

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ СЕМЕНА КУЗНЕЦЯ**

ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

КАФЕДРА ІНФОРМАТИКИ ТА КОМП'ЮТЕРНОЇ ТЕХНІКИ

Рівень вищої освіти
Спеціальність
Освітня програма
Група

Перший (бакалаврський)
Інформаційні системи та технології
Інформаційні системи та технології
6.04.126.010.19.1

ДИПЛОМНИЙ ПРОЄКТ

на тему: «Розроблення інформаційної підсистеми підтримки роботи
магазину комп'ютерної техніки»

Виконала: студентка Марія ЛАЗЕБНИК

Керівник: к.т.н., доцент Наталя БРИНЗА

Рецензент:

к.т.н., доцент кафедри інформаційних систем
ХНЕУ ім. С.Кузнеця
Олег ФРОЛОВ

Харків – 2023 рік

**ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ СЕМЕНА КУЗНЕЦЯ**

| | |
|-------------------------|--|
| Факультет | Інформаційних технологій |
| Кафедра | Інформатики та комп'ютерної техніки |
| Освітній ступінь | Бакалавр |
| Спеціальність | 126 «Інформаційні системи та технології» |

ЗАТВЕРДЖУЮ

Завідувач кафедри

Інформатики та комп'ютерної техніки

_____ проф. Сергій УДОВЕНКО

01 лютого .2023 року

ЗАВДАННЯ

**НА ДИПЛОМНИЙ ПРОЄКТ СТУДЕНТЦІ
ЛАЗЕБНИК Марія Максимівна**

1. Тема проєкту: «Розроблення інформаційної підсистеми підтримки роботи магазину комп'ютерної техніки»

керівник проєкту: Наталя Бринза, к.т.н., доцент

затверджені наказом ректора від 01 лютого .2023 року № 107–С

2. Строк подання студентом проєкту: 01 червня 2023 року

3. Вихідні дані до проєкту: ДСТУ щодо оброблення інформації, літературні джерела, матеріали практики.

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

Розділ 1. Аналіз предметної області та постановка задачі

Розділ 2. Специфікація вимог до модуля «Розроблення інформаційної підсистеми підтримки роботи магазину комп'ютерної техніки»

Розділ 3. Проєктні рішення

Розділ 4. Технічні рішення

5. Перелік графічного матеріалу:

Діаграми: «Розроблення інформаційної підсистеми підтримки роботи магазину комп'ютерної техніки», «Оформлення замовлення», діаграма варіантів використання, фізична, логічна моделі БД.

6. Консультація розділів дипломного проекту

| Розділ | Прізвище, ім'я та посада консультанта | Підпис, дата | |
|--------|---------------------------------------|----------------|------------------|
| | | Завдання видав | Завдання прийняв |
| | | | |

7. Дата видачі завдання: 01 лютого .2023року

КАЛЕНДАРНИЙ ПЛАН

| № з/п | Назва етапів дипломного проекту | Термін виконання етапів проекту | Примітка |
|-------|---|---------------------------------|----------|
| 1 | Розроблення плану дипломного проекту, ознайомлення з літературними джерелами за темою | 01.02.2023 | |
| 2 | Аналіз предметної області | 15.02.2023 | |
| 3 | Розроблення специфікації вимог до модуля | 15.03.2023 | |
| 4 | Розроблення проектних і технічних рішень | 25.04.2023 | |
| 5 | Перевірка чернетки дипломного проекту та внесення зміни до неї керівником | 01.05.2023 | |
| 6 | Оформлення дипломного проекту | 15.05.2023 | |
| 7 | Перевірка якості дипломного проекту на плагіат | 20.05.2023 | |
| 8 | Подання Голові Екзаменаційної комісії щодо захисту дипломного проекту | 1.06.2023 | |

Студент

Марія ЛАЗЕБНИК

Керівник проекту

Наталя БРИНЗА

РЕФЕРАТ

Пояснювальна записка до дипломного проекту: с., рис., табл., додатки 1, джерел.

Тема дипломного проекту: «Розроблення інформаційної підсистеми підтримки роботи магазину комп'ютерної техніки».

Об'єктом дослідження є процес проектування та інформаційної підсистеми підтримки роботи магазину комп'ютерної техніки.

Предметом дослідження є процес розробки та проектування інформаційної підсистеми підтримки роботи магазину комп'ютерної техніки.

Метою дипломного проекту є розроблення інформаційної підсистеми підтримки роботи магазину комп'ютерної техніки.

Модуль розроблений в рамках веб-додатку з використанням наступних засобів проектування та технологій:

У результаті виконання проекту розроблено розроблення інформаційної підсистеми підтримки роботи магазину комп'ютерної техніки

Результати розробки можуть бути використані будь-якою людиною для організації продажі комп'ютерної техніки через інтернет

КОМП'ЮТЕРНА ТЕХНІКА, МАГАЗИН, БАЗА ДАНИХ, РОЗРОБКА, ВЕБСАЙТ, ОНЛАЙН-МАГАЗИН

ABSTRACT

Explanatory note to the diploma project: p., figures, tables, appendices, references.

Theme of the diploma project: "Development of an information subsystem to support the work of a computer store".

The object of research is the process of designing and information subsystem to support the work of a computer store.

The subject of research is the process of development and design of the information subsystem to support the work of a computer store.

The purpose of the diploma project is to develop an information subsystem to support the work of a computer equipment store.

The module is developed within a web application using the following design tools and technologies:

As a result of the project, the development of an information subsystem to support the work of a computer equipment store was developed

The results of the development can be used by anyone to organise the sale of computer equipment via the Internet

COMPUTER EQUIPMENT, STORE, DATABASE, DEVELOPMENT, WEBSITE, ONLINE STORE

ЗМІСТ

| | |
|--|----|
| ВСТУП | 8 |
| 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ «РОЗРОБЛЕННЯ ІНФОРМАЦІЙНОЇ ПІДСИСТЕМИ ПІДТРИМКИ РОБОТИ МАГАЗИНУ КОМП'ЮТЕРНОЇ ТЕХНІКИ» | 9 |
| 1.1. Опис предметної області розроблення | 9 |
| 1.2. Огляд і аналіз існуючих аналогів, що реалізують функції предметної області..... | 14 |
| 2. СПЕЦИФІКАЦІЯ ВИМОГ ДО МОДУЛЯ..... | 17 |
| 2.1. Глосарій..... | 17 |
| 2.2. Розроблення варіантів використання | 18 |
| 2.2.1. Розроблення діаграми варіантів використання..... | 19 |
| 2.2.2. Специфікація варіантів використання..... | 19 |
| 2.2.3. Мокап варіантів використання | 23 |
| 2.3. Специфікація функціональних та нефункціональних вимог | 26 |
| 3. ПРОЕКТНІ ТА ТЕХНІЧНІ РІШЕННЯ | 29 |
| 3.1. Логічна постановка задачі..... | 29 |
| 3.2. Проектування структури бази даних..... | 30 |
| 3.2.1. Концептуальне інфологічне проектування..... | 30 |
| 3.2.2. Проектування глобальної логічної моделі даних..... | 32 |
| 3.2.3. Проектування фізичної моделі бази даних та її програмна реалізація.. | 33 |
| 4. ТЕХНІЧНІ РІШЕННЯ..... | 35 |
| 4.1. Розроблення архітектури програмної системи | 35 |
| 4.2. Тестування програмної системи | 36 |
| 4.3. Розгортання програмного продукту | 41 |
| ВИСНОВКИ..... | 42 |
| СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ..... | 43 |
| ДОДАТКИ..... | 46 |

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

ВВ – варіант використання

ІС – інформаційна система

ПК – персональний комп'ютер

ПП – програмний продукт

ПЗ – програмне забезпечення

ВСТУП

Сучасний ринок комп'ютерної техніки є дуже конкурентним, тому магазини, що продають комп'ютерну техніку, повинні підтримувати високу якість обслуговування та оперативну реакцію на потреби клієнтів. Інформаційні технології можуть стати ключовим фактором в успішному функціонуванні магазину комп'ютерної техніки та забезпечити його конкурентоспроможність на ринку.

Метою даної дипломної роботи є розроблення інформаційної підсистеми, яка допоможе вирішити проблеми, пов'язані з підтримкою роботи магазину комп'ютерної техніки. Застосування даної системи дозволить автоматизувати багато процесів, що виникають у магазинах комп'ютерної техніки, таких як облік товарів, керування складом, управління замовленнями, обробка ремонтів, а також забезпечить швидкий та точний обмін даними з клієнтами та постачальниками.

Дипломна робота буде присвячена вивченню актуальних технологій та методів розробки програмного забезпечення, а також їх практичному застосуванню у створенні інформаційної підсистеми для підтримки роботи магазину комп'ютерної техніки. В результаті виконання дипломної роботи очікується отримання функціональної та ефективної інформаційної підсистеми, яка забезпечить підвищення якості обслуговування та зростання конкурентоспроможності магазину комп'ютерної техніки на ринку.

1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ «РОЗРОБЛЕННЯ ІНФОРМАЦІЙНОЇ ПІДСИСТЕМИ ПІДТРИМКИ РОБОТИ МАГАЗИНУ КОМП'ЮТЕРНОЇ ТЕХНІКИ»

1.1.Опис предметної області розроблення

У зв'язку з швидким розвитком технологій та зміною попиту на комп'ютерну техніку, магазини, які працюють у цій сфері, повинні відповідати на нові виклики та швидко адаптуватися до змінних ринкових умов. Інформаційна підсистема підтримки роботи магазину комп'ютерної техніки може стати важливим інструментом для забезпечення ефективної та безперебійної роботи магазину.

Основними проблемами, з якими стикаються магазини комп'ютерної техніки, є необхідність швидкої обробки великої кількості замовлень та запитів від клієнтів, управління запасами товарів, контроль над збутом та фінансовими показниками. Для вирішення цих проблем може бути розроблена інформаційна підсистема, яка дозволить автоматизувати процеси обліку та управління товарами, забезпечити ефективний контроль за замовленнями та запасами, а також забезпечити звіти про фінансові показники та продажі.

Основними функціями інформаційної підсистеми:

- облік та управління товарами: додавання та видалення товарів, зміна характеристик товарів, моніторинг наявності товарів на складі;
- обробка замовлень: прийняття замовлень від клієнтів, створення замовлень, моніторинг стану замовлень та їх виконання;
- управління запасами: ведення обліку запасів товарів на складі, автоматичне оновлення запасів при продажі товару, автоматичне створення замовлень на додавання нових товарів до запасу;
- фінансовий облік: ведення обліку фінансових показників (наприклад, прибутку, витрат), створення звітів про фінансові показники та продажі;

- керування персоналом: ведення обліку робочого часу працівників, контроль над доступом до інформації та правами користувачів.

Розробка інформаційної підсистеми може допомогти магазину комп'ютерної техніки управляти своєю діяльністю ефективніше та забезпечити більш якісний сервіс для клієнтів. Схема контекстної діаграми бізнес-процесу вебзастосунку на рис. 1.1

Таблиця 1.1

Характеристика процесу «Підтримки роботи магазину комп'ютерної техніки»

| Назва характеристики | Значення характеристики |
|----------------------|---|
| Ім'я бізнес-процесу | Підтримки роботи магазину комп'ютерної техніки |
| Основні учасники | Адміністратор |
| Вхідна подія | Вхід до веб-застосунку |
| Вхідні документи | Складські запаси, Замовлення клієнтів |
| Вихідна подія | Сформоване замовлення |
| Вихідні документи | Підтвердження замовлення, Залишок товарів на складі |

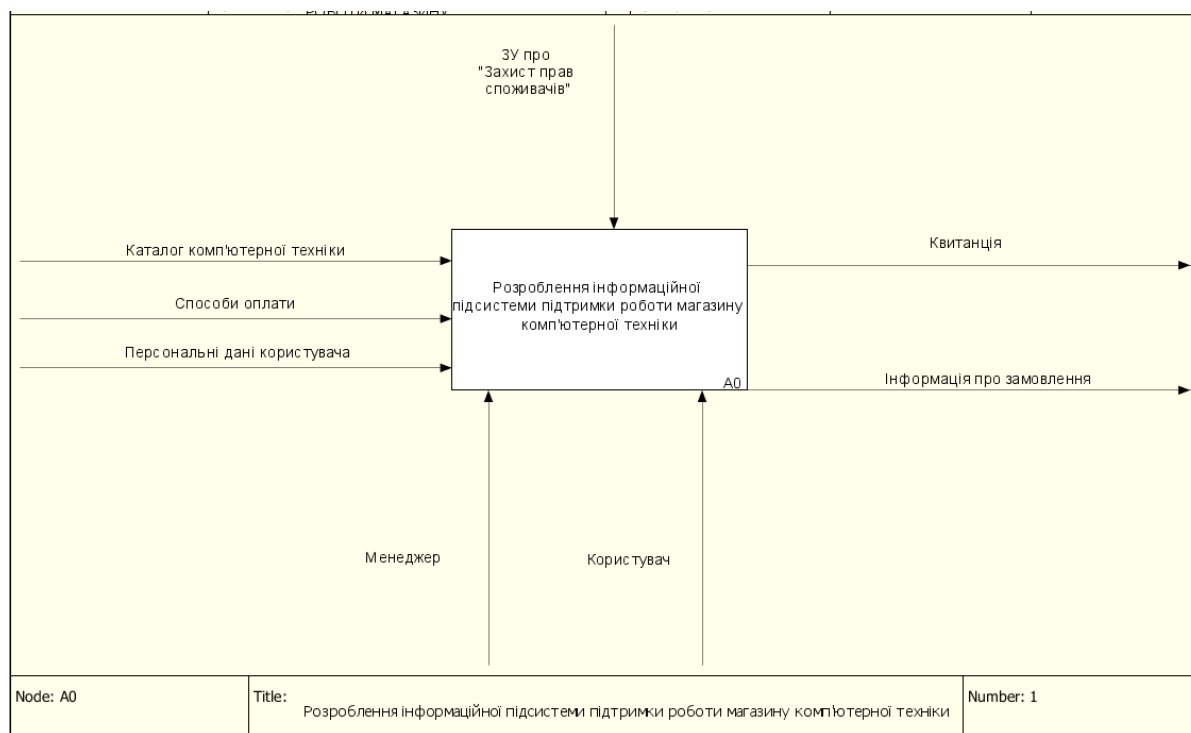


Рисунок 1.1 – Діаграма бізнес-процесу «Розроблення інформаційної підсистеми підтримки роботи магазину комп'ютерної техніки»

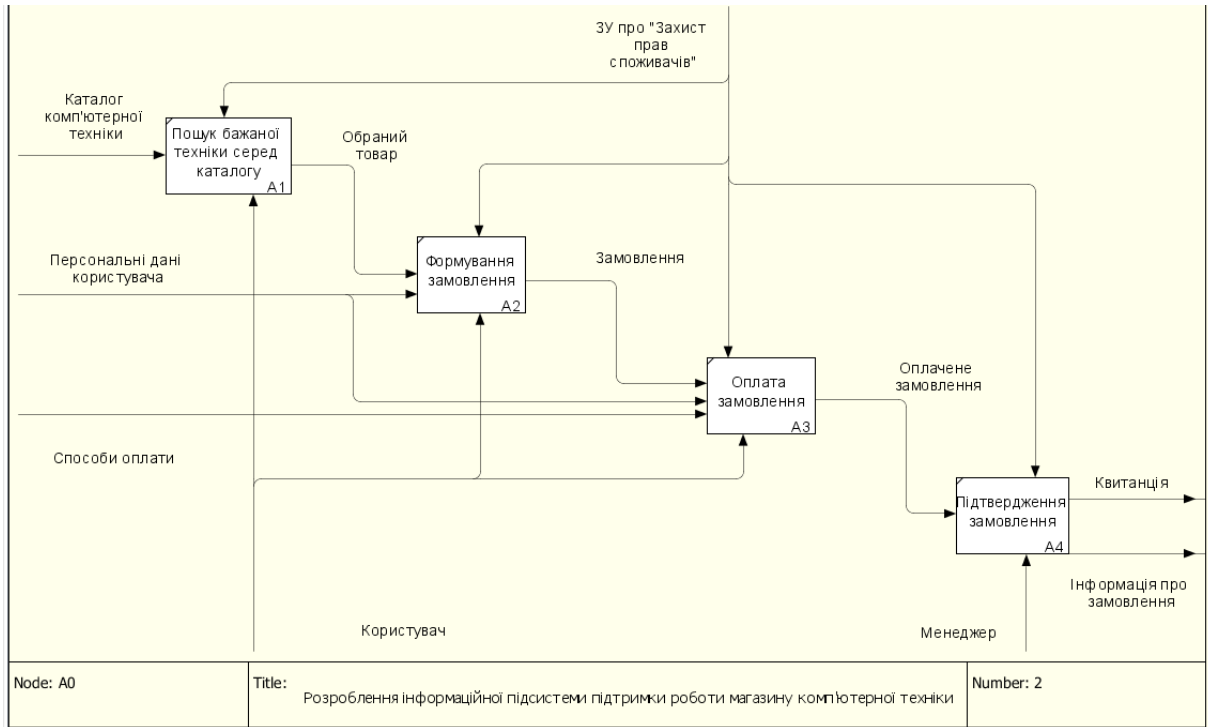


Рисунок 1.2 – Діаграма декомпозиції бізнес-процесу «Розроблення інформаційної підсистеми підтримки роботи магазину комп'ютерної техніки»

Нижче, на рис. 1.3 наведено декомпозицію бізнес-процесу «Формування замовлення», а у табл. 1.2-1.5, наведено детальну характеристику процесів, що було виявлено під час декомпозиції основного бізнес-процесу

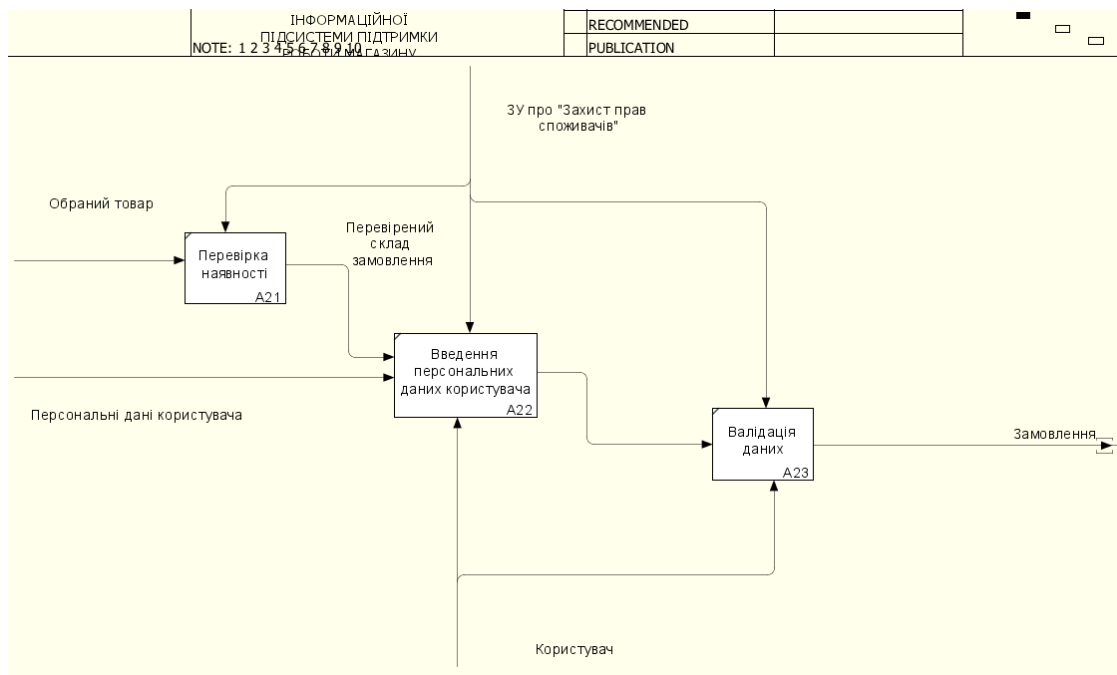


Рисунок 1.3– Діаграма декомпозиції бізнес-процесу «Оформлення замовлення»

Таблиця 1.2

Характеристика процесу «Прийом та розміщення замовлень»

| Назва характеристики | Значення характеристики |
|----------------------|--------------------------------|
| Ім'я бізнес-процесу | Прийом та розміщення замовлень |
| Основні учасники | Адміністратор |
| Вхідна подія | Клієнт замовив товар |
| Вихідна подія | Розміщене замовлення |
| Вихідні документи | Замовлення |

Таблиця 1.3

Характеристика процесу «Обробка замовлення»

| Назва характеристики | Значення характеристики |
|----------------------|---|
| Ім'я бізнес-процесу | Обробка замовлення |
| Основні учасники | Адміністратор |
| Вхідна подія | Розміщене замовлення |
| Вихідна подія | Замовлення оброблено |
| Вихідні документи | Підтвердження замовлення, Товари до відгрузки |

Таблиця 1.4

Характеристика процесу «Відгрузка товарів»

| Назва характеристики | Значення характеристики |
|----------------------|---------------------------|
| Ім'я бізнес-процесу | Відгрузка товарів |
| Основні учасники | Адміністратор |
| Вхідна подія | Замовлення оброблено |
| Вихідна подія | Відправлення замовлення |
| Вихідні документи | Залишок товарів на складі |

Характеристика процесу «Відгрузка товарів»

| Назва характеристики | Значення характеристики |
|----------------------|---------------------------|
| Ім'я бізнес-процесу | Відгрузка товарів |
| Основні учасники | Адміністратор |
| Вхідна подія | Замовлення оброблено |
| Вихідна подія | Відправлення замовлення |
| Вихідні документи | Залишок товарів на складі |

Діаграма потоків даних системи «Розроблення інформаційної підсистеми підтримки роботи магазину комп'ютерної техніки» й зовнішні об'єкти, з якими цей модуль взаємодіє (ці взаємодії позначені за допомогою вхідних і вихідних інформаційних потоків) наведена на рис. 1.4.

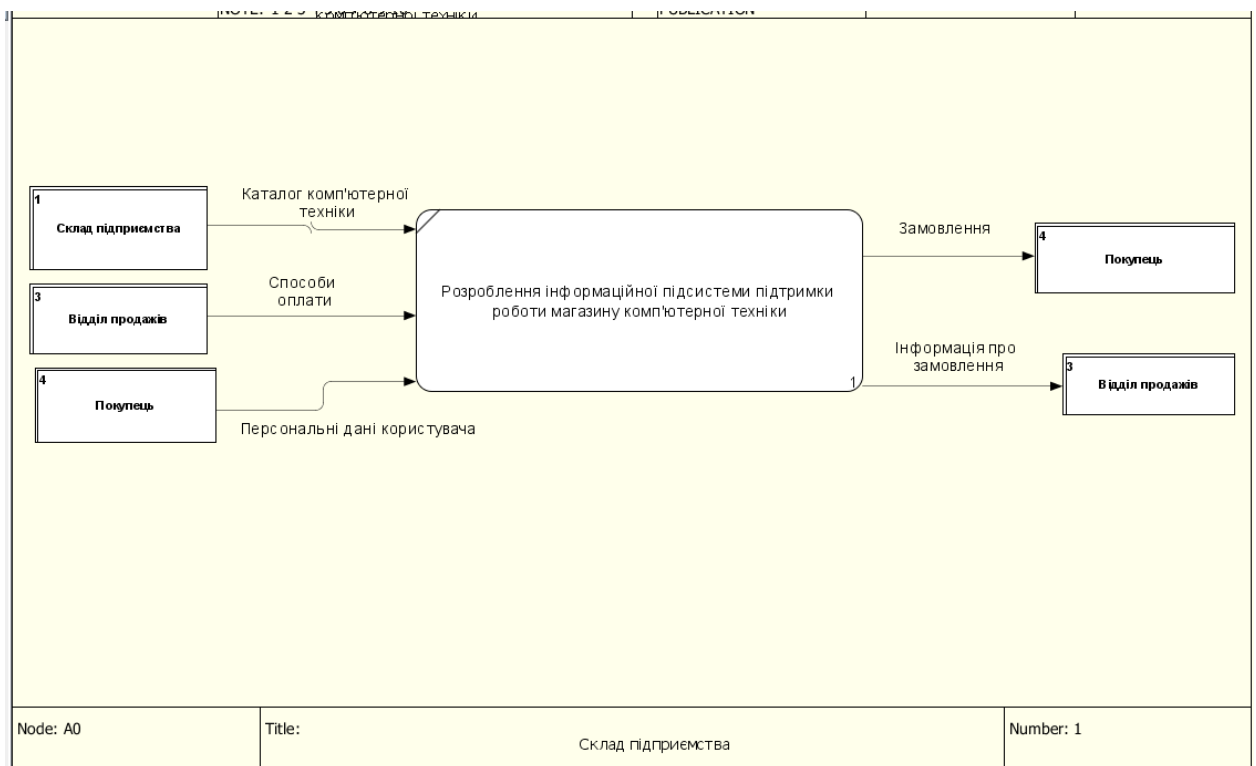


Рисунок 1.4 – Контекстна діаграма у стандарті DFD

Побудову моделі системи «Інтернет-магазин матеріалів для зовнішньої реклами» у стандарті моделювання DFD було виконано на основі тих робіт,

які були попередньо виділені у стандарті моделювання IDEF0. Результати побудови діаграм потоків даних в рамках системи «Інтернет-магазин матеріалів для зовнішньої реклами» подані на рисунку 1.5

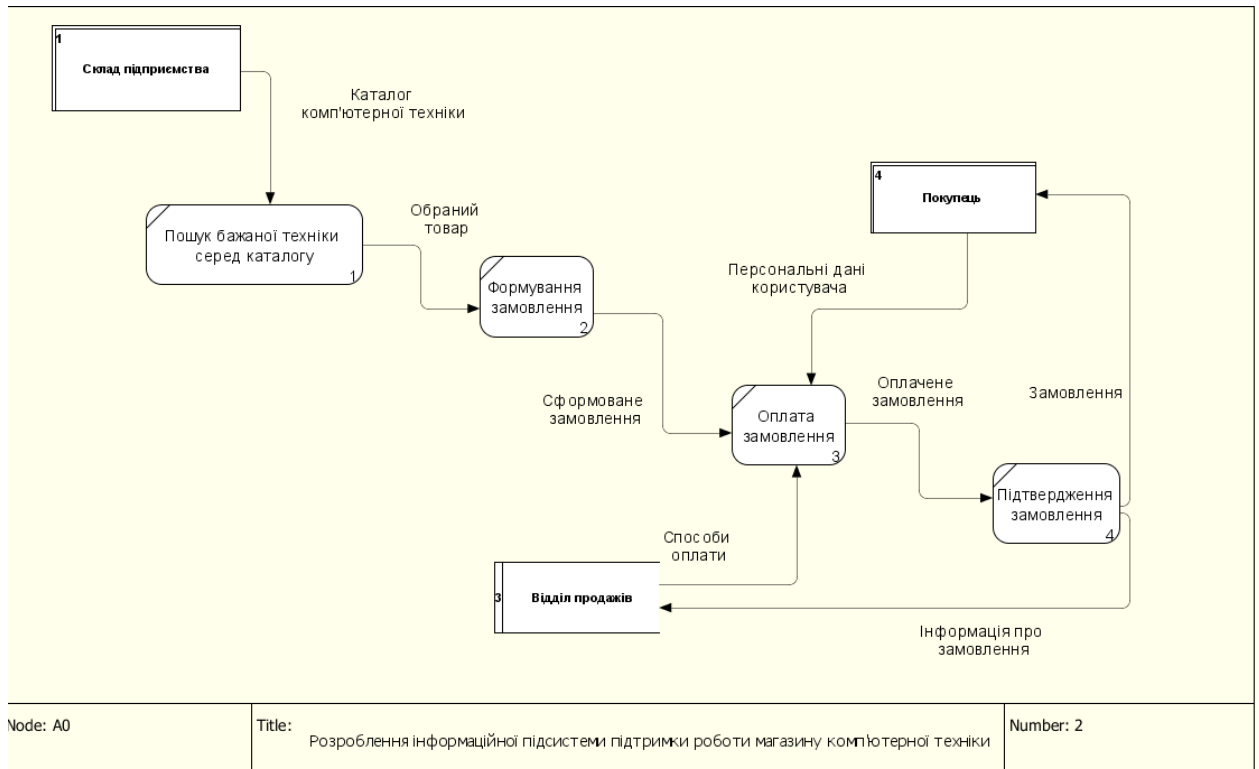


Рисунок 1.5 – Діаграма декомпозиції контекстної діаграми в стандарті DFD

1.2. Огляд і аналіз існуючих аналогів, що реалізують функції предметної області

На вже сформованому ринку веб-додатків з продажу комп'ютерної техніки є ключові представники, що будуть включені до порівняння. Так, слід виділити наступних 3 існуючих рішення. Порівняльну характеристику та скріншоти додатків наведено відповідно у табл. 1.6 та рис. 1.6 – 1.8.

Порівняння характеристик програмних застосунків

| Застосунок | Розетка | Мойо. | Сота |
|--|---------|-------|------|
| 1 | 2 | 3 | 4 |
| Наявність оффлайн магазинів | + | + | + |
| Наявність бонусної програми | + | + | - |
| Наявність різних способів оплати | + | + | + |
| Зберігання даних кредитної карти в базі даних магазину | + | + | - |
| Можливість купити товар в кредит/растрочку | + | + | + |
| Інтуїтивно зрозумілий інтерфейс | + | + | + |
| Наявність блогу | + | + | + |
| Наявність оффлайн магазинів | + | + | + |

Усі інтернет-магазини мають схожий дизайн, білий задній фон та верхній та нижній футери виконані в фірмових кольорах сайту. Sota винесла адреси магазинів, контактний телефон та графік роботи магазинів винесла в верхній футер, в розетці вся інформація знаходиться в нижньому футері в вигляді посилань на інтернет сторінки з необхідною інформацією. В мойо ця інформація міститься в верхньому футері, телефон колл-центру міститься там у вигляді номера, а адреси та графік роботи магазинів в вигляді посилання на окрему веб сторінку.

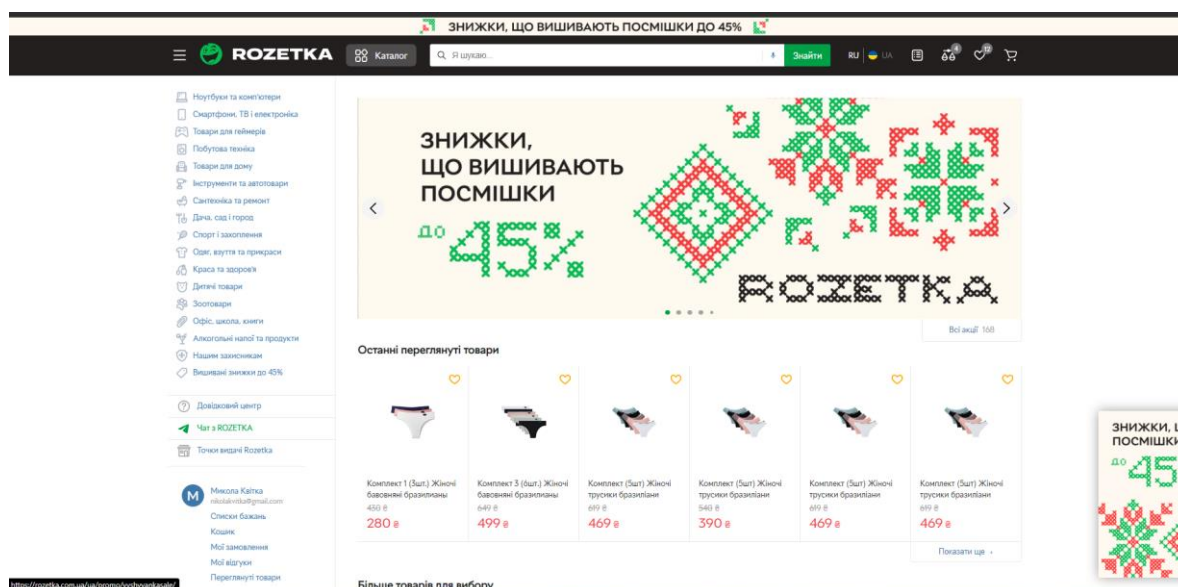


Рисунок 1.6 – Головна сторінка сайту розетка

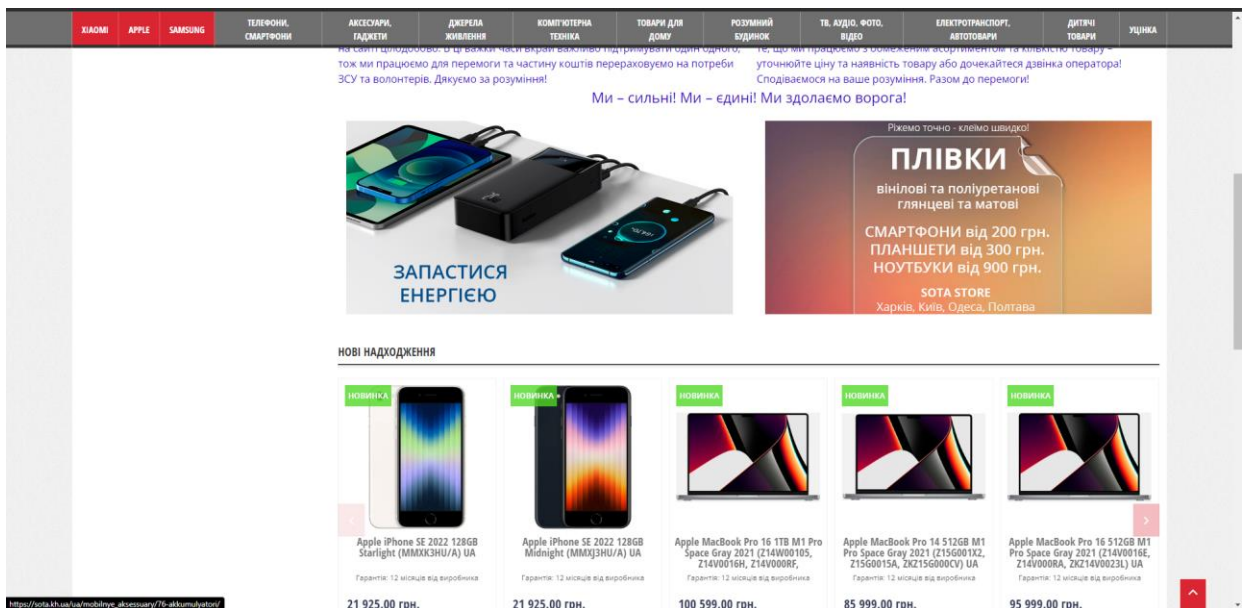


Рисунок 1.7 – Головна сторінка сайту сота

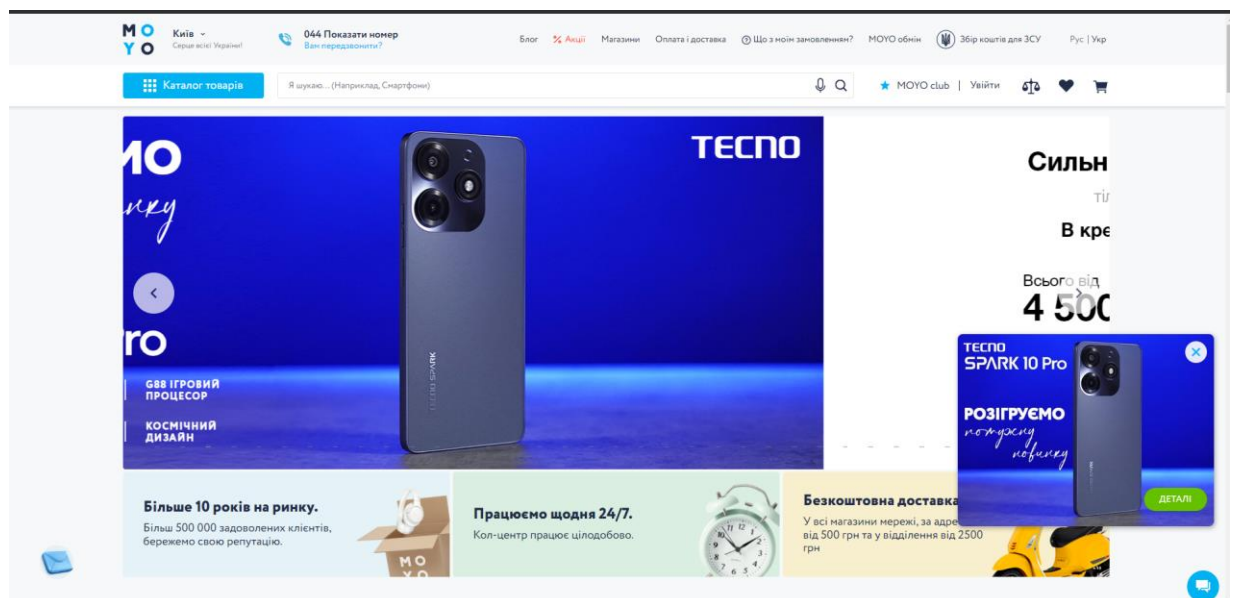


Рисунок 1.8 – Головна сторінка сайту мойо

Проаналізувавши наявні аналоги, можна зробити висновок що на ринку існує велика конкуренція. Задля того, щоб інтернет-магазин був спроможним конкурувати, треба застосувати успішні практики інших магазинів.

- наявність оффлайн магазину або точки видачі в різних містах.
- можливість придбати товар р кредит/рострочку.
- можливість оплатити товар онлайн та при отриманні.
- для безпеки даних клієнтів підключити перевірений стороній сервіс оплати картою онлайн.
- розробити мінімалістичний дизайн, який буде інтуїтивний зрозумілим для користувача.

2. СПЕЦИФІКАЦІЯ ВИМОГ ДО МОДУЛЯ

2.1. Глосарій

Глосарій – список понять в специфічній області знання з їх визначеннями. Словник основних термінів, які використовуються в проєкті, наведено у табл. 2.1.

Таблиця 2.1

Глосарій

| Термін | Опис терміну |
|---|--|
| 1 | 2 |
| 1. Основні поняття та категорії предметної області та проєкту | |
| Вебзастосунок | Це програмне забезпечення, яке розробляється для використання через веб-браузер. У цьому випадку вебзастосунок буде служити як онлайн-каталог моніторів |
| Продукти | це комп'ютерна техніка, яку магазин продає. Продукти можуть включати персональні комп'ютери, ноутбуки, планшети, смартфони, принтери, сканери та інші пристрої |
| Аналітика | це процес збору, аналізу та інтерпретації даних для зроблення прийняття рішень. Аналітика може допомогти зі збором статистичних даних про надання консультацій та виявлення тенденцій, які можуть допомогти поліпшити модуль. |
| Категорії продуктів | продукти можна розподілити на категорії для зручності навігації та організації товарів. Наприклад, категорії можуть включати "Ноутбуки", "Смартфони", "Акcesуари", "Програмне забезпечення" тощо |
| Замовлення | замовлення є основним елементом роботи магазину. Клієнти можуть робити замовлення на продукти, інформація про які зберігається в системі. Замовлення можуть містити деталі, такі як список продуктів, кількість, ціна, адреса доставки та ін |
| Склад | це місце, де зберігаються фізичні продукти. Інформація про наявність продуктів на складі, їх кількість та розташування може бути відстежена в системі |
| Постачальники | це компанії або особи, які постачають продукти магазину. Інформація про постачальників, така як контактні дані, умови постачання та ціни, може бути збережена в системі |
| Система оформлення замовлення | це частина підсистеми, яка дозволяє клієнтам оформити замовлення, вибрати продукти, вказати адресу доставки та провести оплату |
| Звіти та аналітика | ця функція дозволяє генерувати звіти про продажі, відстежувати тренди, аналізувати популярність певних |

| 1 | 2 |
|--------------------------------|---|
| | продуктів, оцінювати ефективність маркетингових акцій та вирішувати стратегічні питання |
| Управління запасами | функція, яка допомагає відстежувати рух продуктів на складі, контролювати наявність запасів, робити замовлення у постачальників та забезпечувати наявність товарів для клієнтів |
| 2. Користувачі системи | |
| Клієнт | особа, яка використовує модуль експертно-консультативних послуг для студентів вищих навчальних закладів на основі веб-технологій. |
| Адміністратор | особа, яка відповідає за налагодження та підтримку роботи модуля. |
| 3. Вхідні та вихідні документи | |
| Персональні дані клієнта | вхідний документ, який містить інформацію про клієнта. |
| Каталог моніторів | це перелік або колекція інформації про різні моделі моніторів, яка надається користувачам для ознайомлення та вибору. В онлайн-каталозі моніторів для персональних комп'ютерів кожен запис представляє окремий монітор і містить докладну інформацію про його технічні характеристики, включаючи розмір екрану, роздільну здатність, тип панелі, підключення, характеристики кольору, ціну та інші важливі параметри. Каталог може включати також фотографії, відео та описи кожного монітора, щоб користувачі мали можливість більш детально ознайомитися з продуктом перед покупкою |
| Сформоване замовлення | вихідний документ, який містить інформацію про замовлення клієнта. |

2.2. Розроблення варіантів використання

Варіант використання – це функціональність системи, що дозволяє користувачеві отримати будь-який значущий для нього, відчутний і вимірний результат. Кожен варіант використання відповідає окремому сервісу, що надається моделюваною системою у відповідь на запит користувача, тобто визначає, як використовувати цю систему .

Дизайн або стандартний елемент варіанта використання UML використовується для визначення загальної поведінки системи або будь-якого

іншого об'єкта предметної області без урахування внутрішньої структури цього об'єкта. Кожен варіант використання визначає послідовність дій, які має виконувати спроектована система при взаємодії з відповідним актором. Схема варіантів може бути доповнена пояснювальним текстом, що розкриває зміст або семантику складових його компонентів. Такий пояснювальний текст називається запискою або сценарієм.

Окремий варіант використання позначається на схемі еліпсом, який містить його коротку назву або назву у вигляді дієслова з пояснювальними словами.

Мета варіанту використання – визначити повний аспект або фрагмент поведінки сутності без розкриття внутрішньої структури об'єкта. Такою сутністю може бути вихідна система або будь-який інший елемент моделі, який має власну поведінку, наприклад підсистема або клас у моделі системи.

2.2.1. Розроблення діаграми варіантів використання.

Діаграма варіантів використання представлена на рис. 2.3.

2.2.2. Специфікація варіантів використання.

Опис основних варіантів використання (ВВ) наведений у табл. 2.2 - 2.8.

Таблиця 2.2

Варіант використання «Пошук товарів»

| Контекст використання | Пошук товарів |
|-----------------------|---|
| 1 | 2 |
| Дійові особи | Клієнт |
| Передумова | Клієнт зайшов на сайт |
| Тригер | Клієнт зайшов на сайт |
| Дійові особи | Клієнт |
| Сценарій | 1 – Клієнт натиснув кнопку «Search». 2 – Ввести свій пошуковий запит. 3 – Натиснути кнопку пошуку |
| Постумова | Веб застосунок відобразив клієнту сторінку з товарами, які підходять до його запиту. |

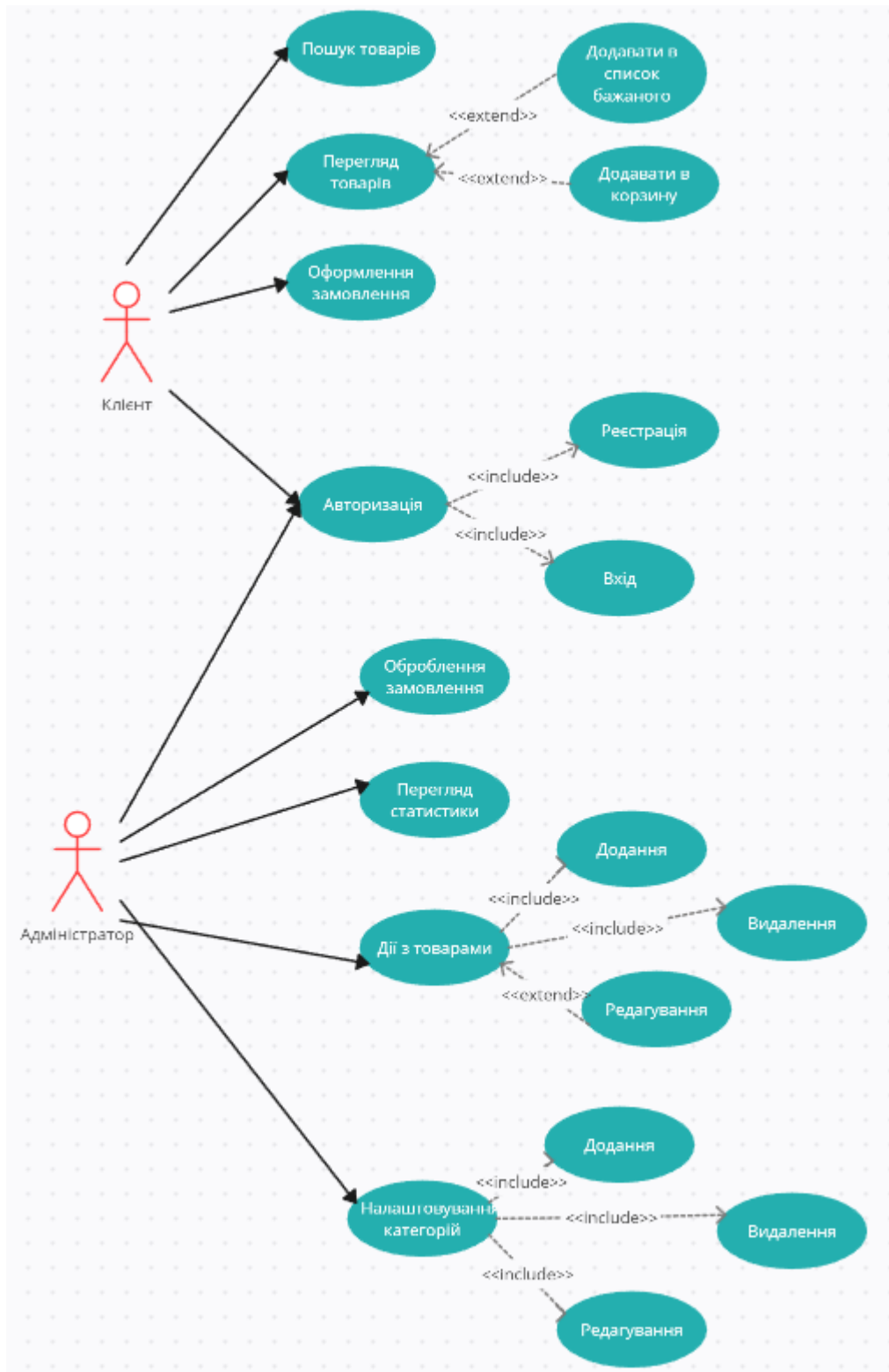


Рис. 2.1. Діаграма варіантів використання наведена

Таблиця 2.3

Варіант використання «Перегляд товарів»

| Контекст використання | Перегляд товарів |
|-----------------------|--|
| 1 | 2 |
| Дійові особи | Клієнт |
| Передумова | Клієнт ввів пошуковий запит |
| Тригер | Клієнт знайшов потрібний товар |
| Сценарій | 1 – Натискання на картинку товару 2 – Перегляд інформації про товар |
| Постумова | Адміністратор згідно з потребами обробив замовлення або виконав операцію з інформацією про монітори. |

Таблиця 2.4

Варіант використання «Оформлення замовлення»

| Контекст використання | Оформлення замовлення |
|-----------------------|--|
| 1 | 2 |
| Дійові особи | Клієнт |
| Передумова | Клієнт зайшов на сайт та додав монітор до кошика |
| Тригер | Натискання кнопки «Оформити замовлення» |
| Сценарій | 1 – Ввести необхідну інформацію |
| | 2 – Натиснути кнопку «Оформити» |
| Постумова | Замовлення з'явилося в списку замовлень адміністратора |

Таблиця 2.5

Варіант використання «Оброблення замовлення»

| Контекст використання | Оброблення замовлення |
|-----------------------|-------------------------------------|
| 1 | 2 |
| Дійові особи | Адміністратор |
| Передумова | Адміністратор авторизувався на сайт |
| Тригер | Пришло нове замовлення |

Закінчення табл.2.5

| 1 | 2 |
|-----------|--|
| Сценарій | 1 - Натискання кнопки «Sales» 2 - Натискання кнопки «Orders» 3 – Обрав нове замовлення 4 – Зміна статусу замовлення |
| Постумова | Інформація про замовлення була внесена в систему й статус замовлення змінився. |

Таблиця 2.6

Варіант використання «Авторизація»

| | |
|-----------------------|--|
| Контекст використання | Авторизація |
| Дійові особи | Клієнт |
| Передумова | Користувач зайшов на сайт |
| Тригер | Натиснення кнопки «My Account» та ввід даних користувача |
| Сценарій | 1 - Натискання кнопки «My Account» 2 - Натискання кнопки «Login» 3 – Ввід даних користувача 4 - Натискання кнопки «Login» |
| Постумова | В разі коли користувач ввів вірний логін та пароль, його буде перенаправлено на особистий кабінет. В іншому випадку, що введена інформація не вірна. |

Таблиця 2.7

Варіант використання «Перегляд статистики»

| | |
|-----------------------|--|
| Контекст використання | Перегляд статистики |
| Дійові особи | Адміністратор |
| Передумова | Адміністратор авторизувався на сайт |
| Тригер | Перегляд статистики |
| Сценарій | 1 – Адміністратор перейшов на сторінку «Dashboard» 2 – Перегляд статистики в полі «Sales Analytics» |
| Постумова | Статистика продажів переглянуто |

Таблиця 2.8

Варіант використання «Дії з товаром»

| | |
|-----------------------|--|
| Контекст використання | Дії з товаром |
| Дійові особи | Адміністратор |
| Передумова | Адміністратор авторизувався на сайт |
| Тригер | Поступив новий товар |
| Сценарій | 1 – Адміністратор перейшов на сторінку «Product» 2 – Натиснення кнопки «Add» 3 – Введення інформації про товар 4 – Натиснення кнопки «Save» |
| Постумова | Товар був успішно доданий до бази даних інтернет магазину |

2.2.3. Мокап варіантів використання

Мокап (або макет) - це візуальна модель, що відображає зовнішній вигляд та компоненти продукту або системи. Він служить для демонстрації ідеї, концепції або дизайну перед фактичною реалізацією проекту.

Мокап може бути створений у формі зображень, скріншотів, прототипів або інтерактивних додатків, що дозволяють користувачам взаємодіяти з макетом. Використання мокапів дозволяє замовникам, дизайнерам та розробникам отримати загальне уявлення про продукт, зрозуміти його функціонал та внутрішню структуру, внести зміни та вдосконалити дизайн до фінальної реалізації.

Мокапи широко використовуються у різних сферах, таких як веб-дизайн, розробка мобільних додатків, інтерфейси користувача, графічний дизайн та інші процеси, де важливим є візуальне представлення продукту. Мокап використання «Головна сторінка» наведена на рис. 2.4.

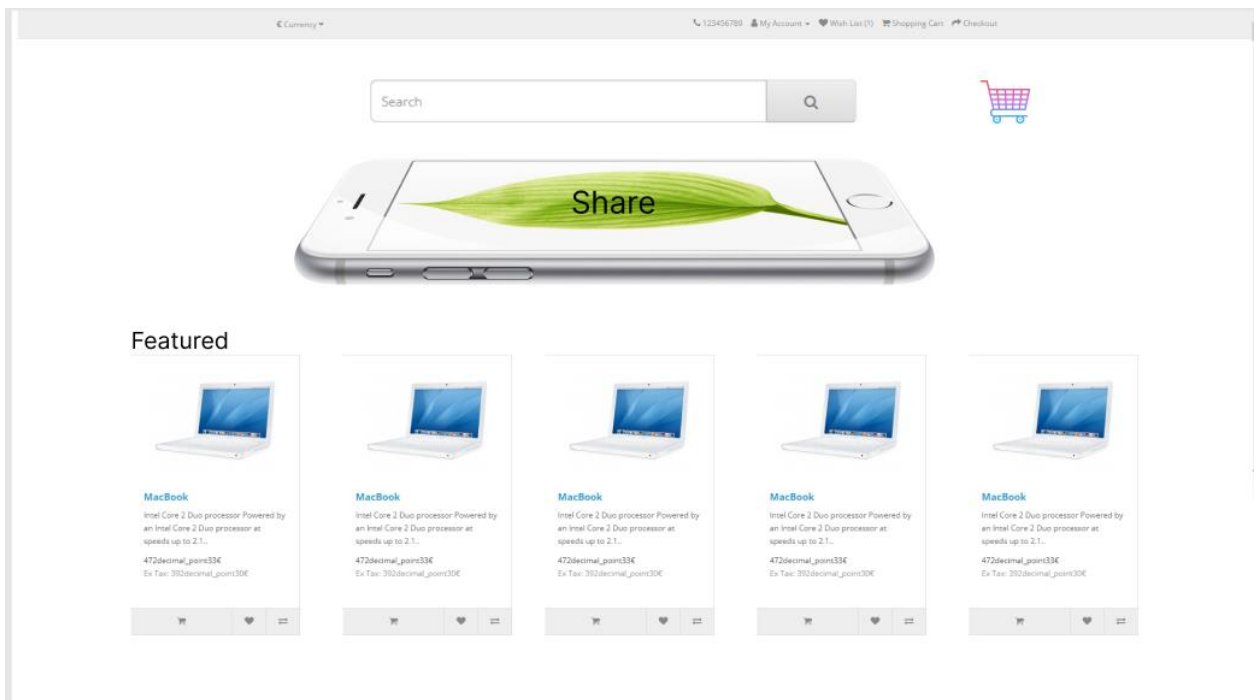


Рис. 2.2. Варфрейм «Головна сторінка»

Структура головного меню програми наведена на рис. 2.4.

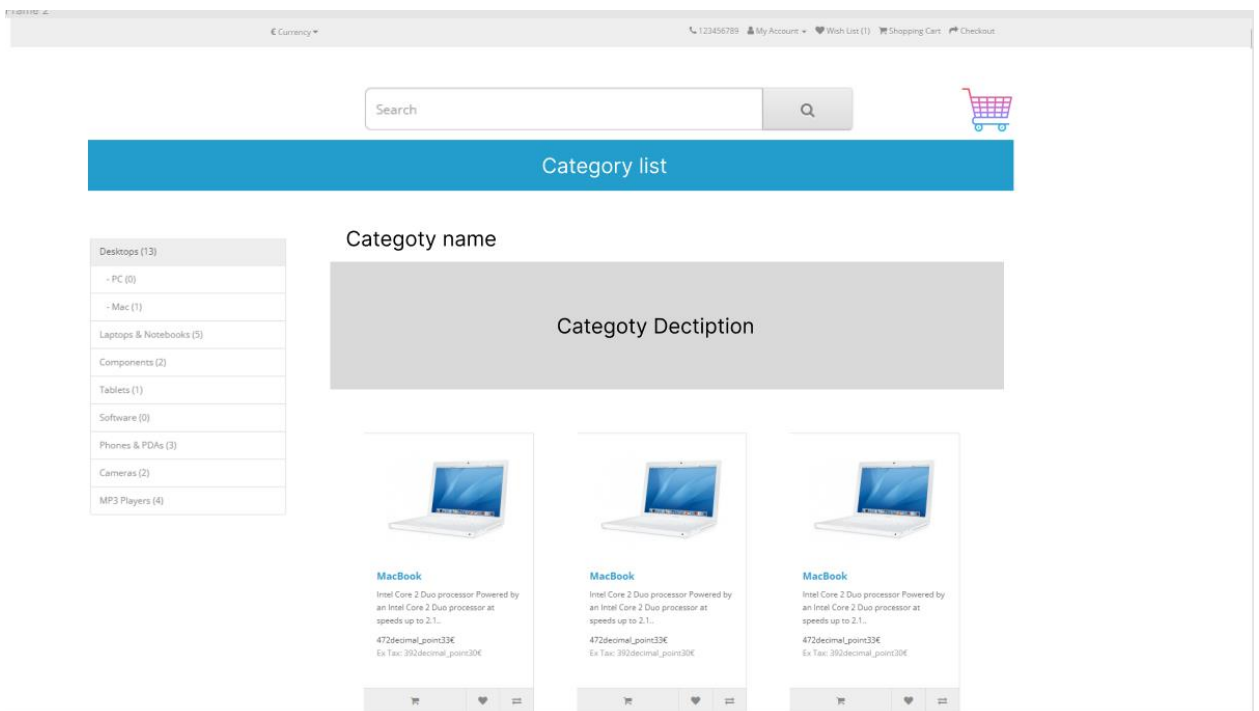


Рис. 2.3. Варфрейм «Сторінка категорії»

Варфрейм основних варіантів використання наведена на рис. 2.5 - 2.6.

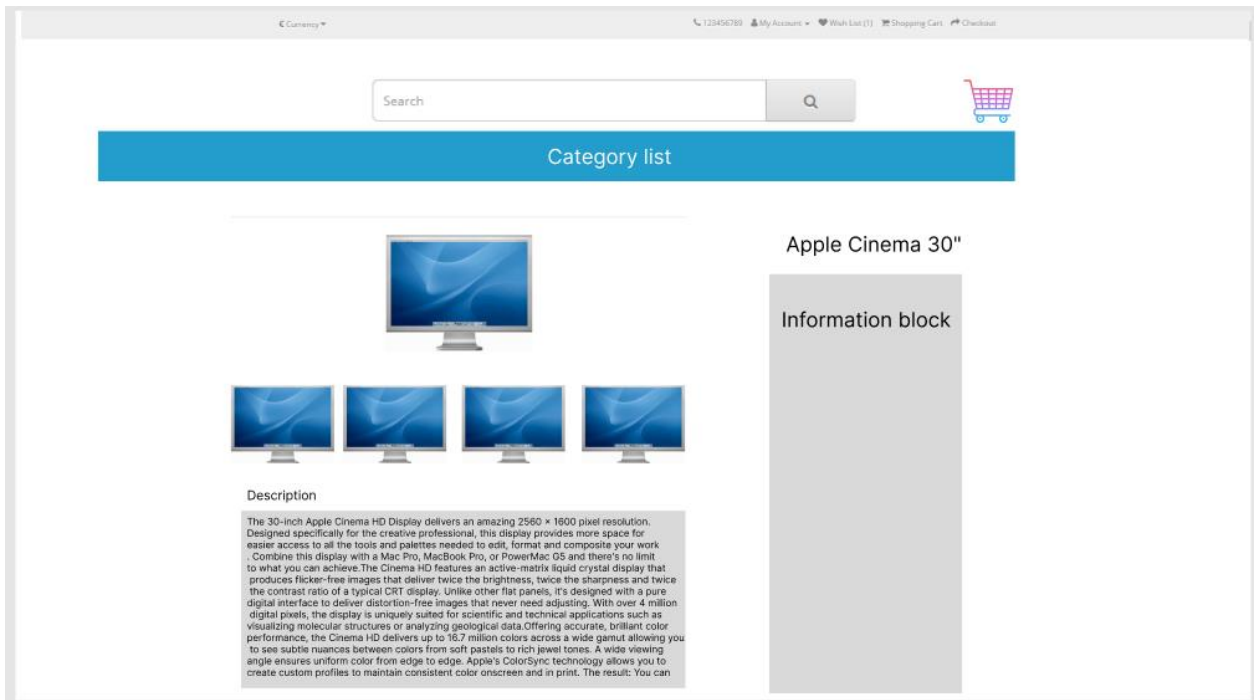


Рис. 2.4. Варфрейм «Інформація про товар»

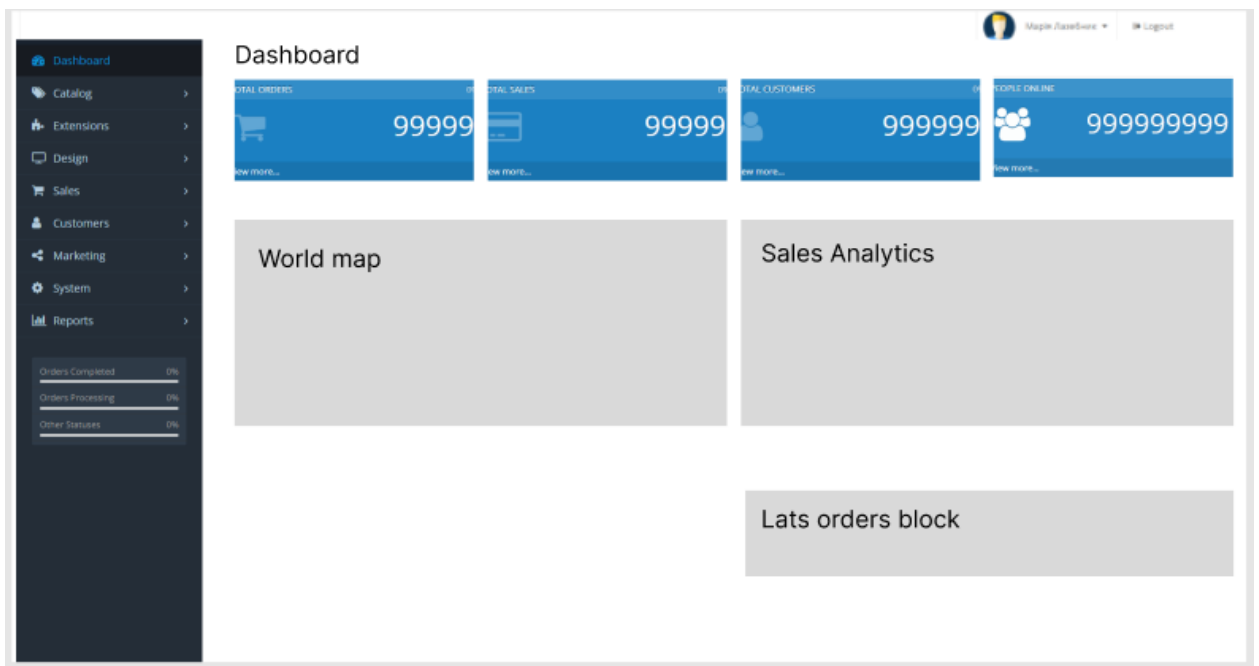


Рис. 2.7. Варфрейм «Dashboard»

2.3. Специфікація функціональних та нефункціональних вимог

Вимоги до програмного забезпечення – це сукупність тверджень відносно атрибутів, властивостей або якостей програмної системи, що підлягає реалізації. Існують функціональні та нефункціональні вимоги до системи.

Специфікація вимог до програмного забезпечення – це повний опис поведінки системи що розробляється. Він включає набір прецедентів, що описують всі взаємодії користувача з системою. Прецеденти також відомі як функціональні вимоги. На додачу до прецедентів, специфікація вимог також містить не функціональні (чи додаткові вимоги).

Не функціональні вимоги є вимогами які накладають обмеження на проект, чи реалізацію (такі як вимоги інженерії продуктивності, стандарти якості, чи обмеження проектування).

Специфікація функціональних вимог наведена в табл. 2.9. Специфікація нефункціональних вимог наведена в табл. 2.10.

Таблиця 2.9

Специфікація функціональних вимог

| Ідентифікатор вимоги | Назва вимоги | Атрибути вимог | |
|----------------------|---|----------------|------------|
| | | Пріоритет | Складність |
| UC-1 | Наявність авторизації та автентифікації | обов'язкова | середня |
| UC-2 | Наявність розподілення користувачів ролями | обов'язкова | середня |
| UC-3 | Додання товарів в корзину | обов'язкова | середня |
| UC-4 | Додавання нових товарів | рекомендована | середня |
| UC-5 | Видалення товарів | обов'язкова | середня |
| UC-6 | Перегляд замовлень | обов'язкова | середня |
| UC-7 | Наявність аналітики продажів | рекомендована | низька |
| UC-8 | Наявність пагінації на сторінках з товарами | рекомендована | низька |

Специфікація нефункціональних вимог

| Ідентифікатор вимоги | Назва вимоги (варіанта використання) | Атрибути вимог | | |
|-------------------------------|--|----------------|-----------|---------------------|
| | | Пріоритет | Трудність | Контакт/ Виконавець |
| 1 | 2 | 3 | 4 | 5 |
| 1. Застосовність | | | | |
| NFR-1 | Вимірний час відгуку для типових завдань – не більше 3 сек. | Високий | Висока | Розробник |
| NFR-2 | Час завантаження веб-сайту – не більше 5 сек. | Високий | Висока | Розробник |
| NFR-3 | Простий та функціональний інтерфейс | Високий | Висока | Розробник |
| NFR-4 | Час, необхідний для навчання звичайних і досвідчених користувачів – не більше 2 год. | Високий | Висока | Розробник |
| 2. Надійність | | | | |
| NFR-5 | Доступність – 20 % | Високий | Середня | Розробник |
| NFR-6 | Середній час без-відмовної роботи – 7 днів | Високий | Висока | Розробник |
| NFR-7 | Середнє напрацювання до ремонту – 7 днів | Високий | Висока | Розробник |
| 3. Робочі характеристики | | | | |
| NFR-8 | Продуктивність – 10 шт. | Середній | Середня | Розробник |
| NFR-9 | Місткість – 10 000/міс. | Середній | Середня | Розробник |
| NFR-10 | Режими зниженої продуктивності – 5 шт. | Середній | Середня | Розробник |
| 4. Експлуатаційна придатність | | | | |
| NFR-11 | Дотримання стандартів кодування, що дозволяє забезпечити сумісність з іншими системами та програмними засобами. | Високий | Висока | Розробник |
| NFR-12 | Використання стандартизованих бібліотек класів, що дозволяє зменшити час розробки та підвищити стійкість до помилок. | Середній | Середня | Розробник |
| 5. Проектні обмеження | | | | |
| NFR-13 | Програмування за допомогою конструктора | Середній | Середня | Розробник |

Закінчення табл. 2.10

| 1 | 2 | 3 | 4 | 5 |
|---|---|----------|---------|-----------|
| 6. Вимоги до документації, призначеної для користувача, і до системи допомоги | | | | |
| NFR-14 | Документація та система допомоги мають бути постійно оновлюваними та забезпечувати користувачів актуальною інформацією. | Низький | Низька | Розробник |
| 7. Інтерфейси | | | | |
| 7.1 Інтерфейси користувача | | | | |
| NFR-15 | Інтерфейс сумісний з різними видами пристроїв | Високий | Висока | Розробник |
| NFR-16 | Зрозуміла та логічна структура, відповідна до природного мислення користувача. | Середній | Середня | Розробник |
| NFR-16 | Інтерфейс надає доступ до всіх функцій системи, які необхідні для виконання завдань користувача. | Високий | Висока | Розробник |
| NFR-17 | Інтерфейс стійкий до помилок користувачів та надійним у виконанні функцій. | Високий | Висока | Розробник |
| 7.2 Апаратні інтерфейси | | | | |
| NFR-18 | Мережевий адаптер з передачею 250 Мбіт/с | Середній | Середня | Розробник |
| NFR-19 | Процесор 700 МГц, або вище. | Середній | Середня | Розробник |
| 7.3 Програмні інтерфейси | | | | |
| NFR-20 | Операційна система будь-яка | Середній | Середня | Розробник |
| NFR-21 | Наявність усіх компонентів для нормальної роботи в Internet | Високий | Висока | Розробник |
| 7.4 Комунікаційні інтерфейси | | | | |
| NFR-22 | Якісне підключення до мережі Internet | Високий | Висока | Розробник |

Під час роботи над другим розділом дипломної роботи було створено глосарій термінів та понять, що використовуються в проекті; проаналізовано основні варіанти використання застосунку та побудовано діаграму варіантів використання з подальшою деталізацією основних варіантів використання; визначено функціональні та нефункціональні вимоги до веб сайту, їх пріоритети та складність реалізації.

3. ПРОЕКТНІ ТА ТЕХНІЧНІ РІШЕННЯ

3.1. Логічна постановка задачі

Звітність та аналітика є важливою частиною інформаційної підсистеми підтримки роботи магазину комп'ютерної техніки. Модель звітності та аналітики включає розробку математичних моделей і алгоритмів для генерації звітів, аналізу даних та отримання цінної інформації для прийняття рішень. Основні складові моделі звітності та аналітики включають наступні елементи:

- Збір та збереження даних: Модель повинна включати механізми для збору та збереження даних про продажі, клієнтів, запаси, вартість товарів тощо. Ці дані можуть бути зібрані з різних джерел, включаючи систему оформлення замовлення, склад, фінансові системи тощо.

- Аналіз даних: Модель повинна включати методи та алгоритми для аналізу накопичених даних. Це можуть бути методи статистичного аналізу, машинного навчання, дашбордів, засобів візуалізації тощо. Аналіз даних може включати пошук трендів, виявлення кореляцій, кластеризацію клієнтів, аналіз популярності продуктів тощо.

- Генерація звітів: Модель повинна включати механізми для генерації різноманітних звітів на основі аналізу даних. Звіти можуть включати зведену інформацію про продажі, прибуток, популярність продуктів, звіти про запаси, звіти про клієнтів тощо. Звіти можуть бути представлені у вигляді тексту, таблиць, графіків або візуалізаційних засобів.

- Прогнозування та стратегічне планування: Модель може включати також методи для прогнозування майбутніх трендів та розробки стратегічних планів на основі аналізу даних. Наприклад, модель може допомагати у визначенні оптимального рівня запасів, оцінці ефективності маркетингових акцій, ідентифікації нових ринкових можливостей тощо.

- Взаємодія з користувачем: Модель може включати інтерфейс, що дозволяє користувачам взаємодіяти з системою звітності та аналітики,

налаштовувати параметри аналізу, генерувати власні звіти, переглядати результати аналізу та приймати рішення на основі цих даних.

Ці елементи моделі звітності та аналітики допомагають забезпечити важливу інформацію для магазину комп'ютерної техніки, що допомагає керівництву в прийнятті рішень, вдосконаленні стратегій, покращенні роботи та досягненні більшої ефективності.

3.2. Проектування структури бази даних

3.2.1. Концептуальне інфологічне проектування

Інфологічне проектування – це процес створення концептуальної моделі предметної області, призначеної для виключення надмірності даних та проектуванні логічної моделі бази даних, що не залежить від специфіки конкретної системи управління базами даних (СУБД).

Концептуальна модель найбільш повно відповідає потребам проектування баз даних. Вона включає в себе опис об'єктів або понять предметної області і зв'язків між ними та опис вимог до допустимих значень даних і до зв'язків між ними. Словник даних наведений в табл 3.1

Таблиця 3.1

Словник даних

| № з/п | Найменування елемента | Ідентифікатор | Тип і довжина | Призначення елемента |
|-------|-----------------------|---------------|---------------|-----------------------|
| 1 | 2 | 3 | 4 | 5 |
| 1 | Пріоритет | Priority | nvarchar(64) | Пріоритет |
| 2 | Ціна опції | Price | Money | Ціна опції |
| 3 | День початку | Date_start | DateTime | День початку |
| 4 | День кінця | Date_end | | День кінця |
| 5 | Посилання на картинку | Image | nvarchar(64) | Посилання на картинку |
| 6 | Номер в списку | Sort_order | | Номер в списку |
| 7 | Значення | Option_value | nvarchar(64) | Значення |
| 8 | Потрібно | Requierd | Bool | Потрібно |
| 9 | Тип | Type | nvarchar(64) | Тип |
| 10 | Назва | Option_name | nvarchar(64) | Назва |

Закінчення табл. 3.1

| 1 | 2 | 3 | 4 | 5 |
|----|-----------------------------|-------------------|--------------|-----------------------------|
| 11 | Кількість | Quantity | Int | Кількість |
| 12 | Вага | Weight | Decimal | Вага |
| 13 | Модел | Model | nvarchar(64) | Модел |
| 14 | Розташування | Location | nvarchar(64) | Розташування |
| 15 | Кількість товарів на складу | Quantity | Int | Кількість товарів на складу |
| 16 | Ціна товару | Price | Money | Ціна товару |
| 17 | Вага товару | Weight | Decimal | Вага товару |
| 18 | Довжина | Length | Decimal | Довжина |
| 19 | Ширина | Width | Decimal | Ширина |
| 20 | Висота | Height | Decimal | Висота |
| 21 | Дата доступності | Date_avalible | DateTime | Дата доступності |
| 22 | Дата додання | Date_added | DateTime | Дата додання |
| 23 | Дата модифікації | Date_modified | DateTime | Дата модифікації |
| 24 | Назва виробника | Manufacture_name | nvarchar(64) | Назва виробника |
| 25 | Картинка виробника | Manufacture_image | nvarchar(64) | Картинка виробника |
| 26 | Автор коменторя | Author | nvarchar(64) | Автор коменторя |
| 27 | Текст коменторя | Text | nvarchar(64) | Текст коменторя |
| 28 | Оцінка | Rating | int | Оцінка |
| 29 | Статус | Status | nvarchar(64) | Статус |

Глобальної інфологічної моделі даних наведена в табл 3.2

Таблиця 3.2

Обмеження атрибутів

| № з/п | Ім'я атрибута | Межі або допустимі значення | Структура (формат) | Умова | Значення за замовчуванням |
|-------|-----------------------|-----------------------------|--------------------|----------|---------------------------|
| 1 | 2 | 3 | 4 | 5 | 6 |
| 1 | Пріоритет | – | – | NOT NULL | – |
| 2 | Ціна опції | – | – | NOT NULL | – |
| 3 | День початку | – | mm/dd/yyyy | NOT NULL | – |
| 4 | День кінця | – | mm/dd/yyyy | NOT NULL | – |
| 5 | Посилання на картинку | – | – | NOT NULL | – |
| 6 | Номер в списку | – | – | NOT NULL | – |
| 7 | Значення | – | – | NOT NULL | – |
| 8 | Потрібно | – | – | NOT NULL | – |

Закінчення табл. 3.2

| 1 | 2 | 3 | 4 | 5 | 6 |
|----|-----------------------------|---|------------|----------|---|
| 9 | Тип | – | – | NOT NULL | – |
| 10 | Назва | – | – | NOT NULL | – |
| 11 | Кількість | – | – | NOT NULL | – |
| 12 | Вага | – | – | NOT NULL | – |
| 13 | Модель | – | – | NOT NULL | – |
| 14 | Розташування | – | – | NOT NULL | – |
| 15 | Кількість товарів на складу | – | – | NOT NULL | – |
| 16 | Ціна товару | – | – | NOT NULL | – |
| 17 | Вага товару | – | – | NOT NULL | – |
| 18 | Довжина | – | – | NOT NULL | – |
| 19 | Ширина | – | – | NOT NULL | – |
| 20 | Висота | – | – | NOT NULL | – |
| 21 | Дата доступності | – | mm/dd/yyyy | NOT NULL | – |
| 22 | Дата додання | – | mm/dd/yyyy | NOT NULL | – |
| 23 | Дата модифікації | – | mm/dd/yyyy | NOT NULL | – |
| 24 | Назва виробника | – | – | NOT NULL | – |
| 25 | Картинка виробника | – | – | NOT NULL | – |
| 26 | Автор коменторя | – | – | NOT NULL | – |
| 27 | Текст коменторя | – | – | NOT NULL | – |
| 28 | Оцінка | – | – | NOT NULL | – |
| 29 | Статус | – | – | NOT NULL | – |

3.2.2. Проектування глобальної логічної моделі даних.

Логічна схема бази даних є абстрактним описом структури та взаємозв'язків між сутностями бази даних, що відображає логічну організацію даних без прив'язки до конкретного Системи управління базами даних (СУБД) або фізичної реалізації. Вона дозволяє визначити сутності, їх атрибути та зв'язки між ними.

Логічна схема бази даних може бути представлена у вигляді діаграми, схеми або набору правил та визначень, які описують структуру бази даних.

Вона зазвичай використовує спеціальні моделі даних, такі як ER-модель (модель сутність-зв'язок) або UML-діаграми класів, для відображення сутностей, атрибутів та взаємозв'язків.

Логічна схема бази даних визначає структуру бази даних на рівні концепційної моделі, яка не залежить від конкретної реалізації. Вона визначає, як дані організовані та як вони пов'язані між собою, що дозволяє забезпечити цілісність даних та ефективний доступ до них.

СУБД-орієнтована модель даних представлена у вигляді логічної схеми бази даних приведена на рис. 3.1.

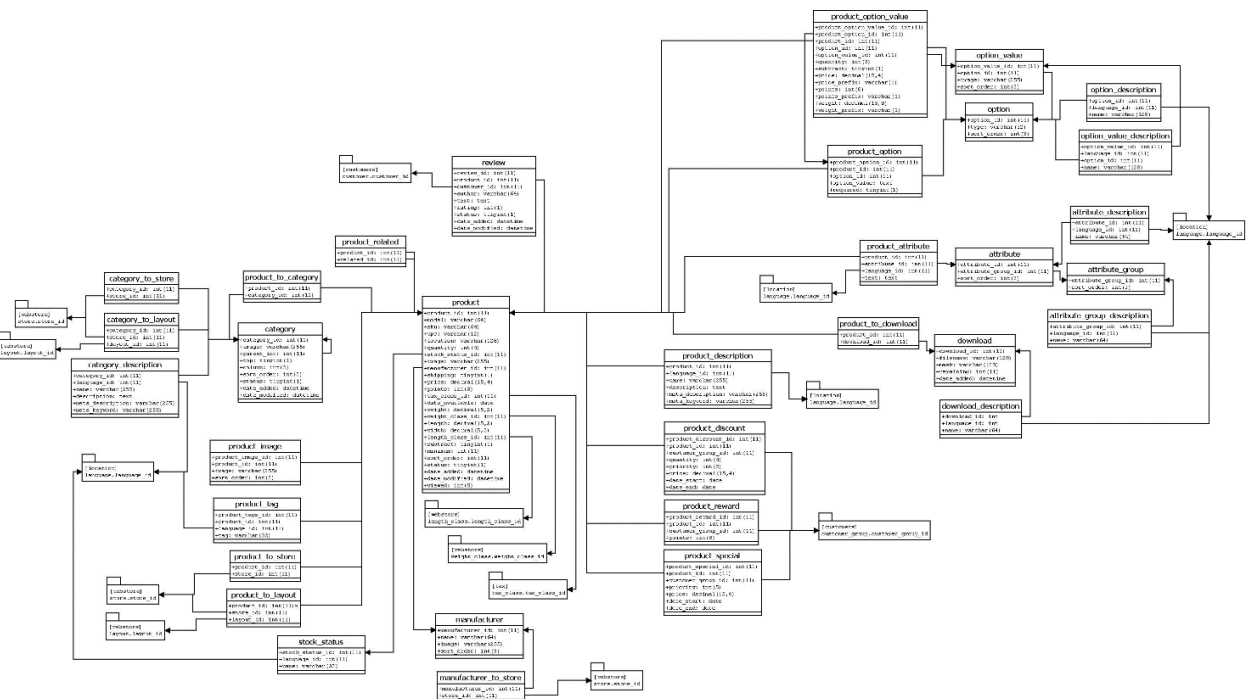


Рис. 3.1. Логічна схема бази даних

3.2.3. Проектування фізичної моделі бази даних та її програмна реалізація.

На основі логічної моделі БД будується фізична модель, яка, окрім сутностей та атрибутів, містить типи даних та їх розмірність. Фізична схема бази даних приведена на рис. 3.2

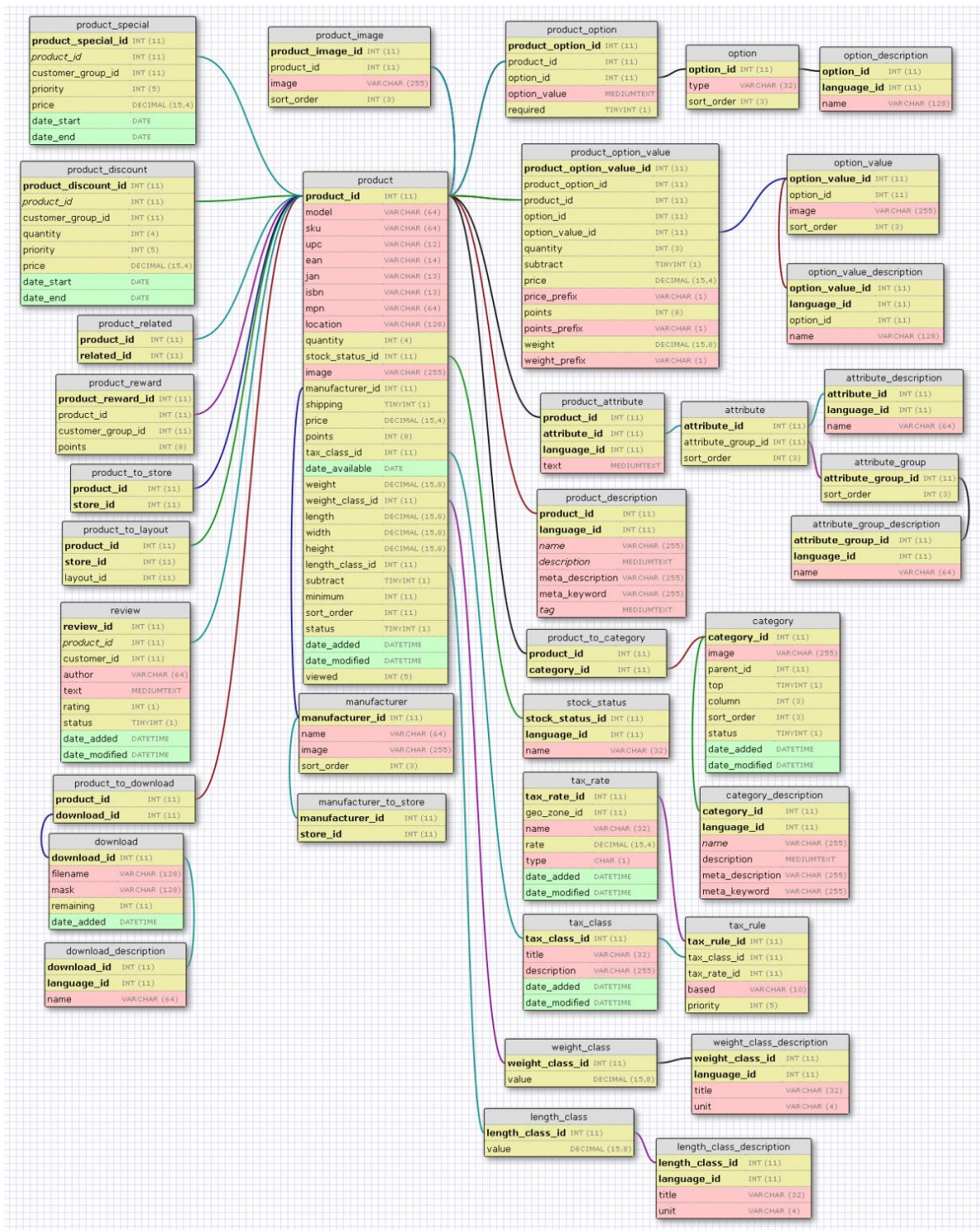


Рис. 3.2. Фізична схема бази даних

4. ТЕХНІЧНІ РІШЕННЯ

4.1. Розроблення архітектури програмної системи

Розробка архітектури програмної системи OpenCart включає наступні аспекти:

- Клієнт-серверна архітектура: OpenCart базується на клієнт-серверній архітектурі, де клієнтська частина відповідає за взаємодію з користувачем, а серверна частина обробляє запити користувачів та управляє базою даних.

- Модульна структура: OpenCart має модульну структуру, що дозволяє легко додавати та розширювати функціональність системи. Кожен модуль виконує конкретну функцію, наприклад, керування продуктами, замовленнями або платежами.

- Модель-вид-контролер (MVC): OpenCart використовує архітектурний шаблон Model-View-Controller (MVC), де модель представляє дані та бізнес-логіку, вид відповідає за відображення інтерфейсу користувача, а контролер обробляє запити користувача та взаємодіє з моделлю та видом.

- Шари додатку: OpenCart розділяє функціональність системи на різні шари, такі як шар доступу до даних (Data Access Layer), бізнес-логіки (Business Logic Layer) та презентаційного шару (Presentation Layer). Це дозволяє зберігати код чистим, легко зрозумілим та підтримуваним.

- База даних: OpenCart використовує реляційну базу даних, таку як MySQL або MariaDB, для зберігання даних про продукти, замовлення, користувачів та інше. База даних моделюється у вигляді таблиць зі зв'язками між ними.

- Розширення та теми: OpenCart надає можливість додавати розширення та використовувати теми для налаштування вигляду та функціональності магазину. Це забезпечує гнучкість та можливість адаптації системи під конкретні потреби.

Під час розробки архітектури програмної системи OpenCart рекомендується дотримуватись кращих практик програмування, таких як розділення обов'язків, чистий код та принципи SOLID, щоб забезпечити читабельність, розширюваність та підтримуваність системи.

На основі проведеного проектування було розроблено програмний продукт. Лістинг програми наведено у додатку А

4.2. Тестування програмної системи

Тестування програми – процес виконання програмного коду, спрямований на виявлення існуючих в ньому дефектів. Під дефектом розуміється ділянка програмного коду, виконання якої за певних умов приводить до несподіваного поведження системи.

Завданням тестування є визначення умов, при яких проявляються помилки, та документування цих умов. Документування процесу тестування полягає у створенні тест-плану сполученим зі звітом про проведення тестування. Тест-план складається з окремих тестових прикладів, кожен з яких перевіряє одну з функцій системи на основі функціональних вимог до програмного продукту.

Опис тестових прикладів наведений в табл. 3.4. Опис та результати виконання тест-прикладів наведені в табл. 3.5 – 3.11.

Таблиця 3.4

Тестовий план

| № з/п | Призначення | Вимоги, що перевіряються | Передумови для тесту | Критерій проходження тесту |
|-------|--|--------------------------|--|---|
| 1 | 2 | 3 | 4 | 5 |
| 1 | Тестування можливості авторизації та автентифікації користувача/ | UC-1, UC-2 | Користувач відкрив застосунок ; користувач має зареєстрований акаунт | Користувач має доступ до такого функціоналу застосунку, що передбачений роллю користувача |

Закінчення табл. 3.4

| 1 | 2 | 3 | 4 | 5 |
|---|--|------|---|---|
| 2 | Тестування можливості додання товару в кошик | UC-3 | Користувач відкрив застосунок: Користувач обрав товар Користувач Натиснув кнопку «Buy» | Товар був доданий в корзину |
| 3 | Тестування можливості додання нових товарів | UC-4 | Користувач авторизувався в ролі «Адміністратор» Користувач перейшов на вкладку «Product» | Товар був доданий до списку товарів системи |
| 4 | Тестування можливості видалення товарів | UC-5 | Користувач авторизувався в ролі «Адміністратор» Користувач перейшов на вкладку «Product» | Товар був видалений зі списку товарів системи |
| 5 | Тестування можливості перегляду замовлень | UC-6 | Користувач авторизувався в ролі «Адміністратор» Користувач перейшов на вкладку «Orders» Користувач натиснув на замовлення | Користувачу відображена сторінка з інформацією про обране замовлення |
| 6 | Тестування можливості перегляду аналіти по продажам | UC-7 | Користувач авторизувався в ролі «Адміністратор» Користувач перейшов на вкладку «Dashdoard» | Користувачу відображена сторінка з інформацією про продажі |
| 7 | Тестування можливості перегляду списку товарів по 20 товарів на сторінку | UC-8 | Користувач відкрив застосунок ; Користувач перейшов на сторінку з переглядом наявних товарів | Користувачу відображено 20 товарі на сторінку |

Таблиця 3.5

Виконання тест-прикладу №1

| № з/п | Крок сценарію | Очікуваний результат | Отриманий результат | Відмітка про проходження кроку сценарію (Так/Ні) |
|---|--|---|---|--|
| 1 | Користувач відкрив застосунок | З'являється вікно авторизації | З'являється вікно авторизації | Так |
| 2 | Користувач вводить пароль та логін, натискаючи «Login» | Користувач отримує доступ до функціоналу застосунку | Користувач отримує доступ до функціоналу застосунку | Так |
| Відмітка про проходження тест-прикладу: | | | | Так |

Таблиця 3.6

Виконання тест-прикладу №2

| № з/п | Крок сценарію | Очікуваний результат | Отриманий результат | Відмітка про проходження кроку сценарію (Так/Ні) |
|---|---|--|--|--|
| 1 | 2 | 3 | 4 | 5 |
| 1 | Обрати товар з списку й натиснути на нього. | Повинна відкритися сторінка інформації про обрані товару. | Відкрилася сторінка інформації про обрані товару. | Так |
| 2 | Натиснути кнопку «Buy» | Повинно відкритися повідомлення про успішне додання в корзину. | Відкрилося повідомлення про успішне додання в корзину. | Так |
| 3 | Перейти на сторінку «Cart». | Доданий товар повинен відображатися в корзині | Доданий товар відображається в корзині | Так |
| Відмітка про проходження тест-прикладу: | | | | Так |

Таблиця 3.7

Виконання тест-прикладу №3

| № з/п | Крок сценарію | Очікуваний результат | Отриманий результат | Відмітка про проходження кроку сценарію (Так/Ні) |
|---|---|--|--|--|
| 1 | 2 | 3 | 4 | 5 |
| 1 | Обрати товари з списку й натиснути кнопку «Delete». | Повинна відкритися вікно з підтвердженням видалення. | Відкрилася вікно з підтвердженням видалення | Так |
| 2 | Натиснути кнопку «ОК» | Обраний товар був видалений з списку товарів | Обраний товар був видалений з списку товарів | Так |
| Відмітка про проходження тест-прикладу: | | | | Так |

Таблиця 3.8

Виконання тест-прикладу №4

| № з/п | Крок сценарію | Очікуваний результат | Отриманий результат | Відмітка про проходження кроку сценарію (Так/Ні) |
|-------|----------------------------|--|---|--|
| 1 | 2 | 3 | 4 | 5 |
| 1 | Натиснути кнопку «Orders». | Повинна відкритися сторінка з списком замовлень. | Відкрилася вікно з сторінка з списком замовлень | Так |
| 2 | Обрати замовлення, | Повинна відкритися сторінка | Відкрилася вікно з сторінка з | Так |

Закінчення табл. 3.8

| 1 | 2 | 3 | 4 | 5 |
|---|--|------------------------------|----------------------------|-----|
| | інформацію про яке потрібно переглянути інформацію | з інформацією про замовлення | інформацією про замовлення | |
| Відмітка про проходження тест-прикладу: | | | | Так |

Таблиця 3.9

Виконання тест-прикладу №5

| № з/п | Крок сценарію | Очікуваний результат | Отриманий результат | Відмітка про проходження кроку сценарію (Так/Ні) |
|---|--|--|--|--|
| 1 | 2 | 3 | 4 | 5 |
| 1 | Користувач вводить пароль та логін, натискаючи «Login» | Користувач отримує доступ до функціоналу застосунку як «Адміністратор» | Користувач отримує доступ до функціоналу застосунку як «Адміністратор» | Так |
| 2 | Натиснути кнопку «Dashdoard». | Повинна відкритися сторінка інформації про статистику продажів. | Відкрилася сторінка інформації про статистику продажів.. | Так |
| Відмітка про проходження тест-прикладу: | | | | Так |

Таблиця 3.10

Виконання тест-прикладу №6

| № з/п | Крок сценарію | Очікуваний результат | Отриманий результат | Відмітка про проходження кроку сценарію (Так/Ні) |
|-------|------------------------------|---|--|--|
| 1 | 2 | 3 | 4 | 5 |
| 1 | Обрати пункт меню «Products» | Повинна бути відображена сторінка з списком товарів | Відображена сторінка з списком товарів | Так |

Закінчення табл. 3.10

| 1 | 2 | 3 | 4 | 5 |
|---|---|---|------------------------------------|-----|
| 2 | Переглянути скільки на сторінці відобразилося ліків | Повинна бути відображена сторінка з 20 товарами | Відображена сторінка з 16 товарами | Так |
| Відмітка про проходження тест-прикладу: | | | | Так |

Тестових прикладів виконано: 6.

Тестових прикладів пройдено: 6.

Тестування програмного продукту показало, що програмний продукт володіє високим ступенем якості згідно визначених функціональних вимог. Тестування надало позитивні результати, що дозволяє провести розгортання програмного продукту.

4.3. Розгортання програмного продукту

Щоб розгорнути програмний продукт OpenCart PHP, знадобиться виконати наступні кроки:

- Завантажити OpenCart: Відвідайте офіційний веб-сайт OpenCart (<https://www.opencart.com/>) та завантажте останню версію програми.
- Розпакувати архів: Після завантаження програмного продукту розпакувати архів з файлами OpenCart на локальному комп'ютері.
- Завантажити файли на сервер: Завантажити файли OpenCart на веб-сервер. Це можна зробити за допомогою FTP-клієнта або через панель керування хостинг-провайдера.
- Створити базу даних: У панелі керування базою phpMyAdmin створіть нову базу даних для OpenCart. Запам'ятати дані доступу до бази даних (назва бази даних, ім'я користувача та пароль).
- Завершити налаштування: Відкрити веб-браузер і перейдіть до URL-адреси сайту, де розгорнуто OpenCart. Слідую інструкціям на екрані для завершення процесу установки. Ввести дані бази даних, адміністраторські налаштування та інші необхідні деталі.
- Налаштувати теми та розширення: Після успішного розгортання OpenCart можна вибрати тему оформлення та встановити додаткові розширення зі адміністраторського розділу OpenCart.

Після завершення цих кроків програмний продукт OpenCart повинен бути успішно розгорнутий на веб-сервері, і можна почати налаштовувати свій магазин.

ВИСНОВКИ

У рамках даного дипломного проекту була розроблена інформаційна підсистема, яка спрямована на підтримку роботи магазину комп'ютерної техніки. Метою проекту було створення зручного та ефективного інструменту для автоматизації процесів, пов'язаних з управлінням товарним асортиментом, замовленнями, продажами та клієнтським обслуговуванням.

В результаті розробки інформаційної підсистеми було досягнуто наступних результатів:

Створено систему каталогізації товарів, яка дозволяє зручно додавати нові товари, вказувати їхні характеристики та фотографії. Це спрощує процес управління асортиментом та допомагає швидко знаходити необхідний товар.

Розроблено систему замовлень, яка дозволяє клієнтам зручно робити замовлення через інтернет або в магазині. Менеджерам також надано інструмент для обробки замовлень та відстеження їх статусу. Це полегшує процес продажу та забезпечує задоволення клієнтів.

Впроваджено систему управління клієнтською базою даних, яка дозволяє зберігати та відстежувати інформацію про клієнтів, їхні замовлення та контактні дані. Це сприяє покращенню обслуговування клієнтів та підвищує рівень їхньої задоволеності.

Розроблено систему аналітики та звітності, яка надає менеджерам зручний інструмент для аналізу продажів, стану асортименту та ефективності магазину. Це допомагає приймати обґрунтовані рішення щодо планування закупівель, маркетингових акцій та розвитку бізнесу.

Розроблена інформаційна підсистема є потужним інструментом для оптимізації роботи магазину комп'ютерної техніки. Її впровадження може значно полегшити управління та підвищити ефективність різних процесів в магазині, забезпечити кращий сервіс для клієнтів та сприяти розвитку бізнесу..

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Дроняк, В. М. Інтерактивна система формування замовлень на продукцію фотостудії [Текст] / В. М. Дроняк // Сучасні комп'ютерні інформаційні технології : матеріали V Всеукр. шк.-семінару молодих вчен. і студ. АСІТ'2015 [м. Тернопіль, 22-23 трав. 2015р.] / відп. за вип. М. П. Дивак. - Тернопіль : ТНЕУ, 2015. - С. 134.
2. ДСТУ 3008:2015. Звіти у сфері науки і техніки. Структура та правила оформлювання: чинний від 01.07.2017. – Київ : ДП "УкрНДНЦ", 2016. – 26 с.
3. ДСТУ 8302:2015. Інформація та документація. Бібліографічне посилання. Загальні положення та правила складання : чинний від 01.07.2016. – Київ : ДП "УкрНДНЦ", 2016. – 17 с.
4. Євдокимова А. В. Головні аспекти проектної діяльності студії інтер'єр-дизайну [Текст] / А. В. Євдокимова, К. О. Овсієнко // Сучасний менеджмент і економічний розвиток. Екологічна економіка: конфлікти та проблеми управління : матеріали постійно діючої міжнародної науково-практичної інтернет-конференції, 1 вересня 2014 р. - 31 серпня 2015 р. – Суми : СумДУ, 2015.
5. Фан Х. М. Проектування веб-сайта для сучасної фотостудії з використанням новітніх досягнень веб-дизайну / Х. М. Фан ; наук. кер. О. Л. Яворський // Наукові розробки молоді на сучасному етапі : тези доповідей XVI Всеукраїнської наукової конференції молодих вчених та студентів (27-28 квітня 2017 р., Київ). - К. : КНУТД, 2017. - Т. 1 : Сучасні матеріали і технології виробництва виробів широкого вжитку та спеціального призначення. - С. 450-451.
6. Дорофєєва О. Удосконалення доступу до галузевого веб-ресурсу / Ольга Дорофєєва // Бібліотечний вісник. – 2010. – № 1. – С. 3-9.
7. Дипломний проєкт. Методичні рекомендації для студентів спеціальності 126 "Інформаційні системи та технології" освітньої програми

"Інформаційні системи та технології" першого (бакалаврського) рівня [Електронний ресурс] / уклад. С. Г. Удовенко, О. О. Тютюник, О. В. Гороховатський, В. А. Затхей; Харківський національний економічний університет ім. С. Кузнеця. – Електрон. текстові дан. (189 КБ). – Харків : ХНЕУ ім. С. Кузнеця, 2023. – 48 с.

8. Інькова Н. А. Створення Web-сайтів : Навчально-методичний посібник (Електронний ресурс) / Інькова Н. А., Зайцева Е. А., Кузьміна Н. В., Толстих С. Г. – URL: <http://club-edu.tambov.ru/methodic/fio/p5.doc>

9. Компанєєтс М.О. Принципи проектування ефективних вебсайтів / М.О. Компанєєтс // Молодий вчений. – 2015. – № 9. – Част. 2. – С. 106-108

10. Основні етапи розробки веб-сайту (Електронний ресурс) / [укл. Юрчак І. Ю.] / Національний університет "Львівська політехніка". – URL: <http://www.victoria.lviv.ua/html/wp/ssteps.html> (дата звернення 25.05.2020).

11. Проектування. Методичні рекомендації до виконання курсового проєкту для студентів спеціальності 126 "Інформаційні системи та технології" першого (бакалаврського) рівня [Електронний ресурс] / уклад. О. О. Тютюник, Н. О. Бринза, І. О. Ушакова; Харківський національний економічний університет ім. С. Кузнеця. - Електрон. текстові дан. (2,16 МБ). - Харків : ХНЕУ ім. С. Кузнеця, 2022. - 28 с.

12. Угрин Д. І. Веб-технології та веб-дизайн: [навч. посіб. для студ. вищ. навч. закл.] / Д. І. Угрин, М. І. Маниліч, Б. Є. Деркач. – Чернівці : Золоті литаври, 2012. – 285 с.: іл., табл. – Бібліогр.: с. 270-282.

13. Наскрізна програма практики для студентів спеціальності 126 "Інформаційні системи та технології" першого (бакалаврського) рівня [Електронний ресурс] / уклад. С. Г. Удовенко, О. О. Тютюник, Н. О. Бринза; Харківський національний економічний університет ім. С. Кузнеця. - Електрон. текстові дан. (81,2 КБ). - Харків : ХНЕУ ім. С. Кузнеця, 2022

14. Методологія IDEF0. [Електронний ресурс] – режим доступу: https://stud.com.ua/87184/ekonomika/metodologiya_idef0

15. Інтернет магазин «Rozetka». [Електронний ресурс] – режим доступу:
<https://rozetka.com.ua/ua/>

16. . Інтернет магазин «Мою». [Електронний ресурс] – режим доступу:
<https://www.mojo.ua/ua/>

17. Інтернет магазин «Sota». [Електронний ресурс] – режим доступу:
<https://sota.kh.ua/ua/>

18. Основні поняття реляційних баз даних [Електронний ресурс] // Bondarenko.UA. - Режим доступу: <https://bondarenko.dn.ua/osnovni-ponyattuarelyatsijnih-bd-normalizatsiya-zv-yazok-ta-klyuchi/>.

19. Тестування програмного забезпечення (UML) [Електронний ресурс] // Вікіпедія : вільна енциклопедія. - Режим доступу:
https://uk.wikipedia.org/wiki/%D0%A2%D0%B5%D1%81%D1%82%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BD%D0%BE%D0%B3%D0%BE_%D0%B7%D0%B0%D0%B1%D0%B5%D0%B7%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%BD%D1%8F

20. Уніфікований локаатор ресурсів. Вікіпедія : вільна енциклопедія. [Електронний ресурс] - Режим доступу:
https://uk.wikipedia.org/wiki/%D0%A3%D0%BD%D1%96%D1%84%D1%96%D0%BA%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B9_%D0%BB%D0%BE%D0%BA%D0%B0%D1%82%D0%BE%D1%80_%D1%80%D0%B5%D1%81%D1%83%D1%80%D1%81%D1%96%D0%B2

21. Тестування безпеки: SQL-ін'єкції [Електронний ресурс] // QATestLab. – Режим доступу: <https://training.qatestlab.com/blog/technical-articles/security-testing-sql-injection/>.

22. Нефункціональні вимоги [Електронний ресурс] // Вікіпедія : вільна енциклопедія. - Режим доступу:
https://uk.wikipedia.org/wiki/Нефункціональні_вимоги .

23. Глосарій [Електронний ресурс] // Вікіпедія : вільна енциклопедія. - Режим доступу: <https://uk.wikipedia.org/wiki/Глосарій>

ДОДАТКИ

ЛІСТИНГ КОДУ ЗАСТОСУНКУ

```

{{ header }}
<div id="account-account" class="container">
  <ul class="breadcrumb">
    {% for breadcrumb in breadcrumbs %}
    <li><a href="{{ breadcrumb.href }}">{{ breadcrumb.text }}</a></li>
    {% endfor %}
  </ul>
  {% if success %}
  <div class="alert alert-success alert-dismissible"><i class="fa fa-check-circle"></i> {{ success
  }}</div>
  {% endif %}
  <div class="row">{{ column_left }}
    {% if column_left and column_right %}
    {% set class = 'col-sm-6' %}
    {% elseif column_left or column_right %}
    {% set class = 'col-sm-9' %}
    {% else %}
    {% set class = 'col-sm-12' %}
    {% endif %}
  <div id="content" class="{{ class }}">{{ content_top }}
    <h2>{{ text_my_account }}</h2>
    <ul class="list-unstyled">
      <li><a href="{{ edit }}">{{ text_edit }}</a></li>
      <li><a href="{{ password }}">{{ text_password }}</a></li>
      <li><a href="{{ address }}">{{ text_address }}</a></li>
      <li><a href="{{ wishlist }}">{{ text_wishlist }}</a></li>
    </ul>
    {% if credit_cards %}
    <h2>{{ text_credit_card }}</h2>
    <ul class="list-unstyled">
      {% for credit_card in credit_cards %}
      <li><a href="{{ credit_card.href }}">{{ credit_card.name }}</a></li>
      {% endfor %}
    </ul>
    {% endif %}
    <h2>{{ text_my_orders }}</h2>
    <ul class="list-unstyled">
      <li><a href="{{ order }}">{{ text_order }}</a></li>
      <li><a href="{{ download }}">{{ text_download }}</a></li>
      {% if reward %}
      <li><a href="{{ reward }}">{{ text_reward }}</a></li>
      {% endif %}
      <li><a href="{{ return }}">{{ text_return }}</a></li>
      <li><a href="{{ transaction }}">{{ text_transaction }}</a></li>
      <li><a href="{{ recurring }}">{{ text_recurring }}</a></li>
    </ul>
    <h2>{{ text_my_affiliate }}</h2>
    <ul class="list-unstyled">

```

```

{% if not tracking %}
  <li><a href="{{ affiliate }}">{{ text_affiliate_add }}</a></li>
  {% else %}
  <li><a href="{{ affiliate }}">{{ text_affiliate_edit }}</a></li>
  <li><a href="{{ tracking }}">{{ text_tracking }}</a></li>
  {% endif %}
</ul>
<h2>{{ text_my_newsletter }}</h2>
<ul class="list-unstyled">
  <li><a href="{{ newsletter }}">{{ text_newsletter }}</a></li>
</ul>
  {{ content_bottom }}</div>
  {{ column_right }}</div>
</div>
{{ footer }}

{{ header }}
<div id="account-address" class="container">
  <ul class="breadcrumb">
    {% for breadcrumb in breadcrumbs %}
      <li><a href="{{ breadcrumb.href }}">{{ breadcrumb.text }}</a></li>
    {% endfor %}
  </ul>
  <div class="row">{{ column_left }}
    {% if column_left and column_right %}
      {% set class = 'col-sm-6' %}
    {% elseif column_left or column_right %}
      {% set class = 'col-sm-9' %}
    {% else %}
      {% set class = 'col-sm-12' %}
    {% endif %}
    <div id="content" class="{{ class }}">{{ content_top }}
      <h2>{{ text_address }}</h2>
      <form action="{{ action }}" method="post" enctype="multipart/form-data" class="form-horizontal">
        <fieldset>
          <div class="form-group required">
            <label class="col-sm-2 control-label" for="input-firstname">{{ entry_firstname }}</label>
            <div class="col-sm-10">
              <input type="text" name="firstname" value="{{ firstname }}" placeholder="{{ entry_firstname }}" id="input-firstname" class="form-control" />
              {% if error_firstname %}
                <div class="text-danger">{{ error_firstname }}</div>
              {% endif %} </div>
            </div>
          <div class="form-group required">
            <label class="col-sm-2 control-label" for="input-lastname">{{ entry_lastname }}</label>
            <div class="col-sm-10">
              <input type="text" name="lastname" value="{{ lastname }}" placeholder="{{ entry_lastname }}" id="input-lastname" class="form-control" />

```



```

<select name="country_id" id="input-country" class="form-control">
  <option value="">{{ text_select }}</option>

  {% for country in countries %}
    {% if country.country_id == country_id %}

      <option value="{{ country.country_id }}" selected="selected">{{ country.name
}}</option>

    {% else %}

      <option value="{{ country.country_id }}">{{ country.name }}</option>

    {% endif %}
  {% endfor %}

</select>
{% if error_country %}
<div class="text-danger">{{ error_country }}</div>
{% endif %} </div>
<div class="form-group required">
  <label class="col-sm-2 control-label" for="input-zone">{{ entry_zone }}</label>
  <div class="col-sm-10">
    <select name="zone_id" id="input-zone" class="form-control">
      </select>
      {% if error_zone %}
        <div class="text-danger">{{ error_zone }}</div>
      {% endif %} </div>
</div>
{% for custom_field in custom_fields %}
  {% if custom_field.type == 'select' %}
    <div class="form-group{% if custom_field.required %} required {% endif %} custom-
field" data-sort="{{ custom_field.sort_order }}">
      <label class="col-sm-2 control-label" for="input-custom-field{{
custom_field.custom_field_id }}">{{ custom_field.name }}</label>
      <div class="col-sm-10">
        <select name="custom_field[{{ custom_field.location }}][{{
custom_field.custom_field_id }}" id="input-custom-field{{ custom_field.custom_field_id }}"
class="form-control">
          <option value="">{{ text_select }}</option>

          {% for custom_field_value in custom_field.custom_field_value %}
            {% if address_custom_field[custom_field.custom_field_id] and
custom_field_value.custom_field_value_id ==
address_custom_field[custom_field.custom_field_id] %}

              <option value="{{ custom_field_value.custom_field_value_id }}"
selected="selected">{{ custom_field_value.name }}</option>

```

```

    <option value="{{ custom_field_value.custom_field_value_id }}">{{
custom_field_value.name }}</option>
        {% endif %}
    {% endfor %}
</select>
    {% if error_custom_field[custom_field.custom_field_id] %}
    <div class="text-danger">{{ error_custom_field[custom_field.custom_field_id]
}}</div>
    {% endif %}</div>
</div>
{% endif %}
{% if custom_field.type == 'radio' %}
<div class="form-group{% if custom_field.required %} required {% endif %} custom-
field" data-sort="{{ custom_field.sort_order }}">
    <label class="col-sm-2 control-label">{{ custom_field.name }}</label>
    <div class="col-sm-10">
        <div> {% for custom_field_value in custom_field.custom_field_value %}
            <div class="radio"> {% if address_custom_field[custom_field.custom_field_id] and
custom_field_value.custom_field_value_id ==
address_custom_field[custom_field.custom_field_id] %}
                <label>
                    <input type="radio" name="custom_field[{{ custom_field.location }}][{{
custom_field.custom_field_id }}" value="{{ custom_field_value.custom_field_value_id }}"
checked="checked" />
                    {{ custom_field_value.name }}</label>
                {% else %}
                <label>
                    <input type="radio" name="custom_field[{{ custom_field.location }}][{{
custom_field.custom_field_id }}" value="{{ custom_field_value.custom_field_value_id }}" />
                    {{ custom_field_value.name }}</label>
                {% endif %} </div>
            {% endfor %} </div>
            {% if error_custom_field[custom_field.custom_field_id] %}
            <div class="text-danger">{{ error_custom_field[custom_field.custom_field_id]
}}</div>
            {% endif %} </div>
        </div>
    {% endif %}
{% if custom_field.type == 'checkbox' %}
<div class="form-group{% if custom_field.required %} required {% endif %} custom-
field" data-sort="{{ custom_field.sort_order }}">
    <label class="col-sm-2 control-label">{{ custom_field.name }}</label>
    <div class="col-sm-10">
        <div> {% for custom_field_value in custom_field.custom_field_value %}
            <div class="checkbox"> {% if address_custom_field[custom_field.custom_field_id]
and custom_field_value.custom_field_value_id in
address_custom_field[custom_field.custom_field_id] %}
                <label>
                    <input type="checkbox" name="custom_field[{{ custom_field.location }}][{{
custom_field.custom_field_id }}]" value="{{ custom_field_value.custom_field_value_id }}"
checked="checked" />
                    {{ custom_field_value.name }}</label>
            </div>
        </div>
    </div>

```

```

{% else %}
    <label>
        <input type="checkbox" name="custom_field[{{ custom_field.location }}][{{
custom_field.custom_field_id }}]" value="{{ custom_field_value.custom_field_value_id }}"
/>
        {{ custom_field_value.name }}</label>
    {% endif %} </div>
{% endfor %} </div>
{% if error_custom_field[custom_field.custom_field_id] %}
    <div class="text-danger">{{ error_custom_field[custom_field.custom_field_id]
}}</div>
{% endif %} </div>
</div>
{% endif %}
{% if custom_field.type == 'text' %}
    <div class="form-group{% if custom_field.required %} required {% endif %} custom-
field" data-sort="{{ custom_field.sort_order }}">
        <label class="col-sm-2 control-label" for="input-custom-field{{
custom_field.custom_field_id }}">{{ custom_field.name }}</label>
        <div class="col-sm-10">
            <input type="text" name="custom_field[{{ custom_field.location }}][{{
custom_field.custom_field_id }}" value="{% if
address_custom_field[custom_field.custom_field_id] %}{{
address_custom_field[custom_field.custom_field_id] }}{% else %}{{ custom_field.value }}{%
endif %}" placeholder="{{ custom_field.name }}" id="input-custom-field{{
custom_field.custom_field_id }}" class="form-control" />
            {% if error_custom_field[custom_field.custom_field_id] %}
                <div class="text-danger">{{ error_custom_field[custom_field.custom_field_id]
}}</div>
            {% endif %} </div>
        </div>
    {% endif %}
{% if custom_field.type == 'textarea' %}
    <div class="form-group{% if custom_field.required %} required {% endif %} custom-
field" data-sort="{{ custom_field.sort_order }}">
        <label class="col-sm-2 control-label" for="input-custom-field{{
custom_field.custom_field_id }}">{{ custom_field.name }}</label>
        <div class="col-sm-10">
            <textarea name="custom_field[{{ custom_field.location }}][{{
custom_field.custom_field_id }}" rows="5" placeholder="{{ custom_field.name }}" id="input-
custom-field{{ custom_field.custom_field_id }}" class="form-control">{% if
address_custom_field[custom_field.custom_field_id] %}{{
address_custom_field[custom_field.custom_field_id] }}{% else %}{{ custom_field.value }}{%
endif %}</textarea>
            {% if error_custom_field[custom_field.custom_field_id] %}
                <div class="text-danger">{{ error_custom_field[custom_field.custom_field_id]
}}</div>
            {% endif %} </div>
        </div>
    {% endif %}
{% if custom_field.type == 'file' %}

```

```

<div class="form-group{% if custom_field.required %} required {% endif %} custom-field"
data-sort="{{ custom_field.sort_order }}">
  <label class="col-sm-2 control-label">{{ custom_field.name }}</label>
  <div class="col-sm-10">
    <button type="button" id="button-custom-field{{ custom_field.custom_field_id }}"
data-loading-text="{{ text_loading }}" class="btn btn-default"><i class="fa fa-upload"></i> {{
button_upload }}</button>
    <input type="hidden" name="custom_field[{{ custom_field.location }}][{{
custom_field.custom_field_id }}" value="{% if
address_custom_field[custom_field.custom_field_id]['code'] %}{{
address_custom_field[custom_field.custom_field_id]['code'] }}{% endif %}" />
    <span>{{ address_custom_field[custom_field.custom_field_id]['name'] }}</span>
    {% if error_custom_field[custom_field.custom_field_id] %}
    <div class="text-danger">{{ error_custom_field[custom_field.custom_field_id]
}}</div>
    {% endif %} </div>
  </div>
  {% endif %}
  {% if custom_field.type == 'date' %}
  <div class="form-group{% if custom_field.required %} required {% endif %} custom-
field" data-sort="{{ custom_field.sort_order }}">
    <label class="col-sm-2 control-label" for="input-custom-field{{
custom_field.custom_field_id }}">{{ custom_field.name }}</label>
    <div class="col-sm-10">
      <div class="input-group date">
        <input type="text" name="custom_field[{{ custom_field.location }}][{{
custom_field.custom_field_id }}" value="{% if
address_custom_field[custom_field.custom_field_id] %}{{
address_custom_field[custom_field.custom_field_id] }}{% else %}{{ custom_field.value }}{%
endif %}" placeholder="{{ custom_field.name }}" data-date-format="YYYY-MM-DD"
id="input-custom-field{{ custom_field.custom_field_id }}" class="form-control" />
        <span class="input-group-btn">
          <button type="button" class="btn btn-default"><i class="fa fa-
calendar"></i></button>
        </span></div>
        {% if error_custom_field[custom_field.custom_field_id] %}
        <div class="text-danger">{{ error_custom_field[custom_field.custom_field_id]
}}</div>
        {% endif %} </div>
      </div>
    {% endif %}
    {% if custom_field.type == 'time' %}
    <div class="form-group{% if custom_field.required %} required {% endif %} custom-
field" data-sort="{{ custom_field.sort_order }}">
      <label class="col-sm-2 control-label" for="input-custom-field{{
custom_field.custom_field_id }}">{{ custom_field.name }}</label>
      <div class="col-sm-10">
        <div class="input-group time">
          <input type="text" name="custom_field[{{ custom_field.location }}][{{
custom_field.custom_field_id }}" value="{% if
address_custom_field[custom_field.custom_field_id] %}{{
address_custom_field[custom_field.custom_field_id] }}{% else %}{{ custom_field.value }}{%
endif %}" placeholder="{{ custom_field.name }}" data-date-format="YYYY-MM-DD"
id="input-custom-field{{ custom_field.custom_field_id }}" class="form-control" />
          <span class="input-group-btn">
            <button type="button" class="btn btn-default"><i class="fa fa-
clock"></i></button>
          </span></div>
          {% if error_custom_field[custom_field.custom_field_id] %}
          <div class="text-danger">{{ error_custom_field[custom_field.custom_field_id]
}}</div>
          {% endif %} </div>
        </div>
      </div>
    {% endif %}
  </div>
  {% endif %}
  </div>

```

```

endif %}" placeholder="{{ custom_field.name }}" data-date-format="HH:mm" id="input-
custom-field{{ custom_field.custom_field_id }}" class="form-control" />
    <span class="input-group-btn">
        <button type="button" class="btn btn-default"><i class="fa fa-
calendar"></i></button>
    </span></div>
    {% if error_custom_field[custom_field.custom_field_id] %}
    <div class="text-danger">{{ error_custom_field[custom_field.custom_field_id]
}}</div>
    {% endif %} </div>
</div>
{% endif %}
{% if custom_field.type == 'datetime' %}
<div class="form-group{% if custom_field.required %} required {% endif %} custom-
field" data-sort="{{ custom_field.sort_order }}">
    <label class="col-sm-2 control-label" for="input-custom-field{{
custom_field.custom_field_id }}">{{ custom_field.name }}</label>
    <div class="col-sm-10">
        <div class="input-group datetime">
            <input type="text" name="custom_field[{{ custom_field.location }}][{{
custom_field.custom_field_id }}" value="{% if
address_custom_field[custom_field.custom_field_id] %}{{
address_custom_field[custom_field.custom_field_id] }}{% else %}{{ custom_field.value }}{%
endif %}" placeholder="{{ custom_field.name }}" data-date-format="YYYY-MM-DD
HH:mm" id="input-custom-field{{ custom_field.custom_field_id }}" class="form-control" />
            <span class="input-group-btn">
                <button type="button" class="btn btn-default"><i class="fa fa-
calendar"></i></button>
            </span></div>
            {% if error_custom_field[custom_field.custom_field_id] %}
            <div class="text-danger">{{ error_custom_field[custom_field.custom_field_id]
}}</div>
            {% endif %} </div>
</div>
{% endif %}
{% endfor %}
<div class="form-group">
    <label class="col-sm-2 control-label">{{ entry_default }}</label>
    <div class="col-sm-10"> {% if default %}
        <label class="radio-inline">
            <input type="radio" name="default" value="1" checked="checked" />
            {{ text_yes }}</label>
        <label class="radio-inline">
            <input type="radio" name="default" value="0" />
            {{ text_no }}</label>
        {% else %}
        <label class="radio-inline">
            <input type="radio" name="default" value="1" />
            {{ text_yes }}</label>
        <label class="radio-inline">
            <input type="radio" name="default" value="0" checked="checked" />
            {{ text_no }}</label>
    </div>
</div>

```

```

{% endif %} </div>
</div>
</fieldset>
<div class="buttons clearfix">
  <div class="pull-left"><a href="{{ back }}" class="btn btn-default">{{ button_back
}}</a></div>
  <div class="pull-right">
    <input type="submit" value="{{ button_continue }}" class="btn btn-primary" />
  </div>
</div>
</form>
{{ content_bottom }}</div>
{{ column_right }}</div>
</div>
<script type="text/javascript"><!--
// Sort the custom fields
$('.form-group[data-sort]').detach().each(function() {
  if ($(this).attr('data-sort') >= 0 && $(this).attr('data-sort') <= $('.form-group').length-2) {
    $('.form-group').eq(parseInt($(this).attr('data-sort'))+2).before(this);
  }

  if ($(this).attr('data-sort') > $('.form-group').length-2) {
    $('.form-group:last').after(this);
  }

  if ($(this).attr('data-sort') === $('.form-group').length-2) {
    $('.form-group:last').after(this);
  }

  if ($(this).attr('data-sort') < -$('.form-group').length-2) {
    $('.form-group:first').before(this);
  }
});
--></script>
<script type="text/javascript"><!--
$('#button[id^='\button-custom-field\"]').on('click', function() {
  var element = this;

  $('#form-upload').remove();

  $('body').prepend('<form enctype="multipart/form-data" id="form-upload"
style="display: none;"><input type="file" name="file" /></form>');

  $('#form-upload input[name='\file\']).trigger('click');

  if (typeof timer != 'undefined') {
    clearInterval(timer);
  }

  timer = setInterval(function() {
    if ($('#form-upload input[name='\file\']).val() != "") {
      clearInterval(timer);
    }
  }, 100);
--></script>

```

```

$.ajax({
    url: 'index.php?route=tool/upload',
    type: 'post',
    dataType: 'json',
    data: new FormData($('#form-upload')[0]),
    cache: false,
    contentType: false,
    processData: false,
    beforeSend: function() {
        $(element).button('loading');
    },
    complete: function() {
        $(element).button('reset');
    },
    success: function(json) {
        $(element).parent().find('.text-danger').remove();

        if (json['error']) {
            $(element).parent().find('input').after('<div
class="text-danger">' + json['error'] + '</div>');
        }

        if (json['success']) {
            alert(json['success']);

            $(element).parent().find('input').val(json['code']);
        }
    },
    error: function(xhr, ajaxOptions, thrownError) {
        alert(thrownError + "\r\n" + xhr.statusText + "\r\n" +
xhr.responseText);
    }
});
}, 500);
});
</script>
<script type="text/javascript"><!--
$('.date').datetimepicker({
    language: '{{ datepicker }}',
    pickTime: false
});

$('.datetime').datetimepicker({
    language: '{{ datepicker }}',
    pickDate: true,
    pickTime: true
});

$('.time').datetimepicker({
    language: '{{ datepicker }}',

```

```

pickDate: false
});
//--></script>
<script type="text/javascript"><!--
$( 'select[name=\'country_id\'] ).on( 'change', function() {
    $.ajax({
        url: 'index.php?route=account/account/country&country_id=' + this.value,
        dataType: 'json',
        beforeSend: function() {
            $( 'select[name=\'country_id\'] ).prop( 'disabled', true );
        },
        complete: function() {
            $( 'select[name=\'country_id\'] ).prop( 'disabled', false );
        },
        success: function( json ) {
            if ( json[ 'postcode_required' ] == '1' ) {
                $( 'input[name=\'postcode\'] ).parent().parent().addClass( 'required' );
            } else {
                $( 'input[name=\'postcode\'] ).parent().parent().removeClass( 'required' );
            }

            html = '<option value="">{{ text_select }}</option>';

            if ( json[ 'zone' ] && json[ 'zone' ] != "" ) {
                for ( i = 0; i < json[ 'zone' ].length; i++ ) {
                    html += '<option value="" + json[ 'zone' ][ i ][ 'zone_id' ] + ""';

                    if ( json[ 'zone' ][ i ][ 'zone_id' ] == '{{ zone_id }}' ) {
                        html += ' selected="selected"';
                    }

                    html += '>' + json[ 'zone' ][ i ][ 'name' ] + '</option>';
                }
            } else {
                html += '<option value="0" selected="selected">{{ text_none
            }}</option>';
        }

        $( 'select[name=\'zone_id\'] ).html( html );
    },
    error: function( xhr, ajaxOptions, thrownError ) {
        alert( thrownError + "\r\n" + xhr.statusText + "\r\n" + xhr.responseText );
    }
    });
});

$( 'select[name=\'country_id\'] ).trigger( 'change' );
//--></script>
{{ footer }}

{{ header }}

```



```

<div id="account-address" class="container">
  <ul class="breadcrumb">
    {% for breadcrumb in breadcrumbs %}
    <li><a href="{{ breadcrumb.href }}">{{ breadcrumb.text }}</a></li>
    {% endfor %}
  </ul>
  {% if success %}
  <div class="alert alert-success alert-dismissible"><i class="fa fa-check-circle"></i> {{ success
  }}</div>
  {% endif %}
  {% if error_warning %}
  <div class="alert alert-warning"><i class="fa fa-exclamation-circle"></i> {{ error_warning
  }}</div>
  {% endif %}
  <div class="row">{{ column_left }}
  {% if column_left and column_right %}
  {% set class = 'col-sm-6' %}
  {% elseif column_left or column_right %}
  {% set class = 'col-sm-9' %}
  {% else %}
  {% set class = 'col-sm-12' %}
  {% endif %}
  <div id="content" class="{{ class }}">{{ content_top }}
  <h2>{{ text_address_book }}</h2>
  {% if addresses %}
  <div class="table-responsive">
    <table class="table table-bordered table-hover">
      {% for result in addresses %}
      <form action="{{ result.delete }}" method="post" enctype="multipart/form-data"
      id="form-address-{{ result.address_id }}">
        <tr>
          <td class="text-left">{{ result.address }}</td>
          <td class="text-right"><a href="{{ result.update }}" class="btn btn-info">{{ button_edit
          }}</a> &nbsp;  
          <a href = "{{ result.delete }}" data-toggle="tooltip" title="{{ button_delete }}"
          class="btn btn-danger" onclick="return confirm('{{ text_confirm }}')">{{ button_delete
          }}</a></td>
        </tr>
      </form>
      {% endfor %}
    </table>
  </div>
  {% else %}
  <p>{{ text_empty }}</p>
  {% endif %}

  <div class="buttons clearfix">
    <div class="pull-left"><a href="{{ back }}" class="btn btn-default">{{ button_back
    }}</a></div>
    <div class="pull-right"><a href="{{ add }}" class="btn btn-primary">{{
    button_new_address }}</a></div>

```

```

    {{ content_bottom }}</div>
    {{ column_right }}</div>
</div>
{{ footer }}

{{ header }}
<div id="account-affiliate" class="container">
  <ul class="breadcrumb">
    {% for breadcrumb in breadcrumbs %}
    <li><a href="{{ breadcrumb.href }}">{{ breadcrumb.text }}</a></li>
    {% endfor %}
  </ul>
  {% if error_warning %}
  <div class="alert alert-danger alert-dismissible"><i class="fa fa-exclamation-circle"></i> {{
error_warning }}</div>
  {% endif %}
  <div class="row">{{ column_left }}
  {% if column_left and column_right %}
  {% set class = 'col-sm-6' %}
  {% elseif column_left or column_right %}
  {% set class = 'col-sm-9' %}
  {% else %}
  {% set class = 'col-sm-12' %}
  {% endif %}
  <div id="content" class="{{ class }}">{{ content_top }}
  <h1>{{ heading_title }}</h1>
  <form action="{{ action }}" method="post" enctype="multipart/form-data" class="form-
horizontal">
    <fieldset>
      <legend>{{ text_my_affiliate }}</legend>
      <div class="form-group">
        <label class="col-sm-2 control-label" for="input-company">{{ entry_company
}}</label>
        <div class="col-sm-10">
          <input type="text" name="company" value="{{ company }}" placeholder="{{
entry_company }}" id="input-company" class="form-control" />
        </div>
      </div>
      <div class="form-group">
        <label class="col-sm-2 control-label" for="input-website">{{ entry_website }}</label>
        <div class="col-sm-10">
          <input type="text" name="website" value="{{ website }}" placeholder="{{
entry_website }}" id="input-website" class="form-control" />
        </div>
      </div>
    </fieldset>
    <fieldset>
      <legend>{{ text_payment }}</legend>
      <div class="form-group">
        <label class="col-sm-2 control-label" for="input-tax">{{ entry_tax }}</label>
        <div class="col-sm-10">

```

```

<input type="text" name="tax" value="{{ tax }}" placeholder="{{ entry_tax }}" id="input-tax"
class="form-control" />
</div>
</div>
<div class="form-group">
<label class="col-sm-2 control-label">{{ entry_payment }}</label>
<div class="col-sm-10">
<div class="radio">
<label>{% if payment == 'cheque' %}
<input type="radio" name="payment" value="cheque" checked="checked" />
{% else %}
<input type="radio" name="payment" value="cheque" />
{% endif %}
{{ text_cheque }}</label>
</div>
<div class="radio">
<label>{% if payment == 'paypal' %}
<input type="radio" name="payment" value="paypal" checked="checked" />
{% else %}
<input type="radio" name="payment" value="paypal" />
{% endif %}
{{ text_paypal }}</label>
</div>
<div class="radio">
<label>{% if payment == 'bank' %}
<input type="radio" name="payment" value="bank" checked="checked" />
{% else %}
<input type="radio" name="payment" value="bank" />
{% endif %}
{{ text_bank }}</label>
</div>
</div>
</div>
<div id="payment-cheque" class="payment">
<div class="form-group required">
<label class="col-sm-2 control-label" for="input-cheque">{{ entry_cheque }}</label>
<div class="col-sm-10">
<input type="text" name="cheque" value="{{ cheque }}" placeholder="{{
entry_cheque }}" id="input-cheque" class="form-control" />
{% if error_cheque %}
<div class="text-danger">{{ error_cheque }}</div>
{% endif %}
</div>
</div>
</div>
<div id="payment-paypal" class="payment">
<div class="form-group required">
<label class="col-sm-2 control-label" for="input-paypal">{{ entry_paypal }}</label>
<div class="col-sm-10">
<input type="text" name="paypal" value="{{ paypal }}" placeholder="{{ entry_paypal
}}" id="input-paypal" class="form-control" />
{% if error_paypal %}

```

```

<div class="text-danger">{{ error_paypal }}</div>
    {% endif %}
</div>
</div>
</div>
<div class="payment" id="payment-bank">
    <div class="form-group">
        <label class="col-sm-2 control-label" for="input-bank-name">{{ entry_bank_name
    }}</label>
        <div class="col-sm-10">
            <input type="text" name="bank_name" value="{{ bank_name }}" placeholder="{{
entry_bank_name }}" id="input-bank-name" class="form-control" />
        </div>
    </div>
    <div class="form-group">
        <label class="col-sm-2 control-label" for="input-bank-branch-number">{{
entry_bank_branch_number }}</label>
        <div class="col-sm-10">
            <input type="text" name="bank_branch_number" value="{{ bank_branch_number }}"
placeholder="{{ entry_bank_branch_number }}" id="input-bank-branch-number" class="form-
control" />
        </div>
    </div>
    <div class="form-group">
        <label class="col-sm-2 control-label" for="input-bank-swift-code">{{
entry_bank_swift_code }}</label>
        <div class="col-sm-10">
            <input type="text" name="bank_swift_code" value="{{ bank_swift_code }}"
placeholder="{{ entry_bank_swift_code }}" id="input-bank-swift-code" class="form-control" />
        </div>
    </div>
    <div class="form-group required">
        <label class="col-sm-2 control-label" for="input-bank-account-name">{{
entry_bank_account_name }}</label>
        <div class="col-sm-10">
            <input type="text" name="bank_account_name" value="{{ bank_account_name }}"
placeholder="{{ entry_bank_account_name }}" id="input-bank-account-name" class="form-
control" />
            {% if error_bank_account_name %}
            <div class="text-danger">{{ error_bank_account_name }}</div>
            {% endif %}
        </div>
    </div>
    <div class="form-group required">
        <label class="col-sm-2 control-label" for="input-bank-account-number">{{
entry_bank_account_number }}</label>
        <div class="col-sm-10">
            <input type="text" name="bank_account_number" value="{{ bank_account_number
    }}" placeholder="{{ entry_bank_account_number }}" id="input-bank-account-number"
class="form-control" />
            {% if error_bank_account_number %}
            <div class="text-danger">{{ error_bank_account_number }}</div>
            {% endif %}
        </div>
    </div>

```

```

{% endif %}
</div>
</div>
</div>
{% for custom_field in custom_fields %}
{% if custom_field.location == 'affiliate' %}
{% if custom_field.type == 'select' %}
<div class="form-group{% if custom_field.required %} required {% endif %} custom-
field" data-sort="{{ custom_field.sort_order }}">
<label class="col-sm-2 control-label" for="input-custom-field{{
custom_field.custom_field_id }}">{{ custom_field.name }}</label>
<div class="col-sm-10">
<select name="custom_field[{{ custom_field.location }}][{{
custom_field.custom_field_id }}" id="input-custom-field{{ custom_field.custom_field_id }}"
class="form-control">
<option value="">{{ text_select }}</option>
{% for custom_field_value in custom_field.custom_field_value %}
{% if affiliate_custom_field[custom_field.custom_field_id] and
custom_field_value.custom_field_value_id ==
affiliate_custom_field[custom_field.custom_field_id] %}
<option value="{{ custom_field_value.custom_field_value_id }}"
selected="selected">{{ custom_field_value.name }}</option>
{% else %}
<option value="{{ custom_field_value.custom_field_value_id }}">{{
custom_field_value.name }}</option>
{% endif %}
{% endfor %}
</select>
{% if error_custom_field[custom_field.custom_field_id] %}
<div class="text-danger">{{ error_custom_field[custom_field.custom_field_id]
}}</div>
{% endif %}
</div>
</div>
</div>
{% endif %}
{% if custom_field.type == 'radio' %}
<div class="form-group{% if custom_field.required %} required {% endif %} custom-
field" data-sort="{{ custom_field.sort_order }}">
<label class="col-sm-2 control-label">{{ custom_field.name }}</label>
<div class="col-sm-10">
<div>
{% for custom_field_value in custom_field.custom_field_value %}
<div class="radio">
{% if affiliate_custom_field[custom_field.custom_field_id] and
custom_field_value.custom_field_value_id ==
affiliate_custom_field[custom_field.custom_field_id] %}
<label>
<input type="radio" name="custom_field[{{ custom_field.location }}][{{
custom_field.custom_field_id }}" value="{{ custom_field_value.custom_field_value_id }}"
checked="checked" />
{{ custom_field_value.name }}</label>
{% else %}

```

```

<label>
    <input type="radio" name="custom_field[{{ custom_field.location }}][{{
custom_field.custom_field_id }}]" value="{{ custom_field_value.custom_field_value_id }}" />
    {{ custom_field_value.name }}</label>
    {% endif %}
</div>
{% endfor %}
</div>
{% if error_custom_field[custom_field.custom_field_id] %}
<div class="text-danger">{{ error_custom_field[custom_field.custom_field_id]
}}</div>
{% endif %}
</div>
</div>
{% endif %}
{% if custom_field.type == 'checkbox' %}
<div class="form-group{% if custom_field.required %} required {% endif %} custom-
field" data-sort="{{ custom_field.sort_order }}">
<label class="col-sm-2 control-label">{{ custom_field.name }}</label>
<div class="col-sm-10">
<div>
    {% for custom_field_value in custom_field.custom_field_value %}
    <div class="checkbox">
        {% if affiliate_custom_field[custom_field.custom_field_id] and
custom_field_value.custom_field_value_id in
affiliate_custom_field[custom_field.custom_field_id] %}
<label>
            <input type="checkbox" name="custom_field[{{ custom_field.location }}][{{
custom_field.custom_field_id }}]" value="{{ custom_field_value.custom_field_value_id }}"
checked="checked" />
            {{ custom_field_value.name }}</label>
        {% else %}
<label>
            <input type="checkbox" name="custom_field[{{ custom_field.location }}][{{
custom_field.custom_field_id }}]" value="{{ custom_field_value.custom_field_value_id }}"
/>
            {{ custom_field_value.name }}</label>
        {% endif %}
    </div>
    {% endfor %}
</div>
    {% if error_custom_field[custom_field.custom_field_id] %}
<div class="text-danger">{{ error_custom_field[custom_field.custom_field_id]
}}</div>
    {% endif %}
</div>
</div>
{% endif %}
{% if custom_field.type == 'text' %}
<div class="form-group{% if custom_field.required %} required {% endif %} custom-
field" data-sort="{{ custom_field.sort_order }}">

```

```

<label class="col-sm-2 control-label" for="input-custom-field{{
custom_field.custom_field_id }}">{{ custom_field.name }}</label>
<div class="col-sm-10">
  <input type="text" name="custom_field[{{ custom_field.location }}][{{
custom_field.custom_field_id }}" value="{% if
affiliate_custom_field[custom_field.custom_field_id] %}{{
affiliate_custom_field[custom_field.custom_field_id] }}{% else %}{{ custom_field.value }}{%
endif %}" placeholder="{{ custom_field.name }}" id="input-custom-field{{
custom_field.custom_field_id }}" class="form-control" />
  {% if error_custom_field[custom_field.custom_field_id] %}
  <div class="text-danger">{{ error_custom_field[custom_field.custom_field_id]
}}</div>
  {% endif %}
</div>
</div>
{% endif %}
{% if custom_field.type == 'textarea' %}
<div class="form-group{% if custom_field.required %} required {% endif %} custom-
field" data-sort="{{ custom_field.sort_order }}">
  <label class="col-sm-2 control-label" for="input-custom-field{{
custom_field.custom_field_id }}">{{ custom_field.name }}</label>
  <div class="col-sm-10">
    <textarea name="custom_field[{{ custom_field.location }}][{{
custom_field.custom_field_id }}" rows="5" placeholder="{{ custom_field.name }}" id="input-
custom-field{{ custom_field.custom_field_id }}" class="form-control">{% if
affiliate_custom_field[custom_field.custom_field_id] %}{{
affiliate_custom_field[custom_field.custom_field_id] }}{% else %}{{ custom_field.value }}{%
endif %}</textarea>
    {% if error_custom_field[custom_field.custom_field_id] %}
    <div class="text-danger">{{ error_custom_field[custom_field.custom_field_id]
}}</div>
    {% endif %}
  </div>
</div>
{% endif %}
{% if custom_field.type == 'file' %}
<div class="form-group{% if custom_field.required %} required {% endif %} custom-
field" data-sort="{{ custom_field.sort_order }}">
  <label class="col-sm-2 control-label">{{ custom_field.name }}</label>
  <div class="col-sm-10">
    <button type="button" id="button-custom-field{{ custom_field.custom_field_id }}"
data-loading-text="{{ text_loading }}" class="btn btn-default"><i class="fa fa-upload"></i> {{
button_upload }}</button>
    <input type="hidden" name="custom_field[{{ custom_field.location }}][{{
custom_field.custom_field_id }}" value="{% if
affiliate_custom_field[custom_field.custom_field_id] %}{{
affiliate_custom_field[custom_field.custom_field_id] }}{% endif %}" />
    {% if error_custom_field[custom_field.custom_field_id] %}
    <div class="text-danger">{{ error_custom_field[custom_field.custom_field_id]
}}</div>
    {% endif %}
  </div>

```

```

</div>
    {% endif %}
    {% if custom_field.type == 'date' %}
    <div class="form-group{% if custom_field.required %} required {% endif %} custom-
field" data-sort="{{ custom_field.sort_order }}">
        <label class="col-sm-2 control-label" for="input-custom-field{{
custom_field.custom_field_id }}">{{ custom_field.name }}</label>
        <div class="col-sm-10">
            <div class="input-group date">
                <input type="text" name="custom_field[{{ custom_field.location }}][{{
custom_field.custom_field_id }}" value="{% if
affiliate_custom_field[custom_field.custom_field_id] %}{{
affiliate_custom_field[custom_field.custom_field_id] }}{% else %}{{ custom_field.value }}{%
endif %}" placeholder="{{ custom_field.name }}" data-date-format="YYYY-MM-DD"
id="input-custom-field{{ custom_field.custom_field_id }}" class="form-control" />
                <span class="input-group-btn">
                    <button type="button" class="btn btn-default"><i class="fa fa-
calendar"></i></button>
                </span></div>
                {% if error_custom_field[custom_field.custom_field_id] %}
                <div class="text-danger">{{ error_custom_field[custom_field.custom_field_id]
}}</div>
            {% endif %}
        </div>
    </div>
    </div>
    {% endif %}
    {% if custom_field.type == 'time' %}
    <div class="form-group{% if custom_field.required %} required {% endif %} custom-
field" data-sort="{{ custom_field.sort_order }}">
        <label class="col-sm-2 control-label" for="input-custom-field{{
custom_field.custom_field_id }}">{{ custom_field.name }}</label>
        <div class="col-sm-10">
            <div class="input-group time">
                <input type="text" name="custom_field[{{ custom_field.location }}][{{
custom_field.custom_field_id }}" value="{% if
affiliate_custom_field[custom_field.custom_field_id] %}{{
affiliate_custom_field[custom_field.custom_field_id] }}{% else %}{{ custom_field.value }}{%
endif %}" placeholder="{{ custom_field.name }}" data-date-format="HH:mm" id="input-
custom-field{{ custom_field.custom_field_id }}" class="form-control" />
                <span class="input-group-btn">
                    <button type="button" class="btn btn-default"><i class="fa fa-
calendar"></i></button>
                </span></div>
                {% if error_custom_field[custom_field.custom_field_id] %}
                <div class="text-danger">{{ error_custom_field[custom_field.custom_field_id]
}}</div>
            {% endif %}
        </div>
    </div>
    </div>
    {% endif %}
    {% if custom_field.type == 'datetime' %}

```



```

<div class="form-group{% if custom_field.required %} required {% endif %} custom-field"
data-sort="{{ custom_field.sort_order }}">
  <label class="col-sm-2 control-label" for="input-custom-field{{
custom_field.custom_field_id }}">{{ custom_field.name }}</label>
  <div class="col-sm-10">
    <div class="input-group datetime">
      <input type="text" name="custom_field[{{ custom_field.location }}][{{
custom_field.custom_field_id }}" value="{% if
affiliate_custom_field[custom_field.custom_field_id] %}{{
affiliate_custom_field[custom_field.custom_field_id] }}{% else %}{{ custom_field.value }}{%
endif %}" placeholder="{{ custom_field.name }}" data-date-format="YYYY-MM-DD
HH:mm" id="input-custom-field{{ custom_field.custom_field_id }}" class="form-control" />
      <span class="input-group-btn">
        <button type="button" class="btn btn-default"><i class="fa fa-
calendar"></i></button>
      </span></div>
      {% if error_custom_field[custom_field.custom_field_id] %}
      <div class="text-danger">{{ error_custom_field[custom_field.custom_field_id]
}}</div>
      {% endif %}
    </div>
  </div>
</div>
{% endif %}
{% endfor %}
</fieldset>
{% if text_agree %}
<div class="buttons clearfix">
  <div class="pull-right">{{ text_agree }}
  {% if agree %}
  <input type="checkbox" name="agree" value="1" checked="checked" />
  {% else %}
  <input type="checkbox" name="agree" value="1" />
  {% endif %}
  &nbsp;
  <input type="submit" value="{{ button_continue }}" class="btn btn-primary" />
</div>
</div>
{% else %}
<div class="buttons clearfix">
  <div class="pull-right">
    <input type="submit" value="{{ button_continue }}" class="btn btn-primary" />
  </div>
</div>
{% endif %}
</form>
{{ content_bottom }}</div>
{{ column_right }}</div>
</div>
<script type="text/javascript"><!--
$(input[name='payment']).on('change', function() {
  $('payment').hide();

```

```

    $('#payment-' + this.value).show();
});

$('input[name='\payment\']:checked').trigger('change');
//--></script>
<script type="text/javascript"><!--
// Sort the custom fields
$('.form-group[data-sort]).detach().each(function() {
    if ($(this).attr('data-sort') >= 0 && $(this).attr('data-sort') <= $('.form-group').length) {
        $('.form-group').eq($(this).attr('data-sort')).before(this);
    }

    if ($(this).attr('data-sort') > $('.form-group').length) {
        $('.form-group:last').after(this);
    }

    if ($(this).attr('data-sort') == $('.form-group').length) {
        $('.form-group:last').after(this);
    }

    if ($(this).attr('data-sort') < -$('.form-group').length) {
        $('.form-group:first').before(this);
    }
});
//--></script>
<script type="text/javascript"><!--
$('button[id^='\button-custom-field\"]').on('click', function() {
    var node = this;

    $('#form-upload').remove();

    $('body').prepend('<form enctype="multipart/form-data" id="form-upload"
style="display: none;"><input type="file" name="file" /></form>');

    $('#form-upload input[name='\file\']).trigger('click');

    if (typeof timer != 'undefined') {
        clearInterval(timer);
    }

    timer = setInterval(function() {
        if ($('#form-upload input[name='\file\']).val() != "") {
            clearInterval(timer);

            $.ajax({
                url: 'index.php?route=tool/upload',
                type: 'post',
                dataType: 'json',
                data: new FormData($('#form-upload')[0]),
                cache: false,
                contentType: false,
                processData: false,

```

```

beforeSend: function() {
    $(node).button('loading');
},
complete: function() {
    $(node).button('reset');
},
success: function(json) {
    $(node).parent().find('.text-danger').remove();

    if (json['error']) {
        $(node).parent().find('input').after("<div class='text-
danger'>' + json['error'] + '</div>");
    }

    if (json['success']) {
        alert(json['success']);

        $(node).parent().find('input').val(json['code']);
    }
},
error: function(xhr, ajaxOptions, thrownError) {
    alert(thrownError + "\r\n" + xhr.statusText + "\r\n" +
xhr.responseText);
}
});
}, 500);
});
<!--></script>
<script type="text/javascript"><!--
$('.date').datetimepicker({
    language: '{{ datepicker }}',
    pickTime: false
});

$('.datetime').datetimepicker({
    language: '{{ datepicker }}',
    pickDate: true,
    pickTime: true
});

$('.time').datetimepicker({
    language: '{{ datepicker }}',
    pickDate: false
});
<!--></script>
{{ footer }}

{{ header }}
<div id="account-download" class="container">
<ul class="breadcrumb">
    {% for breadcrumb in breadcrumbs %}

```

```

<li><a href="{{ breadcrumb.href }}">{{ breadcrumb.text }}</a></li>
  {% endfor %}
</ul>
<div class="row">{{ column_left }}
  {% if column_left and column_right %}
  {% set class = 'col-sm-6' %}
  {% elseif column_left or column_right %}
  {% set class = 'col-sm-9' %}
  {% else %}
  {% set class = 'col-sm-12' %}
  {% endif %}
  <div id="content" class="{{ class }}">{{ content_top }}
  <h2>{{ heading_title }}</h2>
  {% if downloads %}
  <div class="table-responsive">
    <table class="table table-bordered table-hover">
      <thead>
        <tr>
          <td class="text-right">{{ column_order_id }}</td>
          <td class="text-left">{{ column_name }}</td>
          <td class="text-left">{{ column_size }}</td>
          <td class="text-left">{{ column_date_added }}</td>
          <td></td>
        </tr>
      </thead>
      <tbody>

        {% for download in downloads %}
        <tr>
          <td class="text-right">{{ download.order_id }}</td>
          <td class="text-left">{{ download.name }}</td>
          <td class="text-left">{{ download.size }}</td>
          <td class="text-left">{{ download.date_added }}</td>
          <td><a href="{{ download.href }}" data-toggle="tooltip" title="{{ button_download }}"
class="btn btn-primary"><i class="fa fa-cloud-download"></i></a></td>
        </tr>
        {% endfor %}
      </tbody>

    </table>
  </div>
  <div class="row">
    <div class="col-sm-6 text-left">{{ pagination }}</div>
    <div class="col-sm-6 text-right">{{ results }}</div>
  </div>
  {% else %}
  <p>{{ text_empty }}</p>
  {% endif %}
  <div class="buttons clearfix">
    <div class="pull-right"><a href="{{ continue }}" class="btn btn-primary">{{
button_continue }}</a></div>
  </div>

```

```

    {{ content_bottom }}</div>
    {{ column_right }}</div>
</div>
{{ footer }}

{{ header }}
<div id="account-edit" class="container">
  <ul class="breadcrumb">
    {% for breadcrumb in breadcrumbs %}
    <li><a href="{{ breadcrumb.href }}">{{ breadcrumb.text }}</a></li>
    {% endfor %}
  </ul>
  {% if error_warning %}
  <div class="alert alert-danger alert-dismissible"><i class="fa fa-exclamation-circle"></i> {{
error_warning }}</div>
  {% endif %}
  <div class="row">{{ column_left }}
    {% if column_left and column_right %}
    {% set class = 'col-sm-6' %}
    {% elseif column_left or column_right %}
    {% set class = 'col-sm-9' %}
    {% else %}
    {% set class = 'col-sm-12' %}
    {% endif %}
    <div id="content" class="{{ class }}">{{ content_top }}
    <h1>{{ heading_title }}</h1>
    <form action="{{ action }}" method="post" enctype="multipart/form-data" class="form-
horizontal">
      <fieldset>
        <legend>{{ text_your_details }}</legend>
        <div class="form-group required">
          <label class="col-sm-2 control-label" for="input-firstname">{{ entry_firstname }}
</label>
          <div class="col-sm-10">
            <input type="text" name="firstname" value="{{ firstname }}" placeholder="{{
entry_firstname }}" id="input-firstname" class="form-control" />
            {% if error_firstname %}
            <div class="text-danger">{{ error_firstname }}</div>
            {% endif %}
          </div>
        </div>
        <div class="form-group required">
          <label class="col-sm-2 control-label" for="input-lastname">{{ entry_lastname
}}</label>
          <div class="col-sm-10">
            <input type="text" name="lastname" value="{{ lastname }}" placeholder="{{
entry_lastname }}" id="input-lastname" class="form-control" />
            {% if error_lastname %}
            <div class="text-danger">{{ error_lastname }}</div>
            {% endif %}
          </div>
        </div>
      </div>
    </div>

```

```

<div class="form-group required">
    <label class="col-sm-2 control-label" for="input-email">{{ entry_email }}</label>
    <div class="col-sm-10">
        <input type="email" name="email" value="{{ email }}" placeholder="{{ entry_email
}} " id="input-email" class="form-control" />
        {% if error_email %}
        <div class="text-danger">{{ error_email }}</div>
        {% endif %}
    </div>
</div>
<div class="form-group required">
    <label class="col-sm-2 control-label" for="input-telephone">{{ entry_telephone
}}</label>
    <div class="col-sm-10">
        <input type="tel" name="telephone" value="{{ telephone }}" placeholder="{{
entry_telephone }}" id="input-telephone" class="form-control" />
        {% if error_telephone %}
        <div class="text-danger">{{ error_telephone }}</div>
        {% endif %}
    </div>
</div>
{% for custom_field in custom_fields %}
{% if custom_field.type == 'select' %}
    <div class="form-group{% if custom_field.required %} required {% endif %} custom-
field" data-sort="{{ custom_field.sort_order }}">
        <label class="col-sm-2 control-label" for="input-custom-field{{
custom_field.custom_field_id }}">{{ custom_field.name }}</label>
        <div class="col-sm-10">
            <select name="custom_field[{{ custom_field.location }}][{{
custom_field.custom_field_id }}" id="input-custom-field{{ custom_field.custom_field_id }}"
class="form-control">
                <option value="">{{ text_select }}</option>
                {% for custom_field_value in custom_field.custom_field_value %}
                {% if account_custom_field[custom_field.custom_field_id] and
custom_field_value.custom_field_value_id ==
account_custom_field[custom_field.custom_field_id] %}
                <option value="{{ custom_field_value.custom_field_value_id }}"
selected="selected">{{ custom_field_value.name }}</option>
                {% else %}
                <option value="{{ custom_field_value.custom_field_value_id }}">{{
custom_field_value.name }}</option>
                {% endif %}
                {% endfor %}
            </select>
            {% if error_custom_field[custom_field.custom_field_id] %}
            <div class="text-danger">{{ error_custom_field[custom_field.custom_field_id]
}}</div>
            {% endif %}
        </div>
    </div>
{% endif %}
{% if custom_field.type == 'radio' %}

```

```

    <div class="form-group{% if custom_field.required %} required {% endif %} custom-
field" data-sort="{{ custom_field.sort_order }}">
        <label class="col-sm-2 control-label">{{ custom_field.name }}</label>
        <div class="col-sm-10">
            <div>
                {% for custom_field_value in custom_field.custom_field_value %}
                <div class="radio">
                    {% if account_custom_field[custom_field.custom_field_id] and
custom_field_value.custom_field_value_id ==
account_custom_field[custom_field.custom_field_id] %}
                    <label>
                        <input type="radio" name="custom_field[{{ custom_field.location }}][{{
custom_field.custom_field_id }}]" value="{{ custom_field_value.custom_field_value_id }}"
checked="checked" />
                        {{ custom_field_value.name }}</label>
                    {% else %}
                    <label>
                        <input type="radio" name="custom_field[{{ custom_field.location }}][{{
custom_field.custom_field_id }}]" value="{{ custom_field_value.custom_field_value_id }}" />
                        {{ custom_field_value.name }}</label>
                    {% endif %}
                </div>
                {% endfor %}
            </div>
            {% if error_custom_field[custom_field.custom_field_id] %}
            <div class="text-danger">{{ error_custom_field[custom_field.custom_field_id]
}}</div>
            {% endif %}
        </div>
    </div>
    {% endif %}
    {% if custom_field.type == 'checkbox' %}
    <div class="form-group{% if custom_field.required %} required {% endif %} custom-
field" data-sort="{{ custom_field.sort_order }}">
        <label class="col-sm-2 control-label">{{ custom_field.name }}</label>
        <div class="col-sm-10">
            <div>
                {% for custom_field_value in custom_field.custom_field_value %}
                <div class="checkbox">
                    {% if account_custom_field[custom_field.custom_field_id] and
custom_field_value.custom_field_value_id in
account_custom_field[custom_field.custom_field_id] %}
                    <label>
                        <input type="checkbox" name="custom_field[{{ custom_field.location }}][{{
custom_field.custom_field_id }}]" value="{{ custom_field_value.custom_field_value_id }}"
checked="checked" />
                        {{ custom_field_value.name }}</label>
                    {% else %}
                    <label>

```

```

        <input type="checkbox" name="custom_field[{{ custom_field.location }}][{{
custom_field.custom_field_id }}]" value="{{ custom_field_value.custom_field_value_id }}"
/>
        {{ custom_field_value.name }}</label>
    {% endif %}
</div>
{% endfor %}
</div>
{% if error_custom_field[custom_field.custom_field_id] %}
<div class="text-danger">{{ error_custom_field[custom_field.custom_field_id]
}}</div>
{% endif %}
</div>
</div>
{% endif %}
{% if custom_field.type == 'text' %}
<div class="form-group{% if custom_field.required %} required {% endif %} custom-
field" data-sort="{{ custom_field.sort_order }}">
    <label class="col-sm-2 control-label" for="input-custom-field{{
custom_field.custom_field_id }}">{{ custom_field.name }}</label>
    <div class="col-sm-10">
        <input type="text" name="custom_field[{{ custom_field.location }}][{{
custom_field.custom_field_id }}" value="{% if
account_custom_field[custom_field.custom_field_id] %}{{
account_custom_field[custom_field.custom_field_id] }}{% else %}{{ custom_field.value }}{%
endif %}" placeholder="{{ custom_field.name }}" id="input-custom-field{{
custom_field.custom_field_id }}" class="form-control" />
        {% if error_custom_field[custom_field.custom_field_id] %}
        <div class="text-danger">{{ error_custom_field[custom_field.custom_field_id]
}}</div>
        {% endif %}
    </div>
</div>
{% endif %}
{% if custom_field.type == 'textarea' %}
<div class="form-group{% if custom_field.required %} required {% endif %} custom-
field" data-sort="{{ custom_field.sort_order }}">
    <label class="col-sm-2 control-label" for="input-custom-field{{
custom_field.custom_field_id }}">{{ custom_field.name }}</label>
    <div class="col-sm-10">
        <textarea name="custom_field[{{ custom_field.location }}][{{
custom_field.custom_field_id }}" rows="5" placeholder="{{ custom_field.name }}" id="input-
custom-field{{ custom_field.custom_field_id }}" class="form-control">{% if
account_custom_field[custom_field.custom_field_id] %}{{
account_custom_field[custom_field.custom_field_id] }}{% else %}{{ custom_field.value }}{%
endif %}</textarea>
        {% if error_custom_field[custom_field.custom_field_id] %}
        <div class="text-danger">{{ error_custom_field[custom_field.custom_field_id]
}}</div>
        {% endif %}
    </div>
</div>

```



```

    {% endif %}
    {% if custom_field.type == 'file' %}
    <div class="form-group{% if custom_field.required %} required {% endif %} custom-
field" data-sort="{{ custom_field.sort_order }}">
    <label class="col-sm-2 control-label">{{ custom_field.name }}</label>
    <div class="col-sm-10">
    <button type="button" id="button-custom-field{{ custom_field.custom_field_id }}"
data-loading-text="{{ text_loading }}" class="btn btn-default"><i class="fa fa-upload"></i> {{
button_upload }}</button>
    <input type="hidden" name="custom_field[{{ custom_field.location }}][{{
custom_field.custom_field_id }}" value="{% if
account_custom_field[custom_field.custom_field_id]['code'] %}{{
account_custom_field[custom_field.custom_field_id]['code'] }} {% endif %}" />
    <span>{{ account_custom_field[custom_field.custom_field_id]['name'] }}</span>
    {% if error_custom_field[custom_field.custom_field_id] %}
    <div class="text-danger">{{ error_custom_field[custom_field.custom_field_id]
}}</div>
    {% endif %}
    </div>
</div>
</div>
{% endif %}
{% if custom_field.type == 'date' %}
<div class="form-group{% if custom_field.required %} required {% endif %} custom-
field" data-sort="{{ custom_field.sort_order }}">
<label class="col-sm-2 control-label" for="input-custom-field{{
custom_field.custom_field_id }}">{{ custom_field.name }}</label>
<div class="col-sm-10">
<div class="input-group date">
<input type="text" name="custom_field[{{ custom_field.location }}][{{
custom_field.custom_field_id }}" value="{% if
account_custom_field[custom_field.custom_field_id] %}{{
account_custom_field[custom_field.custom_field_id] }}{% else %}{{ custom_field.value }}{%
endif %}" placeholder="{{ custom_field.name }}" data-date-format="YYYY-MM-DD"
id="input-custom-field{{ custom_field.custom_field_id }}" class="form-control" />
<span class="input-group-btn">
<button type="button" class="btn btn-default"><i class="fa fa-
calendar"></i></button>
</span></div>
    {% if error_custom_field[custom_field.custom_field_id] %}
    <div class="text-danger">{{ error_custom_field[custom_field.custom_field_id]
}}</div>
    {% endif %}
    </div>
</div>
</div>
{% endif %}
{% if custom_field.type == 'time' %}
<div class="form-group{% if custom_field.required %} required {% endif %} custom-
field" data-sort="{{ custom_field.sort_order }}">
<label class="col-sm-2 control-label" for="input-custom-field{{
custom_field.custom_field_id }}">{{ custom_field.name }}</label>
<div class="col-sm-10">
<div class="input-group time">

```



```

    {{ column_right }}</div>
</div>
<script type="text/javascript"><!--
// Sort the custom fields
$('.form-group[data-sort]').detach().each(function() {
    if ($(this).attr('data-sort') >= 0 && $(this).attr('data-sort') <= $('.form-group').length) {
        $('.form-group').eq($(this).attr('data-sort')).before(this);
    }

    if ($(this).attr('data-sort') > $('.form-group').length) {
        $('.form-group:last').after(this);
    }

    if ($(this).attr('data-sort') == $('.form-group').length) {
        $('.form-group:last').after(this);
    }

    if ($(this).attr('data-sort') < -$('.form-group').length) {
        $('.form-group:first').before(this);
    }
});
--></script>
<script type="text/javascript"><!--
$('#button[id^=\button-custom-field\]').on('click', function() {
    var element = this;

    $('#form-upload').remove();

    $('body').prepend('<form enctype="multipart/form-data" id="form-upload"
style="display: none;"><input type="file" name="file" /></form>');

    $('#form-upload input[name=\file\']).trigger('click');

    if (typeof timer != 'undefined') {
        clearInterval(timer);
    }

    timer = setInterval(function() {
        if ($('#form-upload input[name=\file\']).val() != "") {
            clearInterval(timer);

            $.ajax({
                url: 'index.php?route=tool/upload',
                type: 'post',
                dataType: 'json',
                data: new FormData($('#form-upload')[0]),
                cache: false,
                contentType: false,
                processData: false,
                beforeSend: function() {
                    $(element).button('loading');

```

```

    },
    complete: function() {
        $(element).button('reset');
    },
    success: function(json) {
        $(element).parent().find('.text-danger').remove();

        if (json['error']) {
            $(element).parent().find('input').after('<div
class="text-danger">' + json['error'] + '</div>');
        }

        if (json['success']) {
            alert(json['success']);

            $(element).parent().find('input').val(json['code']);
        }
    },
    error: function(xhr, ajaxOptions, thrownError) {
        alert(thrownError + "\r\n" + xhr.statusText + "\r\n" +
xhr.responseText);
    }
});
}, 500);
});
<!--></script>
<script type="text/javascript"><!--
$( '.date' ).datetimepicker({
    language: '{{ datepicker }}',
    pickTime: false
});

$( '.datetime' ).datetimepicker({
    language: '{{ datepicker }}',
    pickDate: true,
    pickTime: true
});

$( '.time' ).datetimepicker({
    language: '{{ datepicker }}',
    pickDate: false
});
<!--></script>
{{ footer }}

{{ header }}
<div id="account-reset" class="container">
    <ul class="breadcrumb">
        {% for breadcrumb in breadcrumbs %}
        <li><a href="{{ breadcrumb.href }}">{{ breadcrumb.text }}</a></li>
        {% endfor %}

```

```

</ul>
<div class="row">{{ column_left }}
  {% if column_left and column_right %}
  {% set class = 'col-sm-6' %}
  {% elseif column_left or column_right %}
  {% set class = 'col-sm-9' %}
  {% else %}
  {% set class = 'col-sm-12' %}
  {% endif %}
  <div id="content" class="{{ class }}">{{ content_top }}
    <h1>{{ heading_title }}</h1>
    <form action="{{ action }}" method="post" enctype="multipart/form-data" class="form-
horizontal">
      <fieldset>
        <legend>{{ text_password }}</legend>
        <div class="form-group">
          <label class="col-sm-2 control-label" for="input-password">{{ entry_password
}}</label>
          <div class="col-sm-10">
            <input type="password" name="password" value="{{ password }}" id="input-
password" class="form-control" />
            {% if error_password %}
            <div class="text-danger">{{ error_password }}</div>
            {% endif %} </div>
          </div>
          <div class="form-group">
            <label class="col-sm-2 control-label" for="input-confirm">{{ entry_confirm }}</label>
            <div class="col-sm-10">
              <input type="password" name="confirm" value="{{ confirm }}" id="input-confirm"
class="form-control" />
              {% if error_confirm %}
              <div class="text-danger">{{ error_confirm }}</div>
              {% endif %}</div>
            </div>
          </fieldset>
          <div class="buttons clearfix">
            <div class="pull-left"><a href="{{ back }}" class="btn btn-default">{{ button_back
}}</a></div>
            <div class="pull-right">
              <button type="submit" class="btn btn-primary">{{ button_continue }}</button>
            </div>
          </div>
        </form>
        {{ content_bottom }}</div>
      {{ column_right }}</div>
</div>
{{ footer }}

{{ header }}
<div id="account-password" class="container">
  <ul class="breadcrumb">
    {% for breadcrumb in breadcrumbs %}

```

```

    <li><a href="{{ breadcrumb.href }}">{{ breadcrumb.text }}</a></li>
  {% endfor %}
</ul>
<div class="row">{{ column_left }}
  {% if column_left and column_right %}
  {% set class = 'col-sm-6' %}
  {% elseif column_left or column_right %}
  {% set class = 'col-sm-9' %}
  {% else %}
  {% set class = 'col-sm-12' %}
  {% endif %}
  <div id="content" class="{{ class }}">{{ content_top }}
    <h1>{{ heading_title }}</h1>
    <form action="{{ action }}" method="post" enctype="multipart/form-data" class="form-
horizontal">
      <fieldset>
        <legend>{{ text_password }}</legend>
        <div class="form-group required">
          <label class="col-sm-2 control-label" for="input-password">{{ entry_password
}}</label>
          <div class="col-sm-10">
            <input type="password" name="password" value="{{ password }}" placeholder="{{
entry_password }}" id="input-password" class="form-control" />
            {% if error_password %}
            <div class="text-danger">{{ error_password }}</div>
            {% endif %}
          </div>
        </div>
        <div class="form-group required">
          <label class="col-sm-2 control-label" for="input-confirm">{{ entry_confirm }}</label>
          <div class="col-sm-10">
            <input type="password" name="confirm" value="{{ confirm }}" placeholder="{{
entry_confirm }}" id="input-confirm" class="form-control" />
            {% if error_confirm %}
            <div class="text-danger">{{ error_confirm }}</div>
            {% endif %}
          </div>
        </div>
      </fieldset>
      <div class="buttons clearfix">
        <div class="pull-left"><a href="{{ back }}" class="btn btn-default">{{ button_back
}}</a></div>
        <div class="pull-right">
          <input type="submit" value="{{ button_continue }}" class="btn btn-primary" />
        </div>
      </div>
    </form>
    {{ content_bottom }}</div>
  {{ column_right }}</div>
</div>
{{ footer }}

```

```

{{ header }}
<div id="account-recurring" class="container">
  <ul class="breadcrumb">
    {% for breadcrumb in breadcrumbs %}
      <li><a href="{{ breadcrumb.href }}">{{ breadcrumb.text }}</a></li>
    {% endfor %}
  </ul>
  <div class="row">{{ column_left }}
    {% if column_left and column_right %}
      {% set class = 'col-sm-6' %}
    {% elseif column_left or column_right %}
      {% set class = 'col-sm-9' %}
    {% else %}
      {% set class = 'col-sm-12' %}
    {% endif %}
    <div id="content" class="{{ class }}">{{ content_top }}
      <h2>{{ heading_title }}</h2>
      <div class="table-responsive">
        <table class="table table-bordered table-hover">
          <thead>
            <tr>
              <td class="text-left" colspan="2">{{ text_recurring_detail }}</td>
            </tr>
          </thead>
          <tbody>
            <tr>
              <td class="text-left" style="width: 50%;"><b>{{ text_order_recurring_id }}</b> #{{
order_recurring_id }}<br/>
                <b>{{ text_date_added }}</b> {{ date_added }}<br/>
                <b>{{ text_status }}</b> {{ status }}<br />
                <b>{{ text_payment_method }}</b> {{ payment_method }}</td>
              <td class="text-left" style="width: 50%;"><b>{{ text_order_id }}</b> <a href="{{
order }}">#{{ order_id }}</a><br/>
                <b>{{ text_product }}</b> <a href="{{ product }}">{{ product_name }}</a><br/>
                <b>{{ text_quantity }}</b> {{ product_quantity }}</td>
            </tr>
          </tbody>
        </table>
        <table class="table table-bordered table-hover">
          <thead>
            <tr>
              <td class="text-left">{{ text_description }}</td>
              <td class="text-left">{{ text_reference }}</td>
            </tr>
          </thead>
          <tbody>
            <tr>
              <td class="text-left" style="width: 50%;">{{ recurring_description }}</td>
              <td class="text-left" style="width: 50%;">{{ reference }}</td>
            </tr>
          </tbody>
        </table>

```

```

</div>
<h3>{{ text_transaction }}</h3>
<div class="table-responsive">
  <table class="table table-bordered table-hover">
    <thead>
      <tr>
        <td class="text-left">{{ column_date_added }}</td>
        <td class="text-left">{{ column_type }}</td>
        <td class="text-right">{{ column_amount }}</td>
      </tr>
    </thead>
    <tbody>

      {% if transactions %}
      {% for transaction in transactions %}
      <tr>
        <td class="text-left">{{ transaction.date_added }}</td>
        <td class="text-left">{{ transaction.type }}</td>
        <td class="text-right">{{ transaction.amount }}</td>
      </tr>
      {% endfor %}
      {% else %}
      <tr>
        <td colspan="3" class="text-center">{{ text_no_results }}</td>
      </tr>
      {% endif %}
    </tbody>

  </table>
</div>
  {{ recurring }}{{ content_bottom }}</div>
  {{ column_right }}</div>
</div>
  {{ footer }}

```