

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
Західноукраїнський національний університет  
Факультет комп'ютерних інформаційних технологій  
Кафедра комп'ютерної інженерії

Мірецький Володимир Володимирович

## **Модуль захисту даних веб-ресурсу / Web resource data protection module**

спеціальність: 123 – Комп'ютерна інженерія  
освітньо-професійна програма – Комп'ютерна інженерія

Кваліфікаційна робота

Виконав: студент групи КІ-41  
Мірецький Володимир Володимирович

Науковий керівник  
к.т.н. доцент Дубчак Л.О.

Кваліфікаційну роботу допущено  
до захисту:

"\_\_" \_\_\_\_\_ 20\_\_ р.

Завідувач кафедри  
\_\_\_\_\_ О.Л. Дубчак

ТЕРНОПІЛЬ - 2024

## АНОТАЦІЯ

Кваліфікаційна робота на тему «Модуль захисту даних веб-ресурсу» зі спеціальності 123 «Комп'ютерна інженерія» освітнього ступеня «бакалавр» містить 75 сторінок пояснюючої записки, 7 рисунків, 9 таблиць, 5 додатків. Обсяг графічного матеріалу 2 аркуші формату А3.

Метою даної кваліфікаційної роботи є вивчення методів і засобів розробки веб-сайтів, що використовуються в мережі Інтернет, а також проектування та розробка програмного забезпечення для захисту інформації в глобальній мережі Інтернет.

Методи дослідження включають методи веб-технологій, засобів управління контентом веб-сайту, розробка схеми проекту та його впровадження.

Завданнями даної кваліфікаційної роботи є побудова структурної схеми веб-сайту; розробка функціональної структури сайту; побудова структури бази даних; розробка алгоритмів роботи системи; створення сайту; інформаційне наповнення сайту; створення авторизації сайту; розробка модуля кодування та декодування сторінок на сайті; реалізація управління зв'язком між користувачами; проведення наукових досліджень на місцях; створення карти сайту; імпорт коментарів і контактів в базу даних; веб-майстер.

Ключові слова: ВЕБ-САЙТ, МОДУЛЬ ЗАХИСТУ ІНФОРМАЦІЇ, АУТЕНТИФІКАЦІЯ КОРИСТУВАЧІВ.

## ANNOTATION

The qualification paper on the topic "Web resource data protection module" from specialty 123 "Computer engineering" of the bachelor's degree contains 75 pages of explanatory note, 7 figures, 9 tables, 5 appendices. The amount of graphic material is 2 sheets of A3 format.

The method of this qualification work is the study of methods and means of developing websites that are used on the Internet, as well as the design and development of information protection software on the global Internet.

Research methods include web technology methods, website content management tools, project design and implementation.

The tasks of this qualification work are the construction of the structural scheme of the website; development of the functional structure of the site; construction of the database structure; development of system operation algorithms; website creation; information content of the site; creation of site authorization; development of a module for encoding and decoding pages on the site; implementation of communication management between users; conducting scientific research in the field; creating a site map; import of comments and contacts into the database; webmaster

**Keywords:** WEBSITE, INFORMATION PROTECTION MODULE, USER AUTHENTICATION.

.

## ЗМІСТ

Вступ.....	8
1 Аналіз сучасних технологій створення веб-сайтів.....	10
1.1 Сучасні технології створення веб-сайтів .....	10
1.2 Аналіз засобів керування вмістом веб-сайтів .....	14
1.3 Формування вимог і постановка задачі .....	23
2 Структура та алгоритм функціонування веб-сайту .....	24
2.1 Структура веб-сайту .....	24
2.2 Структура бази даних .....	29
2.3 Алгоритми функціонування веб-сайту .....	32
3 Програмна реалізація модулів та тестування системи .....	34
3.1 Функціональна структура програмного забезпечення .....	34
3.2 Програмна реалізація модуля авторизації .....	36
3.3 Налаштування системи керування вмістом веб-сайту .....	41
Висновки.....	44
Список використаних джерел.....	45
Додаток А Процес створення веб-сайту .....	48
Додаток Б Схема аутентифікації користувачів .....	49
Додаток В Код надсилання запиту на відновлення токенів .....	50
Додаток Г Техніко-економічне обґрунтування проекту .....	52
Додаток Д Світлокопія тез .....	75

					КР.КІ. 07126/20. 00.00.000 ПЗ			
Змн.	Арк.	№ докум.	Підпис	Дата	<b>МОДУЛЬ ЗАХИСТУ ВЕБ-РЕСУРСУ</b>	Літ.	Арк.	Акрушів
Розроб.		Мірецький В.В.				н	7	75
Перевір.		Дубчак Л.О.				ЗУНУ.ФКІТ.КІ-41		
Консуль.		Савка Н.Я.						
Н. Контр.		Дубчак Л.О.						
Затверд.		Дубчак Л.О.						



## ВСТУП

В останні роки Інтернет-технології різко розвинулися. В Інтернеті розміщено понад 100 мільйонів професійних веб-сайтів і так званих домашніх сторінок. Ця тенденція пов'язана з різким зниженням цін на доступ до Інтернету, а також різким зниженням цін на придбання доменних імен і оплату послуг хостингу [1].

Будь-який бізнес, особливо пов'язаний з інформаційними технологіями, не може обійтися без електронної пошти, FTP і, звичайно ж, Інтернету. Всесвітня павутина дозволяє швидко знайти всю необхідну інформацію про діяльність компанії. Тому сьогодні мати власний сайт – це вже не фантазія, а необхідна умова успішного бізнесу.

Його можна використовувати не тільки в рекламних цілях, а й для організації форуму для обговорення діяльності компанії та якості послуг чи продуктів, або з метою замовлення через Інтернет.

Сьогодні існує проблема розробки спеціалізованого модуля з питань захисту інформації відповідно до сучасних потреб, оскільки у всесвітній мережі Інтернет ресурсів є дуже багато конфіденційної інформації користувачів. Тому захист інформації користувачів сайту є актуальним.

Збільшення кількості користувачів Інтернету в інформаційному просторі зумовлює розвиток технологій створення сайтів та інформаційних порталів. Крім того, все більше користувачів Інтернету хочуть отримати конфіденційну інформацію, в тому числі від компаній-конкурентів.

Метою даної кваліфікаційної роботи є вивчення методів і засобів розробки веб-сайтів, що використовуються в мережі Інтернет, а також проектування та розробка програмного забезпечення для захисту інформації в глобальній мережі Інтернет.

Об'єкт дослідження - сайти з користувальницькими модулями.

					ДП.КСМ.07126/20.00.00.000 ПЗ	Арк.
						8
Змн.	Арк.	№ докум.	Підпис	Дата		

Предметом дослідження є модуль захисту інформації веб-ресурсів.

При розробці веб-сайту необхідно пройти всі етапи, а саме аналіз завдань, вибір веб-технологій, засобів управління контентом веб-сайту, розробка схеми проекту та його впровадження.

Завданнями даної кваліфікаційної роботи є побудова структурної схеми веб-сайту; розробка функціональної структури сайту; побудова структури бази даних; розробка алгоритмів роботи системи; створення сайту; інформаційне наповнення сайту; створення авторизації сайту; розробка модуля кодування та декодування сторінок на сайті; реалізація управління зв'язком між користувачами; проведення наукових досліджень на місцях; створення карти сайту; імпорт коментарів і контактів в базу даних; веб-майстер.

					ДП.КСМ.07126/20.00.00.000 ПЗ	Арк.
						9
Змн.	Арк.	№ докум.	Підпис	Дата		

# 1 АНАЛІЗ СУЧАСНИХ ТЕХНОЛОГІЙ СТВОРЕННЯ ВЕБ-САЙТІВ

## 1.1 Сучасні технології створення веб-сайтів

Перед початком створення веб-сайту важливо визначитися з вибором веб-технології для його організації. Якщо ви створюєте сайт тільки за допомогою HTML, вам доведеться розробити велику кількість сторінок, що призведе до статичності сайту та обмежених можливостей. Однак, динамічні сайти, де контент сторінок формується за допомогою скриптів, мають значно більшу функціональність та легше оновлюються.

Сьогодні найпопулярнішими інструментами для динамічної розробки сайтів є PHP, MySQL та JavaScript. PHP - це вбудована мова веб-сценаріїв, яка працює на сервері та дозволяє швидко і ефективно створювати динамічні веб-додатки. Синтаксис PHP запозичує елементи з інших мов, таких як Perl, Java та C++. До важливих можливостей PHP належать підтримка регулярних виразів, робота з масивами, функції, файлові системи, об'єктно-орієнтовані методи та широка підтримка роботи з базами даних.

Основне застосування PHP - це серверні сценарії, що дозволяє виконувати всі функції, які реалізують інші програми CGI, наприклад, обробка даних форми, створення динамічних

сторінок, або робота з файлами cookie. Втім, PHP може виконувати ще багато інших завдань.

При написанні сценаріїв PHP може бути надзвичайно корисним інструментом для створення та керування динамічним контентом у поєднанні з JavaScript, стилями, WML (Wireless Markup Language) та іншими мовами програмування. Завдяки сотням вбудованих функцій PHP здатний вирішувати майже будь-яке завдання, яке постає перед розробником.

PHP підтримує функції для створення та обробки діаграм, математичних обчислень, електронної комерції та інтеграції з популярними

					ДП.КСМ.07596/05.00.00.000 ПЗ	Арк.
						10
Змн.	Арк.	№ докум.	Підпис	Дата		



технологіями, такими як XML (Extensible Markup Language), ODBC (Open Database Connectivity) та Macromedia Shockwave.

Однією з головних переваг PHP є те, що код PHP можна вбудовувати безпосередньо в HTML, що дозволяє програмістам уникати написання складних програм з багатьма командами лише для створення HTML-коду.

Головним принципом розробки мови PHP є її практичність. PHP забезпечує програмістам можливість швидко та ефективно виконувати поставлені завдання. Практичний характер PHP визначається п'ятьма ключовими характеристиками: традиційністю; простотою; ефективністю; безпекою; гнучкістю.

Ще одна важлива перевага PHP полягає в тому, що він безкоштовний. Багато конструкцій PHP взято з мов C та Perl, і часто сам код PHP схожий на код, написаний на C, C++ або Pascal. Це робить його легким для освоєння та розуміння.

PHP-скрипти можуть бути дуже короткими або містити тисячі рядків коду, залежно від завдання. Механізм PHP просто починає виконувати код після першої керуючої послідовності. Ефективність є надзвичайно важливою при програмуванні багатокористувацьких систем.

PHP підтримує розподіл ресурсів, об'єктно-орієнтоване програмування та керування сесіями. Остання версія включає механізм підрахунку посилань, який запобігає надмірному використанню пам'яті.

PHP надає розробникам і адміністраторам гнучкі та ефективні інструменти безпеки, які поділяються на системного та прикладного рівнів. PHP може працювати в безпечному режимі, який обмежує можливості користувача. Наприклад, можна обмежити максимальний час виконання та використання пам'яті, щоб запобігти перевантаженню сервера.

Адміністратори можуть встановлювати обмеження на каталоги, де дозволено виконувати PHP-скрипти, а також використовувати PHP для захисту конфіденційної інформації на сервері. Стандартний набір функцій

					ДП.КСМ.07596/05.00.00.000 ПЗ	Арк.
						11
Змн.	Арк.	№ докум.	Підпис	Дата		

PHP включає надійні механізми шифрування. PHP також сумісний з багатьма сторонніми програмами, що спрощує інтеграцію з безпечними технологіями електронної комерції. Крім того, вихідний код PHP-скриптів не можна переглянути в браузері, оскільки скрипти компілюються перед відправкою користувачу.

MySQL є однією з найпопулярніших СУБД для веб-сайтів. Хоча вона не призначена для роботи з дуже великими обсягами даних, вона оптимально підходить для як великих, так і малих веб-проектів завдяки високій швидкості, надійності та гнучкості. Підтримка MySQL автоматично включена в дистрибутив PHP. Важливим фактором є також те, що MySQL є безкоштовним. MySQL розповсюджується на умовах GNU General Public License (GPL). Взаємодія з базою даних здійснюється за допомогою системи керування базами даних (СУБД), яка декодує запити та виконує операції з інформацією бази даних. Тому правильніше говорити про запити до СУБД і взаємодію СУБД з веб-скриптом [5]. Існують наступні різновиди баз даних:

- ієрархічна;
- реляційна;
- об'єктно-орієнтована;
- гібридна.

Ієрархічні бази даних використовують деревовидну структуру для зберігання інформації, що робить їх схожими на файлову систему комп'ютера. У реляційних базах даних інформація організована в таблиці, що складаються зі стовпців і рядків, де на перетині знаходяться комірки. Запити до такої бази даних повертають таблицю, яку можна використовувати в наступних запитах. Дані в різних таблицях часто взаємопов'язані, що пояснює назву "реляційні".

В об'єктно-орієнтованих базах даних інформація зберігається у вигляді об'єктів, що зручно для об'єктно-орієнтованого програмування. Однак, вони

					ДП.КСМ.07596/05.00.00.000 ПЗ	Арк.
						12
Змн.	Арк.	№ докум.	Підпис	Дата		

поки що не досягли такої популярності, як реляційні бази даних, через нижчу продуктивність.

Гібридні СУБД поєднують можливості реляційних і об'єктно-орієнтованих баз даних.

Основні характеристики реляційних баз даних включають:

- дані зберігаються в таблицях, що складаються зі стовпців і рядків;
- на перетині кожного стовпця і рядка знаходиться лише одне значення;
- кожен стовпець має ім'я, яке є його назвою, і всі значення в одному стовпці мають один тип (наприклад, у стовпці `id_forum` всі значення можуть бути цілочисельними, а в стовпці `name` – текстовими);
- стовпці розташовані в певному порядку, визначеному при створенні таблиці, тоді як рядки можуть розташовуватися довільно. У таблиці може не бути жодного рядка, але обов'язково має бути хоча б один стовпець;
- запити до бази даних повертають результат у вигляді таблиць, які також можуть бути об'єктами запитів.

Індекс — це впорядкований список значень полів, призначений для прискорення пошуку в базі даних. Унікальний індекс — це список значень, де кожне значення унікальне. Наприклад, у таблиці з паспортними даними для поля "номер паспорта" можна створити унікальний індекс, оскільки кожен номер паспорта унікальний. Первинний ключ є прикладом унікального індексу, який використовується для унікальної ідентифікації записів у таблиці, тому жодні два записи не можуть мати однакове значення первинного ключа. Первинний ключ скорочено називається РК і використовується для організації зв'язків між таблицями.

Наприклад, у таблиці тем первинний ключ називається `id_theme`, а в таблиці публікацій – `id_post`. Поле `id_theme` присутнє в обох таблицях, і в

					ДП.КСМ.07596/05.00.00.000 ПЗ	Арк.
						13
Змн.	Арк.	№ докум.	Підпис	Дата		

таблиці публікацій воно виступає як зовнішній ключ, що посилається на первинний ключ у таблиці тем. Зовнішні ключі скорочено позначають як FK. Як видно на рисунку, зовнішній ключ вказує на первинний ключ таблиці тем, встановлюючи логічний зв'язок один-до-одного між елементами таблиць тем і публікацій.

Первинні ключі можна розділити на логічні (природні) та сурогатні (штучні). Логічний ключ вибирається з даних, що природно ідентифікують запис, наприклад, номер паспорта в базі даних паспортів. Якщо природного ключа немає, використовується сурогатний ключ.

Альтернативний ключ — це додаткове поле в базі даних, що може слугувати як первинний ключ. Нормалізація схеми бази даних — це процес усунення надмірності даних, що знижує ймовірність помилок, зменшує використання дискового простору тощо.

Інший тип динамічної сторінки можна створити за допомогою JavaScript. Особливістю такого підходу є те, що сторінка повністю завантажується в браузері, а потім певні дії можуть виконуватися без підключення до сервера.

## 1.2 Аналіз засобів керування вмістом веб-сайтів

Сьогодні існує багато систем управління контентом для веб-сайтів, таких як Joomla, Drupal, Byteflow, MODx, Zope. Система керування вмістом (CMS) — це програмне забезпечення, яке організовує веб-сайти або інші джерела інформації в Інтернеті чи на персональному комп'ютері.

Byteflow — це блогова платформа з відкритим вихідним кодом, що розповсюджується за ліцензією BSD. Вона написана на Python за допомогою веб-фреймворку Django.

					ДП.КСМ.07596/05.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		14

### Основні функції Byteflow:

- підтримка різних мов розмітки для написання статей: Markdown, BBCode, простий текст, HTML;
- використання заданої в налаштуваннях мови розмітки для коментарів;
- деревовидні коментарі;
- надсилання та прийняття pfgbnsd;
- автоматична реєстрація користувача під час написання першого запиту;
- можливість коментаторам налаштовувати свій профіль;
- теги для статей, перегляд повідомлень за тегами;
- перегляд статей;
- OpenID-аутентифікація з підтримкою кількох OpenID для одного користувача;
- OpenID-сервер;
- імпорт повідомлень з бази даних Wordpress;
- підтримка Gravatar (сервіс аватарів);
- підтримка XFN у блогрол-посиланнях;
- блок посилань для додавання у соціальні каталоги посилань;
- опціональна капча у формі коментування з підтримкою сервісу reCaptcha.

Zope — це об'єктно-орієнтований сервер додатків, написаний на Python. Він розроблений на основі власної ліцензії ZPL з відкритим кодом. Zope, зокрема його версія Zope2, зазвичай використовується як система керування контентом (CMS). Однак для цього був створений програмний фреймворк CMF (Content Management Framework) - набір бібліотек для створення видавничої системи на базі Zope. На основі CMF створено видавничу систему Plone.

Zope має багато корисних функцій для розробників: база даних об'єктів містить не лише вміст, зображення, спеціальні дані, але й динамічні шаблони

					ДП.КСМ.07596/05.00.00.000 ПЗ	Арк.
						15
Змн.	Арк.	№ докум.	Підпис	Дата		

HTML/XML, сценарії, інструменти пошуку, запити до реляційних баз даних (RDBMS) та програмний код.

Zore забезпечує надійну модель веб-розробки, що дозволяє вносити зміни до веб-сайту з будь-якої точки світу завдяки глибоко інтегрованій моделі безпеки, яка базується на концепції «безпечного делегування контролю». Це дозволяє делегувати контроль над певними частинами веб-сайту іншим організаціям або особам. Модель транзакцій застосовується як до бази даних об'єктів Zore (ZODB), так і до конекторів багатьох реляційних баз даних, забезпечуючи потужну інтеграцію даних. Ця модель транзакцій працює автоматично, гарантуючи успішне збереження всіх даних, поки хтось не відновить транзакцію за допомогою веб-браузера або іншої клієнтської програми.

Для розширення базових функцій побудови сайту існує багато плагінів (продуктів за термінологією Zore). Це можуть бути об'єкти зі спеціальним вмістом, конектори до реляційних баз даних або зовнішніх даних, розширені засоби управління вмістом, програмні рішення для e-commerce, документообігу, управління вмістом. Zore має власні HTTP, FTP, WEBDAV і XML-RPC сервіси, але також може використовуватися разом з Apache або іншими серверами.

Zore був створений корпорацією Zore. Частина Zore стали відкритими у 1996 році. Все серверне програмне забезпечення стало доступним у вигляді відкритого коду у 1998 році. Бачення Zore Corporation щодо розробки Zore залишається головним пріоритетом для всієї спільноти розробників. Проте з кожною новою версією Zore все більше розробників з усього світу вносять цінні зміни, пропонують вдосконалення системи та перевіряють нововведення.

Існує спеціальний сайт для розробників (Zore Development Site), де можна знайти детальну документацію поточних проєктів, а також пропозиції щодо майбутнього та аналіз минулого системи Zore. Наприкінці 2005 року

					ДП.КСМ.07596/05.00.00.000 ПЗ	Арк.
						16
Змн.	Арк.	№ докум.	Підпис	Дата		

вийшла нова версія Zore — Zore3. Це повністю перероблена версія, розробка якої тривала кілька років і ґрунтується на досвіді попередніх версій Zore. З випуском Zore3 розробники отримали потужний сервер додатків, який дозволяє створювати не лише системи управління контентом (CMS), а й складніші системи, такі як автоматизація бізнес-процесів і системи документообігу.

MODx — система управління контентом з відкритим вихідним кодом та відкритою ліцензією, написана на мові програмування PHP і використовує для зберігання даних СУБД MySQL.

Основні характеристики MODx:

- інтегровано механізм функціонування;
- підтримка дружніх URL;
- керування інтерфейсами користувачів;
- поділ прав доступу менеджерів і веб-користувачів;
- можливість створення груп користувачів;
- можливість створення груп документів з обмеженим доступом для відвідувачів і менеджерів;
- зручна структура документів у вигляді дерева;
- можливість публікації і зняття з публікації документів у вказаний час;
- підтримка документів різних форматів (MIME);
- можливість створення додаткових параметрів різних типів;
- повний контроль над HTML-кодом для реалізації будь-якого дизайну;
- вбудований SEO-інструментарій (керування ключовими словами і мета-тегами);
- інтуїтивно зрозумілий інтерфейс системи управління контентом (CMS);
- висока розширюваність (за допомогою плагінів і модулів);
- простота мов шаблонів.

					ДП.КСМ.07596/05.00.00.000 ПЗ	Арк.
						17
Змн.	Арк.	№ докум.	Підпис	Дата		

У стандартному наборі CMS пропонує кілька дуже гнучких розширень, таких як створення кількох списків сторінок із розбивкою на сторінки. Дозволяє відображати як звичайний список документів у вигляді посилань, так і список, відсортований за певним ключовим словом. Дозволяє використовувати будь-які додаткові параметри та відображати ім'я автора публікації, кількість коментарів, ціну товару тощо.

Wayfinder – органайзер меню. Надає можливість впорядковувати як прості меню, так і багаторівневі розкриті меню. Легко змінити. eForm – керує роботою форм в CMS. Він може просто надіслати введену інформацію поштою або обробити її для отримання результатів.

Jot – механізм зворотного зв'язку з користувачем. Breadcrumbs – створить ланцюжок посилань на сайті.

AjaxSearch – пошук веб-сайтів за допомогою Ajax. MODx поширюється за ліцензією GNU/GPL, що дозволяє використовувати цю CMS безкоштовно без обмежень і дозволяє працювати з відкритим кодом.

Вимоги:

- PHP 4.3.10 або старше. PHP 5 характеризується простотою;
- MYSQL 4.1.x або старше [7].

Drupal – система управління сайтом (CMS), написана на мові PHP, яка використовує як сховище даних реляційну базу даних (підтримуються, PostgreSQL та інші). Drupal захищена ліцензією.

Drupal пропонує наступні рішення:

- механізми категоризації. Кожен документ на сайті може бути частиною однієї або кількох категорій. Самі категорії можуть формувати списки або складні ієрархічні структури з численними предикатами і взаємопосиланнями елементів;
- готові рішення для типових завдань. Новинний сайт, блог або форум – це можливості, що надаються вбудованими модулями системи керування контентом сайту. інтеграція всіх компонентів.

					ДП.КСМ.07596/05.00.00.000 ПЗ	Арк.
						18
Змн.	Арк.	№ докум.	Підпис	Дата		



- можлива наскрізна категоризація для всіх типів документів на сайті (наприклад, спільний список ключових слів для форумів і блогів). Форум з відображенням цікавих новин на головній сторінці або новинний сайт з блогами і форумами виглядатимуть як єдиний сайт, а не як окремі сторінки, пов'язані лише загальним дизайном;

- навігація і пошук. Книга сайту дозволяє групувати цікаві матеріали, які варто виділити і зберегти для майбутнього. Для зручного доступу до архівних матеріалів служать категоризація контенту і доступ до архіву за датою публікації. Документи завжди мають постійні посилання (перманентні посилання). Крім того, за допомогою коротких посилань і псевдонімів сайт отримує короткі та легко запам'ятовувані посилання на розділи та окремі сторінки, які не містять спеціальних символів і добре індексуються пошуковими системами.

Таксономія (класифікація) — оригінальна концепція визначення структури веб-сторінки, що відокремлює структуру від представлення [7]. Використовуючи таксономію, ви можете визначити довільну кількість тем, за якими будуть розміщені майбутні елементи сайту. Вони можуть бути представлені у вигляді плоских списків або довільних вкладених ієрархій (як у вигляді дерева, коли елемент має лише одного батька в ієрархії, так і необов'язково, коли елемент може мати декілька батьківських елементів одночасно).

В результаті ми отримуємо таку діаграму: документів (вузлів) різних типів (node types) пов'язані з темами (термінами), які, в свою чергу, розділені на словники. Така схема дає можливість побудувати на ділянці кілька незалежних структур, пов'язуючи одні і ті ж документи з різними структурами.

Theme - контент веб-сайту в Drupal відокремлений від дизайну. Використовуючи різні теми, ви можете змінювати дизайн, не впливаючи на вміст і структуру веб-сайту.

					ДП.КСМ.07596/05.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		19

Drupal не прив'язаний до механізму розгортання теми, розробник сайту може вибрати найбільш зручні способи створення дизайну сайту. Але Drupal пропонує кілька готових рішень у вигляді движків тем, таких як xtemplate (тема з шаблонами XML) або phptemplate (тема з шаблонами PHP).

Однак інколи значних змін у дизайні можна досягти шляхом маніпулювання стилями.

Коментарі – будь-який документ сайту відвідувачі можуть коментувати. Для анонімних відвідувачів передбачені поля для їх ідентифікації (ім'я, email, адреса сайту), але ця можливість також відключається.

Ведення блогів – популярне інтернет-хобі під назвою «ведення блогів» також доступне в Drupal. Ведення журналів, можливість коментування їх іншими користувачами, а також підтримка через XML-RPC зовнішніх програм-клієнтів для нотаток в журналах. Site book (книга для співпраці) – ідея дещо схожа на Wiki, де кожен користувач може додавати та редагувати текст. Під час розгортання Drupal документи можуть проходити через чергу модерації.

Більшість матеріалів веб-сайту можна включити як сторінки книги. Наприклад, це можна використовувати для архівування дискусій на форумі, де найцікавіші дискусії можна зібрати в архівну книгу. При цьому документи в книзі залишаються в своїх частинах, як і раніше.

Форум - має інтегровану систему управління контентом сайту форуму, яка добре вписується в структуру всього сайту. Оскільки структура форумів базується на таксономії, категорії та вкладені форуми можна створювати довільно. Основні функції форуму включені, але немає інструменту масової модерації, тому у випадку створення активних сторінок форуму з сотнями тисяч або мільйонів повідомлень, форуми Drupal будуть непрактичними - в цьому випадку це краще використовувати система керування вмістом форуму (наприклад, phpbb).

					ДП.КСМ.07596/05.00.00.000 ПЗ	Арк.
						20
Змн.	Арк.	№ докум.	Підпис	Дата		

Offline News (Content Syndication) - Drupal експортує більшість документів у RDF/RSS, Atom та деякі інші, що зручно для перегляду в автономному режимі за допомогою зовнішніх програм або для автоматичної публікації на інших сторінках.

Колекція новин – Новини надаються як RDF/RSS з інших сайтів і можуть збиратися Drupal і відображатися на цьому сайті.

Призначення прав доступу - на основі призначення однієї або кількох ролей користувачам права доступу до різних функцій веб-сайту призначаються безпосередньо ролям. Модулі визначають, які з їхніх функцій надають доступ до певних ролей.

Чіткі, логічні діаграми, включені в стандартний дистрибутив Drupal, достатні для більшості завдань. Вбудовані скрипти - в Drupal можна включити скрипти PHP в будь-який документ на веб-сайті. У сценарії ви можете отримати доступ до функцій API Drupal — це дозволяє швидко створювати динамічні сторінки з функціями, які не надає механізм, без необхідності писати окремі модулі.

Інтернаціоналізація - Drupal надає різні механізми для інтернаціоналізації та локалізації зовнішнього вигляду (але не вмісту) веб-сайту. Зокрема, можливість редагування перекладів через веб-інтерфейс (для перекладів, що зберігаються в базі).

Ще одна приємна особливість архітектури Drupal — використання Unicode. Весь текст зберігається у форматі UTF-8, новини та повідомлення електронної пошти передаються в такому ж кодуванні. Така уніфікація дозволяє публікувати документи на сайті різними мовами, не вимагаючи зміни кодування в браузері.

Drupal не є універсальною CMS: функціональність його окремих частин може бути кращою або гіршою порівняно з іншими продуктами. Вибір залежить від конкретних завдань, які стоять перед розробником сайту. Зрозумівши це, розробники CMS Drupal не прагнули створити універсальний

					ДП.КСМ.07596/05.00.00.000 ПЗ	Арк.
						21
Змн.	Арк.	№ докум.	Підпис	Дата		

продукт, а зосередилися на сайтах, орієнтованих на спілкування, таких як блоги, форуми та засоби колективної розробки. Однак це не обмежує використання Drupal лише цими завданнями, оскільки додаткові модулі значно розширюють його можливості. У розробках з відкритим кодом лише масовий попит і бажання самих розробників можуть призвести до появи нових функцій. [9].

Joomla – це система керування контентом, написана на мовах PHP та JavaScript, яка використовує базу даних MySQL для збереження даних. Joomla – це безкоштовне програмне забезпечення, захищене ліцензією GPL.

Основні переваги Joomla:

- Joomla CMS повністю функціонує на базі даних, використовуючи PHP та MySQL;
- Багаторівневий доступ для зареєстрованих користувачів до адміністративної та фронтальної частин сайту;
- Зручна структура розділів та категорій для ефективного організування контенту;
- Легке створення та редагування матеріалів;
- Контент може додаватися адміністраторами та менеджерами через адміністративну частину, а також публікаторами, редакторами, авторами та звичайними зареєстрованими користувачами через фронтальну частину сайту;
- Повне керування блоками меню та модулів, зокрема відображення останніх новин, популярних матеріалів, пошуку, форми авторизації та голосування;
- Редагування матеріалів за допомогою візуального редактора, подібного до текстових редакторів, таких як Word;
- Підтримка програмних продуктів інших розробників;

					ДП.КСМ.07596/05.00.00.000 ПЗ	Арк.
						22
Змн.	Арк.	№ докум.	Підпис	Дата		

- Працює на системах Linux, FreeBSD, WinNT, Win2K, MacOSX, Solaris, AIX, SCO та інших.

### 1.3 Формування вимог і постановка задачі

Головною метою цього дипломного проекту є дослідження методів і засобів розробки веб-сайтів, що використовуються в Інтернеті, а також розробка програмного забезпечення для веб-сайту "Захист інформації в Інтернеті". У більшості випадків цільовою аудиторією можуть бути клієнти, тренери та лікарі, які можуть зацікавитися сайтом, тому важливо захищати їхні конфіденційні дані. На веб-сайті буде представлена структурована інформація про методи, форми, засоби, загрози та проблеми захисту інформації в мережі.

Форум на сайті дозволить обговорювати різні питання, пов'язані з захистом інформації, а будь-який матеріал, розміщений на сайті, може бути прокоментований. Таким чином, під час розробки буде відображена інформація про структуру сайту, роботу скриптів шифрування, можливість пошуку, реєстрацію, форум, контактну інформацію, зареєстрованих користувачів, коментарі та карту сайту.

У цьому розділі буде представлений огляд відомих інструментів і технологій для створення веб-сайтів, а також аналіз доступних рішень і ресурсів на основі Joomla CMS. Висновок зроблено на підставі аналізу, що в Інтернеті дуже мало ресурсів, присвячених захисту інформації. Тому виникає необхідність розробки спеціалізованого веб-сайту, що відповідає сучасним потребам і дозволяє обмінюватися досвідом щодо захисту конфіденційної інформації.

					ДП.КСМ.07596/05.00.00.000 ПЗ	Арк.
						23
Змн.	Арк.	№ докум.	Підпис	Дата		

## 2 СТРУКТУРА ТА АЛГОРИТМ ФУНКЦІОНУВАННЯ ВЕБ-САЙТУ

### 2.1 Структура веб-сайту

Архітектурою програмної системи можна вважати її внутрішню організацію, яка включає в себе представлення програмного продукту як системи, що складається з різних взаємодіючих частин. Зазвичай ці частини відображаються у вигляді окремих модулів. Розробка архітектури є важливим етапом, оскільки вона дозволяє розбити програмну систему на окремі блоки для полегшення розробки. Цей підхід сприяє кращому розумінню всієї системи, її складових та взаємозв'язків між ними.

Основними завданнями розробки архітектури програмної системи є визначення програмних підсистем та відображення їхніх зовнішніх функцій, а також визначення способів взаємодії між цими підсистемами. З урахуванням прийнятих на цьому етапі рішень проводиться подальше уточнення та функціональна специфікація.

Розробка структури веб-сайту охоплює все, що стосується його вмісту та логічної організації. Хоча користувача перш за все цікавить зовнішнє оформлення та враження від сайту, надійна та ретельно продумана схема створить основу для подальшого розвитку та підтримки сайту. Проте важливо пам'ятати, що без якісного інформаційного наповнення жоден сайт не зможе стати популярним.

CMS Joomla відрізняється чітким розділенням інформації на контент і дизайн. Контент може бути представлений статтями або статичним вмістом, а для організації статей Joomla пропонує ієрархічну структуру з розділами і категоріями. Узагальнена структура інформаційного наповнення сайту в CMS виглядає так: Розділ→Категорія→Стаття.

Нижче наведено приклад структури інформаційного наповнення веб-сайту "RAPID health optimization".

					ДП.КСМ.07596/05.00.00.000 ПЗ	Арк.
						24
Змн.	Арк.	№ докум.	Підпис	Дата		

THE RAPID HEALTH PROTOCOL INCLUDES:



A health analysis of thousands of data points and hundreds of biomarkers to decode every aspect of your health and performance.



Personalized protocols designed to optimize your nutrition, supplementation, training, sleep, recovery, and lifestyle.



Customized, confidential, and hassle-free testing done at home, on your schedule.



With a full team of operators, coaches, and subject matter experts led by Program Directors Dr. Andy Galpin and Dan Garner to get you feeling and performing your best.

Рисунок 2.1 – Структура інформаційного наповнення сайту

Отже, всередині кожного розділу виділяються категорії, до яких прив'язуються статті. Кожна стаття обов'язково повинна бути призначена до якоїсь категорії, але лише до однієї. Розділ, в свою чергу, для того, щоб вміщувати статті, повинен мати хоча б одну категорію.

Почати наповнення нового сайту контентом слід зі створення інформаційної структури з розділів і категорій.

При створенні нового розділу необхідно вказати його заголовок, назву, рівень доступу і статус публікації. Також можна додати зображення та детальний опис до розділу. Заголовок буде використовуватися у публічній частині сайту для позначення розділу, а назва – у інтерфейсі суперадміністратора. Заголовок і назва можуть бути однаковими.

					ДП.КСМ.07596/05.00.00.000 ПЗ	Арк.
						25
Змн.	Арк.	№ докум.	Підпис	Дата		

Процес створення категорії аналогічний створенню розділу: потрібно вказати заголовок, назву та інші параметри. Однак особливість полягає в можливості вибору розділу, до якого відноситься нова категорія.

Розділ "Менеджер медіагалереї" у інтерфейсі адміністратора призначений для керування зображеннями, що розміщуються на сайті. Перед тим, як додавати зображення до статей, їх слід завантажити на сайт. Зручно організувати медіафайли у папках на сервері, наприклад, зображення для ілюстрації статей можна зберігати в папці "stories".

За допомогою "Менеджера користувачів" суперадміністратор може додавати нових користувачів до системи, вказавши їхні дані та призначивши їх до відповідних груп (зарєєстрованих, авторів, менеджерів тощо). Блокуючи користувача, суперадміністратор може заборонити йому доступ до системи, а також видалити користувача.

Під меню в CMS Joomla розуміється ієрархія "пунктів", яка пов'язана з елементами контенту. Система меню відображає структуру розділів, категорій і статей у навігаційному меню. Joomla дозволяє створювати різні меню і керувати їх відображенням на сайті.

Управління меню здійснюється з "Менеджера меню" у інтерфейсі адміністратора. На сторінці з переліком пунктів меню можна налаштовувати різні параметри, такі як стан публікації, сортування, рівень доступу та ідентифікатори елементів.

У Joomla існує чітко визначений набір типів елементів, які можуть бути пов'язані з пунктом меню. Наприклад, пункт "Відео контент" відноситься до компонента CMS "Система автентифікації користувача", тому при кліці на цей пункт завжди відкриватиметься вікно автентифікації користувача сайту.

Нижче наведено приклад структури меню веб-сайту "RAPID health optimization" (рисунок 2.2).

					ДП.КСМ.07596/05.00.00.000 ПЗ	Арк.
						26
Змн.	Арк.	№ докум.	Підпис	Дата		



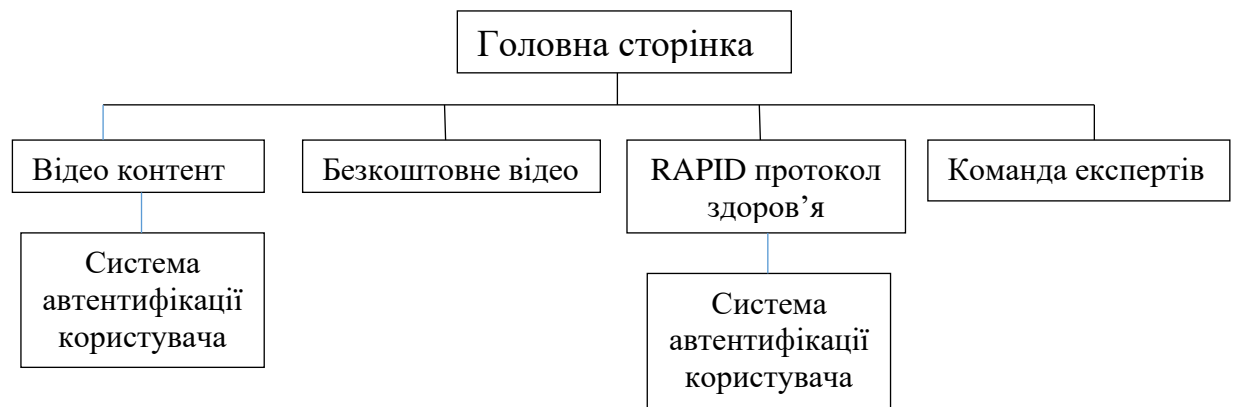


Рисунок 2.2 – Структура веб-сайту

При створенні нового пункту меню необхідно обрати тип елемента. Кожен тип пунктів відрізняється тим, які дані відображаються при кліці на пункт меню і який результат відображається. Наприклад, пункт "Безкоштовне відео" відображає безкоштовне коротке відео про протокол здоров'я RAPID. А пункт "Команда експертів" містить посилання на всіх експертів, представлених на веб-сайті.

Для CMS Joomla розроблені додаткові компоненти та модулі. Компоненти є ключовими елементами функціональності Joomla і зазвичай розміщуються в основній частині шаблону сайту, тобто mainbody.php. Модулі розширюють можливості Joomla і можуть бути розміщені в будь-якій частині сайту.

Управління модулями та компонентами завжди проводиться через панель адміністратора, яка включає в себе внутрішній (back-end) інтерфейс.

Завантажений у базу даних контент Joomla відображається з використанням шаблонів оформлення. Шаблони служать формою для вмісту, і вони можуть бути дуже легко встановлені всередині CMS Joomla за допомогою функцій управління шаблонами.

Встановлення шаблону можна здійснити через головне меню в адміністраторському інтерфейсі, використовуючи опцію "Встановити\знищити→шаблон".

Структура веб-сайту "RAPID" показана в додатку А і включає такі основні модулі:

- Головне меню;
- Перехід на відео контент для зареєстрованих користувачів;
- Опис протоколу здоров'я;
- Вхід/реєстрація користувачів;
- Меню експертів проекту.

Головне меню забезпечує швидкий перехід між розділами сайту, а модуль "автентифікація користувачів" відображає дані про зареєстрованих користувачів. Зареєстровані користувачі поділяються на дві категорії:

- Зовнішні користувачі (Front-end Users);
- Внутрішні користувачі (Back-end Users).

Зовнішні користувачі отримують певні привілеї над гостями, такі як можливість створювати запити на сайті. Внутрішні користувачі включають менеджерів, адміністраторів і суперадміністраторів, які мають доступ до зовнішнього та внутрішнього інтерфейсів сайту і різні рівні привілеї.

Є два способи, за якими гості можуть стати зареєстрованими користувачами сайту: вони можуть самостійно зареєструватися або суперадміністратор може додати їх в систему через адміністративну панель. Форма входу за замовчуванням містить посилання для реєстрації, яке може бути вимкнено суперадміністратором.

Модуль "вхід/реєстрація" показує зареєстрованого користувача, його активність, статус, останній візит, оновлення профілю та інші дані. Він взаємодіє з модулем користувача, який дозволяє надсилати повідомлення та файли зареєстрованим користувачам.

Наповнення, вигляд і сприйняття веб-сайту, його графічний інтерфейс – це перші речі, які бачить користувач. При розробці важливо брати до уваги вимоги та інтереси користувачів на різних рівнях.

З цієї причини на сайті рекомендується розмістити такі блоки:

					ДП.КСМ.07596/05.00.00.000 ПЗ	Арк.
						28
Змн.	Арк.	№ докум.	Підпис	Дата		

- Назва веб-сайту;
- Опис діяльності;
- Доступ до відео контенту;
- Опис запропонованого протоколу здоров'я;
- Вхід/реєстрація;
- Інформація про експертів.

Для кожного елемента сторінки необхідно визначити стиль форматування, включаючи шрифт, кольори, фон, вирівнювання і т.д.

Таким чином, в цьому розділі визначено структуру програмної системи, основні модулі і візуальну структуру сайту, що дозволить оптимально відображати інформацію, організувати інтерфейс користувача та полегшити обслуговування.

## 2.2 Структура бази даних

База даних представляє собою організовану структуру даних, керування якою здійснюється системою управління базами даних (СУБД). СУБД оптимізує доступ до даних та їхню маніпуляцію для швидкого та ефективного використання, у відмінну від текстових файлів, де дані зберігаються без структури. СУБД дозволяє швидко виконувати пошук та маніпулювати великими обсягами даних.

Основу проекту будь-якої бази даних складає аналіз вимог та її проектування. Вдалі бази даних рідко формуються автоматично, і часто виникають проблеми, такі як надмірність. Процес проектування бази даних розпочинається з визначення вимог, а потім переходить до створення даталогічної моделі та графічного представлення моделі. Наступним етапом є

					ДП.КСМ.07596/05.00.00.000 ПЗ	Арк.
						29
Змн.	Арк.	№ докум.	Підпис	Дата		

реалізація бази даних, під час якої важливо правильно вибрати типи атрибутів таблиць.

База даних, що є невід’ємною частиною Joomla, є добре продуманою та структурованою. Вона успішно виконує поставлені завдання.

Структура бази даних складається з різних таблиць, кожна з яких виконує свою функцію. Таблиці можна розділити на:

- 7- Загальна інформація про налаштування та структуру веб-сайту;
- Інформаційне наповнення сайту;
- Встановлені модулі та компоненти;
- Конфігурація модулів та компонентів.

База даних складається з взаємопов’язаних таблиць, атрибути яких доцільно привести до третьої нормальної форми, щоб уникнути дублювання даних та забезпечити зручність їх редагування. Ключові поля в таблицях є унікальними та використовуються як ідентифікатори рядків. При проектуванні бази даних важливо визначити оптимальні типи та розміри полів для економії пам’яті.

Таблиця 2.1 — Структура таблиці jos\_content

Поле	Тип MySQL	Атрибути	По замовчуванні
* id	int(11)	UNSIGNED	NULL, auto_increment
Title	varchar(255)		
Alias	varchar(255)		
title_alias	varchar(255)		
introtext	mediumtext		
fulltext	mediumtext		
State	tinyint(3)		0
sectionid	int(11)	UNSIGNED	0
Mask	int(11)	UNSIGNED	0

Продовження таблиці 2.1

Catid	int(11)	UNSIGNED	0
created	datetime		0000-00-00 00:00:00
created_by	int(11)	UNSIGNED	0
created_by_alias	varchar(255)		
modified	datetime		0000-00-00 00:00:00
modified_by	int(11)	UNSIGNED	0
checked_out	int(11)	UNSIGNED	0
checked_out_time	datetime		0000-00-00 00:00:00
publish_up	datetime		0000-00-00 00:00:00
publish_down	datetime		0000-00-00 00:00:00
images	text		
Urls	text		
attribs	text		
version	int(11)	UNSIGNED	1
parentid	int(11)	UNSIGNED	0
ordering	int(11)	UNSIGNED	0
metakey	text		
metadesc	text		
access	int(11)	UNSIGNED	0
Hits	int(11)	UNSIGNED	0
metadata	text		

Однією з ключових таблиць є jos\_components. Ця таблиця відведена для зберігання інформації про компоненти, які встановлені на сайті, а також їхнє розташування. Давайте розглянемо поле enabled – це поле використовується для зберігання даних про те, чи слід відобразити компонент на сайті.

В базі даних також є інші таблиці, які містять налаштування компонентів або модулів, і вони мають схожі назви з компонентами або модулями відповідно.

Як видно з вище наведених таблиць, більшість атрибутів мають текстовий (char, varchar) або числовий (int, tinyint) формат. Для великих обсягів тексту використовується двійковий тип даних (blob). Великі двійкові дані, такі як фотографії або архіви, зберігаються на жорсткому диску, а в базі даних зберігаються лише посилання на ці файли.

### 2.3 Алгоритми функціонування веб-сайту

Виходячи з аналізу структури сайту та бази даних, який був проведений у попередніх розділах, потрібно розробити алгоритми, за якими будуть формуватися сторінки.

Генерація веб-сторінок повинна відповідати візуальній структурі. Тому процес формування сторінки має відбуватися крок за кроком, залежно від того, який елемент дизайну потрібно відобразити.

Основні етапи у процесі генерації сторінок включають наступні:

- Завантаження конфігураційного файлу.
- Встановлення з'єднання з базою даних.
- Перевірка режиму адміністрування.
- Завантаження каскадних таблиць стилів (CSS).
- Формування та відображення верхньої частини сторінки.
- Створення меню.
- Завантаження модулів відповідно до обраної категорії.
- Відображення контактної інформації.

					ДП.КСМ.07596/05.00.00.000 ПЗ	Арк.
						32
Змн.	Арк.	№ докум.	Підпис	Дата		

Якщо користувач обирає пункт з головного меню, відображається інформація відповідного розділу. У разі переходу до режиму адміністрування, спочатку перевіряється доступ до цього розділу. Потім, залежно від введених даних, відбувається оновлення (додавання, зміна, видалення) інформації розділу, і показується результат або формується запитувана сторінка. Якщо користувач не ідентифікований або не має потрібних прав, виводиться повідомлення про помилку.

Крім того, розділи веб-сайту поділяються на ті, що зберігають інформацію в базі даних, і ті, дані яких містяться у HTML-файлах на жорсткому диску або зчитуються з бази даних (тобто статичні та динамічні сторінки).

Алгоритм формування вмісту розділів, які взаємодіють з базою даних, включає такі етапи:

- Визначення вхідних параметрів.
- Запит інформації з бази даних.
- Формування та відображення результатів.

					ДП.КСМ.07596/05.00.00.000 ПЗ	Арк.
						33
Змн.	Арк.	№ докум.	Підпис	Дата		

## 3 ПРОГРАМНА РЕАЛІЗАЦІЯ МОДУЛІВ ТА ТЕСТУВАННЯ СИСТЕМИ

### 3.1 Функціональна структура програмного забезпечення

Розглянемо функціональну схему програмного забезпечення. Для зручності розділимо його на дві складові: інтерфейс користувача та систему керування веб-сайтом.

Спочатку розглянемо інтерфейс користувача та принципи його створення. Програмний інтерфейс користувача реалізується у файлах `index.php` та `template.css`, які розташовані у папці з шаблоном нашого сайту.

Панель управління дозволяє отримати доступ до всіх доступних функцій внутрішнього інтерфейсу. Надані параметри меню можуть відрізнитися залежно від статусу користувача, наприклад менеджера, адміністратора або суперадміністратора. Додаткові компоненти та модулі доступні онлайн для таких систем керування.

Компоненти є основними елементами функціональності Joomla. Зазвичай вони розміщуються в основному тексті шаблону веб-сайту, тобто `mainbody.php`. Модулі розширюють можливості Joomla.

Модуль — це невеликий фрагмент вмісту, який можна розмістити будь-де на веб-сайті, якщо це дозволяє шаблон. Важливою особливістю Joomla CMS є чітке розмежування інформаційної складової сайту між контентом і дизайном. Вміст — це важливий вміст веб-сайту. Наприклад: зміст статті, ілюстрації до статті, список ділових контактних телефонів, список публікацій.

Дизайн є формою зовнішнього контенту. Наприклад, тип і колір шрифту, який використовується для відображення тексту; графічне оформлення сторінок; логотип компанії в лівому верхньому куті; Елементи HTML визначають структуру тексту на сторінці [10].

					ДП.КСМ.07596/05.00.00.000 ПЗ	Арк.
						34
Змн.	Арк.	№ докум.	Підпис	Дата		



Алгоритм роботи системи можна розділити на наступні етапи:

- підключення файлу index.php з кореневого каталогу сайту;
- підключення конфігураційних файлів;
- підключення файлу-шаблону.

Файл index.php є головним на сайті, він структурує та наповнює контентом веб-сайт. Він відповідає за розміщення елементів контенту та їх відображення. Цей файл створюється веб-дизайнером і визначає місце для кожного об'єкта та меню на сайті.

Перший файл, який підключається - це defines.php, в якому описуються константи, що визначають шляхи до папок сайту.

Файл framework.php використовується для перевірки конфігураційних файлів, наявності Joomla та завантаження бібліотек та конфігураційного файлу configuration.php.

Файл configuration.php містить основні константи, такі як параметри підключення до бази даних, параметри хоста та електронна адреса адміністратора сайту.

Файл index.php в папці templates відповідає за вигляд сайту. Зміни в цьому файлі дозволяють змінювати розташування елементів на веб-сторінці.

Файл template.css відповідає за стильове оформлення, таке як розмір, колір та розміщення тексту, параметри меню та кольори фону. Тут задаються параметри відображення окремих елементів, наприклад, оформлення меню та зовнішній вигляд посилань.

Далі розглянемо можливості користувача. Є головна сторінка, з якої можна перейти до розділу безкоштовного відеоконтенту уроків та особистого протоколу здоров'я.

Авторизація дозволяє зареєструватись на сайті, керувати зв'язками з іншими користувачами, редагувати власний профіль, писати відгуки та коментарі від власного імені.

					ДП.КСМ.07596/05.00.00.000 ПЗ	Арк.
						35
Змн.	Арк.	№ докум.	Підпис	Дата		

Розглянемо структуру програмного забезпечення адміністраторської частини сайту. Основний файл, який обробляє частину адміністратора - index.php у папці administrator.

Цей файл описує константи та підключає відповідні файли:

'defines.php' - містить опис констант;

'framework.php' - відповідає за підключення основних файлів вмісту веб-сайту;

'helper.php' - викликає допомогу;

'toolbar.php' - описує функції панелі інструментів.

Розглянемо папку administrator та її вміст:

- backups - для зберігання резервних копій;
- cache - для кешування даних;
- components - для зберігання компонентів та їх налаштувань;
- help - містить матеріали допомоги по керуванню Joomla;
- images - зображення іконок частини адміністратора;
- includes - бібліотеки додаткових функцій;
- language - файли мов сайту;
- modules - модулі та їх налаштуван

### 3.2 Програмна реалізація модуля авторизації

При створенні веб-сайту «RAPID health optimization» була реалізована сторінка авторизації користувача. На головній сторінці сайту для отримання доступу до відео-матеріалу користувач використовує кнопку «Watch free lab analysis and performance protocol» (рисунок 3.1).

					ДП.КСМ.07596/05.00.00.000 ПЗ	Арк.
						36
Змн.	Арк.	№ докум.	Підпис	Дата		

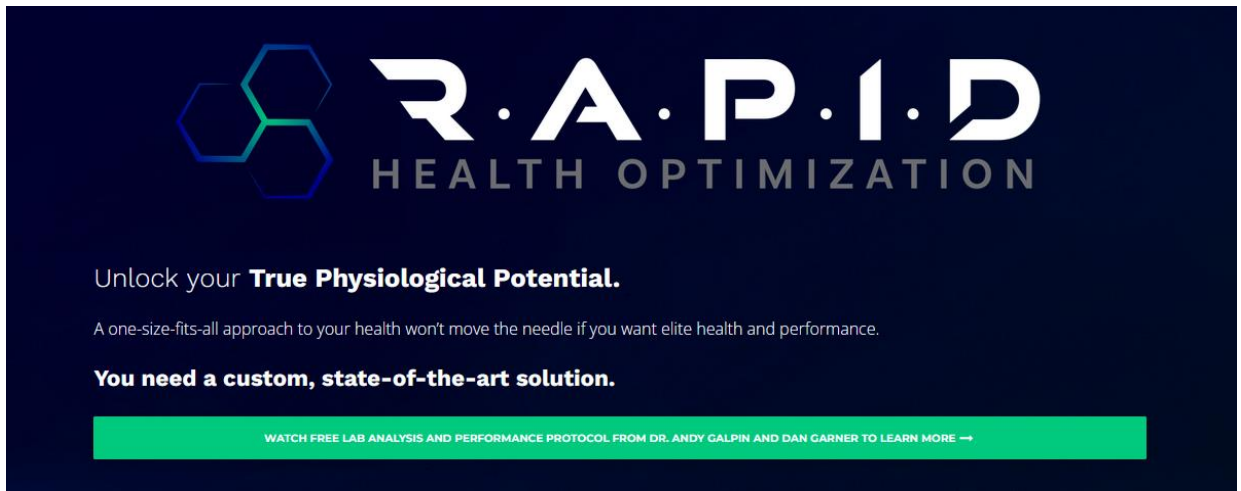


Рисунок 3.1 – Кнопка доступу до відео-контенту

При натисканні кнопки впливає вікно реєстрації (рисунок 3.2).

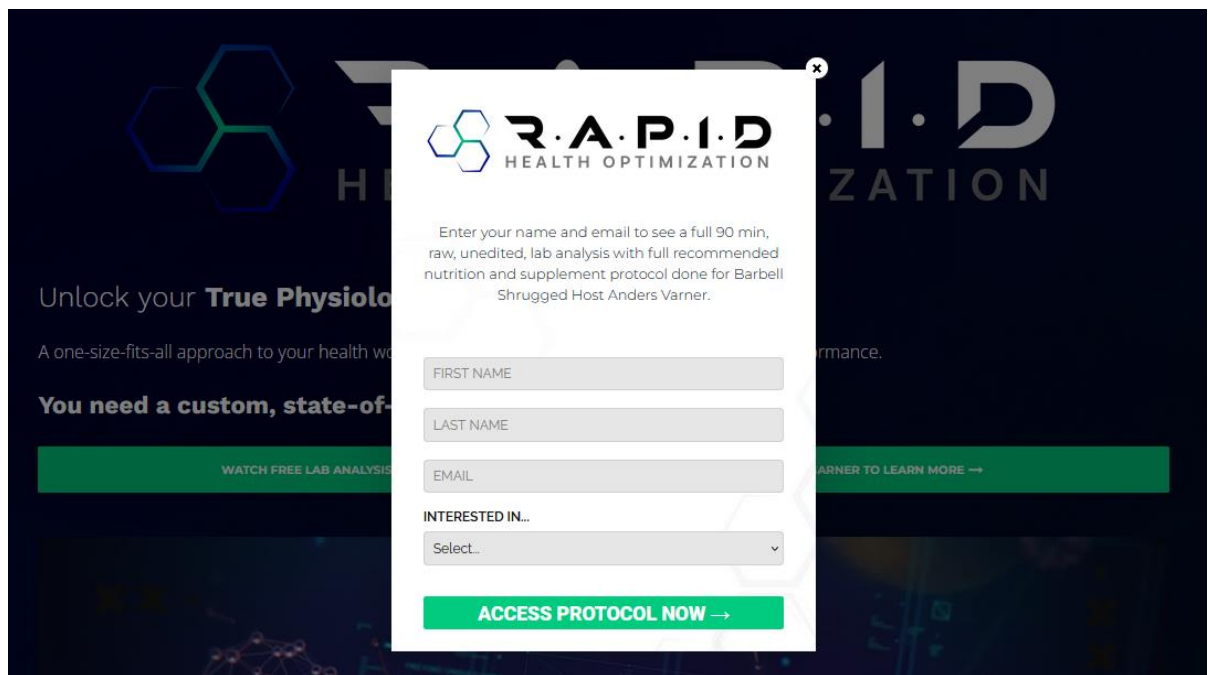


Рисунок 3.2 – Вікно реєстрації користувача

Для перевірки користувача виконується наступний код:

```
const submitHandler = async (values: values, formikHelpers:
FormikHelpers<any>) => {
  try {
```

						ДП.КСМ.07596/05.00.00.000 ПЗ	Арк.
							37
Змн.	Арк.	№ докум.	Підпис	Дата			

```

const { access_token, refresh_token } = await axios({
  method: 'post',
  url: API_AUTH_ROUTES.LOGIN,
  body: values,
});

setAccessRefreshCookies(access_token, refresh_token);
} catch (e) {
};

```

Коли користувач натискає на кнопку "Log in" (рисунок 3.3) цей код надсилає дані з полів логін форми на сервер. Якщо email і пароль є вірними, то в результат отримуються два токена (access\_token, refresh\_token) та записуються в cookie.

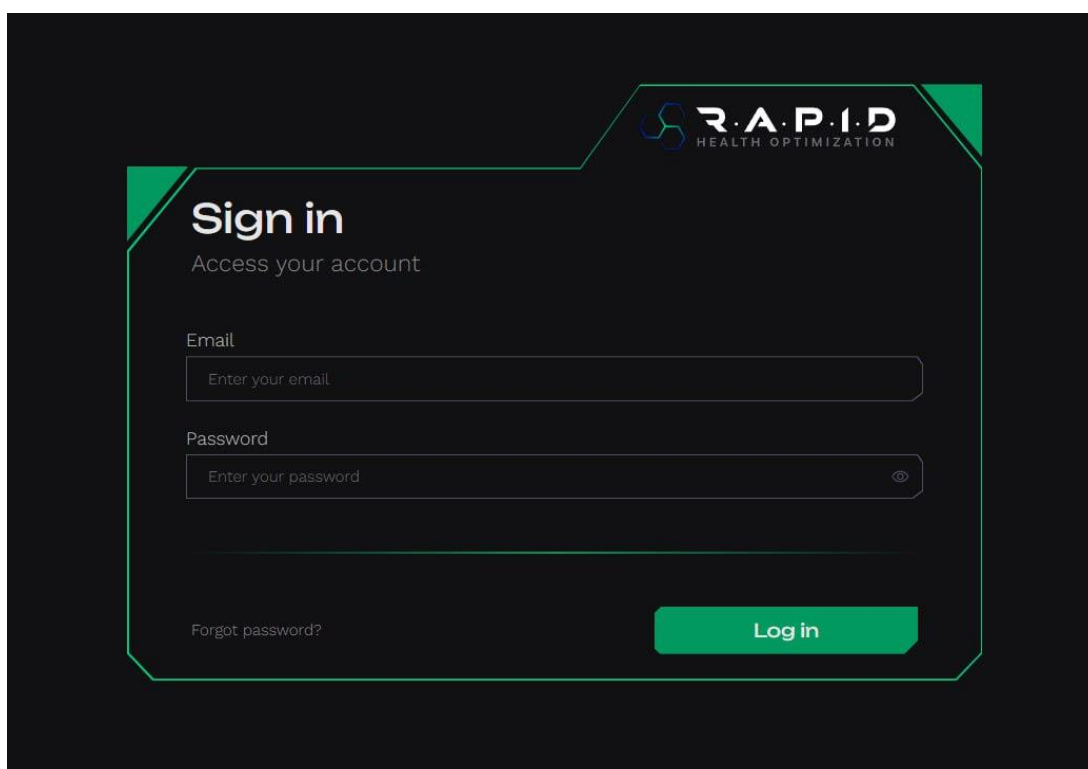


Рисунок 3.3 – Вікно автентифікації користувача

access\_token - це ключ спілкування клієнта з сервером. Записується він в хедер запити. Сервер в свою чергу отримавши цей запит перевіряє чи токен дійсний, якщо так, то дає відповідь з тими даними, задля яких робили реквест. Таким чином сервер є захищеним від запитів клієнта, в якого немає відповідного access\_token.

Логіка з реквестом даних та записом access\_token відбувається при ініціалізації інстанса axios в його інтерцепторі:

```
const axiosInstance = axios.create({
  baseURL: API_URL,
});
// Request interceptor
axiosInstance.interceptors.request.use(getConfig, (error) => {
  return Promise.reject(error);
});
```

refresh\_token використовується для того, щоб оновити цю пару токенів. access\_token є валідним протягом короткого періоду часу, для прикладу 5хв. refresh\_token в свою чергу може бути валідним 30 діб. Коли access\_token стає недійсним, сервер про це повідомляє і робиться новий запит на відновлення access\_token та refresh\_token. Ключем цього запиту вже виступає refresh\_token. Як би добре не був зашифрований токен доступу до сервера, все одно, можна знайти алгоритм його злому, тому існують оці часові рамки, які знижують можливість створення такого алгоритму, чим вони коротші, тим надійніше. Логіка з надсиланням запити на відновлення токенів знаходиться в інтерцепторі (додаток В).

Акаунт користувача, який проходить протокол тренувань виглядає наступним чином (рисунок 3.4):

					ДП.КСМ.07596/05.00.00.000 ПЗ	Арк.
						39
Змн.	Арк.	№ докум.	Підпис	Дата		

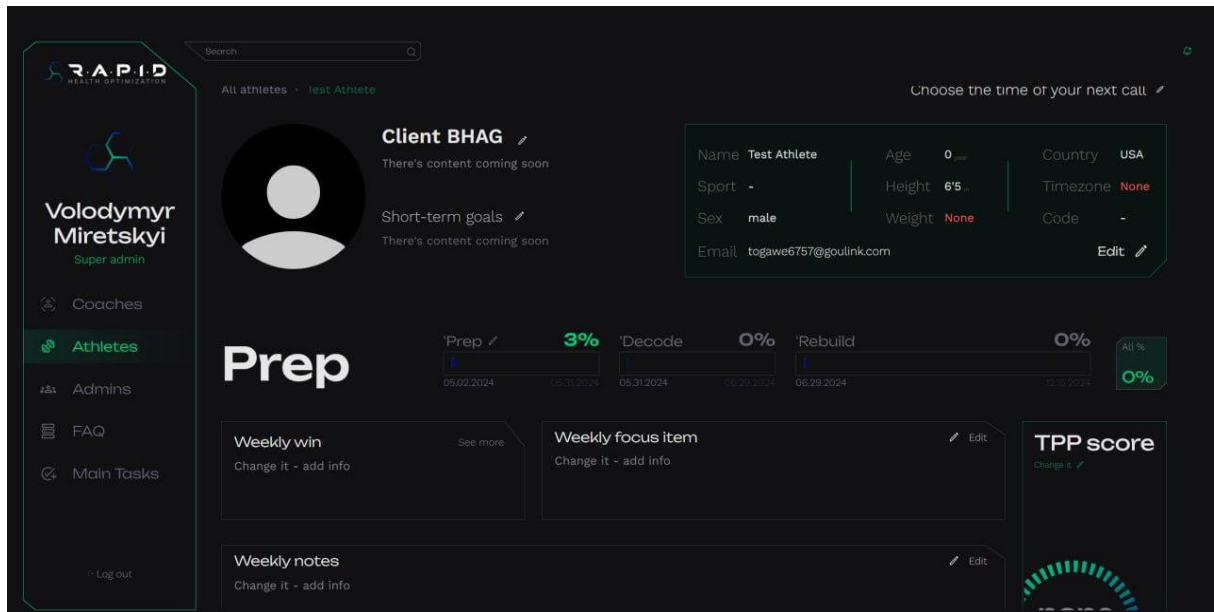


Рисунок 3.4 – Вікно акаунту користувача

Тренер (лікар) має вищі привілеї доступу, ніж звичайний користувач. Він може переглядати прогрес користувача, здійснювати корекцію програми тренувань, залишати коментарі. Для цього тренер здійснює доступ до бази даних користувача, яка в його профілі має вигляд звичайних папок (рисунок 3.5), що дуже зручно для перегляду.

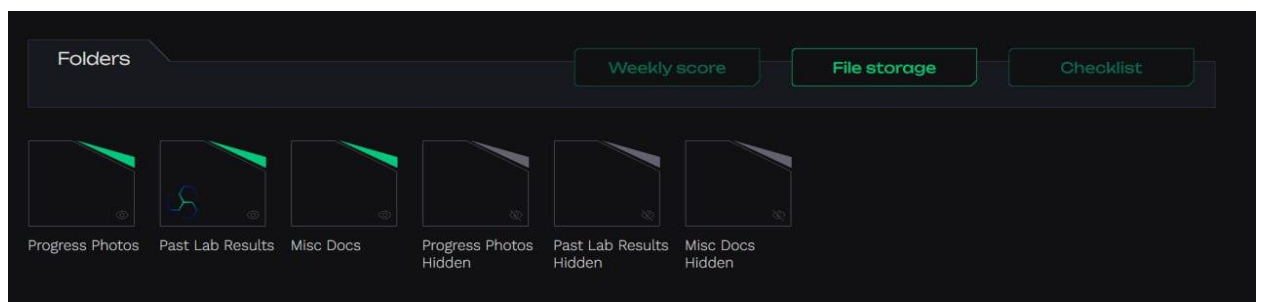


Рисунок 3.5 – База файлів користувача

					ДП.КСМ.07596/05.00.00.000 ПЗ	Арк.
						40
Змн.	Арк.	№ докум.	Підпис	Дата		

### 3.3 Налаштування системи керування вмістом веб-сайту

Перед початком встановлення та налаштування системи керування контентом веб-сайту необхідно переконатися, що локальний сервер відповідає таким системним вимогам:

- PHP 4.2.x або вище;
- MySQL 3.23.x або вище;
- Apache 1.13.19 або вище.

PHP повинен мати вбудовану підтримку MySQL, XML і Zlib.

Налаштування системи керування контентом веб-сайту включає збереження її параметрів та вмісту. Для цього необхідно створити базу даних на сервері MySQL. Префікс таблиць за замовчуванням – jos\_. Ім'я хоста залежить від налаштувань, наданих провайдером хостингу, зазвичай це localhost. Для захисту бази даних від несанкціонованого доступу створюється користувач з паролем.

Під час налаштування системи потрібно вказати e-mail суперадміністратора для зв'язку з користувачами та вибрати обробник електронної пошти для надсилання листів із сайту.

Опція Site Offline дозволяє включати і вимикати зовнішній інтерфейс сайту під час оновлення модулів або інформації, щоб користувачі не могли відвідувати сайт до завершення робіт. У разі, якщо сайт знаходиться в офлайн-режимі, можна показати повідомлення: «Сайт зараз закритий на технічне обслуговування. Будь ласка, зайдіть пізніше».

Під час створення вмісту сайту можна додавати компоненти, модулі, плагіни, шаблони. Система керування вмістом веб-сайту дозволяє розробнику вибрати який з тестових редакторів буде використовуватися при редагуванні об'єктів контенту та кількість елементів в списках, які відображаються у внутрішньому (backend) інтерфейсі.

					ДП.КСМ.07596/05.00.00.000 ПЗ	Арк.
						41
Змн.	Арк.	№ докум.	Підпис	Дата		

Коли налаштовується систему управління вмістом веб-сайту, потрібно ввести опис сайту, який використовується пошуковою системою. Деякі пошукові системи рекомендують обмежити опис до 20 слів або менше. Опис має бути коротким і безпосередньо пов'язаним зі змістом сайту. Ви також можете використовувати ключові слова або словосполучення в описі. Обмеження в 20 слів використовується не всіма пошуковими системами, тому опис може бути більш докладним. Проте, основну його частину краще викласти в 20 словах.

Під час перегляду вмісту можна ввімкнути або виключити відображення мета-Тегу Title цільового тегу та мета-тегу Author цільового тегу у вихідному коді[10].

Для підвищення продуктивності сайту необхідно налаштувати кеш. Щоб кеш працював коректно, каталог повинен бути доступний для редагування, повинен бути обраний обробник і вказано максимальний час (в секундах), протягом якого файл буде зберігатися в каталозі кешу перед оновленням.

Система управління вмістом веб-сайту Joomla дозволяє налаштувати власні налаштування та "медіа-галерею".

Коли налаштовуються параметри користувача, супер адміністратор може:

- дозволити або заборонити реєстрацію користувача самостійно;
- заблокувати або видалити користувача;
- встановити рівень доступу для нових користувачів при реєстрації в передній частині сайту;
- встановить активацію облікового запису нового користувача, він отримає повідомлення по електронній пошті з посиланням.

Для активації свого облікового запису дозвольте або не дозволяйте вибрати веб-сайт, який допоможе йому налаштувати свій профіль після входу

					ДП.КСМ.07596/05.00.00.000 ПЗ	Арк.
						42
Змн.	Арк.	№ докум.	Підпис	Дата		



в мовний, візуальний редактор і інтерфейс.;дозволити або заборонити одержувати системну пошту;

Надає користувачеві певний рівень доступу до сайту (реєстрація, автор, редактор, видавець, менеджер і т.д.).

При налаштуванні параметрів медіагалереї суперадміністратор може:

- Встановлювати розширення;
- Встановлювати максимальний розмір завантажуваних файлів (в байтах);
- Дозволити або заборонити використання MIME Magic або Fileinfo для перевірки файлів;
- Встановлювати власні розширення зображень;
- Дозволити або заборонити використання flash downloader для завантаження медіафайлів;
- Обмежити розмір завантажуваних користувачем файлів.

Під час налаштування системи управління вмістом для веб-сайту ви можете встановити тривалість автоматичного виходу користувача, якщо він був неактивним протягом введеного часу, вибрати обробник сеансу, в якому Joomla ідентифікує користувача, як тільки ви підключитесь до веб-сайту за допомогою нестабільних файлів cookie.

Система управління контентом веб-сайту Joomla дозволяє управляти всіма параметрами статей і вибирати мову, що використовується на сайті за замовчуванням.

Якщо мова відрізняється від англійської, необхідно заздалегідь встановити мовний файл за допомогою меню установки.

					ДП.КСМ.07596/05.00.00.000 ПЗ	Арк.
						43
Змн.	Арк.	№ докум.	Підпис	Дата		

## ВИСНОВКИ

В результаті виконання дипломного проекту були реалізовані поставлені завдання:

1) розроблено структурну схему веб-сайту. Розроблена структура програмної системи, визначено основні модулі, а також спроектовано візуальну структуру сайту. Це дозволяє оптимально відображати інформацію на сайті, організувати інтерфейс користувача і полегшити подальше обслуговування сайту;

2) розроблено структуру бази даних, що дало можливість здійснювати розподіл та керування інформацією кожного користувача;

3) наведено алгоритм роботи сценаріїв для отримання інформації з бази даних. Першим кроком є встановлення з'єднання з системою управління базами даних. Для цього використовуються коди та паролі доступу, які зберігаються у файлі конфігурації;

4) реалізовано модуль авторизації користувачів веб-сайту, що дозволило здійснювати керування зв'язками між користувачами, захист конфіденційної інформації та адміністрування;

5) проведено техніко-економічне обґрунтування проекту, що підтверджує його економічну вигоду (додаток Г).

Результати проведеного дослідження апробовано на науково-практичній конференції (додаток Д).

					ДП.КСМ.07596/05.00.00.000 ПЗ	Арк.
						44
Змн.	Арк.	№ докум.	Підпис	Дата		

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Бернерс-Лі Заснування павутини = Weaving the web. The original design and ultimate destiny of the world wide web : З чого починалася і до чого прийде Всесвітня мережа / Тім Бернерс-Лі разом з Марком Фічетті; пер. з англ. А. Іщенко. – Київ : Києво-Могилянська академія, 2007. – 208с.
2. Цеслів О.В. WEB-програмування : навч. посібник / О.В. Цеслів ; М-во освіти і науки, молоді та спорту України, Нац. техн. ун-т України “Київ. політехн. ін-т”. – Київ : НТУУ “КПІ”, 2011. – 296, с. .
3. Шмідт Я. Нова мережа: ознаки, практики і наслідки веб 2.0 = Das Neue Nets Markmale, Praktiken und Folgen des Web 2.0 : посібник для вузів / Ян Шмідт ; [пер. з нім. В. Климченко ; за заг. ред. В. Іванова]. – Київ : Академія Української Преси, Центр Вільної Преси, 2013. – 283
4. Куленко М.Я. Основи графічного дизайну : підручник для студентів вищих навч. закладів / Михайло Куленко; МОНУ; Київський нац. ун-т будівництва і архітектури. – 2-ге вид., виправл. та доп. – Київ : Кондор, 2007. – 492с.
5. Adam D. Scott Building Web Apps for Everyone. Copyright, O’Reilly Media, 2016, 245 p.
6. David Upton. CodeIgniter for Rapid PHP Application Development. Packt Publishing, 2007. 244 p.
7. Casteleyn S., Daniel F., Dolog P., Matera M. Engineering Web Applications. Berlin: Springer-Verlag, 2009, 363p.
8. Cody Lindley. Front-End Developer Handbook 2017. 2017. URL: <https://frontendmasters.gitbooks.io/front-end-handbook-2017/content>.
9. Лось В.О. Web-програмування: методичні рекомендації до виконання лабораторних робіт для здобувачів ступеня вищої освіти

					ДП.КСМ.07596/05.00.00.000 ПЗ	Арк.
						45
Змн.	Арк.	№ докум.	Підпис	Дата		

бакалавра спеціальності «Економіка» освітньо-професійної програми «Економічна кібернетика». Запоріжжя: ЗНУ, 2018, 71 с.

10. Матвієнко О.В., Бородкіна І.Л. Internet - технології: проектування Web-сторінки : навч. посіб. Київ : Альтерпрес, 2003, 132 с.

11. Микитишин А.Г., Митник М.М., Стухляк П.Д., Пасічник В.В. Комп'ютерні мережі : навч. посіб. Львів : Магнолія, 2013, 250 с.

12. Пасічник В. В., Пасічник О. В., Угрин Д. І. Веб-технології : підручник. Львів : Магнолія, 2013, 215 с.

13. Oliver James. Html & CSS is hard (But it doesn't have to be). A friendly web development tutorial for complete beginners. 2017. URL: <https://internetingishard.com/html-and-css>.

14. Jeremy Thomas. MarkSheet. A free HTML and CSS tutorial. 2015-2017. URL: <https://marksheet.io>.

15. Shay Howe. Learn To Code HTML & CSS. Develop & Style Websites. 2014-2017. URL: <https://learn.shayhowe.com/html-css>

16. Shay Howe. Learn To Code Advanced HTML & CSS. Develop & Style Websites. 2014-2017. URL: <https://learn.shayhowe.com/advanced-html-css>.

17. Shklar L., Rosen Wiley R. Web Application Architecture: Principles, Protocols and Practices, 2009 – 440 p.

18. Welling L., Thomson L. PHP and MySQL Web Development (4th Edition), Addison-Wesley Professional, 2008 – 1008 p.

19. Josh Marinacci. HTML Canvas Deep Dive. A Travelogue. 2011. URL: <https://joshondesign.com/p/books/canvasdeepdive/title.html>.

20. Mark Pilgrim. Dive Into HTML5. 2010. URL: <http://diveintohtml5.info/index.html>

21. Harry Roberts. CSS Guidelines. 2017. URL: <https://cssguidelin.es>

22. Mikito Takada. Learn CSS Layout. The Pedantic Way. 2015. URL: <http://book.mixu.net/css/index.html>

23. Козловський А.В., Паночишин Ю.М., Погріщук Б.В. Комп'ютерна техніка та інформаційні технології : навч. посіб. 2-ге вид., стер. Київ : Знання, 2012. 463 с.

24. Манако В. Основи будування сайтів: Науково-виробниче видання / В. Манако, Д. Манако, О. Данилова, О. Войченко. – К.: Вид. дім «Шкіл. світ»: Вид. Л. Галіцина, 2006. – 120 с., [4] арк. – (Б-ка «Шкіл. світу»).

25. Ian Lloyd Own website the right way using HTML & CSS. SitePoint Pty.Ltd, Australia, 2011. – 515 p.

26. Ткаченко В., Шварц Б., Зайцев П. MySQL по максимуму монографія / В. Ткаченко, Б. Шварц, П. Зайцев – Санкт-Петербург: Издательство Питер, 2018. – 20 с.

27. 100+ питань для співбесід та технічних інтерв'ю – бази даних, зокрема SQL (частина 1) [Електронний ресурс] / Електрон. текст. дані. – Режим доступу : <https://www.quality-assurance-group.com/super-puper-shpargalka-100-pytan-dlya-spivbesid-ta-tehnichnyh-intervyu-bazy-danyh-zokrema-sql-chastyna-1/>.

28. Ічанська Н. В. Оптимальний вибір методів організації Інтернет-ресурсів. Системи управління, навігації та зв'язку. Збірник наукових праць. 2019. Т. 3, №55. С. 104–109. URL: <https://doi.org/10.26906/sunz.2019.3.104>

29. Бондаренко О. О., Ластовецький В. В., Пилипчук О. П., Шестопапов Є. А. Інформатика (рівень стандарту): підручник. Харків : Ранок, 2019. 174 с.

30. Web-технології та Web-дизайн: застосування мови HTML для створення електронних ресурсів : навч. посіб. / І. Л. Бородкіна, Г. О. Бородкін.– Київ: Видавництво Ліра-К, 2020. – 212 с.