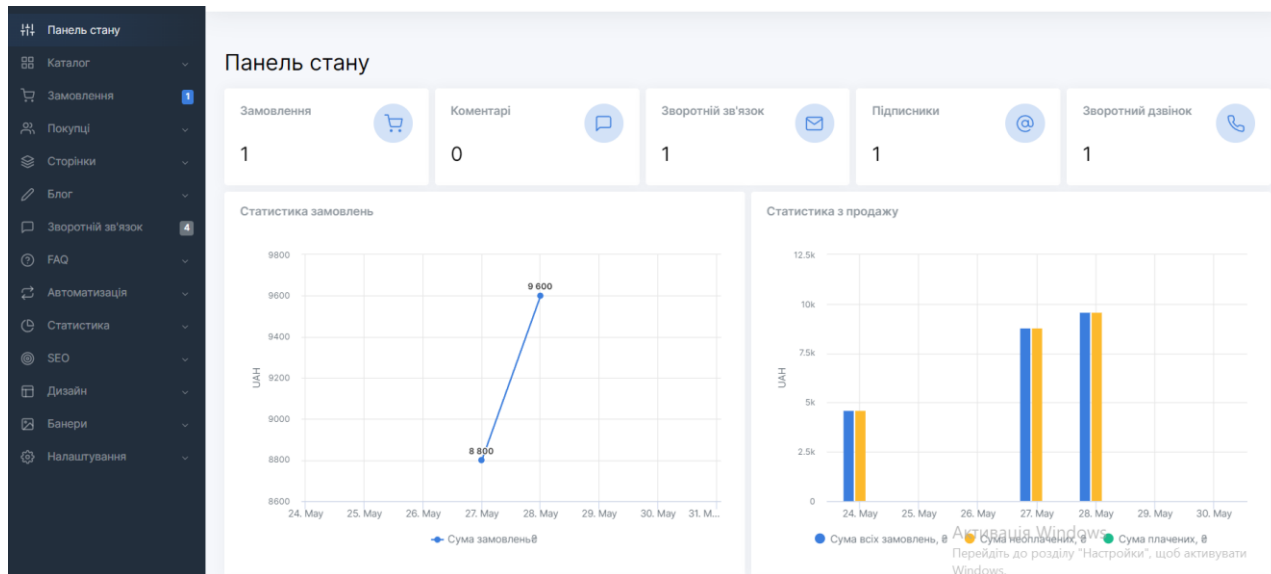


Рисунок В.1 – Панель стану та категорії панелі адміністратора

Товари - 7 [+ Додати товар](#)

Всі товари Всі категорії Всі бренди

Фото	Назва	Ціна	К-сть	Активність	Дії
	Антицур (1000 грам) Варіанти <input type="text"/>	20.00 ₴	∞ ШТ	<input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
	Тандем (5 літрів) Бренд Ukravit Варіанти <input type="text"/>	15.00 \$	∞ ШТ	<input checked="" type="checkbox"/>	<input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
	Дезарал (5 літрів)	40.00 \$	∞ ШТ	<input checked="" type="checkbox"/>	<input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
	Варяг (5 літрів) Бренд Ukravit	180.00 \$	∞ ШТ	<input checked="" type="checkbox"/>	<input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
	Ципервіт	15.00 \$	∞ ШТ	<input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>

Рисунок В.2 – Фрагменти сторінки редагування товарів

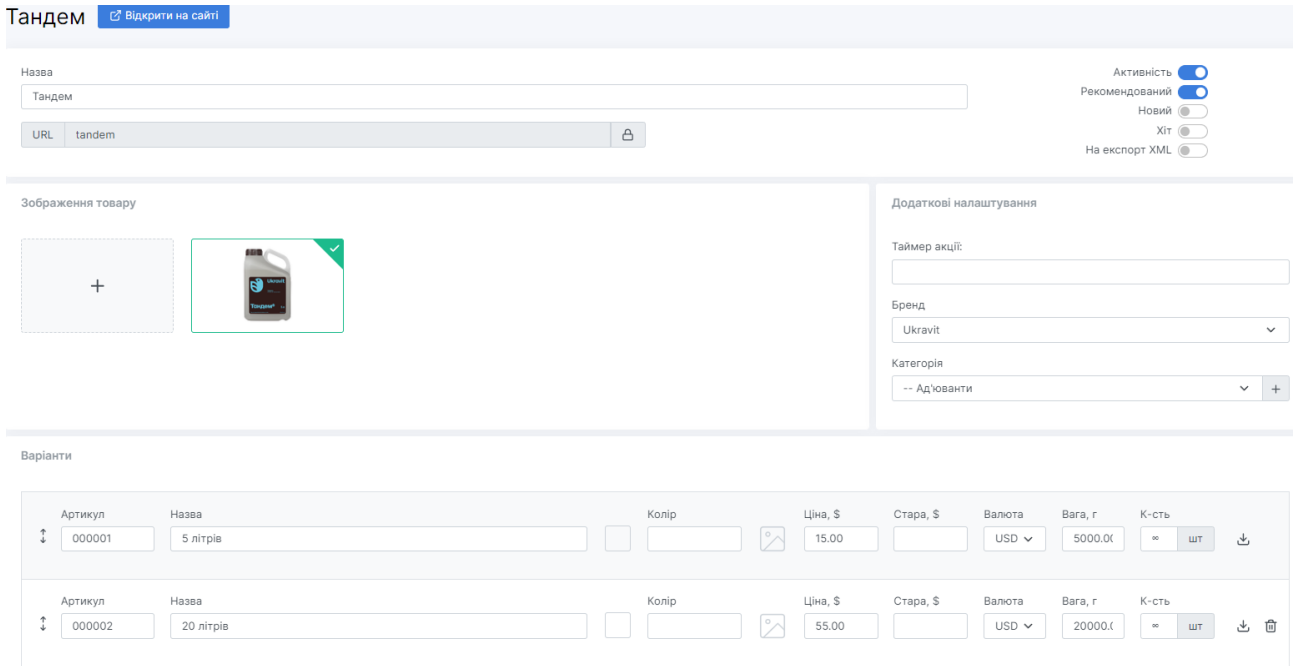


Рисунок В.3 – Фрагменти сторінки редагування товару

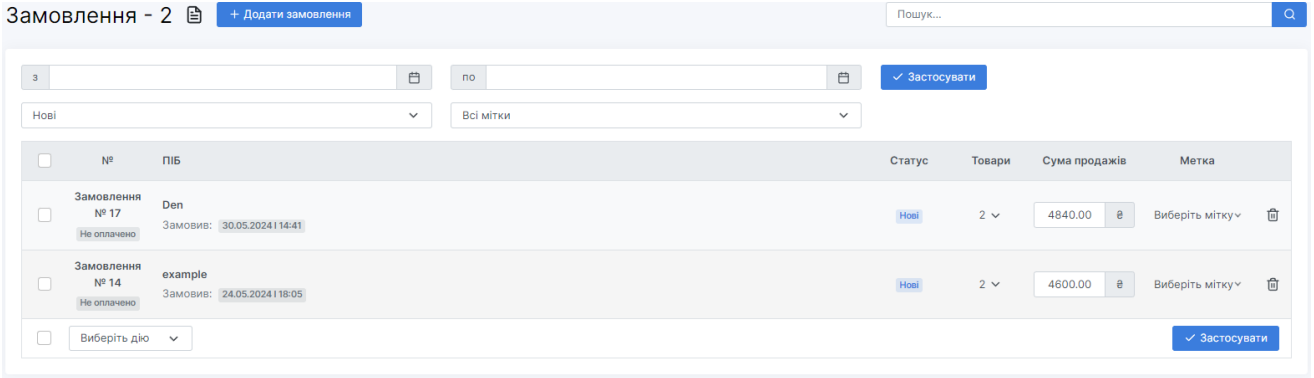


Рисунок В.4 – Фрагменти сторінки замовлень

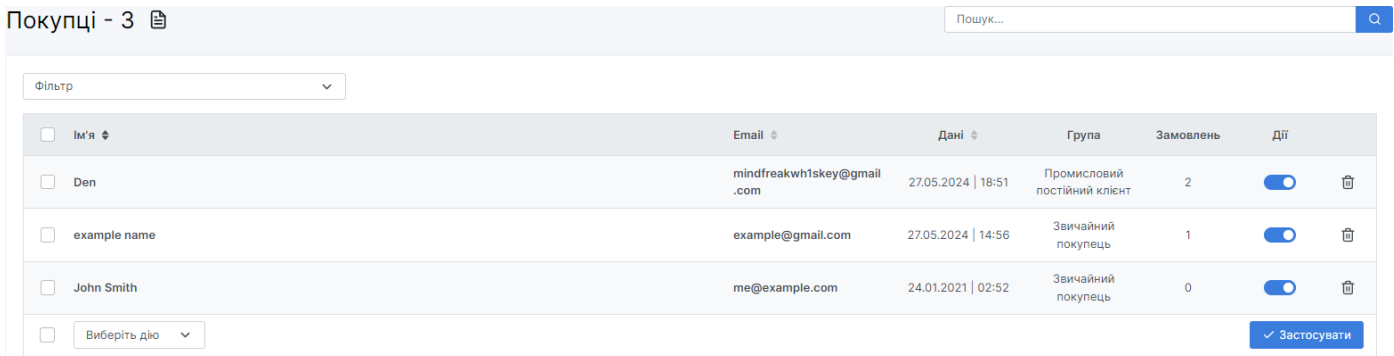


Рисунок В.5 – Фрагменти сторінки покупців



Рисунок В.6 – Фрагменти сторінки блогу

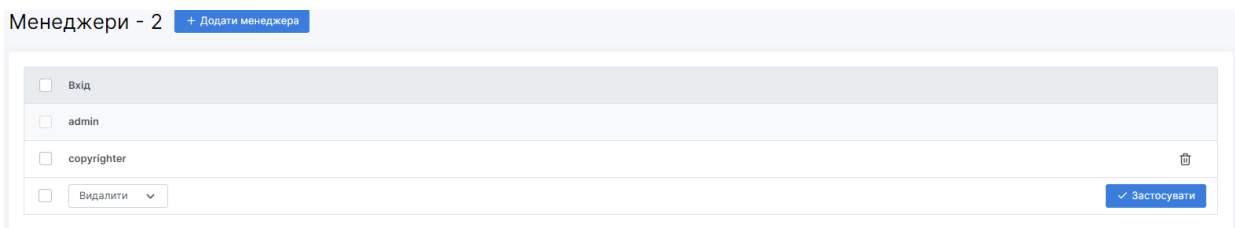


Рисунок В.7 – Фрагменти сторінки персоналу адмін панелі

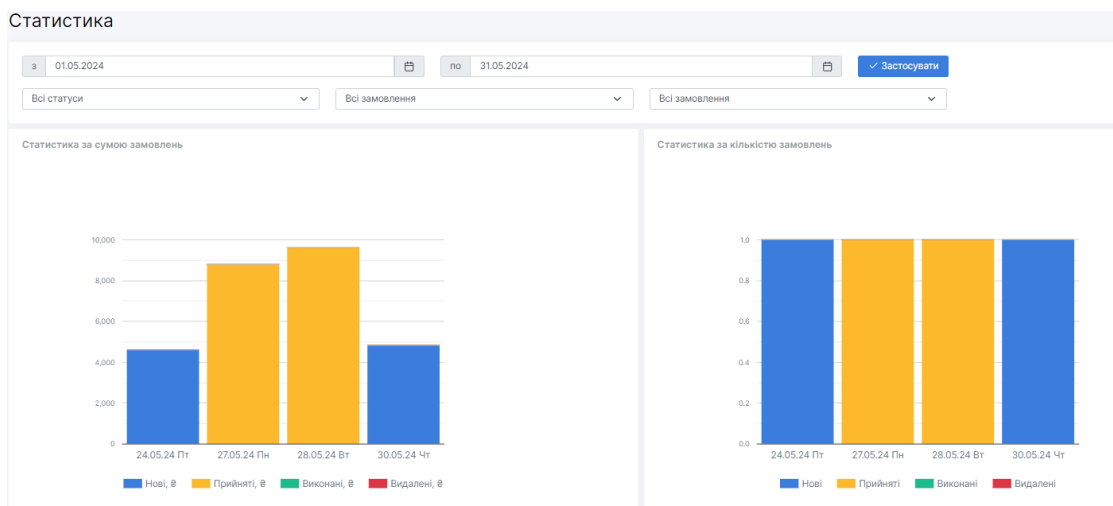


Рисунок В.8 – Фрагменти сторінки статистики замовлень

The screenshot shows the main page of the Ukravit website. At the top, there is a navigation bar with links for 'Оплата', 'Доставка', 'Відгуки', 'Блог', and 'Контакти'. The main header features the Ukravit logo, contact information (phone number (063) 488-54-54 and address in Cherkasy), and a 'Замовити дзвінок' button. A search bar is located below the header. On the left side, there are filters for 'Промислові товари', 'Приватний сектор', 'Професійні деззасоби', and 'Усі товари'. Below these are 'Всі бренди' (listing kvs, kimagran, RACT, Ukravit) and a 'Валюта' section set to 'UAH'. The main content area features a large advertisement for 'Професійний деззасіб Бродівіт від €680' with a 'Купити' button. Below this, there are sections for 'Рекомендовані товари' (listing Tandem, Dezaral, and Varyal) and 'Хіти продажів'.

Рисунок В.9 – Фрагмент головної сторінки сайту Ukravit.pp.ua





The screenshot shows the 'Професійні деззасоби' category page. It includes a breadcrumb trail 'Головна / Каталог / Професійні деззасоби' and a 'Сортувати за' dropdown menu. The page displays three product cards:

- Циперівіт**: Price 600.00 ₴. Description: 'Препарат інсектицидно-акарицидної дії для знищення широкого спектра побутових комах (тарганів, мурашок, бліх, клопів, блощиць, мух, ос, молі, комарів, кліщів, тощо)'. Active ingredient: Циперметрин, 25%. Form: Концентрат емульсії.
- Бродівіт**: Price 680.00 ₴. Description: 'Розчин для приготування отруйних принад і використання їх з метою знешкодження шкідливих гризунів'. Active ingredient: Бродіфакум, 0.25%. Form: Розчин.
- Хлорпіривіт**: Price 700.00 ₴. Description: 'Комбінований (двокомпонентний) контактний інсектицид широкого спектра з чітко вираженою системною дією, акарицидними та репелентними властивостями для знищення синантропних комах, членистоногих і кліщів у середовищі життєдіяльності людини'. Active ingredients: Хлорпірифос, 50%, циперметрин, 5%. Form: Концентрат емульсії.

Рисунок В.10 – Фрагмент товару групи “Професійні деззасоби”

Кошик (6)

[Очистити кошик](#)

	Назва	Ціна	Кількість	Сума	
	Ципервіт	600.00 ₴	4 шт	2 400.00 ₴	
	Бродівіт	680.00 ₴	2 шт	1 360.00 ₴	

[Застосувати купон](#)

Сума: 3 760.00 ₴
 Знижка: 9 %
 Доставка: 800.00 ₴
Разом: 4 221.60 ₴

Рисунок В.11 – Фрагмент кошику

Блог

[Переглянути всі >](#)**Про Ukravit Institute**

20.05.2024

Ukravit Institute – унікальний інноваційний R&D центр, покликаний оптимізувати працю аграрія і допомогти йому у прийнятті ряду...

[Далі →](#)**Допомога у придбанні ресурсів (за підтримки USAID АГРО)**

30.04.2024

Забезпечте своє агровиробництво Компанія Ukravit пропонує товари із відтермінуванням оплати для мікро-, малих і середніх виробників, які...

[Далі →](#)**Агрокредитування на вигідних умовах**

25.04.2024

Отримайте агрокредит без будь-якої застави та з вигідною відсотковою ставкою лише 0.01% річних протягом перших півроку. Господар земельного...

[Далі →](#)

Статті

[Переглянути всі >](#)**Як ефективно підтримати посіви за обмежених ресурсів**20.04.2024 [Ukravit СР](#) [Підтримка посівів](#)

Грамотний вибір мікродобрив для позакореневого підживлення в комплексі з дотриманням технології вирощування дають змогу суттєво поліпшити...

[Далі →](#)

Рисунок В.12 – Фрагмент блогу та статей

Відгуки

★★★★☆ 4.7

3 Коментаря

[Популярні](#) [По порядку](#)





-
-  **Олена**
18.01.2023 в 05:10
★★★★☆
Продукцію купуємо гуртом уже не перший рік. Все завжди якісно.
[Відповісти](#) ^ 1 v
-  **Іван Сергійович**
26.05.2024 в 18:08
★★★★★
Хороший виробник. Продукція якісна, ціни адекватні. Рекомендую.
[Відповісти](#) ^ 0 v
-  **ТОВ Благо**
11.03.2024 в 05:08
★★★★★
Наші найкращі рекомендації. З компанією працюємо давно та у великих обсягах, все завжди чітко і швидко. Дуже хороша консультація від персоналу, завжди підбираємо чітко за нашими потребами.
[Відповісти](#) ^ -1 v
-  Коментувати

Рисунок В.13 – Фрагмент відгуків

Додаток С. Основний код для маніпуляції з товарами

Цей скрипт здійснює пошук товарів на основі ключових слів, отримує їх варіанти (включаючи наявність та ціну), а потім формує список пропозицій, який відправляється у форматі JSON.:

```
<?php
```

```
session_start();
```

```
require_once '../api/Turbo.php';
```

```
$turbo = new Turbo();
```

```
$limit = 30;
```

```
if (!$turbo->managers->access('orders')) {
    return false;
}
```

```
$langId = $turbo->languages->langId();
```

```
$px = ($langId ? 'l' : 'p');
```

```
$langSql = $turbo->languages->getQuery(array('object' => 'product', 'px' => 'p'));
```

```
$keyword = $turbo->request->get('query', 'string');
```

```
$keywords = explode(' ', $keyword);
```

```
$keywordSql = "";
```

```
foreach ($keywords as $keyword) {
    $kw = $turbo->db->escape(trim($keyword));
```

```

    $keywordSql .= $turbo->db->placeholder(
        "AND (
            $px.name LIKE '%$kw%'
            OR $px.meta_keywords LIKE '%$kw%'
            OR p.id in (SELECT product_id FROM __variants WHERE sku LIKE
'%$kw%')
        )"
    );
}

```

```

$turbo->db->query(
    "SELECT
        p.id,
        $px.name,
        i.filename AS image

```

```

FROM __products p
  $langSql->join
LEFT JOIN __images i ON i.product_id=p.id AND i.position=(SELECT
MIN(position) FROM __images WHERE product_id=p.id LIMIT 1)
LEFT JOIN __variants pv ON pv.product_id=p.id
WHERE 1
  $keywordSql
ORDER BY $px.name
LIMIT ?",
$limit
);

foreach ($turbo->db->results() as $product) {
  $products[$product->id] = $product;
}

$langSql = $turbo->languages->getQuery(['object' => 'variant', 'px' => 'pv']);

$variants = [];

if (!empty($products)) {
  $turbo->db->query(
    "SELECT
      pv.id,
      pv.name,
      pv.color,
      pv.sku,
      pv.price,
      IFNULL(pv.stock, ?) AS stock,
      (pv.stock IS NULL) AS infinity,
      pv.product_id,
      $langSql->fields
    FROM __variants pv
      $langSql->join
    WHERE
      pv.product_id IN(?@)
      AND (pv.stock IS NULL OR pv.stock>0)
      AND pv.price>0
    ORDER BY pv.position",
    $turbo->settings->max_order_amount,
    array_keys($products)
  );

  $variants = $turbo->db->results();
}

```

```

foreach ($variants as $variant) {
    if (isset($products[$variant->product_id])) {
        $products[$variant->product_id]->variants[] = $variant;
    }
}

$suggestions = [];

foreach ($products as $product) {
    if (!empty($product->variants)) {
        $suggestion = new stdClass();

        if (!empty($product->image)) {
            $product->image = $turbo->design->resizeModifier($product->image, 35,
35);
        }

        $suggestion->value = $product->name;
        $suggestion->data = $product;
        $suggestions[] = $suggestion;
    }
}

$res = new stdClass();
$res->query = $keyword;
$res->suggestions = $suggestions;

header('Content-Type: application/json; charset=UTF-8');
header('Cache-Control: must-revalidate');
header('Pragma: no-cache');
header('Expires: -1');

echo json_encode($res);

```

Цей PHP-скрипт здійснює експорт статистичних даних про товари у CSV-файл. Він реалізує клас ExportAjax, який наслідує клас Turbo та використовує його методи для роботи з базою даних, запитів та маніпулювання даними.:

```
<?php
```

```
session_start();
```

```
require_once '../api/Turbo.php';
```

```
class ExportAjax extends Turbo
```

```

{
    private $columnsNames = [
        'category_name' => 'Category',
        'product_name' => 'Name',
        'sum_price' => 'Price',
        'amount' => 'Amount'
    ];

    private $columnDelimiter = ',';
    private $statCount = 100;
    private $exportFilesDir = '../files/export/';
    private $filename = 'export_stat_products.csv';

    public function fetch()
    {
        if (!$this->managers->access('stats')) {
            return false;
        }

        $this->db->query('SET NAMES cp1251');

        $page = $this->request->get('page');

        if (empty($page) || $page == 1) {
            $page = 1;

            if (is_writable($this->exportFilesDir . $this->filename)) {
                @unlink($this->exportFilesDir . $this->filename);
            }
        }

        $f = fopen($this->exportFilesDir . $this->filename, 'ab');

        if ($page == 1) {
            fputcsv($f, $this->columnsNames, $this->columnDelimiter);
        }

        $filter = [];

        $dateFilter = $this->request->get('date_filter');

        if (!empty($dateFilter)) {
            $filter['date_filter'] = $dateFilter;
        }
    }
}

```

```
if ($this->request->get('date_from') || $this->request->get('date_to')) {
    $dateFrom = $this->request->get('date_from');
    $dateTo = $this->request->get('date_to');
}

if ($this->request->get('category')) {
    $filter['category_id'] = $this->request->get('category');
}

if (!empty($dateFrom)) {
    $filter['date_from'] = date("Y-m-d 00:00:01", strtotime($dateFrom));
}

if (!empty($dateTo)) {
    $filter['date_to'] = date("Y-m-d 23:59:00", strtotime($dateTo));
}

$status = $this->request->get('status', 'integer');

if (!empty($status)) {
    $filter['status'] = $status;
}

$sortProd = $this->request->get('sort_prod');

if (!empty($sortProd)) {
    $filter['sort_prod'] = $sortProd;
} else {
    $filter['sort_prod'] = 'price';
}

$filter['page'] = max(1, $this->request->get('page', 'integer'));
$filter['limit'] = 40;

$tempFilter = $filter;

unset($tempFilter['limit']);
unset($tempFilter['page']);

$totalCount = $this->reportstat->getReportPurchasesCount($tempFilter);

if ($this->request->get('page') == 'all') {
    $filter['limit'] = $totalCount;
}
```

```

$totalSumm = 0;
$totalAmount = 0;

$reportStatPurchases = $this->reportstat->getReportPurchases($filter);

$catFilter = $this->request->get('category');

if (isset($filter['category_id'])) {
    $cat = $this->categories->getCategory(intval($filter['category_id']));
}

foreach ($reportStatPurchases as $id => $r) {
    if (!empty($r->product_id)) {
        $tmpCat = $this->categories->getCategories(['product_id' => $r->product_id]);
        $tmpCat = reset($tmpCat);

        if (!empty($catFilter) && !in_array($catFilter, (array)$tmpCat->path[$cat->levelDepth - 1]->children)) {
            unset($reportStatPurchases[$id]);
        } else {
            $reportStatPurchases[$id]->category_name = $tmpCat->name;
        }
    }
}

foreach ($reportStatPurchases as $u) {
    $totalSumm += $u->sum_price;
    $totalAmount += $u->amount;
    $str = [];

    foreach ($this->columnsNames as $n => $c) {
        if (isset($u->$n)) {
            $str[] = $u->$n;
        } else {
            $str[] = "";
        }
    }

    fputcsv($f, $str, $this->columnDelimiter);
}

$total = ['category_name' => 'Total', 'product_name' => '', 'price' =>
$totalSumm, 'amount' => $totalAmount];

```

```

fputcsv($f, $total, $this->columnDelimiter);

fclose($f);

if ($this->statCount * $page < $totalCount) {
    return ['end' => false, 'page' => $page, 'totalpages' => $totalCount / $this->statCount];
} else {
    return ['end' => true, 'page' => $page, 'totalpages' => $totalCount / $this->statCount];
}
}
}

```

```
$exportAjax = new ExportAjax();
```

```
$data = $exportAjax->fetch();
```

```

if ($data) {
    header("Content-type: application/json; charset=utf-8");
    header("Cache-Control: must-revalidate");
    header("Pragma: no-cache");
    header("Expires: -1");

    echo json_encode($data);
}

```

Цей PHP-скрипт здійснює пошук товарів у базі даних на основі ключових слів, переданих у запиті, і повертає результати у форматі JSON. Він використовується для надання пропозицій товарів під час пошуку:

```
<?php
```

```
session_start();
```

```
require_once '../api/Turbo.php';
```

```
$turbo = new Turbo();
```

```
$limit = 30;
```

```
$langId = $turbo->languages->langId();
```

```
$px = ($langId ? 'l' : 'p');
```

```
$langSql = $turbo->languages->getQuery(['object' => 'product', 'px' => 'p']);
```

```
$keyword = $turbo->request->get('query', 'string');
```

```
$keywords = explode(' ', $keyword);
$keywordSql = "";
```

```
foreach ($keywords as $kw) {
    $kw = $turbo->db->escape(trim($kw));
    $keywordSql .= $turbo->db->placeholder("AND ($px.name LIKE '%$kw%' OR
$px.meta_keywords LIKE '%$kw%' OR p.id IN (SELECT product_id FROM
__variants WHERE sku LIKE '%$kw%')) ");
}
```

```
$query =
"SELECT
    p.id,
    $px.name,
    i.filename AS image
FROM __products p
    $langSql->join
    LEFT JOIN __images i ON i.product_id = p.id AND i.position = (
        SELECT MIN(position)
        FROM __images
        WHERE product_id = p.id
        LIMIT 1
    )
WHERE 1
    $keywordSql
ORDER BY
    $px.name
LIMIT ?";
```

```
$turbo->db->query($query, $limit);
```

```
$products = $turbo->db->results();
```

```
$suggestions = [];
```

```
foreach ($products as $product) {
    if (!empty($product->image)) {
        $product->image = $turbo->design->resizeModifier($product->image, 35, 35);
    }

    $suggestion = new stdClass();
    $suggestion->value = $product->name;
    $suggestion->data = $product;
    $suggestions[] = $suggestion;
}
```

```
$res = new stdClass();
$res->query = $keyword;
$res->suggestions = $suggestions;
```

```
header('Content-Type: application/json; charset=UTF-8');
header('Cache-Control: must-revalidate');
header('Pragma: no-cache');
header('Expires: -1');
```

```
echo json_encode($res);
```

Код функціоналу Cart:

```
<?php
```

```
require_once 'Turbo.php';
```

```
class Cart extends Turbo
```

```
{
```

```
    /**
```

```
     * отримати Cart
```

```
     */
```

```
    public function getCart()
```

```
    {
```

```
        $cart = new stdClass();
```

```
        $cart->purchases = [];
```

```
        $cart->total_cart = 0;
```

```
        $cart->total_price = 0;
```

```
        $cart->total_products = 0;
```

```
        $cart->coupon = null;
```

```
        $cart->discount = 0;
```

```
        $cart->coupon_discount = 0;
```

```
        $cart->total_weight = 0;
```

```
        if (!empty($_SESSION['shopping_cart'])) {
```

```
            $sessionItems = $_SESSION['shopping_cart'];
```

```
            $variants = $this->variants->getVariants(['id' => array_keys($sessionItems)]);
```

```
            if (!empty($variants)) {
```

```
                $items = [];
```

```
                $productsIds = [];
```

```
                foreach ($variants as $variant) {
```

```
                    $item = new stdClass();
```

```
                    $item->variant = $variant;
```

```

        $item->amount = $sessionItems[$variant->id];
        $items[$variant->id] = $item;
        $productsIds[] = $variant->product_id;
    }

    $products = [];

    foreach ($this->products->getProducts(['id' => $productsIds, 'limit' =>
count($productsIds)]) as $p) {
        $products[$p->id] = $p;
    }

    $images = $this->products->getImages(['product_id' => $productsIds]);

    foreach ($images as $image) {
        $products[$image->product_id]->images[$image->id] = $image;
    }

    foreach ($items as $variantId => $item) {
        if (!empty($products[$item->variant->product_id])) {
            $purchase = new stdClass();
            $purchase->product = $products[$item->variant->product_id];
            $purchase->variant = $item->variant;
            $purchase->amount = $item->amount;

            if (!empty($purchase->product->sale_to) && isset($purchase-
>variant) && strtotime($purchase->product->sale_to) <= time() && $purchase-
>variant->compare_price) {
                $purchase->variant->price = $purchase->variant->compare_price;
                $purchase->variant->compare_price = 0;
                $v = new stdClass();
                $v->price = $purchase->variant->price;
                $v->compare_price = 0;
                $this->variants->updateVariant($purchase->variant->id, $v);
            }

            $cart->purchases[] = $purchase;
            $cart->total_cart += $item->variant->price * $item->amount;
            $cart->total_price += $item->variant->price * $item->amount;
            $cart->total_weight += $item->variant->weight * $item->amount;
            $cart->total_products += $item->amount;
        }
    }

    $cart->discount = 0;

```

```

        if (isset($_SESSION['user_id']) && $user = $this->users->getUser((int)
$_SESSION['user_id'])) {
            $cart->discount = $user->discount;
        }

        $cart->total_price *= (100 - $cart->discount) / 100;

        if (isset($_SESSION['coupon_code'])) {
            $cart->coupon = $this->coupons-
>getCoupon($_SESSION['coupon_code']);

            if ($cart->coupon && $cart->coupon->valid && $cart->total_price >=
$cart->coupon->min_order_price) {
                if ($cart->coupon->type == 'absolute') {
                    $cart->coupon_discount = $cart->total_price > $cart->coupon-
>value ? $cart->coupon->value : $cart->total_price;
                    $cart->total_price = max(0, $cart->total_price - $cart->coupon-
>value);
                } else {
                    $cart->coupon_discount = $cart->total_price * ($cart->coupon-
>value) / 100;
                    $cart->total_price = $cart->total_price - $cart->coupon_discount;
                }
            } else {
                unset($_SESSION['coupon_code']);
            }
        }
    }

    return $cart;
}

/**
 * додавання
 */
public function addItem($variantId, $amount = 1)
{
    $amount = max(1, $amount);

    if (isset($_SESSION['shopping_cart'][$variantId])) {
        $amount = max(1, $amount + $_SESSION['shopping_cart'][$variantId]);
    }
}

```

```

$variant = $this->variants->getVariant($variantId);

if (!empty($variant) && ($variant->stock > 0)) {
    $amount = min($amount, $variant->stock);
    $_SESSION['shopping_cart'][$variantId] = (int) $amount;
}
}

/**
 * оновлення
 */
public function updateItem($variantId, $amount = 1)
{
    $amount = max(1, $amount);

    $variant = $this->variants->getVariant($variantId);

    if (!empty($variant) && $variant->stock > 0) {
        $amount = min($amount, $variant->stock);
        $_SESSION['shopping_cart'][$variantId] = (int) $amount;
    }
}

/**
 * видалення
 */
public function deleteItem($variantId)
{
    unset($_SESSION['shopping_cart'][$variantId]);
}

/**
 * опустощення кошику
 */
public function emptyCart()
{
    unset($_SESSION['shopping_cart']);
    unset($_SESSION['coupon_code']);
}

/**
 * застосувати купон
 */
public function applyCoupon($couponCode)
{

```

```

$coupon = $this->coupons->getCoupon((string) $couponCode);

if ($coupon && $coupon->valid) {
    $_SESSION['coupon_code'] = $coupon->code;
} else {
    unset($_SESSION['coupon_code']);
}
}
}

```

Код скрипту мінімальної суми в кошику:

```
{* Мінімальна сума замовлення в кошику *
```

```

<script>
    // Функція приймає будь-що, а виводить тільки число - прибираєм
    букви, дублі крапок, коми і т.д. і виводимо завжди число.
    const stringSumToNumber = (str) => {
        const arr = str.split("");
        let correct_arr = [];

        arr.forEach(el => {
            if (el === ' ') return;
            if (el === ';' || el === '.') {
                if (correct_arr.includes('.')) {
                    return;
                }
                return correct_arr.push('.');
            }
            const elToNumber = +el;
            if (!Number.isNaN(elToNumber)) {
                correct_arr.push(elToNumber);
            }
        });
        const res = +correct_arr.join("");
        return res;
    };
    const notificationsOfMinSum = document.querySelectorAll('.min-sum-
msg');
    const btnSubmit = document.querySelector('#btn-submit');
    const totalCartSum = {$cart->total_cart};
    const minOrderSum = stringSumToNumber('{ $lang->min_order_sum }');

    if (totalCartSum >= minOrderSum) {

```

```

        notificationsOfMinSum.forEach(msg => {

            msg.classList.remove('active');
        });

        btnSubmit.classList.remove('active');
    }
</script>

```

Код маніпуляції над товаром:

```

/**
 * Оновлення товару
 */
public function updateProduct($id, $product)
{
    $product = (object) $product;

    $query = $this->db->placeholder("UPDATE __products SET ?%,
last_modified=NOW() WHERE id IN(?@) LIMIT ?", $product, (array) $id,
count((array) $id));

    if ($this->db->query($query)) {
        $result = $this->languages->getDescription($product, 'product');

        if (!empty($result->description)) {
            $this->languages->actionDescription($id, $result->description, 'product',
$this->languages->langId());
        }

        return $id;
    } else {
        return false;
    }
}

* Додавання товару
public function addProduct($product)
{
    $product = (array) $product;

    if (empty($product['url'])) {
        $product['url'] = preg_replace("/[\s]+/ui", '-', $product['name']);
        $product['url'] = strtolower(preg_replace("/[^0-9a-za-я-]+/ui", "",
$product['url']));
    }
}

```

```

while ($this->getProduct((string)$product['url'])) {
    if (preg_match('/(.+)_([0-9]+)$/', $product['url'], $parts)) {
        $product['url'] = $parts[1] . '_' . ($parts[2] + 1);
    } else {
        $product['url'] = $product['url'] . '_2';
    }
}

$product = (object) $product;

$result = $this->languages->getDescription($product, 'product');

if (!empty($result->data)) {
    $product = $result->data;
}

if ($this->db->query("INSERT INTO __products SET ?", $product)) {
    $id = $this->db->insertId();
    $this->db->query("UPDATE __products SET last_modified=NOW(),
position=id WHERE id=?", $id);

    if (!empty($result->description)) {
        $this->languages->actionDescription($id, $result->description, 'product');
    }

    return $id;
} else {
    return false;
}
}

```

Код функціоналу відгуків:

```

/**
 * додати відгук
 */
public function addComment($comment)
{
    $query = $this->db->placeholder("INSERT INTO __comments SET ?,
date=NOW()", $comment);

    if (!$this->db->query($query)) {
        return false;
    }

    $id = $this->db->insertId();

```

```

$comment = (array) $comment;

if (isset($comment['approved'], $comment['object_id']) &&
    $comment['approved'] == 1) {
    switch ($comment['type']) {
        case 'blog':
            $this->db->query("UPDATE __blog SET last_modified=NOW()
WHERE id=?", (int) $comment['object_id']);
            break;
        case 'product':
            $this->db->query("UPDATE __products SET last_modified=NOW()
WHERE id=?", (int) $comment['object_id']);
            break;
        case 'article':
            $this->db->query("UPDATE __articles SET last_modified=NOW()
WHERE id=?", (int) $comment['object_id']);
            break;
        case 'review':
            $this->settings->lastModifyReviews = date('Y-m-d H:i:s');
            break;
    }
}

return $id;
}

/**
 * оновити відгук
 */
public function updateComment($id, $comment)
{
    $query = $this->db->placeholder("UPDATE __comments SET ?% WHERE id
IN(?@) LIMIT 1", $comment, (array) $id);
    $this->db->query($query);

    return $id;
}

/**
 * видалити відгук
 */
public function deleteComment($id)
{
    if (!empty($id)) {

```

```

    $this->db->query("SELECT object_id, type, approved FROM __comments
WHERE id=?", (int) $id);

    $comment = $this->db->result();

    if ($comment->approved == 1) {
        switch ($comment->type) {
            case 'blog':
                $this->db->query("UPDATE __blog SET last_modified=NOW()
WHERE id=?", (int) $comment->object_id);
                break;
            case 'product':
                $this->db->query("UPDATE __products SET last_modified=NOW()
WHERE id=?", (int) $comment->object_id);
                break;
            case 'article':
                $this->db->query("UPDATE __articles SET last_modified=NOW()
WHERE id=?", (int) $comment->object_id);
                break;
        }
    }

    $query = $this->db->placeholder("DELETE FROM __comments WHERE id=?
LIMIT 1", (int) $id);
    $this->db->query($query);
}
}
}
Код для методів блогу:
/**
 * Додати пост блогу
 */
public function addPost($post)
{
    if (!isset($post->date)) {
        $dateQuery = ', date=NOW()';
    } else {
        $dateQuery = "";
    }

    $post = (object) $post;
    $result = $this->languages->getDescription($post, 'blog');

    if (!empty($result->data)) {
        $post = $result->data;
    }
}

```

```

    }

    $this->settings->lastModifyPosts = date("Y-m-d H:i:s");

    $query = $this->db->placeholder("INSERT INTO __blog SET
last_modified=NOW(), ?% $dateQuery", $post);

    if (!$this->db->query($query)) {
        return false;
    }

    $id = $this->db->insertId();

    if (!empty($result->description)) {
        $this->languages->actionDescription($id, $result->description, 'blog');
    }

    return $id;
}

/**
 * Оновити пост блогу
 */
public function updatePost($id, $post)
{
    $post = (object) $post;

    $result = $this->languages->getDescription($post, 'blog');

    if (!empty($result->data)) {
        $post = $result->data;
    }

    $this->settings->lastModifyPosts = date("Y-m-d H:i:s");

    $query = $this->db->placeholder("UPDATE __blog SET last_modified=NOW(),
?% WHERE id IN(?@) LIMIT ?", $post, (array) $id, count((array) $id));
    $this->db->query($query);

    if (!empty($result->description)) {
        $this->languages->actionDescription($id, $result->description, 'blog', $this-
>languages->langId());
    }

    return $id; }

```

Код для методів категорій:

```

/**
 * Додати категорію
 */
public function addCategory($category)
{
    $category = (array) $category;

    if (empty($category['url'])) {
        $category['url'] = preg_replace("/[\s]+/ui", '_', $category['name']);
        $category['url'] = strtolower(preg_replace("/[^0-9a-za-я_]+/ui", "",
$category['url']));
    }

    while ($this->getCategory((string) $category['url'])) {
        if (preg_match('/(.+)_([0-9]+)$/', $category['url'], $parts)) {
            $category['url'] = $parts[1] . '_' . ($parts[2] + 1);
        } else {
            $category['url'] = $category['url'] . '_2';
        }
    }

    $category = (object) $category;

    $result = $this->languages->getDescription($category, 'category');

    if (!empty($result->data)) {
        $category = $result->data;
    }

    $this->db->query("INSERT INTO __categories SET ?", $category);
    $id = $this->db->insertId();
    $this->db->query("UPDATE __categories SET position=id WHERE id=?",
$id);

    if (!empty($result->description)) {
        $this->languages->actionDescription($id, $result->description, 'category'); }

    unset($this->categoriesTree);
    unset($this->allCategories);
    return $id; }
/**
 * Оновити категорію
 */

```

```

public function updateCategory($id, $category)
{
    $category = (object) $category;

    $result = $this->languages->getDescription($category, 'category');

    if (!empty($result->data)) {
        $category = $result->data;
    }

    $query = $this->db->placeholder("UPDATE __categories SET
last_modified=NOW(), ?% WHERE id=? LIMIT 1", $category, (int) $id);
    $this->db->query($query);

    if (!empty($result->description)) {
        $this->languages->actionDescription($id, $result->description, 'category',
$this->languages->langId());
    }

    unset($this->categoriesTree);
    unset($this->allCategories);

    return (int) $id;
}

```

Код для методів функціоналу користувачів:

```

/**
 * Додати користувача
 */
public function addUser($user)
{
    $user = (array) $user;

    if (isset($user['password'])) {
        $user['password'] = md5($this->salt . $user['password'] .
md5($user['password']));
    }

    $query = $this->db->placeholder("SELECT COUNT(*) AS count FROM __users
WHERE email=?", $user['email']);
    $this->db->query($query);

    if ($this->db->result('count') > 0) {
        return false;
    }
}

```

```

    }

    $query = $this->db->placeholder("INSERT INTO __users SET ?", $user);
    $this->db->query($query);

    return $this->db->insertId();
}

/**
 * Оновити користувача
 */
public function updateUser($id, $user)
{
    $user = (array) $user;
    if (isset($user['password'])) {
        $user['password'] = md5($this->salt . $user['password'] .
md5($user['password']));
    }
    $query = $this->db->placeholder("UPDATE __users SET ?% WHERE id=?
LIMIT 1", $user, (int) $id);
    $this->db->query($query);

    return $id;
}

```