Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет _____Комп'ютерних наук_____
(повна назва)

Кафедра _____Інформаційних управляючих систем_____
(повна назва)

**КВАЛІФІКАЦІЙНА РОБОТА**
**Пояснювальна записка**

рівень вищої освіти _____другий (магістерський)_____

_____Дослідження методів розробки сайтів інтернет-магазинів_____
(тема)

Виконав:
студент _2_ курсу, групи _ІУСТм-21-1____
_____Владислав КУКІЛЬ_____
(власне ім'я, прізвище)

Спеціальність ____122 Комп'ютерні_____
_____науки_____
(код і повна назва спеціальності)
Тип програми ____освітньо-професійна_____
(освітньо-професійна або освітньо-наукова)

Освітня програма Інформаційні управляючі системи та технології_____
(повна назва освітньої програми)

Керівник ___доц. каф. ІУС Аліна МІХНОВА
(посада, власне ім'я, прізвище)

Допускається до захисту

Зав. кафедри _____ ___Костянтин ПЕТРОВ__
(підпис) (власне ім'я, прізвище)

2022 р.

Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук

Кафедра Інформаційних управляючих систем

Рівень вищої освіти другий (магістерський)

Спеціальність 122 Комп'ютерні науки
(код і повна назва)

Тип програми освітньо-професійна
(освітньо-професійна або освітньо-наукова)

Освітня програма Інформаційні управляючі системи та технології
(повна назва)

ЗАТВЕРДЖУЮ

Зав. кафедри __ __
(підпис)

«_21_» листопада 20 22 р.

# ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

Студентові Кукіль Владиславу Олеговичу
(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження методів розробки сайтів інтернет-магазинів
затверджена наказом університету від _14_ __листопада__2022_ р. № _1490Ст_
2. Термін подання студентом роботи до екзаменаційної комісії _13_ __грудня_ 2022р.
3. Вихідні дані до роботи Звітні матеріали передатестаційної практики, типові структури сайтів інтернет-магазинів, науково-технічна література, публікації, інформація з інтернет-ресурсів щодо підходів та методики розробки сайтів інтернет-магазинів.
4. Перелік питань, що потрібно опрацювати в роботі Огляд і аналіз особливостей розробки сайтів інтернет-магазинів, аналіз вимог до розробки сайтів інтернет магазинів, аналіз існуючих підходів розробки сайтів інтернет-магазинів, постановка задач дослідження, побудова методу розробки сайтів інтернет-магазинів, обґрунтування використання інструментальних засобів, формування критерію оцінювання методу розробки сайтів інтернет-магазинів, методика використання методу, оцінювання ефективності методу розробки сайтів інтернет-магазині, апробація методу розробки сайтів інтернет-магазині

# КАЛЕНДАРНИЙ ПЛАН

| № | Назва етапів роботи | Терміни виконання етапів роботи | Примітка |
|---|---|---|---|
| 1 | Отримання завдання на кваліфікаційну роботу | 10.10.2022 | Виконано |
| 2 | Аналіз предметної галузі та постановка задачі | 18.10.2022 – 23.10.2022 | Виконано |
| 3 | Огляд і аналіз існуючих технологій, методів розробки сайтів інтернет-магазинів | 24.10.2022 – 26.10.2022 | Виконано |
| 4 | Дослідження технологій і методів розробки сайтів інтернет-магазинів | 27.10.2022 – 5.11.2022 | Виконано |
| 5 | Апробація мадифікованого методу розробки сайтів інтернет-магазинів | 6.11.2022 – 7.11.2022 | Виконано |
| 6 | Підготовка пояснювальної записки та графічного матеріалу | 8.10.2022 – 13.11.2022 | Виконано |

Дата видачі завдання 21 листопада 2022 р.

Студент _____
(підпис)

Керівник роботи _____ доц. каф. ІУС Аліна МІХНОВА
(підпис) (посада, власне ім'я, прізвище)

# РЕФЕРАТ

Пояснювальна записка містить: 79 сторінок, 20 рисунків, 18 джерел. 5 таблиць

АТАКИ НА ВЕБ-САЙТИ, БЕЗПЕКА, ЗАХИСТ, ІНТЕРНЕТ-МАГАЗИН, МЕТОДИ РОЗРОБКИ, МОДЕЛЮВАННЯ, САЙТ, ФРЕЙМВОРКИ.

Предметом дослідження є методи розробки сайтів інтернет-магазинів.

Метою даної роботи є дослідження методів розробки, планування, проектування та тестування сайтів інтернет-магазинів та пропонування покращення одного або кількох з аспектів цих методів.

В роботі досліджено існуючі методи розробки сайтів інтернет-магазинів та запропоновано покращення у вигляді модифікованого методу створення сайтів інтернет-магазинів з урахуванням захисту від загроз. Використання модифікованого методу допомагають покращити захист сайтів інтернет-магазинів від існуючих загроз зламу та викрадення даних клієнтів чи компанії. В роботі проведено апробацію модифікованого методу розробки сайтів інтернет-магазинів з урахуванням захисту від загроз.

# ABSTRACT

The thesis note contains 79 pages, 20 figures, 18 references, 5 tables.

DEVELOPMENT METHODS, FRAMEWORKS, METHODS, ONLINE STORE, PROTECTION, SITE, SECURITY, WEBSITE ATTACKS.

The subject of the study is methods of developing online store sites.

The purpose of this work is to investigate the methods of development, planning, design and testing of online store sites and to propose the improvement of one or more of the aspects of these methods.

The work examines the existing methods of developing online store sites and suggests improvements in the form of a modified method of creating online store sites, taking into account protection against threats. The use of the modified method helps to improve the protection of online store sites against existing threats of hacking and theft of customer or company data. In the work, the modified method of developing websites of online stores, taking into account protection against threats, was tested.

**ЗМІСТ**

# СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

CMS - Content Manage System;

XSS - Cross Site Scripting;

CSRF - Cros Site Request Forgey;

БД - база даних;

ІС - інформаційна система;

SaaS - Software as a Service;

DoS - denial-of-service;

DDoS - distributed denial-of-service;

HTTP - Hypertext Transfer Protocol.

# ВСТУП

На сьогоднішній день веб додатки стали невід’ємною частиною життя мільйонів людей. Завдяки ним не має жодної проблеми знайти будь-яку інформацію в мережі інтернет, купити будь-який товар або послугу. Зараз майже кожен бізнес в світі з продажу товарів має онлайн версію свого магазину або навіть веде справи тільки через мережу інтернет. Тому з кожним днем росте попит на створення власного сайту. Бізнеси різних масштабів наймають IT-компанії для того, щоб вони створили унікальну систему, яка б допомогла розширити бізнес, залучити нових клієнтів та підвищити продажі.

В свою чергу IT-компанія повинна правильно організувати розробку програмного забезпечення, щоб вкластись в поставлені терміни та оптимізувати етапи розробки. Саме з цим допомагають методи розробки проектів. Кожен з них має свої особливості, набір етапів розробки. Тому кожен метод більше підходить до певного типу сайтів.

Для того, щоб правильно обрати метод для розробки веб сайту потрібно врахувати на такі фактори як складність проекту, його масштаб, цілі клієнта, зацікавлені сторони, бізнес-ризики, склад команди інструменти. Також потрібно обрати зручний та простий у використанні інструмент для управління проектами.

# 1 ОГЛЯД І АНАЛІЗ ІСНУЮЧИХ МЕТОДІВ РОЗРОБКИ САЙТІВ ІНТЕРНЕТ-МАГАЗИНІВ

## 1.1 Огляд та аналіз особливостей розробки сайтів інтернет-магазинів

Зростаюче поширення Інтернету можна помітити через велику кількість веб-додатків, розроблених за останні роки. Ця категорія програм використовується для підтримки різноманітних сегментів, таких як торгівля, дослідження та медична діяльність. У більшості з них ми можемо ідентифікувати особливості, які зазвичай безпосередньо впливають на процес розробки програмного забезпечення. Крім того, деякі фактори, такі як високий зв'язок, безперервна еволюція та орієнтація на вміст, безпосередньо впливають на спосіб створення веб-додатків.

Веб-програми можна класифікувати різними способами. Наприклад, існує класифікація сайтів як інформаційних сторінки (з формами даних або без них), додатки, орієнтовані на базу даних (з доступом до даних за запитами та/або використанням інформації профілів) і додатки для підтримки бізнес-транзакцій [3].

Додатки для підтримки бізнес-транзакцій також називаються інтернет-магазинами. Їх основна задача - підвищити прибуток компанії завдяки просуванню та продажу товарів через мережу інтернет. Щоб це зробити зазвичай створюється бізнес-модель сайту на основі основних цілей та вимог компанії [3].

Існує 3 основних бізнес-моделі інтернет-магазинів:

- рекламна;
- підтримка існуючого бізнесу;
- створення нового бізнесу.

Рекламна бізнес-модель сайту створюється в тому випадку, коли до нього потрібно залучити постійну аудиторію відвідувачів. Остання за потребою може бути максимально широкою або чітко сегментованою. Контакт з цією аудиторією продається рекламодавцям або спонсорам [3].

Для цього слід вирішити такі завдання:

– початкове залучення відвідувачів на сайт;

– стимулювання наступних відвідувань;

– збільшення часу проведеного відвідувачами на сайті;

– залучення користувачів до життя сайту (дискусії, конкурси, опитування);

– залучення користувачів до розвитку та просування ресурсу [3].

Бізнес-модель підтримки існуючого бізнесу застосовується у випадку, коли у компанії вже є реальний бізнес і за допомогою рекламних та комерційних заходів в інтернеті бажає збільшити кількість клієнтів та продажів [3].

У випадку бізнес-моделі підтримки бізнесу потрібно вирішити наступні завдання:

– просування та реклама компанії та її товарів або послуг;

– забезпечення клієнтів найповнішою, найактуальнішою та об’єктивною інформацією про компанію та її товари або послуги;

– підвищення прибутку за допомогою створення майданчика для продажу товарів чи послуг через мережу інтернет [3].

Бізнес моделі для створення нового бізнесу засновані на великих можливостях мережі інтернет. Ними можуть бути інтернет-магазини, торговельні майданчики, інтернет-аукціони, сайти за надання різних послуг, електронні біржі тощо [3].

В залежності від особливостей та спеціалізації бізнес-діяльності кожна модель може бути розбита на певну кількість підмоделей або один сайт може мати одночасно риси різних бізнес-моделей [3].

Існує кілька систем класифікації інтернет-магазинів:

– за методом роздрібного продажу товарів у мережі: інтернет-магазини, веб-вітрини, торгові системи, торгові ряди;

– за бізнес-моделлю: онлайновий магазин та суміщення оффлайн бізнесу з онлайн (інтернет-магазин створюються на основі вже діючої торгової структури);

– за взаємовідносинами з постачальниками: магазини, які володіють власним складом, магазини що працюють за договорами з постачальниками;

– за ступенем автоматизації: серед торгових систем електронних магазинів розрізняють веб-вітрини, інтернет-магазини та торгові інтернет-системи [3].

Веб-вітрина — сукупність товарного каталогу, системи навігації та оформлення замовлення з наступною передачею його менеджеру для оформлення. Той у свою чергу зв’язується зі складом, організовує доставку товару покупцеві, контролює процес оплати за товар [3].

Інтернет-магазин відрізняється від веб-вітрини повною автоматизацією системи обробки замовлень, завдяки чому можна працювати індивідуально з кожним зареєстрованим клієнтом [3].

Торгові інтернет системи мають спільну рису з інтернет-магазинами - це можливість здійснювати повний торговий цикл у режимі підключення до мережі. При цьому вона додатково інтегрована в систему внутрішнього документообігу компанії. Неавтоматизованими для інтернет-магазину та торгової інтернет систем залишаються системи доставки товару [3]. Всі ці типи інтернет магазинів є елементами інформаційної системи торгової компанії, як показано на рисунку 1.1 в типовій схемі інформаційної системи торгової компанії.

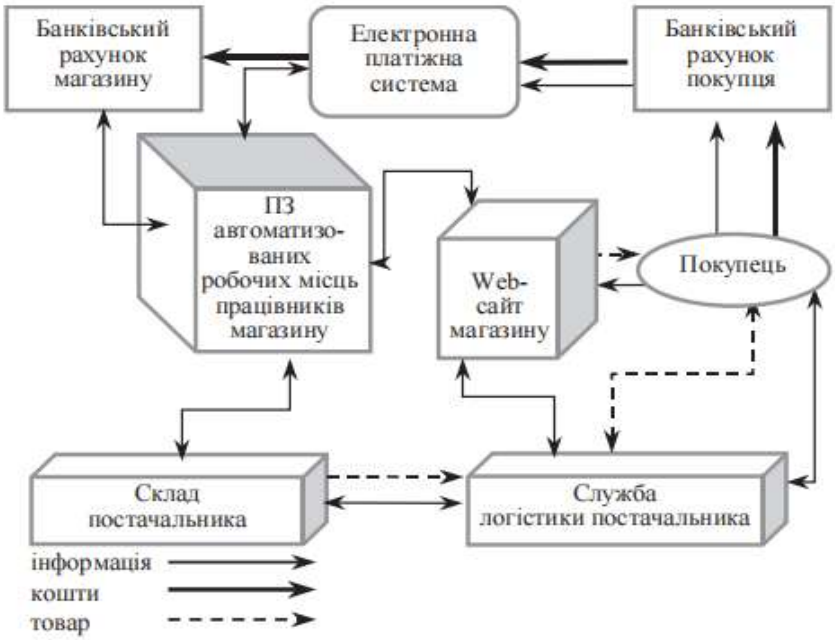


Рисунок 1.1 - Типова схема інформаційної системи торгової компанії

В будь-якій інформаційній системі однією із найважливіших вимог є захист конфіденційних даних клієнтів та компанії, через те що в наш час існує багато способів зламу. Компанії намагаються захистити свою інформаційну систему від спроб зловмисників отримати доступ до неї та інформації компанії. Тому створюються функції захисту для захисту всіх елементів інформаційної системи, в тому числі і веб-сайту компанії.

## 1.2 Аналіз вимог до розробки сайтів інтернет-магазинів

Незважаючи на загальноприйняту думку, основна частина розробки та дизайну веб-сайту не є обов’язковою для процесу кодування. Дійсно, такі технології, як HTML, CSS і JavaScript, надають відомій нам формі Інтернету та визначають спосіб взаємодії з інформацією. Але те, що зазвичай залишається за кадром і, водночас, залишається вирішальною частиною життєвого циклу розробки веб-сайту, — це етапи збору попередньої інформації, детального планування та обслуговування після запуску [4].

Загальна кількість етапів розробки зазвичай коливається від п'яти до восьми, але кожного разу вся картина залишається майже однаковою [4].

Основні етапи розробки сайтів інтернет-магазинів:

- збір інформації;
- планування;
- проектування;
- написання та складання вмісту;
- написання коду програми;
- тестування функціоналу;
- підтримка та обслуговування проекту [4].

Перший етап це збір інформації. Цей етап, етап відкриття та дослідження, визначає, як виглядатимуть наступні кроки. Найважливішим завданням на цьому етапі є чітке уявлення про цілі вашого майбутнього сайту, основні цілі, які ви хочете досягти, і цільову аудиторію, яку ви хочете залучити на свій сайт. Така анкета розробки сайту допомагає розробити найкращу стратегію подальшого управління проектом [4].

Новинний портал відрізняється від розважальних сайтів, а онлайн-ресурси для підлітків виглядають інакше, ніж сайти для дорослих. Різні типи веб-сайтів надають відвідувачам різні функціональні можливості, а це означає, що різні технології повинні використовуватися відповідно до цілей. Добре описаний і детальний план, заснований на цих даних перед розробкою, може захистити вас від витрачання додаткових ресурсів на вирішення неочікуваних проблем, таких як зміна дизайну або додавання функціональності, яка спочатку не була запланована [4].

На цьому етапі циклу розробки веб-сайту розробник створює дані, які дозволяють клієнту судити про те, як виглядатиме весь сайт. Мапа сайту інтернет-магазину повинна описувати зв’язки між основними областями вашого веб-сайту. На основі інформації, зібраної на попередньому етапі, створюється мапа сайту інтернет-магазину [4]. Приклад мапи сайту інтернет-магазину зображений на рисунку 1.2.

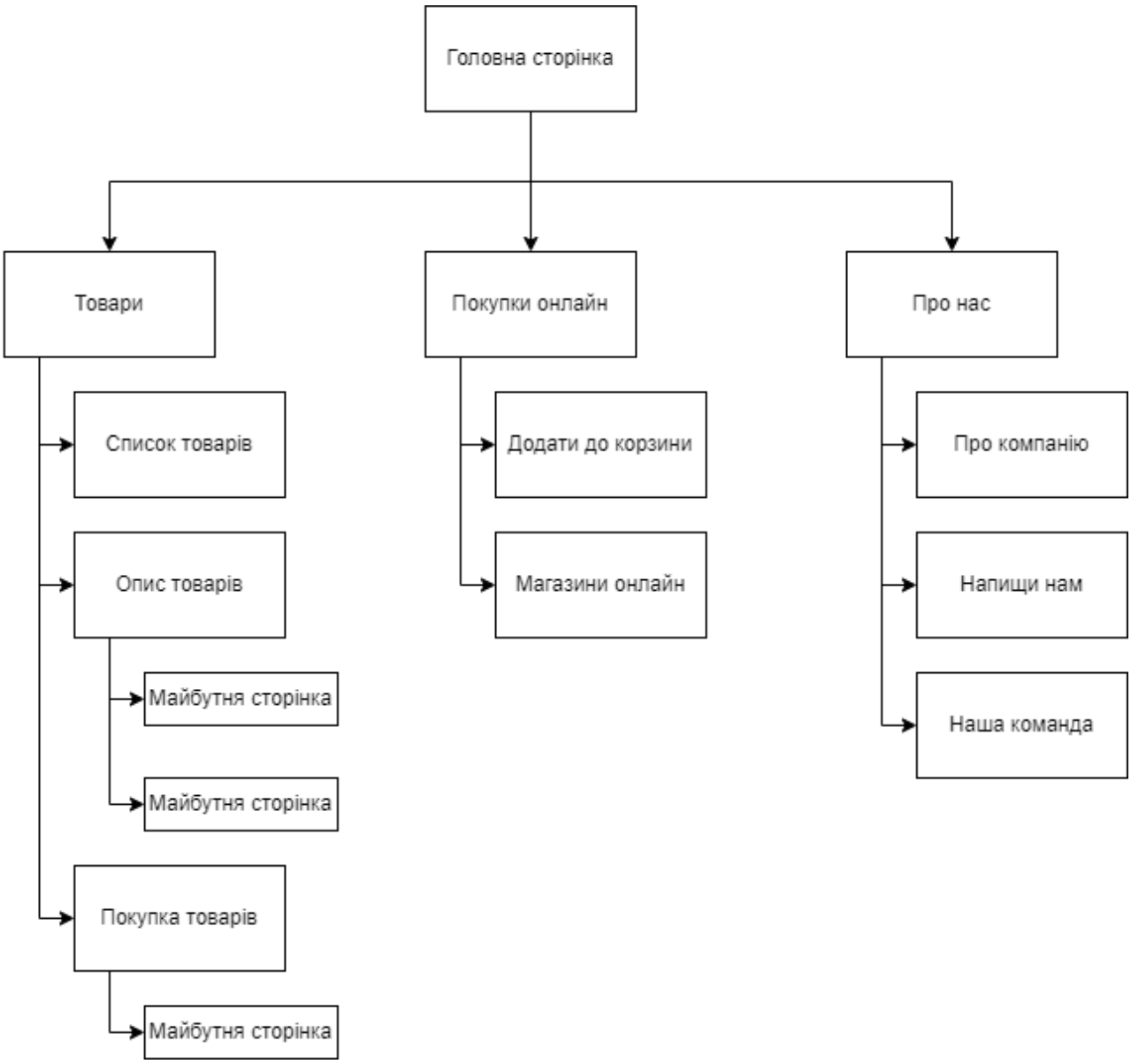


Рисунок 1.2 - Мапа сайту інтернет-магазину

На етапі проектування створюється весь візуальний вміст, наприклад зображення, фотографії та відео. Знову ж таки, уся інформація, яку було зібрано на першому етапі, має вирішальне значення. Під час роботи над дизайном необхідно пам’ятати про замовника та цільову аудиторію [4].

Макет сайту – це результат роботи дизайнера. Це може бути графічний ескіз або справжній графічний дизайн. Основною функцією макета є представлення інформаційної структури, візуалізація вмісту та демонстрація основної функціональності. Макети містять кольори, логотипи, зображення і можуть дати загальне уявлення про майбутній продукт [4].

Написання та компіляція контенту зазвичай збігається з іншими етапами створення веб-сайту, і їх роль не можна недооцінювати. На цьому кроці необхідно письмово викласти саму суть, яку ви хочете донести до аудиторії вашого веб-сайту, і додати заклики до дії. Написання контенту також передбачає створення заголовків, які вловлюють, редагування тексту, написання нового тексту, компіляцію існуючого тексту тощо, що потребує часу та зусиль. Як правило, клієнт зобов'язується надати контент сайту, готовий для міграції на сайт. Краще, коли весь вміст веб-сайту надається до або під час кодування веб-сайту [4].

Після виконання попередніх кроків можна переходити до написання коду. На цьому кроці ви нарешті можете почати створювати сам веб-сайт. Графічні елементи, які були розроблені на попередніх етапах, повинні бути використані для створення реального сайту. Зазвичай спочатку створюється домашня сторінка, а потім додаються всі підсторінки відповідно до ієрархії веб-сайту, яка була раніше створена у формі карти сайту. Необхідно впровадити фреймворки та системи управління контентом (CMS, Content Management System), щоб гарантувати, що сервер зможе безперебійно виконувати інсталяцію та налаштування. На цьому етапі також часто починають впроваджувати захист веб-сайту від загроз зламу. Розробники намагаються захистити свій веб-додаток від найвідоміших та найчастіших загроз. Якщо використовується CMS, то часто створення захисту обходиться лише встановленням декількох плагінів. Це дуже дешевий та швидкий підхід, але з цим не дуже надійний, так як за час розвитку веб-розробки розвивались і способи отримання доступу до сайту зловмисниками [4].

Тестування, мабуть, найбільш рутинна частина процесу. Слід перевірити кожне посилання, щоб переконатися, що серед них немає зламаних. Ви повинні перевірити кожну форму, кожен сценарій, запустити програму перевірки орфографії, щоб знайти можливі помилки. Використовуйте засоби перевірки коду, щоб перевірити, чи відповідає ваш код поточним веб-стандартам. Дійсний код необхідний, наприклад, якщо кросбраузерність має вирішальне значення для

вас. Також, на цьому етапі сайт перевіряється на захист від зламу. Існують різні способи перевірки і тестувальники самі обирають потрібний спосіб перевірки для різних функцій [4]. На рисунку 1.3 зображено схема етапів розробки сайтів інтернет-магазинів, як елемента інформаційної системи торговельної компанії.

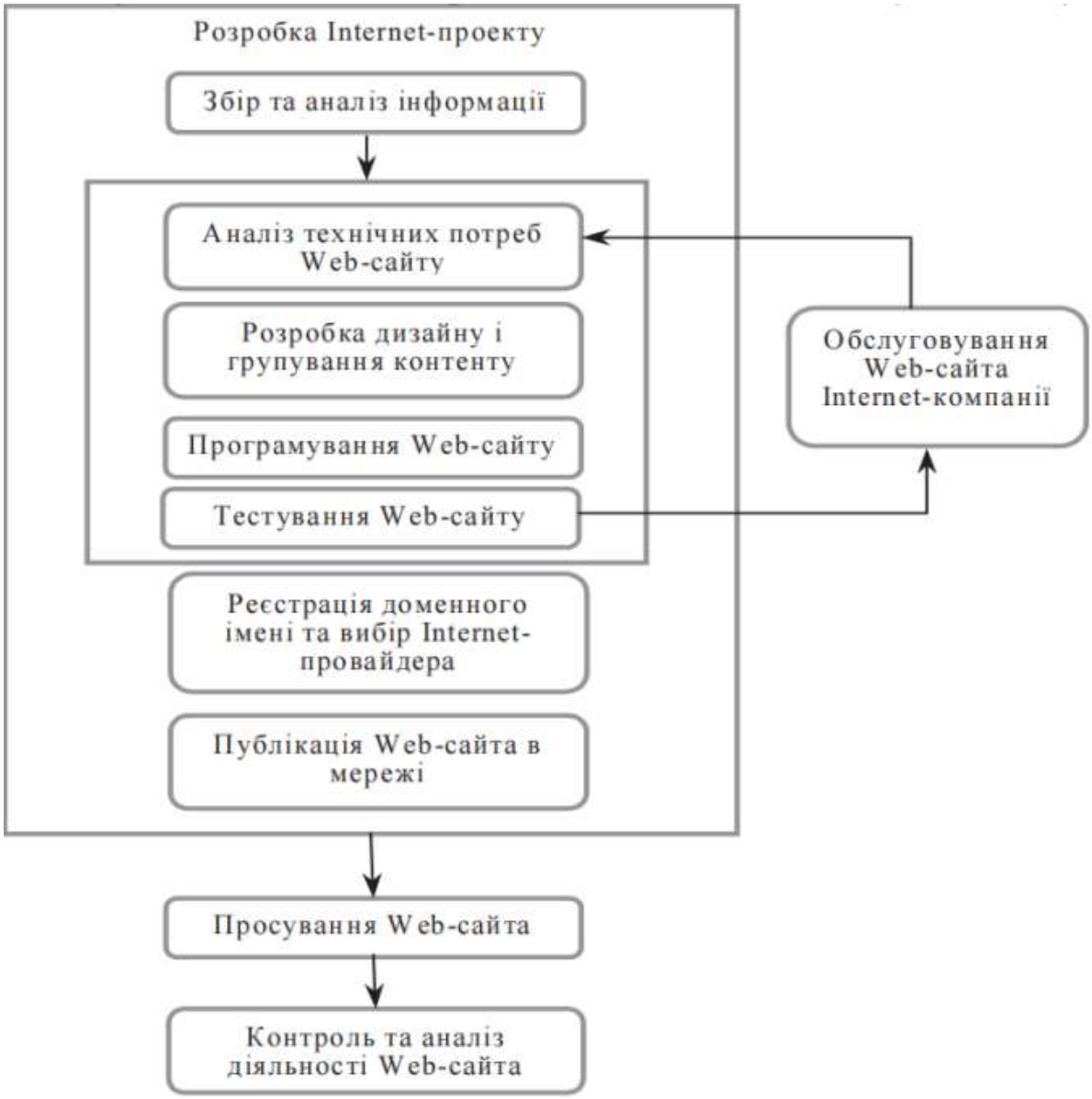


Рисунок 1.3 - Схема етапів розробки сайтів інтернет-магазинів, як елемента інформаційної системи торговельної компанії

Ще одна важлива річ — підтримувати веб-сайт в актуальному стані. Якщо використовується систему управління контентом, регулярні оновлення запобігатимуть помилкам і зменшать ризики безпеки [4].

## 1.3 Аналіз існуючих підходів розробки сайтів інтернет-магазинів

Існує багато підходів розробки сайтів. Правильний вибір підходу розробки сайту залежить від вимог до самого сайту. В інформаційних системах торгових компаній часто вже є інші готові елементи, такі як платіжні системи чи системи постачання. Від цього також може залежати вибір підходу до розробки сайту, бо різні підходи використовують різні програмні засоби, які можуть мати різні рівні підтримки вже існуючих елементів інформаційної системи. Можна виділити 3 основних програмних підходу розробки сайтів [9]:

- за допомогою CMS;
- з використанням популярних фреймворків;
- з використанням SaaS-платформ у Cloud.

CMS або система керування контентом - це програмний пакет, який забезпечує певний рівень автоматизації завдань, необхідних для ефективного керування вмістом сайту [8].

CMS зазвичай є багатокористувацьким програмним забезпеченням на основі сервера, яке взаємодіє з вмістом, що зберігається в репозиторії. Цей репозиторій може бути розташований на тому самому сервері, як частина того самого пакету програмного забезпечення або повністю в окремому сховищі [8].

CMS дозволяє редакторам створювати новий вміст, редагувати наявний вміст, виконувати редакційні процеси над вмістом і, зрештою, робити цей вміст доступним для використання іншими людьми [8].

Логічно, CMS складається з багатьох частин. Інтерфейс редагування, репозиторій, механізми публікації тощо можуть бути окремими, автономними частинами системи за лаштунками. Однак для нетехнічного редактора всі ці частини зазвичай розглядаються як єдине, монолітне ціле [8].

Усі CMS написані на мові програмування PHP, тому вони можуть добре підійти, якщо в інформаційній системі торгової компанії вже є елементи, які

написані на цій мові програмування. Так вийде зекономити час та витрати на впровадження веб-сайту в інформаційну систему торгової компанії.

Найпопулярніші CMS в наш час це WordPress, Drupal, Joomla та Magento.

На рисунку наведена узагальнена структурна схема CMS.

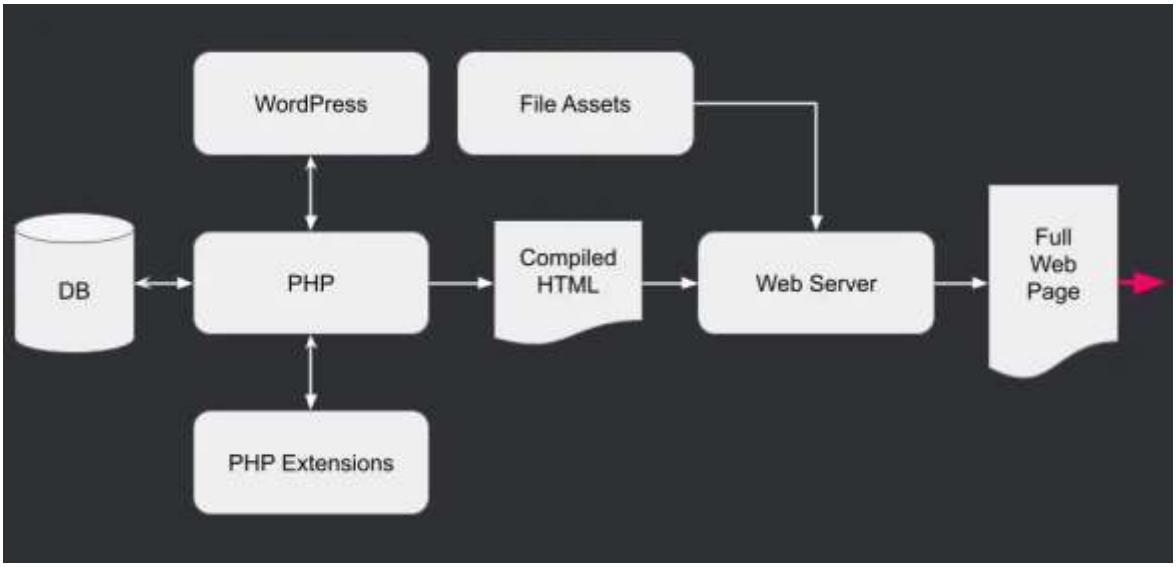


Рисунок 1.4 - Узагальнена структурна схема CMS

Ще один популярний спосіб розробки сайтів інтернет-магазинів це використання фреймворків.

Фреймворк веб-програми — це тип фреймворка або основи, спеціально розроблений, щоб допомогти розробникам створювати веб-програми. Ці структури зазвичай забезпечують основні функції, загальні для більшості веб-додатків, такі як керування сеансами користувачів, збереження даних і системи шаблонів. Використовуючи відповідну структуру, розробник часто може заощадити значну кількість часу на створенні веб-сайту [7].

Популярні фреймворки для PHP - Laravel, Symfony, Yii. Всі вони використовують MVC шаблон та використовуються для розробки невеликих та середніх веб-проектів. Гарно підходять для розробки інтернет-магазинів та торгових інтернет систем завдяки своєму багатому функціоналу [7].

Існує дуже багато веб-фреймворків, для різних мов програмування. Найпопулярніші мови програмування для веб розробки це PHP, Python, Java. Тому при виборі одного з них треба звертати увагу не тільки на базовий

функціонал, підтримку баз даних та націленість фреймворку, а і на мову програмування, яка вже була використана в інших елементах інформаційної системи, для якої розробляється веб-сайт. Також необхідно приділити увагу функціям захисту, які надає той чи інший фреймворк, так як всі вони надають різний рівень захисту. На рисунку 1.5 наведено структурну схему фреймворку Laravel.

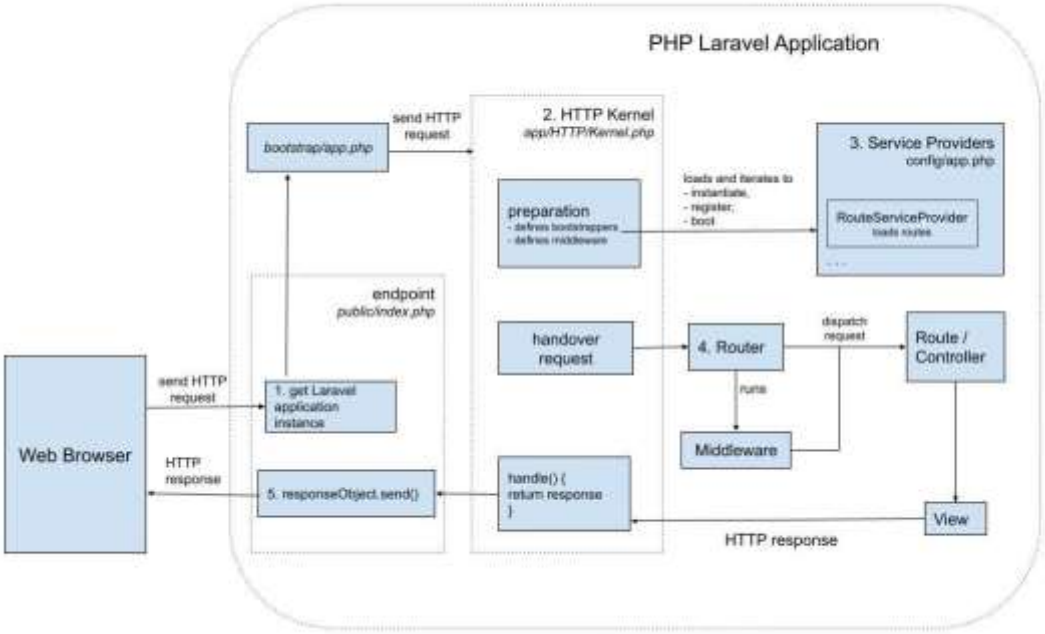


Рисунок 1.5 - Структурна схема фреймворку Laravel

Для Python існують 2 популярних веб фреймворка - це Django та Flask.

Django — MVC фреймворк, як і Laravel та Symfony, є їхньою альтернативою. Його цінують за дуже велику кількість функціоналу, тому використовуючи його для створення веб-додатків можна сильно зекономити час та зменшити бюджет. Також підходить для написання середніх проектів, таких як інтернет-магазинів та торгових інтернет систем. На рисунку 1.5 наведено структурну схему фреймворку Django [18].

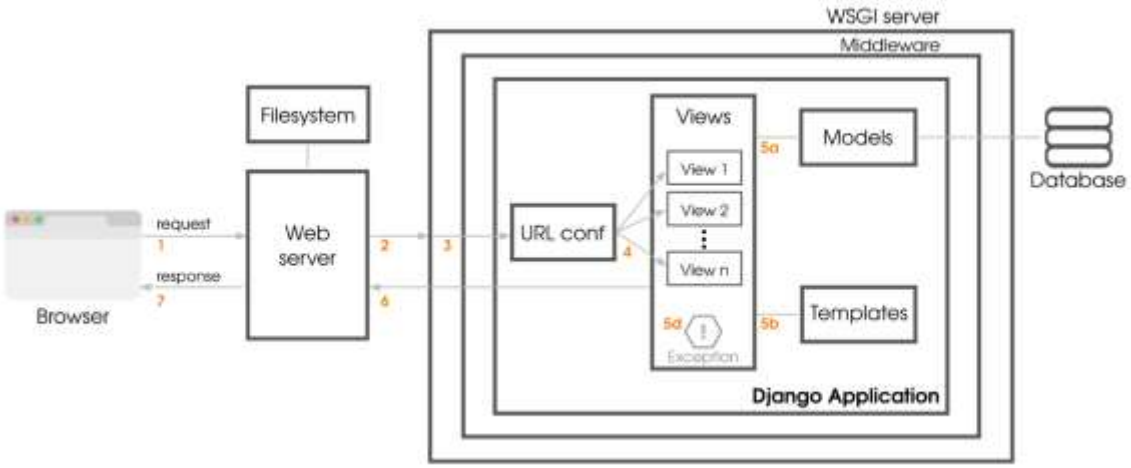


Рисунок 1.6 - Структурна схема фреймворку Django

Flask — це невеликий фреймворк, достатньо малий, щоб його можна було назвати «мікрофреймворком». Він досить малий, щоб, як тільки ви з ним ознайомилися, ви, ймовірно, змогли прочитати та зрозуміти весь його вихідний код. Але те, що він малий, не означає, що він робить менше, ніж інші фреймворки. Flask був розроблений як розширювана структура з нуля; він забезпечує надійне ядро з основними послугами, тоді як розширення забезпечують решту. Оскільки ви можете вибрати пакети розширень, які вам потрібні, ви отримаєте мініатюрний стек, який не роздутий і виконує саме те, що вам потрібно. На рисунку 1.5 наведено структурну схему фреймворку Flask [7].

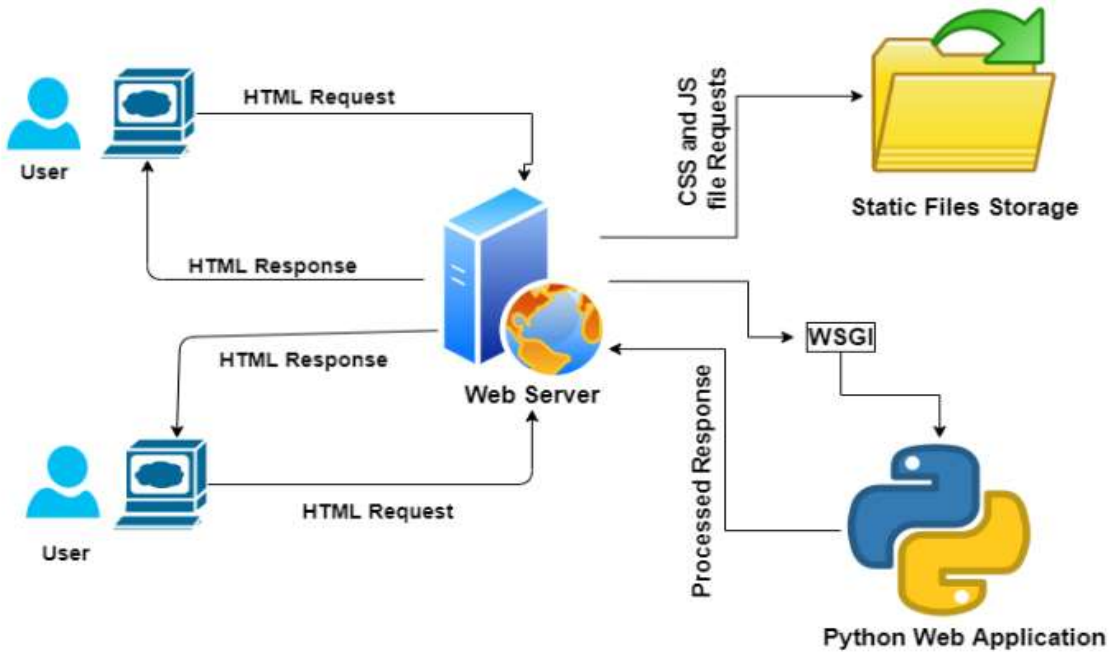


Рисунок 1.7 - Структурна схема фреймворку Flask

Для мови програмування Java найпопулярнішим фреймворком є Spring.

Spring Framework — це фреймворк програми з відкритим кодом і інверсія контейнера керування для платформи Java. Основні функції фреймворку можуть використовуватися будь-якою програмою Java, але існують розширення для створення веб-програм на основі платформи Java EE. Хоча фреймворк не нав’язує жодної конкретної моделі програмування, він став популярним у спільноті Java як альтернатива, заміна або навіть доповнення до моделі Enterprise JavaBeans (EJB) [10].

SaaS-платформи для створення сайтів - це можливість запустити досить простий веб-проект дуже швидко і досить дешево (ще і на умовах оренди і без необхідності хостінгу). Щось подібне до Гугл сайтів, але зі значно більшими можливостями. Рішення підходить для простих сайтів, тимчасових проектів і для перевірки бізнес-ідей. SaaS-платформи, як і CMS, бувають специфічними (наприклад, тільки для інтернет-магазинів) і універсальними (для всіх типових видів сайтів). Цей метод зараз активно розвивається та використовується [9].

Мінуси SaaS-платформ для створення сайту:

Фактично це шаблонний дизайн - оформлення сайтів на SaaS-платформах проводиться за готовими шаблонами (часто не дуже високої якості), які можна тільки «розфарбувати» і на деяких платформах можна окремі блоки місцями поміняти. Для сайтів, до яких пред'являються вимоги до оформлення, такі рішення не підходять [9].

Рамки функціональних можливостей - якщо платформа «не має якихось можливостей», то це ніяк не виправити. Програмний продукт типовий і його налаштування під індивідуальні побажання обмежена. Якщо проект відразу або в перспективі повинен вирішувати специфічні завдання і гнучко налаштовуватися, то SaaS-платформа для його розробки не підходить [9].

Низький рівень захисту від загроз злому. В основному всі SaaS-платформи намагаються зробити розробку сайтів простішою, дешевшою та швидшою. Тому за допомогою цих платформ можна реалізувати лише базовий функціонал, який

може підійти для невеличкого сайту. Також SaaS-платформи не пропонують досконалих функцій захисту [9].

Приклади SaaS-платформ:

UMI, WIX, InSales, Shopify, Setup, uCoz - деякі з цих платформ специфічні (тільки для простих сайтів або тільки для інтернет-магазинів), а деякі - досить універсальні. Якщо ваш проект не має ніяких істотних вимог до дизайну і до функціональності, то має сенс звернути увагу на ці SaaS-рішення. В іншому випадку, вибирайте іншу платформу для створення сайтів [9]. На рисунку 1.6 наведено структурну схему SaaS.


Рисунок 1.8 - Структурна схема SaaS-платформи

## 1.4 Постановка задач дослідження

Об’єктом дослідження є інформаційні системи сайтів інтернет-магазинів

Предметом дослідження є методи розробки сайтів інтернет-магазинів з урах.

Метою даної роботи є дослідження методів розробки, планування, проектування та тестування сайтів інтернет-магазинів та пропонування покращення одного або кількох з аспектів цих методів.

Для досягнення цієї мети необхідно:

– провести аналіз вимог до розробки сайтів інтернет-магазинів задля оцінювання можливості підвищення рівня захисту існуючих проектних рішень;

– провести дослідження існуючих підходів розробки сайтів інтернет-магазинів;

– побудувати метод розробки сайтів інтернет-магазинів з урахуванням підвищення рівня захисту на основі досліджень із попередніх пунктів;

– обґрунтувати вибір відповідних інструментальних засобів для реалізації модифікованого методу з урахуванням підвищення рівня захисту;

– сформувати методику використання модифікованого методу з урахуванням підвищення рівня захисту;

– провести апробацію модифікованого методу розробки сайтів інтернет-магазинів з урахуванням підвищення рівня захисту.

# 2 ДОСЛІДЖЕННЯ ТА ПОБУДОВА МЕТОДУ РОЗРОБКИ САЙТІВ ІНТЕРНЕТ-МАГАЗИНІВ

## 2.1 Дослідження існуючих методів розробки сайтів інтернет-магазинів

За весь час розвитку веб-додатків було створено чимало методів їх розробки. Всі вони виконують основну функцію, а саме забезпечення керування життєвим циклом програмного забезпечення, включаючи розробку та підтримку. Вони повинні поєднувати традиційні методи та принципи розробки програмного забезпечення з конкретними аспектами Інтернету [6].

Загалом, поточні методи розширюють класичні шляхи додавання деякої навігаційної моделі для визначення навігаційних характеристик веб-додатку. Подібна процедура також використовується для розширення деяких методів розробки або мов моделювання до контексту веб-додатків. Ці моделі визначають навігаційні види системи та пов’язуються з певними групами користувачів. Як правило, навігаційні описи представлені графіками, які визначають представлення даних програми та визначену функціональність у структурних моделях. Вузли цих графіків представляють системи, і вони можуть бути пов’язані навігаційними посиланнями [6].

В останні роки кілька намагалися розширити деякі відомі методи, такі як ObjectOriented(OO), EntityRelationship(E-R) і Data Flow Diagram(DFD), для контексту веб-додатків. Крім того, деякі характеристики, такі як миттєвість або короткий час, необхідний для його розробки, можуть запропонувати використання інших методів, наприклад гнучких, оскільки серед їхніх принципів є швидка доставка запущених версій продукту [6].

У таблиці 2.1 представлені наступні атрибути існуючих методів розробки сайтів інтернет-магазинів.

Таблиця 2.1 - Атрибути існуючих методів розробки сайтів інтернет-магазинів

| Назва методу | Спосіб моделювання | Нотація | Інструменти |
|---|---|---|---|
| OOHD | OO | UML та власна | OOHDM-Web |
| UWE | OO | UML та власна | ArgoUWE |
| WebML | OO | E-R, UML, власна | WebRatio |
| OO-H | OO | UML та власна | CASE Tool |
| WAE | OO | UML та власна | |
| OOWS | OO | UML | |
| HDM | E-R | E-R | |
| RMM | E-R | E-R та власна | |
| SWM | DFD | DFD | ASCENT |
| W2000 | E-R та OO | HDM, UML, власна | |

Важливо зауважити, що така оцінка може опускати деякі важливі аспекти. Деталі, які метод пропонує для конкретної діяльності, і інформація, зібрана з нього, можуть бути дуже різними. Деякі методи розглядають лише набір вказівок у текстовому вигляді, тоді як інші надають інструменти для підтримки етапів розробки. Як приклад можна розглянути метод OOHDM, який складається з графічних зображень для етапів архітектури, навігації та проектування інтерфейсу. OOHDM пропонує чітко визначену процедуру розробки веб-додатків [6].

Hypertext Design Model (HDM) — це модель структурованого дизайну гіпертекстових програм. Тому він описує модель проектування, а не модель процесу. На відміну від HDM, OOHDM пропонує чітко визначену процедуру розробки гіпермедіа-додатків [2]. На рисунку 2.1 надано етапи розробки сайту інтернет-магазину з використанням OOHDM методу

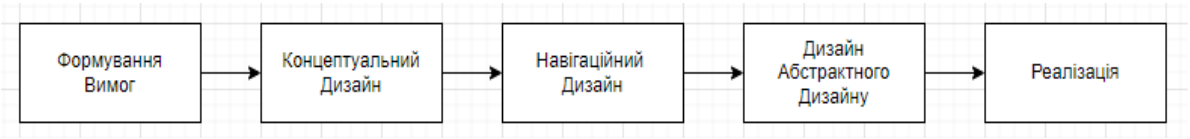


Рисунок 2.1 - Етапи розробки сайту інтернет-магазину з використанням OOHDM методу

Методологія OOHDM включає чотири різні види діяльності. як показано на рисунку 1.

Перший етап це збір вимог. Він передбачає збір вимог зацікавлених сторін за допомогою варіантів використання, які представлені за допомогою діаграми взаємодії користувача [2].

Другий етап - концептуальний дизайн. Концептуальна модель відповідає традиційній об’єктно-орієнтованій моделі та описує сутності дизайну за допомогою нотацій UML [2].

Третій етап це навігаційний дизайн. На цьому етапі визначається навігаційна модель, яку можна розглядати як вигляд концептуальної моделі. Навігаційна модель складається зі схеми навігаційного класу та схеми навігаційного контексту [2].

Четвертий етап - абстрактний дизайн інтерфейсу. Цей крок стосується об’єктів інтерфейсу користувача. Для опису інтерфейсу користувача використовується підхід до проектування абстрактного представлення даних [2].

Реалізація стосується відображення розробленої програми до реалізації програми [2].

Інший популярний метод це WebML — візуальна нотація для розробки складних веб-додатків із інтенсивним використанням даних. Вона надає графічні, але формальні специфікації, втілені в повний процес проектування, якому можуть допомогти інструменти візуального дизайну, такі як WebRatio [2]. Метод WebML має 5 моделей.

Структурна модель: це типова концептуальна модель даних, яка базується на моделі ER (ERM), UML і ODMG. Але автори WebML віддають перевагу моделі UML [17].

Концептуальна модель: вона описує, як структуру можна розширити за допомогою похідних даних, щоб додати надлишкову інформацію. Іншими словами, це схоже на VIEWS у моделюванні бази даних. Як VIEW в Oracle або MySQL. Для кожної сторінки є одна абстрактна таблиця даних. Але вона об'єднаний з інших таблиць. Ця модель використовує WebML-OQL (WebML Object Query Language) [17].

Модель композиції: описує розподіл вмісту на сторінках програми.

Навігаційна модель: показує навігацію між сторінками за допомогою посилань (контекстних, неконтекстних). Моделює, як користувач переміщується в Інтернеті [17].

Презентаційна модель: презентаційне моделювання пов’язане з фактичним виглядом і відчуттям сторінок, визначених моделюванням композиції. Сторінки WebML відображаються відповідно до таблиці стилів [17].

WebML надає набір методів і допоміжних інструментів для систематичного проектування та розробки веб-додатків, вирішує різні проблеми, використовуючи окремі моделі (вміст, навігація, презентація, бізнес-логіка тощо), а також надає компілятори моделей, які створюють більшість логіки та веб-програм. сторінок програми з цих моделей. Однак ці пропозиції також мають деякі обмеження, особливо щодо обміну моделями або представлення подальших проблем моделювання, таких як архітектурні стилі, технологічна незалежність. На рисунку 2.2 надано приклад моделі WebML.

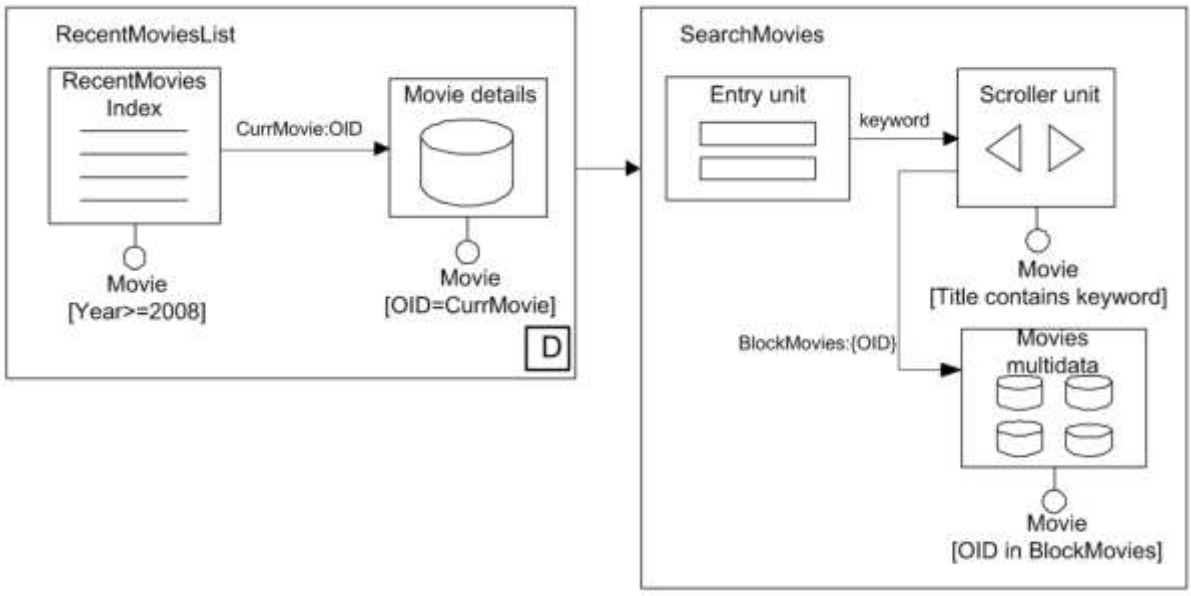


Рисунок 2.2 - Приклад презентаційної моделі WebML

W2000 — це повна нотація для моделювання складних веб-додатків. Вона запозичує концепції з різних областей та об’єднує їх у однорідне рішення. W2000 походить від моделі гіпертекстового дизайну (Hypertext Design Model, HDM), тобто від гіпермедіа та орієнтованих на дані веб-додатків, але також запозичує UML (Unified Modeling Language) для підтримки концепції бізнес-процесів. W2000 дозволяє дизайнеру моделювати всі аспекти веб-додатків, від веб-сторінок до бізнес-транзакцій, узгодженим та інтегрованим способом. Він також використовує підхід, керований моделлю, щоб дозволити дизайнерам поступово вдосконалювати свої моделі та плавно переходити від специфікації до дизайну [5].

Повна модель W2000 організована за чотирма моделями: Інформаційна - визначає дані, які використовуються програмою та сприймаються користувачем, навігаційна - необхідно визначити, як організовано вміст для реалізації, службова - визначає елемент керування та презентаційна - як дані та послуги представлені користувачеві [5]. На рисунку 2.3 приведено ієрархічну структуру методу W2000.

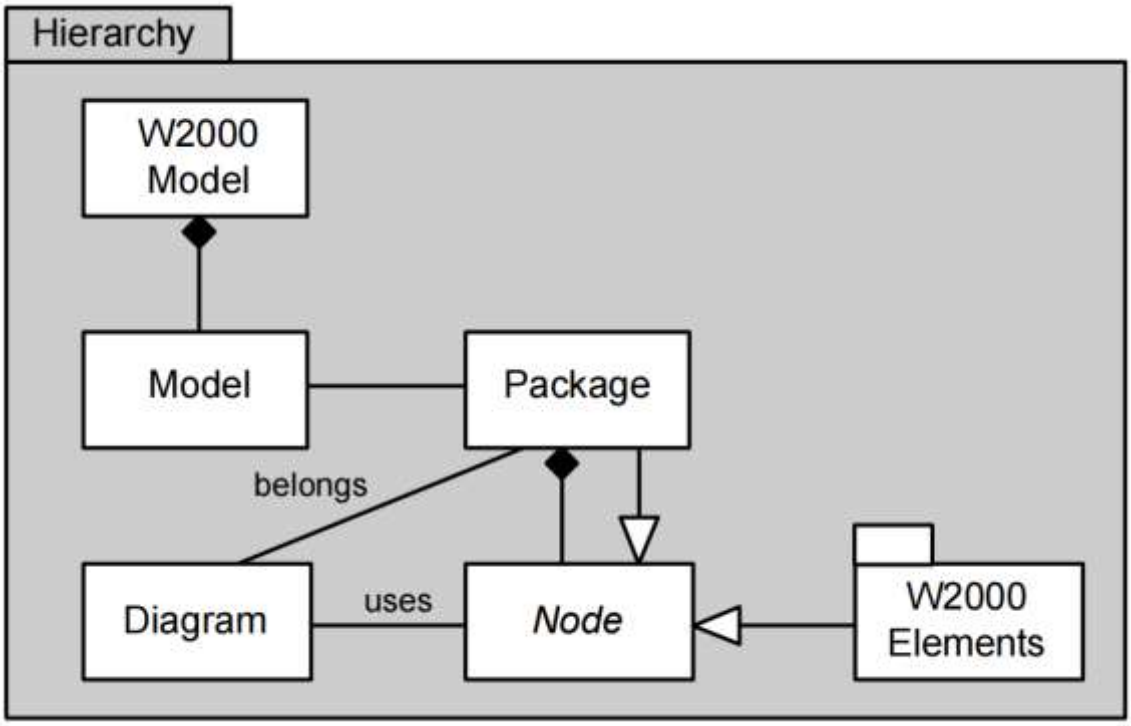


Рисунок 2.3 - Ієрархічна структура методу W2000

Таблиця 2.2 також показує, що більшість методів веб-розробки надають незначну підтримку багатьом етапам розробки, зокрема підтримку тестової діяльності. Через це при розробці можуть бути допущено велика кількість помилок, наприклад в безпеці додатку чи в коректності роботи функціоналу.

Таблиця 2.2 - Підтримка етапів розробки сайту інтернет-магазину відповідними методами

| Назва методу | Формування вимог | Планування | Аналіз | Проектування архітектури | Проектування навігації | Проектування інтерфейсу | Розробка | Тестування | Підтримка |
|---|---|---|---|---|---|---|---|---|---|
| HDM | | | | + | + | | | | |
| RMM | | | | + | + | + | + | | |
| OOHDM | | | | + | + | + | + | | |
| UWE | | | + | + | + | + | | | |
| WebML | | | | | + | + | + | | |
| OO-H | | | + | + | + | + | + | | |
| W2000 | | | + | + | + | + | | | |
| WAE | | + | + | + | + | + | + | + | |
| SWM | + | + | + | + | + | + | + | + | |
| OOWS | | | + | + | + | + | + | | |

Проаналізувавши 3 метода розробки сайтів можна побачити, що всі вони мають певні плюси та мінуси, а також підходять для різних цілей. Головна проблема розглянутих методів це поганий підхід до функцій безпеки додатків. Через приділення недостатньої уваги проектуванню функцій захисту веб-сайту на первинних етапах розробки сайту, можуть виникнути серйозні слабкості в системі захисту. Тому є великий ризик отримання доступу зловмисниками та викрадання даних клієнтів чи компанії.

Існують багато різних види загроз, але не всі вони використовуються хакерами для зламу веб-сайтів. Різні загрози використовуються для зламу різних типів додатків через їхню різну архітектуру та використання різних інструментальних засобів розробки. Тому для захисту веб-сайтів інтернет магазинів потрібно визначити основні загрози зламу та прийняти міри захисту від цих загроз. Якщо інформаційна система, для якої розроблюється веб-сайт має унікальні елементи, дані чи функції, то можливо з’явиться необхідність розроблювати кастомні системи захисту чи використовувати захисні функції, які рідко застосовуються для захисту веб-сайтів.

## 2.2 Дослідження можливих загроз при роботі сайтів інтернет-магазинів

Через проблему недостатньої кількості етапів розробки, коли основна мета це зробити продукт якомога швидше, а не якомога якісніше, дуже часто початкові етапи розробки відходять на другий план, або взагалі ігноруються. Через це впровадження заходів безпеки додатку відбувається набагато пізніше ніж повинно, наприклад на стадії тестування чи розгортання сайту.

За даними IBM, що вартість виправлення помилки, виявленої після випуску продукту, була в 4-5 разів більшою, ніж помилка, виявлена під час проектування, і до 100 разів більша, ніж помилка, виявлена на етапі обслуговування. Дослідження Stake показали, що в середньому організація

виявляла лише чверть прогалин у безпеці свого програмного забезпечення та зазвичай мала сім значних помилок у корпоративному програмному забезпеченні. Їхні висновки підтвердили, що виправлення тих самих дефектів на етапі тестування коштувало приблизно в сім разів менше, ніж після розгортання. Вони прийшли до висновку, що впровадження безпеки в розробку програмного забезпечення на етапі проектування дасть 21% ROSI (повернення інвестицій у безпеку IT); очікування на етапі впровадження зменшить це до 15%, а на етапі тестування ROSI впаде до 12%. Інтеграція безпеки на ранній стадії життєвого циклу розробки програм створює більш безпечні, надійні програми за нижчою ціною [2].

Це призводить до великої кількості слабких місць, через якими зловмисники можуть скористатися для отримання даних клієнтів, такі як паролі, їх адреси, телефонні номери, документи, тощо.

Найпопулярніші види атак, які застосовуються для зламу веб-сайтів:

CSRF – це тип зловмисного використання веб-сайту або веб-програми, коли несанкціоновані команди подаються від користувача, якому веб-програма довіряє. Є багато способів, за допомогою яких шкідливий веб-сайт може передавати такі команди; Наприклад, спеціально створені теги зображень, приховані форми та вибірка JavaScript або XMLHttpRequests можуть працювати без взаємодії або навіть відома користувача. На відміну від міжсайтового сценарію (XSS), який використовує довіру користувача до певного сайту, CSRF використовує довіру сайту до браузера користувача. Під час атаки CSRF зловмисник обманом змушує невинного кінцевого користувача надіслати веб-запит, який він не мав на меті. Це може призвести до виконання дій на веб-сайті, які можуть включати ненавмисний витік даних клієнта або сервера, зміну стану сеансу або маніпулювання обліковим записом кінцевого користувача [11]. На рисунку 2.4 наведено діаграма роботи CSRF атаки.

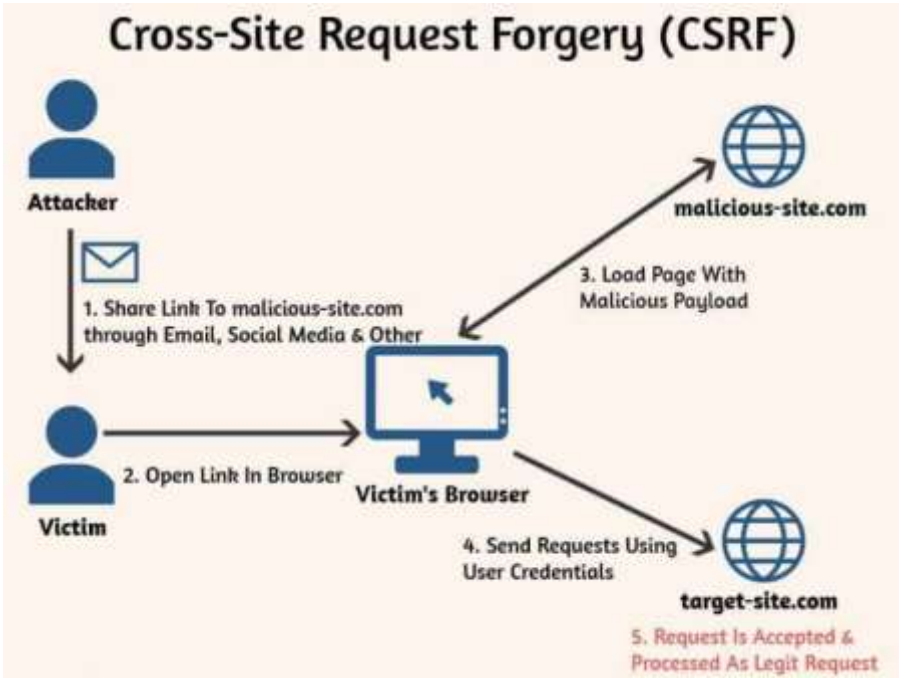


Рисунок 2.4 - Діаграма роботи CSRF атаки

XSS – атаки, які дозволяють користувачеві впроваджувати сценарії на стороні клієнта в браузери інших користувачів. Зазвичай це досягається шляхом зберігання зловмисних сценаріїв у базі даних, де вони будуть отримані та відображені іншим користувачам, або шляхом спонукання користувачів клацати посилання, яке призведе до виконання JavaScript зловмисника у браузері користувача. Однак атаки XSS можуть виникати з будь-якого ненадійного джерела даних, наприклад файлів cookie або веб-служб, якщо дані недостатньо оброблені перед додаванням на сторінку [12].

SQL-ін’єкція – це тип атаки, коли зловмисник може виконати довільний код SQL у базі даних. Це може призвести до видалення записів або витоку даних [16].

Клікджекінг — це вид маніпуляції, коли хакер розміщує на сайті невидиме вікно з потрібною йому командою поверх іншого веб-контенту. Тобто користувачі думають, що натискають на кнопку, яку бачать, але насправді клікають на вікно, яке підготував хакер. [15].

DoS (Denial of Service - Відмова в обслуговуванні) – атака, що має за мету змусити сервер не відповідати на запити. Такий вид атаки не передбачає

отримання деякої секретної інформації, але іноді буває допоміжною в ініціалізації інших атак. Наприклад, деякі програми через помилки в своєму коді можуть викликати виняткові ситуації, і при вимиканні сервісів здатні виконувати код, наданий зловмисником або атаки лавинного типу, коли сервер не може обробити величезну кількість вхідних пакетів [14].

DDoS (Distributed Denial of Service) – має таку ж мету що і DoS, але проводяться не з одного комп'ютера, а з кількох комп'ютерів в мережі. В DDoS-атаках використовується або виникнення помилок, що призводять до відмови сервісу, або спрацьовування захисту, що приводить до блокування роботи сервісу, а в результаті і до відмови в обслуговуванні. DDoS використовується там, де звичайний DoS є неефективним. Для цього кілька комп'ютерів об'єднуються, і кожен здійснює DoS атаку на систему жертви. Разом це називається DDoS-атака [15].

У таблиці 2.3 показано загрози, які можуть виникнути через через поганий дизайн додатку.

Таблиця 2.3 – Загрози, що можуть виникнути через через поганий дизайн додатку

| Категорія | Загрози |
|---|---|
| Перевірка даних введення | Переповнення буферу, SQL ін’єкції, кроссайтовий скриптінг, канонізація |
| Авторизація | Підвищення привілеїв, отримання конфіденційних даних, підробка даних, атаки наживками |
| Конфігурація | Отримання доступу до інтерфейсу адміністратора, отримання доступу до сховища конфігураційних даних |
| Конфіденційні дані | Отримання доступу до конфіденційних даних |
| Сесія | Перехоплення сеансу, повтор сеансу |
| Криптографія | Погана генерація ключів або управління ключами, слабке шифрування |

Кінець таблиці 2.3

| | |
|---|---|
| Маніпуляції з параметрами | Маніпуляції з запитами, маніпуляції з cookie, маніпуляція із заголовками HTTP |
| Виняткові ситуації | Отримання інформації, відмова в обслуговуванні |
| Автентифікація | Відмова виконання операції користувачем, використання додатку без можливості відстеження зловмисника |

## 2.3 Побудова методу розробки сайтів інтернет-магазинів з урахуванням захисту від загроз

Модифікований метод розробки пропонує використовувати всі первинні етапи розробки, такі як формування вимог, планування, аналіз та впроваджувати заходи безпеки на цих етапах. Це допоможе врахувати всі можливі вразливості сайту та правильно спланувати їх розробку та тестування.

Модифікований метод поєднує в собі наступні етапи моделювання з розглянутих методів:

Парадигма моделювання: можна розглядати парадигму гіпертексту та бази даних із методології WebML, оскільки в деяких веб-додатках дані повинні зберігатися в базі даних для адміністрування або прийняття рішень, а гіпертекст для доступу користувачів [2].

Моделювання вимог: методи аналізу структурованих систем і документ із специфікацією вимог до програмного забезпечення можна використовувати для визначення основних вимог бізнесу. OOHDM та W2000 багаті на моделювання вимог [2].

Моделювання вмісту: WebML може фіксувати детальний вміст. Найвидатнішими особливостями WebML є розгляд робочих процесів, у яких задіяно більше одного актора (і, отже, як можуть бути синхронізовані відповідні

гіперподання), можливість контролювати певні типи потоків діяльності шляхом призначення об’єктів [2] .

Моделювання навігації: WebML явно визначив структуру класу, яка підтримує визначені структури. UWE, WebML і OO-H визначають нову концептуальну діаграму процесу. І структура, і навігація процесами. OOHDM розширює як свою структуру, так і навігаційну модель для підтримки нових концепцій. Навігаційна доріжка WSDM і навігаційна діаграма OOHDM є більш потужними для моделювання навігації, для детальної структури [2].

На рисунку 2.5 показано схему алгоритму для покращення безпеки веб-додатків.

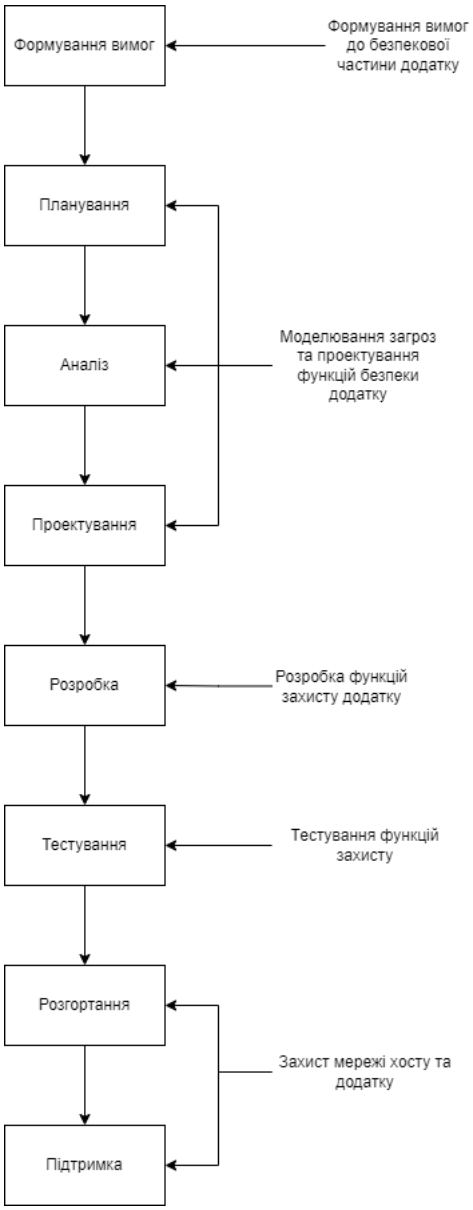


Рисунок 2.5 - Схему алгоритму для покращення безпеки веб-додатків

Моделювання загроз і оцінка безпеки застосовуються, коли створюється нові веб-програми або переглядаєте існуючі програми. Для забезпечення безпеки при створенні нового веб-додатку слід на ранніх етапах провести моделювання загроз, створення вимог. Якщо ціль - це покрашення безпеки вже існуючуго додатку то також потрібно провести оцінку вже існуючих функцій безпеки.

Для того щоб розробити надійну програму, стійку до злому, необхідно використати систематичний та цілісний підхід до захисту мережі, хосту та додатку. На кожному рівні питання безпеки розглядаються на мережевому рівні, рівні хоста та рівні додатків. На малюнку 2.6 також показано категорії конфігурації, які використовуються для організації різних параметрів конфігурації безпеки, які застосовуються до хоста та мережі, а також категорії вразливості програми, які використовуються для структурування розгляду програми.

Безпека веб-додатків цілісного підходу повинна розглядатися на різних рівнях додатків. Зловмисник може використовувати слабкі місця на будь-якому рівні. З цієї причини цілісний підхід до безпеки додатків потрібно застосовувати на всіх трьох рівнях. Цей цілісний підхід до безпеки показано на малюнку 2.6.

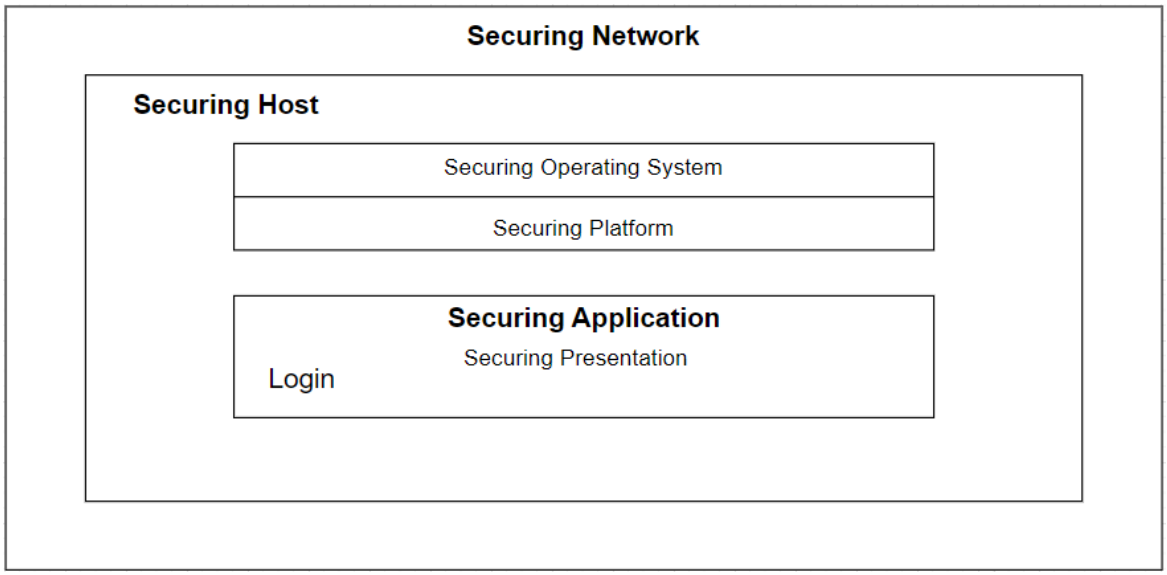


Рисунок 2.6 - Цілісний підхід до безпеки веб-сайтів

Хост включає в себе операційну систему та .NET Framework, а також відповідні служби та компоненти. Незалежно від того, чи є хостом веб-сервер, на якому запущено IIS, сервер додатків, на якому запущено Enterprise Services, чи сервер бази даних, на якому запущено SQL Server, існує певна методологія, спільна для різних ролей і типів серверів.

Існує кілька наборів категорій уразливості додатків, які допоможуть вам розробити та створити безпечні веб-додатки та оцінити безпеку існуючих додатків. Це загальні категорії, які охоплюють кілька технологій і компонентів у багаторівневій архітектурі. Ці категорії є предметом обговорення під час проектування, будівництва та оцінки безпеки. Категорії вразливості додатків. Анулювання введення, автентифікація, авторизація, керування конфігурацією, конфіденційні дані, керування сеансами, криптографія, маніпулювання параметрами та керування винятками.

Веб-додатки ставлять перед дизайнерами та розробниками багато проблем. Природа HTTP без збереження стану означає, що відстеження стану сеансу кожного користувача стає обов’язком програми. Як попередник цього, програма повинна мати можливість ідентифікувати користувача за допомогою певної форми автентифікації. Враховуючи, що всі подальші рішення щодо авторизації базуються на ідентифікації користувача, дуже важливо, щоб процес автентифікації був безпечним, а механізм обробки сеансу, який використовується для відстеження автентифікованих користувачів, був однаково добре захищений. Розробка безпечної автентифікації та механізмів керування сеансами — це лише кілька проблем, з якими стикаються дизайнери та розробники веб-додатків. Інші проблеми виникають через те, що вхідні та вихідні дані передаються через загальнодоступні мережі. Запобігання маніпуляції з параметрами та розголошенню конфіденційних даних є іншими основними проблемами. Деякі з найпоширеніших проблем, які необхідно вирішити за допомогою методів безпечного проектування, показано на малюнку 11(Схема робочого процесу для веб-системи керування конференцією).

Традиційно безпека вважалася проблемою мережі, де брандмауер є основним захистом (модель фортеці) або чимось, що системні адміністратори вирішують шляхом блокування головні комп'ютери. Архітектори та розробники додатків традиційно розглядають безпеку як запізнілу думку або як функцію, яку слід розглядати, коли дозволяє час — зазвичай після розгляду міркувань продуктивності. Підвищення безпеки веб-додатків: загрози та засоби протидії Проблема брандмауера або моделі фортеці полягає в тому, що атаки можуть проходити через захист мережі безпосередньо до програми. Типовий брандмауер допомагає обмежити трафік до HTTP, але трафік HTTP може містити команди, які використовують вразливі місця програми. Повністю покладатися на блокування своїх хостів — ще один невдалий підхід. Хоча декільком загрозам можна ефективно протистояти на рівні хоста, атаки на програми становлять серйозну проблему безпеки, яка зростає. Ще одна область, де виникають проблеми з безпекою, – розгортання. Знайомий сценарій — коли програма дає збій під час її розгортання в заблокованому робочому середовищі, що змушує адміністратора послабити налаштування безпеки. Це часто призводить до нових вразливостей безпеки. Крім того, відсутність політики безпеки або вимог до програми, які не відповідають політиці, можуть поставити під загрозу безпеку.

## 2.4 Формування критерію оцінювання щодо використання методу розробки сайтів інтернет-магазинів з урахуванням захисту від загроз

Визначивши основні типи атак на веб-сайти, можна побудувати критерій оцінювання щодо ефективності роботи модифікованого методу. Для цього слід визначити рівні захисту веб-сайту. В таблиці 2.4 наведені рівні захисту сайтів.

Таблиця 2.4 - Рівні захисту сайтів

| | |
|---|---|
| Перший рівень | Система захисту сайту покриває менше 30% видів атак |
| Другий рівень | Система захисту сайту покриває від 30% до 50 % видів атак |
| Третій рівень | Система захисту сайту покриває від 50% до 70% видів атак |
| Четвертий рівень | Система захисту сайту покриває від 70% до 90% видів атак |
| П’ятий рівень | Система захисту сайту покриває від більше 90% видів атак |

Для того, щоб обрати потрібний рівень захисту веб-сайту інтернет магазину, необхідно вирахувати коефіцієнт ефективності системи захисту(S), який розраховується за формулою 2.1

$$S = \frac{L}{T * P},$$

(2.1)

де L – рівень захисту;

$T$ – час розробки в годинах;

$P$ – вартість розробки в доларах.

Також потрібно визначити загрози, від яких треба захистити веб-додаток інформаційної системи торгової компанії. Існує багато різних видів зламу програмного забезпечення та додатків. Але не всі вони використовуються для зламу веб-сайтів. Тому немає сенсу намагатися захистити веб-сайт систему від усіх існуючих видів загроз, бо на це піде велика кількість ресурсів. Але і обирати найнижчій рівень захисту теж не дуже вдала ідея. Так, це знизить витрату ресурсів на розробку, але і зламати такий сайт буде дуже просто.

Вибір рівня безпеки сайту інформаційної системи торгової компанії залежить від вже існуючих елементів, обраної мови програмування та інших інструментальних засобів, складності системи. Всі ці показники впливають на час та вартість розробки.

# 3 МЕТОДИКА ВИКОРИСТАННЯ МЕТОДУ РОЗРОБКИ САЙТІВ ІНТЕРНЕТ-МАГАЗИНІВ З УРАХУВАННЯМ ЗАХИСТУ ВІД ЗАГРОЗ

## 3.1 Обґрунтування вибору інструментальних засобів для забезпечення захисту

На кожному етапі розробки сайтів використовуються різні інструментальні засоби для досягнення мети кожного з етапів.

Для створення стійкого до зламу додатку, необхідно обрати правильні інструментальні засоби. Два найпопулярніших типи інструментальних засобів для розробки сайтів інтернет-магазинів це системи управління контентом(CMS) та фреймворки(Framework).

CMS зазвичай обирають через їхню простоту в розробці веб-додатків. Дуже часто за допомогою CMS можна створити сайт без написання коду. CMS побудовано на базовій структурі. Це дозволяє користувачам встановлювати теми, плагіни тощо, не впливаючи на основні функції сайту. Більшість CMS базується на модулях, що дозволяє спростити розробку сайтів, бо не обов’язково наймати програмістів.

Але у CMS також є недоліки. Один з них, який є найважливішим - це низький рівень безпеки. Через те, що CMS в основному це програмне забезпечення з відкритим кодом, будь яка людина може прочитати його, та знайти якісь діри в захисті, які потім зможе використати для зламу сайтів на цій CMS. Також через встановлення або оновлення розширень з ненадійних ресурсі можуть поставити під загрозу безпеку веб-сайту [9].

Ще одна причина низького рівня захисту сайтів, які розроблені на CMS - це низький бюджет. Компанії, які замовляють або розроблюють свої сайти за допомогою CMS обирають їх в основному через низьку ціну і намагються зменшити вартість розробки на всіх етапах. Тому часто виходить так, що команда розробників не включає формування та проектування функцій захисту веб-додатку на ранніх етапах розробки. Частіше за все просто обираються декілька

плагінів для захисту сайту від базових видів атак, а кастомні функції захисту не розробляються, щоб спростити розробку та зменшити її вартість.

Фреймворки, натомість, пропонують набагато кращий захист веб-додатків.

По-перше, багато фрейморків не дають доступ до вихідного коду, або в значній мірі обмежують його для осіб, які ніяк не пов’язані з розробкою фреймворку. Через це зловмисникам набагато складніше знайти прогалини в захисті сайтів, написаних за допомогою фреймворків.

Веб-сайти, які працюють на фреймворках, мають власні спеціальні коди, створені на основі фреймворків. Веб-продукти на основі фреймворків важче атакувати, оскільки стороннім особам важче знайти в них недоліки безпеки. Крім того, більшість сучасних фреймворків забезпечують набір вбудованих функцій безпеки, які захищають рішення від найпоширеніших кіберзагроз. Багато фреймворків мають такі вбудовані функції, як захист від XSS і SQL ін’єкцій, захист CSRF, шифрування даних, що дозволяє шифрувати та захищати веб-сайти від найпоширеніших типів атак [10].

Через більшу складність та вартість розробки сайтів за допомогою фреймоврків, їх частіше використовують великі компанії, вимоги яких набагато складніші ніж у тих хто обирає CMS. В таких випадках часто можуть збиратися більш важливіші дані від користувачів або зберігатися важливі дані компанії в базах даних додатків. Через це потреби у захисті додатку набагато більші, ніж у тих клієнтів, що обирають CMS. Також такі проекти потребують більшої кількості кастомного функціоналу, що дуже часто не можуть запропонувати CMS.

Найбільш популярні фреймворки в наш час для розробки веб-додатків інтернет магазинів - це Laravel, Django, Symfony, Flask.

Laravel пропонує велику кількість функцій для захисту від зламу прямо “із коробки”, таких як хешування паролів, захист маршрутів, захист від CSRF атак, можливості для надійного шифрування AES через розширення PHP mcrypt, перевірка, відновлення паролів, перевірка даних користувача, тощо.

Фреймворк Django також має багато фунцій захису веб-додатків, таких як захист від XSS атак, захист від CSRF атак, SQL ін’єкції, захист від клікджекінгу, валідація заголовків хосту.

З міркувань безпеки завжди краще розгорнути свій сайт за протоколом HTTPS. Без цього зловмисні користувачі мережі можуть перехопити облікові дані автентифікації або будь-яку іншу інформацію, що передається між клієнтом і сервером, а в деяких випадках – активні мережеві зловмисники – можуть змінити дані, які надсилаються в будь-якому напрямку. Усі сучасні фреймворки підтримують цей протокол.

Головна ціль модифікованого методу – підвищити захист сайтів інтернет-магазинів. Для цього потрібно не тільки впровадити проектування функцій захисту на ранніх етапах а й обрати зручні інструментальні засоби для реалізації цих функцій. Фреймворки для розробки сайтів – це дуже зручний інструмент не тільки для розробки сайтів з використанням базових функцій захисту від зламу, за допомогою них також зручно впроваджувати кастомний функціонал. Для застосування модифікованого методу розробки сайтів інтернет-магазинів рекомендується використовувати один із розглянутих фреймворків.

## 3.2 Опис методики щодо використання методу розробки сайтів інтернет-магазинів з урахуванням захисту від загроз

Технологія використання модифікованого методу розробки сайтів інтернет-магазинів з урахуванням захисту виконується у кілька етапів:

- формування вимог;
- аналіз існуючих загроз для сайту на основі вимог;
- з концептуальний дизайн сайту;
- навігаційний дизайн сайту;
- проектування архітектури;

– створення інтерфейсу;

– розробка сайту;

– тестування сайту;

– підтримка сайту.

Докладна схема роботи модифікованого методу розробки сайтів інтернет-магазинів з урахуванням захисту від загроз наведена на рисунку 3.1

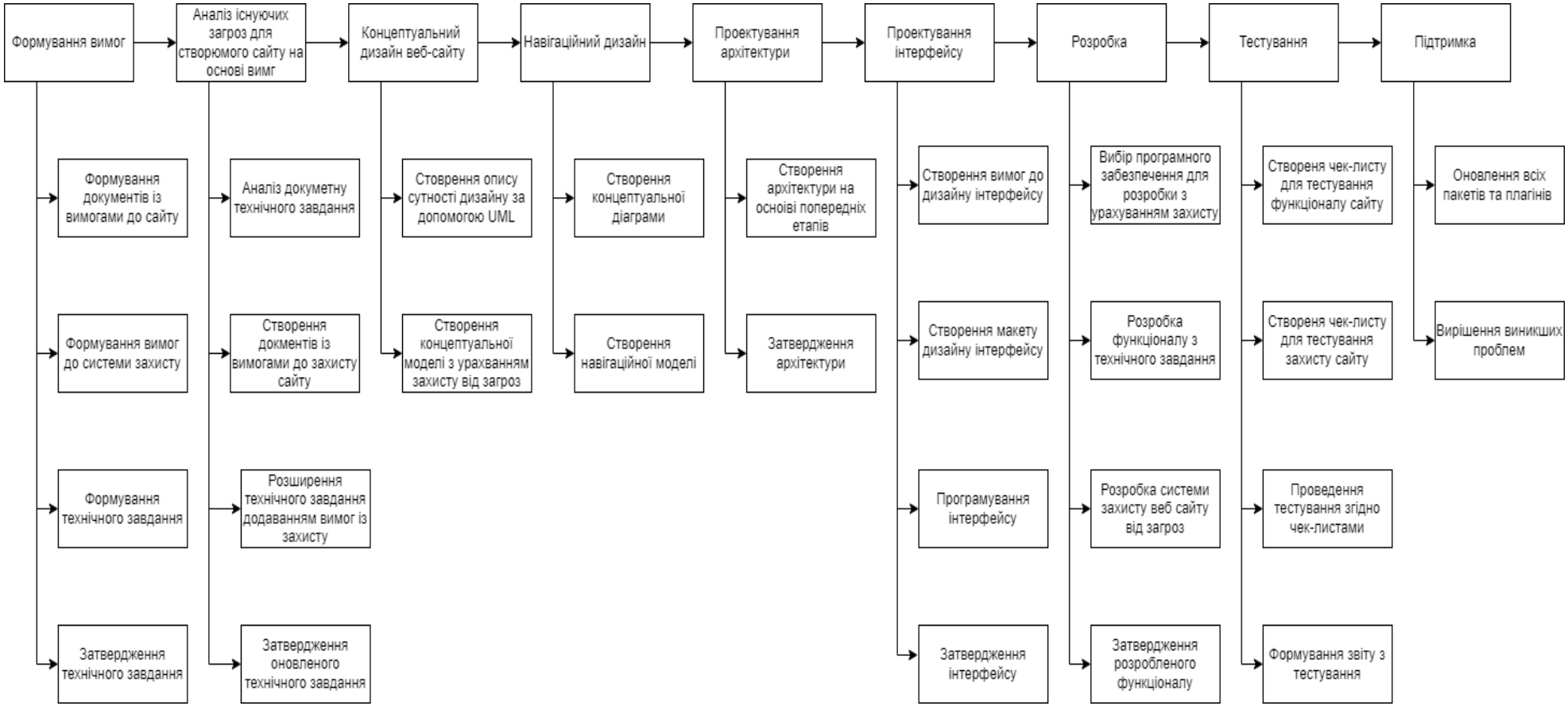


Рисунок 3.1 - Докладна схема роботи модифікованого методу розробки сайтів інтернет-магазинів з урахуванням захисту від загроз

На етапі формування вимог представник команди розробників та менеджер компанії обговорюють потреби замовника до сайту, що буде створюватись. Формуються основні вимоги до функціоналу, а також до системи захисту веб-сайту. Обговорюються інші елементи інформаційної системи На цьому етапі формуються документ технічного завдання. Також представник компанії замовника мають затвердити створений документ технічного завдання.

На етапі аналізу загроз команда, яка відповідає за реалізацію захисту повинна на основі вже створених вимог на попередньому етапі провести аналіз можливих загроз зламу, обрати необхідні засоби захисту від них та розширити технічне завдання вимогами до захисту від загроз.

Концептуальний дизайн - це етап на якому створюються концептуальні моделі сайту, що розроблюється з урахуванням захисту від загроз. Також розроблюється опис сутності за допомогою UML.

На етапі створення навігаційного дизайну необхідно розробити навігаційну модель сайту, а також концептуальну діаграму.

Після усіх цих етапів команда архітекторів приступає до проектування архітектури веб-сайту з урахуванням захисту від загроз. На цьому етапі головний архітектор компанії проводить багато зустрічей з командами розробників. На цьому етапі розробляється архітектура на основі вже існуючих елементів в інформаційній системі замовника та на основі попередніх етапів. В кінці формуються діаграми архітектури сайту. Також головний архітектор та представник компанії замовника повинні затвердити архітектуру.

Після створення архітектури команда розробників приступає до створення інтерфейсу. На цьому етапі створюються вимоги до інтерфейсу та розроблюється макет дизайну. Макет дизайну створюють дизайнери компанії. Після затвердження дизайну розробники приступають до програмування дизайну. В кінці етапу представник компанії замовника затверджує створений дизайн інтерфейсу.

Етап розробки сайту також починається після етапу проектування архітектури. На цьому етапі обираються програмне забезпечення та

інструментальні засоби з урахуванням захисту. Після затвердження інструментальних засобів починається розробка сайту. Також розробляється система захисту від загроз. Після виконання всіх етапів керівник команди та представник компанії замовника затверджують розроблений функціонал.

На етапі тестування розробляються чек-листи для тестування функціоналу та захисту сайту. Після проведення тестів згідно чек-листам створюється звіт із всіма знайденими помилками, які відправляються команді розробників на виправлення.

Етап підтримки це фінальний етап модифікованого методу. На цьому етапі виконуються невеличкі доробки системи, періодично оновлюються елементи системи, такі як встановлені пакети та плагіни, а також проводится вирішення виниклих проблем з функціоналом чи функціями захисту.

# 4 АПРОБАЦІЯ МЕТОДУ РОЗРОБКИ САЙТІВ ІНТЕРНЕТ-МАГАЗИНІВ З УРАХУВАННЯМ ЗАХИСТУ ВІД ЗАГРОЗ

## 4.1 Опис та аналіз вимог для побудови сайту інтернет-магазину електронної техніки

Апробація модифікованого методу буде проводитись на прикладі інтернет-магазину інформаційної системи торгової компанії електронікою.

Вимоги до створення сайту інтернет магазину з продажу електроніки з урахуванням захисту від загроз:

– провести аналіз існуючих загроз для сайту з продажу електроніки;

– створити список загроз, від яких треба захистити веб-сайт;

– створити концептуальний дизайн інтернет-магазину;

– створити навігаційний дизайн;

– створити інтерфейс сайту;

– вибір архітектури сайту з урахуванням захисту від загроз;

– обрати програмне забезпечення для створення сайту з урахуванням захисту від загроз.

Як написано в попередніх пунктах, загроз зламу додатків існує дуже багато, але не всі вони використовуються для зламу веб-сайтів. Тому слід виділити найпопулярніші загрози, які використовуються для зламу сайтів інтернет-магазинів і впровадити проти них захист. Список загроз може бути змінений, в залежності від різних факторів, наприклад якщо потрібно розробити складний, незвичний функціонал, і для нього може знадобитись кастомний захист від злому. Список загроз, від яких треба захистити веб-сайт інтернет-магазину приведено в таблиці 4.1.

Таблиця 4.1 - Список загроз, від яких треба захистити веб-сайт інтернет-магазину з продажу електроніки

| Категорія | Назва можливих загроз |
|---|---|
| Захист даних введення | Переповнення буферу, SQL ін’єкції, кроссайтовий скриптінг, |
| Захист форм авторизації | Підвищення привілеїв, отримання конфіденційних даних, підробка даних, атаки наживками |
| Захист конфігурації | Отримання доступу до інтерфейсу адміністратора, отримання доступу до сховища конфігураційних даних |
| Шифрування конфіденційних дані | Отримання доступу до конфіденційних даних |
| Захист сесії | Перехоплення сеансу, повтор сеансу |
| Криптографія | Погана генерація ключів або управління ключами, слабке шифрування |
| Захист від маніпуляції з параметрами | Маніпуляції з запитами, маніпуляції з cookie, маніпуляція із заголовками HTTP |
| Захист серверу | DoS-атаки, DDoS-атаки |

Концептуальна діаграма сайту інтернет-магазину з продажу електроніки приведено на рисунку 4.2

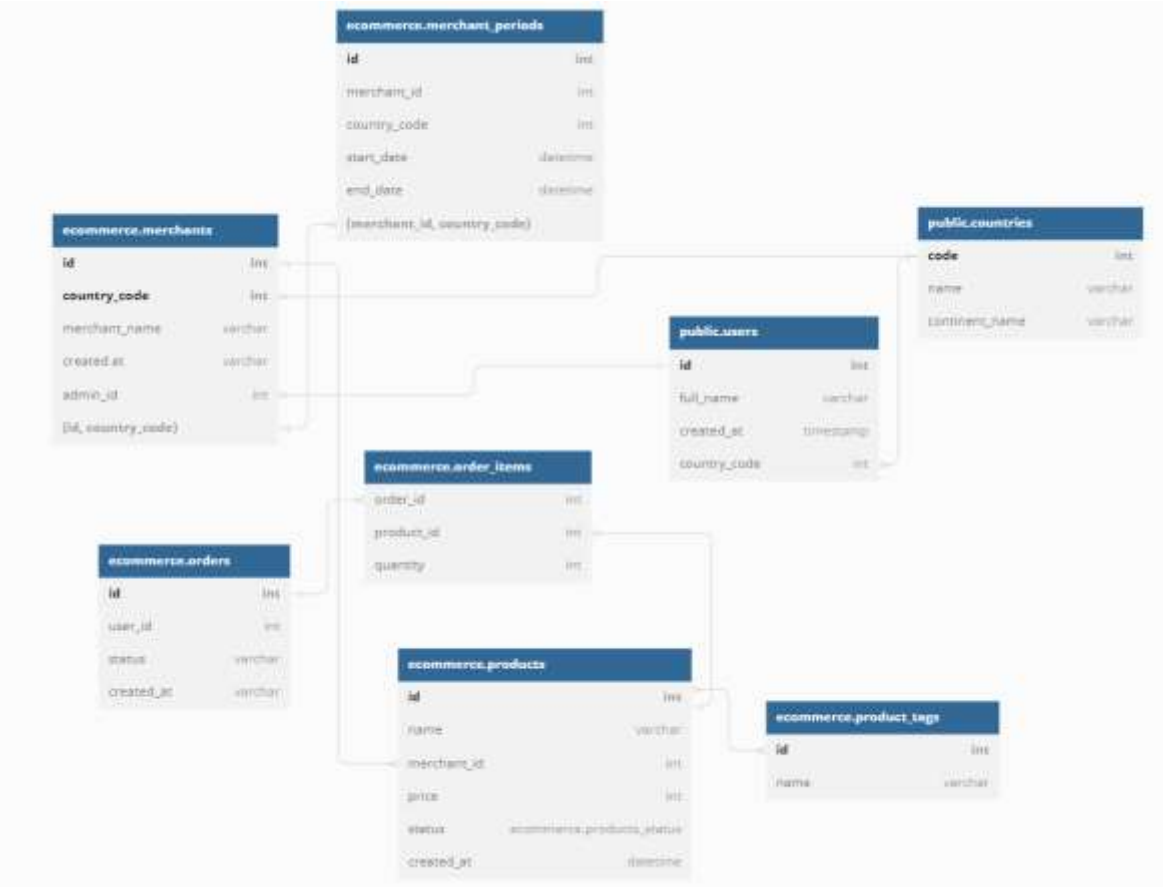


Рисунок 4.2 - Концептуальна діаграма сайту інтернет-магазину з продажу електроніки

Навігаційний дизайн сайту інтернет-магазину з продажу електроніки приведено на рисунку 4.2

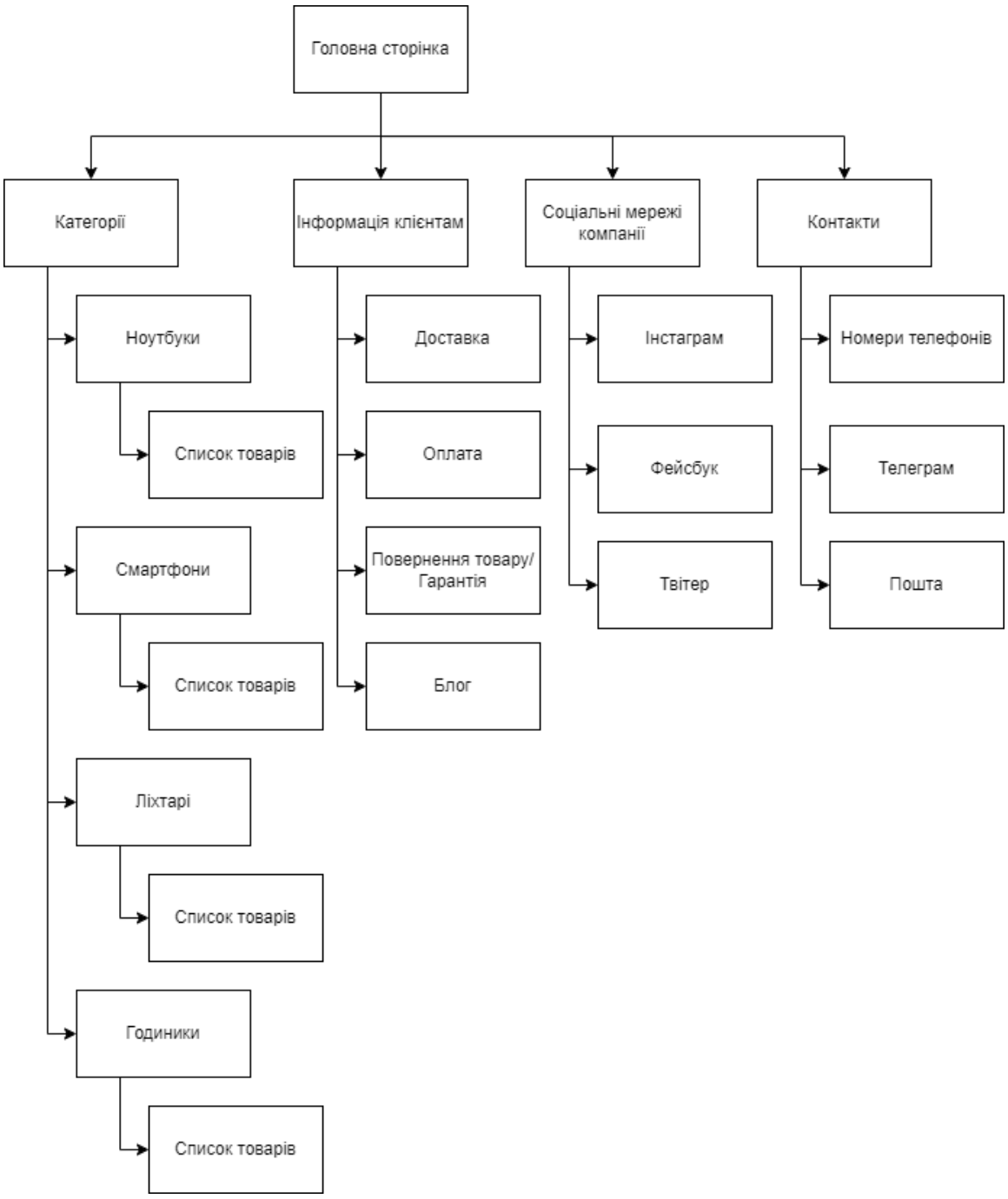


Рисунок 4.2 - Навігаційний дизайн сайту інтернет-магазину з продажу електроніки

Для розробки сайту інтернет-магазину з продажу електроніки найкраще підійде MVC архітектура, бо найпопулярніші фреймворки для розробки веб-додатків підтримують саме цю архітектуру. Завдяки цьому можна легше впроваджувати функції захисту сайту. Це також знизить час та вартість розробки

системи захисту веб сайту. MVC архітектура сайту інтернет-магазину з продажу електроніки показана на рисунку 4.3.

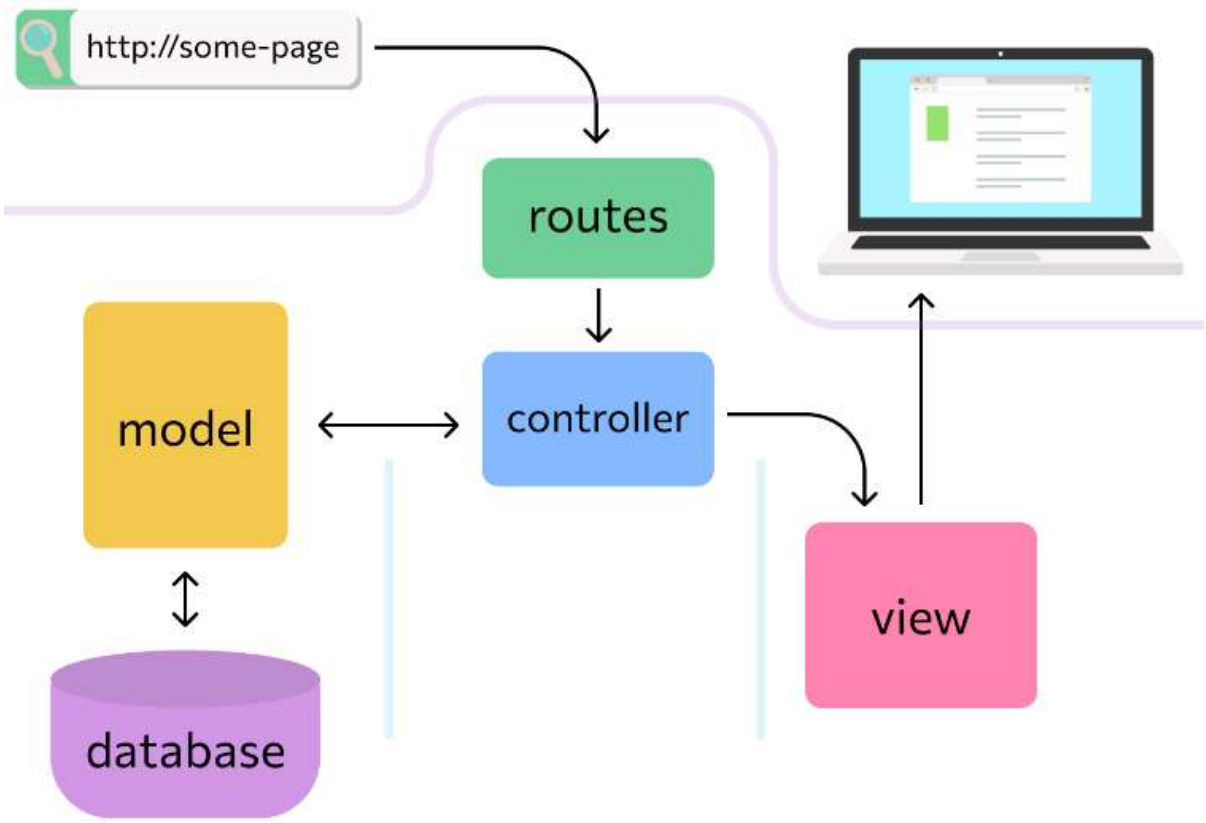


Рисунок 4.3 - MVC архітектура сайту інтернет-магазину з продажу електроніки

Вибір програмного забезпечення для створення сайту з урахуванням захисту від загроз. Для розробки сайту інтернет-магазину найкраще підійде один із фреймворків PHP або Python. Вони мають широкий функціонал як з розробки функціоналу MVC функціоналу, так і з розробки функцій захисту від загроз. Для реалізації сайту інтернет-магазину електроніки пропонується використовувати Laravel. Це самий популярний PHP фреймворк з великою підтримкою, постійними оновленнями. Він підтримує всі відомі JavaScript фреймворки. Також в нього є вбудовані функції захисту від багатьох загроз, таких як CSRF-атак чи SQL-ін’єкцій.

В якості бази даних пропонується MySQL. Це одна з найпопулярніших баз даних, яка має підтримку зі сторони Laravel.

## 4.2 Результати апробації методу розробки сайту інтернет-магазину електронної техніки з урахуванням захисту від загроз

При створенні сайту сайту інтернет-магазину з продажу електроніки було використано модифікований метод розробки сайтів інтернет-магазинів з урахуванням захисту від загроз. На його прикладі було показано приклади діаграм, які створюватимуться при розробці сайтів з використанням модифікованого методу. Також було визначено список найважливіших можливих атак на сайт інтернет-магазину з продажу електроніки та впроваджено захист проти них. Також було приведено приклад вибору програмного забезпечення для створення сайту інтернет-магазину з продажу електроніки. Вибір був зроблений на основі зручності розробки, кількості наявного функціоналу, підтримки різного програмного забезпечення, баз даних, а також на основі системи захисту від загроз.

Також була обрана архітектура сайту та приведена її діаграма. Одним із критеріїв вибору була можливість розробки системи захисту сайту інтернет-магазину з продажу електроніки. Приклади сайту інтернет-магазину з продажу електроніка приведені на рисунках 4.4, 4.5 та 4,6.

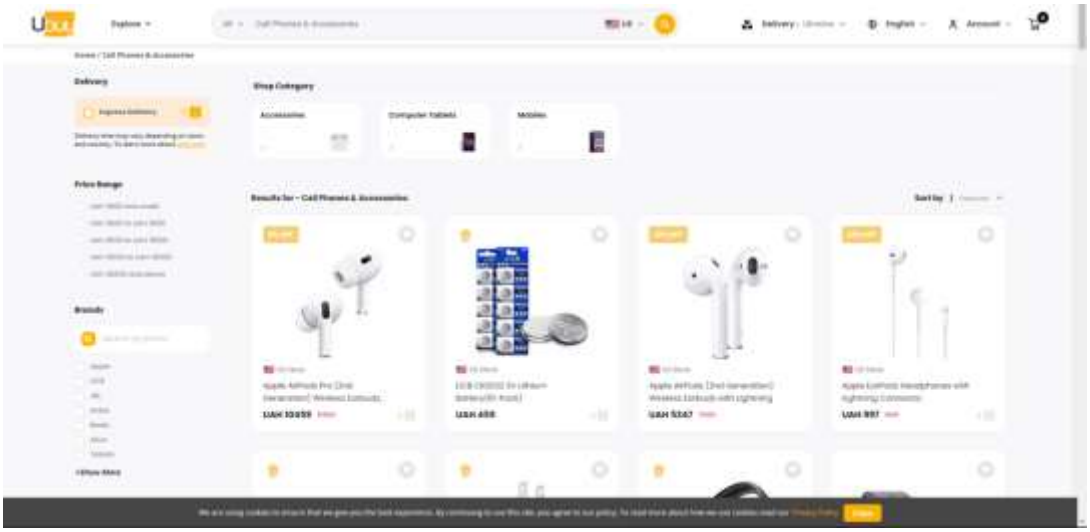

Рисунок 4.4 - Сторінка аксесуарів сайту інтернет-магазину з продажу електроніки

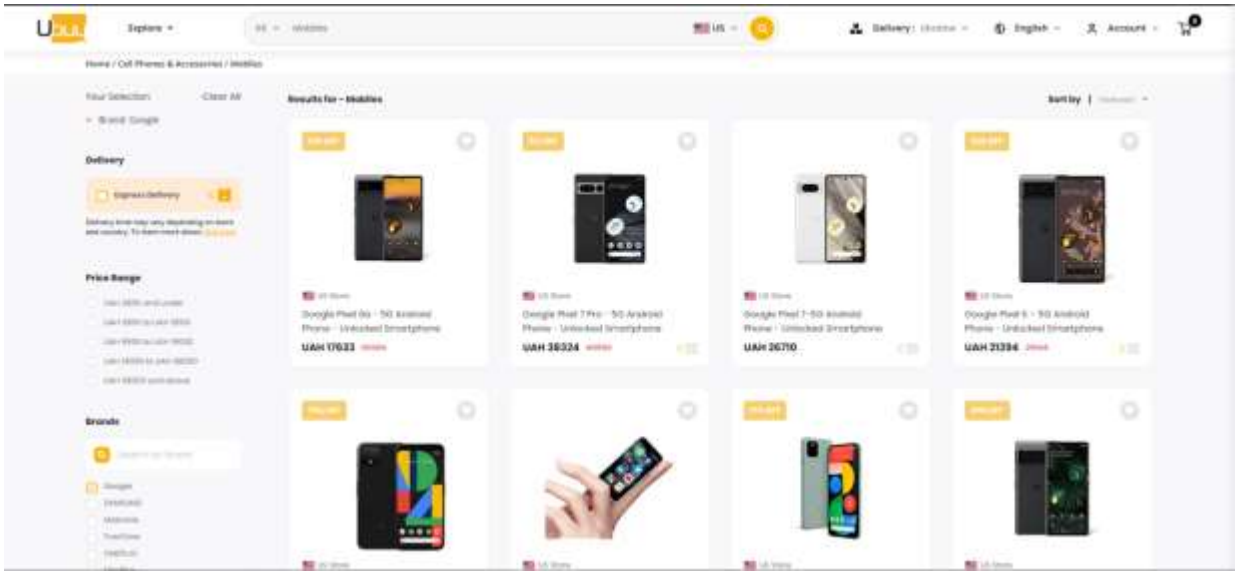

Рисунок 4.5 - Сторінка смартфонів сайту інтернет-магазину з продажу електроніки

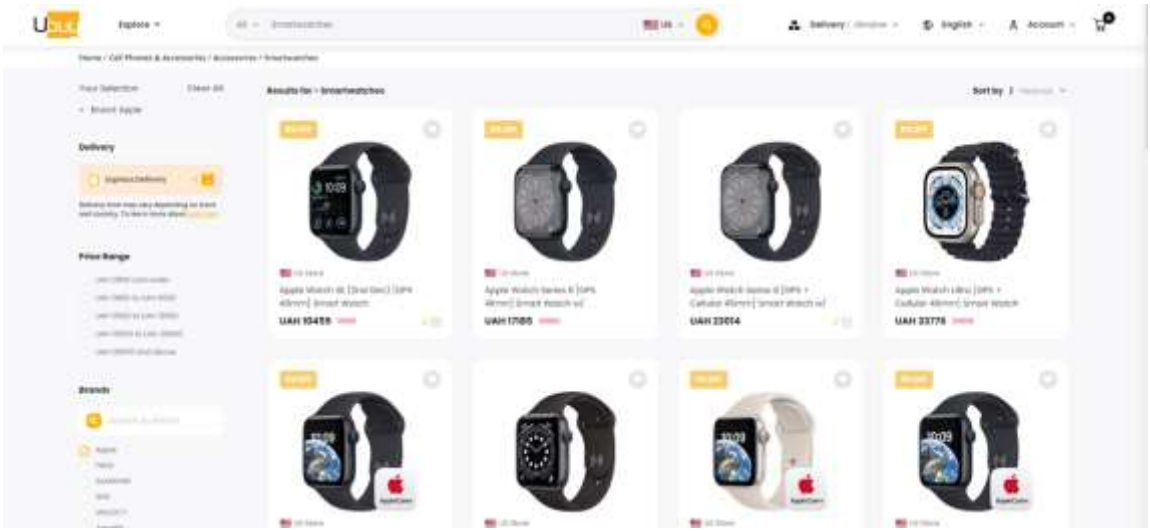

Рисунок 4.6 - Сторінка смарт-годинників сайту інтернет-магазину з продажу електроніки

# ВИСНОВКИ

Кваліфікаційну роботу виконано згідно з методичними вказівками до виконання та захисту кваліфікаційної роботи другого (магістерського) рівня вищої освіти [1].

Метою роботи є вивчення та аналіз існуючих методів розробки сайтів інтернет-магазинів для покращення захисту від загроз.

Для виконання поставленої мети в роботі:

– досліджено існуючі методи розробки сайтів інтернет-магазинів;

– досліджено особливості розробки сайтів інтернет-магазинів;

– досліджено підходи розробки сайтів інтернет-магазинів;

– досліджено види атак, які можуть бути застосовані для зламу сайтів інтернет-магазинів;

– створено модифікований метод розробки сайтів інтернет-магазинів з урахуванням захисту від загроз;

– проведено аналіз існуючих інструментальних засобів для створення сайтів інтерне-магазинів для використання з модифікованим методом розробки сайтів інтернет-магазинів з урахуванням захисту від загроз;

– проведено апробацію модифікованого методу розробки сайтів інтернет-магазинів з урахуванням захисту від загроз.

# ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Методичні вказівки щодо розробки та оформлення кваліфікаційної роботи (для студентів усіх форм навчання другого (магістерського) рівня вищої освіти спеціальності 122 Комп'ютерні науки освітньо-професійної програми «Інформаційні управляючі системи та технології») / Упоряд.: Петров К.Е., Левикін В.М., Чалий С.Ф., Євланов М.В., Саєнко В.І., Міхнов Д.К., Міхнова А.В., Чала О.В. – Харків: ХНУРЕ, 2021. – 30 с.

2. Ingle D.R. Hybrid Analysis and Design Model for Building Web Information // International Journal of Computer Science Issues, 2012. – 518c.

3. Шалева О.І. Електронна комерція [Навчальний посібник] // Центр учбової літератури, Київ. - 2011. - 15с.

4. Website Development Process: Full Guide in 7 Steps [Електронний ресурс] / Svetlana Gordiyenko - Режим доступу: https://xbsoftware.com/blog/website-development-process-full-guide - 23.10.2022р. - загл. з екрану.

5. Luciano Baresi Sandro Morasca W2000: A Modeling Notation for Complex Web Applications [Текст] / Luciano Baresi, Luca Mainetti // International Journal of Computer Science Issues, 2006. - 25с

6. André L.S. Domingues Web application development methods: a comparison [Текст] / André L.S., Sandro L. Bianchini, Marcella L.S. Costa, Fabiano C. Ferrari, José C. Maldonado // WebMedia, Brazil. - 2007. - 2с.

7. Miguel Grinberg Flask Web Development [Текст] / Miguel Grinberg // O'REILLY, Sebastopol, - 2014. - 20с.

8. Deane Barker Web Content Management [Текст] / Deane Barker // O'REILLY, Sebastopol, - 2016. - 45с.

9. Методи розробки Web-сайтів [Електронний ресурс] / Google Sites - Режим доступу: https://sites.google.com/site/tz5103voinovakateryna/metodi-rozrobki-web-sajtiv - 21.11.2022р. - загл. з екрану.

10. Java Code Geeks. Spring Framework Cookbook [Текст] / Java Code Geeks // Exelixis Media. - 2017. - 76с.

11. Cross Site Request Forgery(CSRF) [Електронний ресурс] / Owasp. - Режим доступу: https://owasp.org/www-community/attacks/csrf - 18.11.2022р. - загл. з екрану.

12. Cross Site Request Scripting(XSS) [Електронний ресурс] / Owasp. - Режим доступу: https://owasp.org/www-community/attacks/xss/ - 18.11.2022р. - загл. з екрану.

13. Denial of Service(DoS) [Електронний ресурс] / Owasp. - Режим доступу: https://owasp.org/www-community/attacks/Denial_of_Service - 18.11.2022р. - загл. з екрану.

14. What is DDoS attack and how it work? [Електронний ресурс] / Comptia - Режим доступу: https://www.comptia.org/content/guides/what-is-a-ddos-attack-how-it-works - 18.11.2022р. - загл. з екрану.

15. Що таке клікджекінг та як від нього захиститися [Електронний ресурс] / Netfreedom - Режим доступу: https://netfreedom.org.ua/article/shcho-take-klikdzheking-ta-yak-vid-nogo-zahistitisya - 15.10.2022р. - загл. з екрану.

16. SQL-ін’єкція. Що це таке [Електронний ресурс] / pro-computer - Режим доступу: http://pro-computer.pp.ua/4257-sql-nyekcya-scho-ce-take-kompyuter76.html - 9.11.2022р. - загл. з екрану.

17. Dr. Kovita. Modelling Techniques of Web Architecture for Improvement of Web Applications [Текст] / Dr. Kovita, Sonia Sachdeva // International Journal of Emerging Technology and Advanced Engineering. - 2017. - 225с.

18. Django Web Framework [Електронний ресурс] / djangoproject - Режим доступу: https://www.djangoproject.com/ - 19.10.2022р. - загл. з екрану.