

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДОНЕЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ВАСИЛЯ СТУСА

МОГИЛА НАЗАР АНАТОЛІЙОВИЧ

Допускається до захисту:
в. о. завідувача кафедри
інформаційних технологій,
канд. техн. наук, доцент
_____ О. В. Зелінська
« ____ » _____ 2024 р.

ЕЛЕКТРОНА СИСТЕМА УПРАВЛІННЯ ДІЯЛЬНІСТЮ КАФЕДРИ

Спеціальність 122 «Комп'ютерні науки»

Кваліфікаційна (магістерська) робота

Науковий керівник:
Т. В. Січко, доцент кафедри
інформаційних технологій,
канд. техн. наук, доцент

(підпис)

Оцінка: ____/____/_____
(бали за шкалою ЄКТС/за національною шкалою)

Голова ЕК: _____
(підпис)

Вінниця 2024

ЗМІСТ

ВСТУП.....	4
РОЗДІЛ 1 ТЕХНОЛОГІЧНИЙ ОГЛЯД ПЛАТФОРМ ДЛЯ УПРАВЛІННЯ КАФЕДРОЮ	8
1.1 Визначення проблеми та мета дослідження	8
1.2 Переваги електронної системи управління	9
1.3 Вимоги до електронної системи управління	11
1.4 Структура електронної системи	12
1.5 Вибір серверної платформи для електронної системи управління діяльністю кафедри	13
1.6 Порівняння методів розробки сайту	15
Огляд популярних CMS	24
Обґрунтування вибору CMS WordPress	26
РОЗДІЛ 2 ВИМОГИ ДО ВЕБ-САЙТУ	29
2.1 Опис серверного середовища та хостингу для WordPress	29
2.2 Забезпечення безпеки веб-сайту	31
2.3 Оптимізація продуктивності сайту	32
РОЗДІЛ 3 ПРАКТИЧНА РЕАЛІЗАЦІЯ ЕЛЕКТРОННОЇ СИСТЕМИ УПРАВЛІННЯ ДІЯЛЬНІСТЮ КАФЕДРИ	35
3.1 Огляд серверного оточення	35
3.2 Встановлення та налаштування CMS WordPress	40
3.3 Оптимізація веб-сайту	50
3.4 Налаштування безпеки веб-сайту	55
ВИСНОВКИ	60
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ	62

АНОТАЦІЯ

Могила Н.А. Електронна система управління діяльністю кафедри. Спеціальність 122 «Комп'ютерні науки», освітня програма «Комп'ютерні технології обробки даних (Data Science)». Донецький національний університет імені Василя Стуса, Вінниця, 2024.

У кваліфікаційній роботі досліджено електронну систему управління діяльністю кафедри. Показано розробку вебсайту для управління діяльністю кафедри. Встановлено підвищення ефективності впровадження інформаційної системи у діяльність кафедри.

Ключові слова: вебсайт, системи управління, WordPress

50 с., 30 рис., 10 джерел.

Mohyla N.A. Electronic system for managing the faculty's activities. Specialty 122 “Computer Science”, Programme “Computer data processing technologies”. Vasyl' Stus Donetsk National University, Vinnytsia, 2024.

The qualification work investigates the information system for managing the activities of the department. The development of a website for managing the activities of the department is shown. The efficiency of the information system implementation in the department's activities is determined.

Keywords: website, management systems, WordPress

50 p., 30 fig., 10 sources.

ВСТУП

Ефективне управління діяльністю кафедри в сучасних умовах є важливим завданням для забезпечення якісного освітнього процесу та належної організації наукової роботи. У цьому контексті запровадження електронних систем управління стає необхідністю, що обумовлено як зростанням обсягу даних, так і потребою в автоматизації рутинних завдань. Використання сучасних інформаційних технологій дозволяє не лише оптимізувати адміністративні процеси, а й створює можливості для інтеграції навчального та наукового потенціалу. Актуальність теми дослідження визначається зростаючою роллю цифрових платформ у сфері освіти, які слугують основою для побудови прозорих, адаптивних і зручних у використанні систем. Особливо важливим є створення таких рішень, що враховують специфічні потреби освітніх установ, їхню організаційну структуру та вимоги до захисту даних.

Об'єктом дослідження є процес управління діяльністю кафедри як невід'ємної частини закладу вищої освіти, що включає планування, контроль та координацію діяльності всіх її структурних елементів.

Предметом дослідження є розробка електронної системи управління діяльністю кафедри, що дозволяє автоматизувати управлінські процеси та покращити взаємодію між її учасниками.

Метою дослідження є створення ефективної електронної системи управління, яка спрощує виконання адміністративних завдань, полегшує доступ до інформації, підвищує продуктивність роботи кафедри та забезпечує її функціонування у відповідності до сучасних вимог. Ця мета визначає необхідність аналізу існуючих методів розробки, вибору оптимального інструментарію для побудови системи та реалізації практичної частини, яка включає створення вебсайту.

Наукова новизна дослідження полягає у створенні інтегрованого підходу до розробки електронної системи управління діяльністю кафедри на основі сучасної CMS WordPress. У роботі здійснено комплексний аналіз переваг використання

цього інструменту для створення спеціалізованих веб-рішень в освітній сфері. Зокрема, було розглянуто специфічні потреби організаційного процесу в університеті, які можуть бути задоволені завдяки впровадженню електронної системи, а також запропоновано оптимальні засоби підвищення її функціональності, безпеки та продуктивності. Отримані результати дозволяють зробити внесок у теорію й практику впровадження автоматизованих систем управління в навчальних закладах.

Практична значущість роботи полягає в можливості впровадження створеного вебсайту в реальну діяльність кафедри. Запропоновані рішення щодо оптимізації продуктивності сайту, забезпечення його безпеки та адаптації до вимог користувачів створюють передумови для ефективного функціонування платформи. Це може сприяти спрощенню управлінських процесів, підвищенню прозорості діяльності кафедри, а також покращенню комунікації між викладачами, студентами. Крім того, отримані результати можуть бути використані як модель для розробки аналогічних рішень у інших підрозділах університету або в установах з подібними потребами.

Дослідження включає порівняння методів розробки вебсайтів, а також практичну реалізацію програмних рішень. На етапі проектування вебсайту використовувалися методи системного аналізу для визначення вимог до системи та вибору відповідного програмного забезпечення. У процесі реалізації рішення застосовувалися практичні методики веброзробки, зокрема, встановлення та налаштування CMS WordPress, інтеграція плагінів та модулів, оптимізація продуктивності та забезпечення безпеки. Це дозволило створити функціональний і зручний у використанні інструмент, що відповідає вимогам сучасних стандартів веброзробки.

Характеристика структури роботи відображає послідовність вирішення поставлених завдань та логіку дослідження. Перша частина роботи присвячена аналізу основних аспектів проблематики, зокрема визначенню актуальності створення електронної системи управління діяльністю кафедри і формулюванню

вимог до майбутньої системи. У цьому розділі також проведено порівняння сучасних методів розробки вебресурсів, здійснено огляд популярних CMS, зокрема їхніх функціональних можливостей, і обґрунтовано вибір платформи WordPress як базової для реалізації проєкту. Цей вибір було аргументовано поєднанням простоти використання, широких можливостей для кастомізації та відповідності заявленим вимогам до сайту.

У другій частині роботи детально розглянуто вимоги до створення вебсайту кафедри. Зокрема, акцент зроблено на особливостях серверного середовища та хостингу, необхідного для стабільної роботи WordPress. Увагу приділено питанням забезпечення безпеки вебсайту, що є критично важливим аспектом для будь-якої електронної системи управління, особливо в освітньому середовищі. Розглянуто шляхи для покращення швидкодії вебсайту, що дозволяють забезпечити швидкість завантаження сторінок і стабільну роботу навіть за умов високого навантаження. Також проаналізовано додаткові інструменти та техніки, які сприяють створенню зручного й інтуїтивно зрозумілого інтерфейсу для кінцевих користувачів.

Третій розділ зосереджений на практичній реалізації розробленого рішення. Він охоплює опис процесу встановлення та налаштування CMS WordPress, інтеграції необхідних модулів і плагінів, а також тестування функціональності системи. Особливу увагу приділено заходам із підвищення безпеки, включаючи налаштування SSL-сертифікатів і регулярне оновлення компонентів системи. У цьому розділі також подано результати оптимізації сайту, зокрема заходи зі скорочення часу завантаження, підвищення стабільності роботи та забезпечення мобільної адаптивності. Це дозволило отримати завершений функціональний продукт, який відповідає як технічним вимогам, так і потребам користувачів.

Таким чином, виконане дослідження довело важливість використання сучасних інформаційних технологій в освітньому процесі. Результати роботи мають як теоретичну, так і практичну значущість. Теоретично вони сприяють подальшому розвитку підходів до управління освітніми установами за допомогою електронних систем. Практично — забезпечують кафедрам ефективний інструмент для

управління, зменшуючи адміністративне навантаження та сприяючи автоматизації рутинних процесів.

Поставлені завдання:

1. Формулювання проблем управління діяльністю кафедри та визначення мети дослідження
2. Дослідження переваг електронних систем управління
3. Розробка вимог до функціональності електронної системи управління
4. Порівняльний аналіз серверних платформ і методів розробки для вибору оптимального рішення
5. Розробка та впровадження вебсайту як складової електронної системи управління кафедрою
6. Узагальнення результатів дослідження

Апробація результатів дослідження. Результати кваліфікаційної (магістерської) роботи апробовано на III-ій Міжнародній науково-практичній конференції «Прикладні аспекти сучасних міждисциплінарних досліджень», яка відбулась 01 листопада 2024 року на базі Донецького національного університету імені Василя Стуса. Тези на тему: «Забезпечення конфіденційності даних на сайтах освітніх установ».

РОЗДІЛ 1

ТЕХНОЛОГІЧНИЙ ОГЛЯД ПЛАТФОРМ ДЛЯ УПРАВЛІННЯ КАФЕДРОЮ

1.1 Визначення проблеми та мета дослідження

Ефективне управління діяльністю кафедри є критично важливим для забезпечення високої якості навчального процесу та наукових досліджень у вищих навчальних закладах. Традиційні методи управління, які здебільшого базуються на паперовому документообігу та ручних процесах, мають численні недоліки, що негативно впливають на продуктивність і ефективність роботи кафедри. Однією з головних проблем є значні часові витрати на ведення документації, планування розкладів, підготовку звітів та інші адміністративні завдання. Ці процедури займають значний час у викладачів та адміністративного персоналу, що може негативно впливати на їх основну діяльність – навчання та наукову роботу.

Окрім цього, відсутність автоматизації призводить до підвищеного ризику помилок у документації та звітності. Ручне введення даних і обробка інформації підвищують ймовірність виникнення неточностей та помилок, що може призвести до непорозумінь та затримок у роботі кафедри. Низька ефективність комунікації між викладачами, студентами та адміністрацією є ще однією проблемою, пов'язаною з традиційними методами управління. Відсутність централізованої платформи для обміну інформацією ускладнює координацію та співпрацю між учасниками навчального процесу.

Також традиційні методи не завжди забезпечують швидкий і зручний доступ до необхідної інформації, що ускладнює процес прийняття рішень. Обмежений доступ до актуальних даних може стати перешкодою для ефективного управління та оперативного реагування на зміни в навчальному процесі. Відсутність інтегрованих систем управління також ускладнює планування та оптимізацію використання ресурсів, таких як аудиторії,

обладнання, час викладачів, що знижує загальну ефективність діяльності кафедри.

Метою дослідження є реалізація електронних систем управління діяльністю кафедри у закладах вищої освіти. Для досягнення цієї мети важливо провести всебічне дослідження основних концепцій, типів та функціональних можливостей електронних систем управління. Це включає вивчення сучасних тенденцій та технологій у сфері електронних систем управління, аналіз останніх розробок та їх вплив на організацію навчального процесу.

Також дослідження передбачає оцінку ефективності впровадження електронних систем управління на прикладі кафедри. Це дозволить виявити переваги та недоліки використання таких систем, зокрема їх вплив на продуктивність, точність документації та комунікацію. На основі отриманих результатів будуть сформульовані практичні рекомендації щодо впровадження та оптимізації електронних систем управління діяльністю кафедри, враховуючи специфіку різних навчальних закладів.

Окрему увагу буде приділено вивченню проблем, з якими стикаються навчальні заклади під час впровадження електронних систем управління, та пропозиції шляхів їх подолання. Ідентифікація та аналіз основних проблем дозволить розробити стратегії для успішного впровадження та ефективного використання електронних систем управління діяльністю кафедри. Це, у свою чергу, сприятиме підвищенню загальної ефективності та якості освіти у вищих навчальних закладах.

1.2 Переваги електронної системи управління

Електронна система управління діяльністю кафедри — це інтегрована інформаційна платформа, яка спрямована на автоматизацію адміністративних і навчально-організаційних процесів кафедри. Вона забезпечує централізоване зберігання та обробку даних, оптимізує управлінські завдання та підвищує загальну ефективність функціонування кафедри.

Ця система має численні переваги, які значно підвищують ефективність організаційних та навчальних процесів у вищих навчальних закладах. Нижче розглянуто основні з них:

Однією з найбільших переваг електронних систем управління є автоматизація рутинних адміністративних завдань. Це включає автоматичне створення та зберігання документації, планування розкладів занять, управління студентськими записами та обробку заявок. Автоматизація дозволяє значно знизити часові витрати на виконання цих завдань, що звільняє час для викладачів та адміністративного персоналу для виконання більш значущих і творчих завдань.

Електронні системи управління суттєво знижують ймовірність помилок, які можуть виникати при ручному введенні даних та обробці документів. Системи можуть автоматично перевіряти дані на відповідність і цілісність, що знижує ризик виникнення помилок та підвищує точність і надійність інформації.

Електронні системи управління забезпечують централізовану платформу для обміну інформацією та комунікації між викладачами, студентами та адміністрацією. Це сприяє швидшому та ефективнішому обміну інформацією, зменшує кількість непорозумінь і затримок у комунікації. Викладачі та студенти можуть швидко отримувати необхідну інформацію, обмінюватися документами та спільно працювати над проектами в режимі реального часу.

Електронні системи управління забезпечують зручний доступ до актуальної інформації в режимі реального часу. Це дозволяє викладачам і адміністрації швидко отримувати необхідні дані для прийняття обґрунтованих рішень. Студенти також можуть отримувати актуальну інформацію про розклад занять, оцінки, завдання та іншу важливу інформацію в будь-який час і з будь-якого місця.

Завдяки автоматизації рутинних завдань та підвищенню точності даних, електронні системи управління сприяють підвищенню загальної продуктивності кафедри. Викладачі можуть більше часу присвячувати навчальному процесу та

науковим дослідженням, а адміністративний персонал – більш ефективно управляти ресурсами та підтримувати організаційну діяльність.

Електронні системи управління дозволяють більш ефективно планувати та використовувати ресурси кафедри, такі як аудиторії, обладнання та час викладачів. Системи можуть автоматично визначати найкращий розклад занять, враховуючи наявність ресурсів та потреби студентів і викладачів. Це сприяє зниженню витрат та оптимізації використання ресурсів.

Запровадження електронних систем управління може сприяти поліпшенню якості навчання завдяки більш ефективному управлінню навчальним процесом та підвищенню оперативності прийняття рішень. Викладачі можуть використовувати сучасні технології для інтерактивного навчання, проведення дистанційних занять та моніторингу успішності студентів. Це дозволяє краще адаптувати навчальний процес до потреб студентів та підвищити їхню мотивацію до навчання.

Електронні системи управління забезпечують високий рівень безпеки даних завдяки використанню сучасних технологій захисту інформації. Це включає шифрування даних, контроль доступу, резервне копіювання та інші заходи безпеки, що дозволяють захистити конфіденційну інформацію від несанкціонованого доступу та втрат.

Загалом, електронні системи управління діяльністю кафедри мають значний потенціал для підвищення ефективності, точності та якості управлінських та навчальних процесів у вищих навчальних закладах. Вони сприяють автоматизації рутинних завдань, знижують ймовірність помилок, покращують комунікацію та забезпечують доступ до актуальної інформації, що робить їх незамінними інструментами для сучасних освітніх установ.

1.3 Вимоги до електронної системи управління

Аналіз потреб кафедри є важливим етапом у розробці електронної системи управління, оскільки дозволяє визначити функціональні та технічні вимоги до

сайту, який буде ефективно виконувати свою роль та відповідати потребам користувачів. Основні розділи сайту включають розділ "Про нас", який містить опис кафедри, склад кафедри та досягнення. У розділі "Інформація для вступника" буде розміщена інформація про напрями навчання, спеціальності, вимоги до вступу та відгуки студентів. "Інформація для студента" включатиме документи, рейтингові списки, розклад занять. У розділі "Інше" будуть представлені відео, фото, презентації та інформація про конференції. Розділ "Контакти" міститиме контактну інформацію, форму зворотного зв'язку.

Додаткові вимоги до сайту включають мультимовність для зручності іноземних студентів та викладачів, інтерактивний календар подій з інформацією про майбутні заходи, розділ новин та анонсів, аналітику та звітність для збору та аналізу даних про відвідуваність сайту, мобільну версію сайту для зручного користування на мобільних пристроях, а також систему опитувань та анкетування для збору зворотного зв'язку.

Технічні вимоги включають забезпечення безпеки та конфіденційності даних користувачів, використання SSL-сертифікатів, захист від атак типу XSS та CSRF, забезпечення швидкого завантаження сторінок за допомогою кешування та оптимізації бази даних, гарантію коректної роботи сайту в усіх популярних веб-браузерах, можливість інтеграції з корпоративними системами університету, такими як системи обліку, LMS, ERP, та автоматичне створення резервних копій сайту для забезпечення відновлення даних у разі аварії. Аналіз потреб кафедри дозволяє створити електронну систему управління, яка не лише забезпечує виконання основних функцій, але й відповідає вимогам сучасності, забезпечуючи зручність та ефективність для всіх користувачів.

1.4 Структура електронної системи

Електронна система управління діяльністю кафедри складається з інтегрованих компонентів, які забезпечують ефективне функціонування всіх адміністративних і навчально-організаційних процесів. Основою системи є

вебсайт, що відповідає за відображення й збереження даних, взаємодію між різними підсистемами та користувачами.

Функціональна частина системи включає інтерфейс для різних викладачів і студентів. Викладачі можуть використовувати систему для створення та редагування розкладу, контролю виконання навчальних планів, внесення інформації про результати студентів, а також для доступу до навчальних матеріалів. Студенти отримують можливість переглядати розклад, оцінки, а також залишати запити чи звернення до викладачів через форму зворотного зв'язку.

Система функціонує на основі серверного середовища, оптимізованого для обробки великого обсягу запитів. Особлива увага приділяється захисту даних: використання SSL/TLS шифрування, розширена автентифікація. Ці міри мінімізують ризик несанкціонованого доступу або втрати інформації. Для забезпечення швидкодії та зручності користування застосовуються оптимізаційні механізми, як от кешування й використання сучасних форматів для мультимедійного контенту, що в результаті скорочує час завантаження.

1.5 Вибір серверної платформи для електронної системи управління діяльністю кафедри

При розробці та впровадженні електронної системи управління діяльністю кафедри важливим аспектом є вибір серверної платформи. Основними варіантами для цього є Windows Server та Linux. Обидві платформи мають свої переваги та недоліки, однак у більшості випадків Linux є кращим вибором для серверної платформи. Розглянемо детальніше особливості обох платформ та причини, чому слід обрати саме Linux.

Windows Server є комерційною операційною системою, яка широко використовується в корпоративному середовищі. Вона відрізняється зручним графічним інтерфейсом, високою сумісністю з програмним забезпеченням Microsoft і підтримкою великої кількості комерційних програм. Основні переваги Windows Server включають:

- Інтуїтивно зрозумілий графічний інтерфейс спрощує налаштування та управління сервером.
- Висока сумісність з іншими продуктами Microsoft, такими як MSSQL Server, Exchange Server та Active Directory.
- Велика кількість комерційних програм розробляється під Windows Server.

Однак, Windows Server має і ряд недоліків, таких як висока вартість ліцензій, обмежена гнучкість налаштувань та вища ймовірність вірусних атак через популярність платформи.

Linux є безкоштовною та відкритою операційною системою. Linux має ряд суттєвих переваг, що робить його привабливим для впровадження електронної системи управління діяльністю кафедри:

- Більшість дистрибутивів Linux є безкоштовними, що значно знижує загальні витрати на впровадження та підтримку серверної інфраструктури.
- Linux відомий своєю високою безпекою. Відкритий вихідний код дозволяє спільноті швидко виявляти та виправляти вразливості, що знижує ризик вірусних атак і забезпечує стабільну роботу сервера.
- Linux пропонує велику кількість конфігураційних можливостей, що дозволяє налаштовувати систему відповідно до специфічних вимог проекту.
- Linux-сервери відомі своєю стабільністю та високою продуктивністю, що є критично важливим для забезпечення безперервної роботи електронної системи управління.
- Існує велика кількість безкоштовних та відкритих програмних продуктів, сумісних з Linux, що дозволяє використовувати різноманітні інструменти без додаткових витрат.

Обираючи між Windows Server та дистрибутивах Linux перевагу слід надати Linux з кількох причин.

По-перше, використання Linux дозволяє значно знизити загальні витрати на впровадження та підтримку серверної інфраструктури завдяки відсутності

ліцензійних платежів. Це особливо важливо для навчальних закладів, які часто мають обмежені бюджети.

По-друге, висока безпека та стабільність Linux роблять його оптимальним вибором для серверних рішень. Відсутність частих перезавантажень та менша ймовірність вірусних атак забезпечують надійну та безперебійну роботу системи, що є критично важливим для ефективного управління діяльністю кафедри.

По-третє, гнучкість налаштувань Linux дозволяє адаптувати сервер до специфічних потреб проекту. Це включає можливість встановлення та налаштування різноманітного програмного забезпечення, що забезпечує додаткову функціональність та підвищує ефективність роботи системи.

Нарешті, підтримка великої кількості програмних продуктів з відкритим кодом надає широкі можливості для використання сучасних інструментів без додаткових витрат. Це сприяє розвитку інноваційних рішень та підвищенню якості управління навчальним процесом.

Таким чином, обрання Linux як серверної платформи для електронної системи управління діяльністю кафедри є обґрунтованим вибором, що забезпечує високу ефективність, безпеку та економічну вигоду.

1.6 Порівняння методів розробки сайту

Система управління контентом (CMS) є поширеним інструментом для створення вебсайтів, що надає користувачам можливість керувати контентом без глибоких знань програмування. Для реалізації сайту кафедри інформаційних технологій можливим рішенням є використання таких популярних CMS, як WordPress, Joomla, або Drupal. Кожна з цих платформ має свої особливості. До переваг використання CMS можна віднести:

- Головна перевага CMS є швидкість і легкість їхнього впровадження. Встановлення WordPress, Joomla або Drupal може бути виконане навіть без спеціалізованих знань завдяки наявності детальних інструкцій та підтримки

хостинг-платформ. Зазвичай достатньо кількох кроків для інсталяції системи і налаштування основних параметрів сайту.

- WordPress, Joomla, та Drupal мають велику кількість безкоштовних та платних тем для дизайну сайту, що дозволяє адаптувати його до потреб конкретного сайту без створення дизайну з нуля. Крім того, в цих CMS доступна велика кількість плагінів для розширення функціоналу, таких як управління користувачами, інтеграція з соціальними мережами, безпека та інші специфічні можливості.
- CMS дозволяють легко керувати правами доступу до сайту, що є важливою перевагою для кафедри, де різні групи користувачів (адміністрація, викладачі, студенти тощо) можуть мати різні рівні доступу до інформації. Можливість обмеження прав для різних категорій користувачів забезпечує ефективне управління контентом та безпеку даних.
- Вони надають інтуїтивно зрозумілий інтерфейс для керування контентом, що дозволяє користувачам без знання програмування додавати, редагувати і видаляти інформацію на сайті. Це особливо важливо для академічних сайтів, де необхідно регулярно оновлювати дані про викладачів, навчальні програми та заходи.
- Завдяки популярності платформ WordPress, Joomla та Drupal та інших існує активна спільнота розробників і користувачів, яка може надавати підтримку та допомогу при вирішенні технічних проблем.

З недоліків можна виділити наступні аспекти:

- Хоча й CMS пропонують широкий вибір плагінів та тем, їх можливості можуть бути обмеженими для специфічних завдань. Наприклад, якщо кафедра потребує дуже індивідуальний функціонал або унікальний дизайн, реалізація цих вимог може виявитися складною або дорогою в рамках існуючої CMS.
- Внаслідок популярності CMS вони часто стають мішенню для хакерських атак. Стандартні установки CMS можуть бути вразливими, якщо своєчасно не

оновлювати системи та плагіни. Оскільки WordPress є найпопулярнішою CMS у світі, вона також є найуразливішою до атак, тому кафедра повинна приділяти значну увагу питанням кібербезпеки, зокрема регулярному оновленню та резервуванню даних.

- Попри те, що плагіни можуть значно розширити функціонал сайту, їх надмірна кількість може призвести до конфліктів між ними або сповільнення роботи сайту. Використання багатьох плагінів одночасно також збільшує ризик виникнення проблем із безпекою, оскільки не всі плагіни мають високий рівень підтримки та надійності.
- CMS потребують регулярних оновлень, що стосуються як ядра системи, так і плагінів. Відсутність таких оновлень може призвести до вразливостей, погіршення роботи сайту або навіть його несправності. У разі великих оновлень можуть виникати проблеми зі сумісністю між плагінами або темами, що вимагає додаткових зусиль для їх вирішення.
- CMS, такі як WordPress, добре підходять для невеликих і середніх сайтів, у разі значного збільшення обсягу контенту або кількості відвідувачів можуть виникати проблеми з продуктивністю. Для сайтів з великою відвідуваністю необхідно буде оптимізувати серверне оточення, використовувати кешування та інші методи для підвищення швидкості роботи сайту.
- Оскільки багато тем і плагінів розробляються сторонніми розробниками, може виникнути ситуація, коли підтримка певного плагіна припиняється. Це може вимагати пошуку альтернативних рішень або навіть зміни функціональності сайту, що може створити додаткові труднощі.

Використання фреймворків

Фреймворки є потужним інструментом для розробки вебсайтів, оскільки вони забезпечують розробників готовими компонентами, бібліотеками та структурою для організації коду. Використання фреймворків, таких як Django (Python), Laravel (PHP) та Ruby on Rails (Ruby), надає значно більше контролю та

гнучкості під час створення вебдодатків порівняно з CMS. Особливо це важливо, коли йдеться про розробку специфічних, масштабованих або високопродуктивних сайтів. У контексті розробки сайту для кафедри інформаційних технологій використання фреймворків має свої переваги, але також вимагає від розробників глибших знань програмування.

Однією з найбільших переваг використання фреймворків є їх гнучкість. На відміну від CMS, які мають певні обмеження та стандартизовані функції, фреймворки дозволяють розробникам створювати індивідуальні рішення, налаштовуючи кожен елемент сайту відповідно до вимог проекту. Це особливо корисно для створення нестандартних функціональних можливостей, які можуть бути необхідні для кафедри, наприклад, інтеграція з внутрішніми університетськими системами або розробка специфічного механізму керування персоналом та навантаженням викладачів. Крім того, фреймворки забезпечують високу продуктивність і масштабованість. Django, Laravel та Ruby on Rails розроблені з урахуванням принципів модульності та оптимізації, що дозволяє будувати потужні вебсайти, здатні обробляти велику кількість запитів і даних. Також однією з особливостей фреймворків є їх строгість у структурі проекту. Це допомагає підтримувати чистоту коду та забезпечувати зручність подальшого обслуговування. Наприклад, у Django використовуються моделі об'єктно-реляційного відображення (ORM), що спрощує роботу з базами даних, тоді як Laravel пропонує елегантний синтаксис для роботи з контролерами та маршрутизацією. Завдяки цьому фреймворки сприяють створенню більш організованих і зрозумілих проектів. Ще однією перевагою є підтримка сучасних стандартів безпеки. Фреймворки, такі як Django та Laravel, забезпечують вбудовані засоби для запобігання поширеним вразливостям, таким як SQL-ін'єкції, міжсайтові скриптові атаки (XSS) та підробка міжсайтових запитів (CSRF). Для академічних сайтів, які обробляють персональні дані, це є важливим аспектом, оскільки безпека користувацької інформації має першочергове значення. Також варто відзначити широку підтримку фреймворків з боку спільноти розробників.

Популярні фреймворки мають значну кількість доступної документації, навчальних матеріалів та бібліотек для розширення функціональності. Це дозволяє швидко знаходити рішення для виникаючих проблем або використовувати готові модулі для реалізації складних функцій.

Попри численні переваги, використання фреймворків для розробки сайту також має певні недоліки. Перш за все, фреймворки вимагають від розробників глибоких технічних знань і навичок програмування. Якщо CMS орієнтовані на користувачів, які можуть не мати досвіду у веброботці, то робота з фреймворками передбачає повноцінне розуміння таких технологій, як мови програмування (Python, PHP, Ruby), бази даних та принципи архітектури вебдодатків. Це може призвести до збільшення часу на розробку, особливо якщо команда розробників має обмежений досвід роботи з конкретним фреймворком. Іншим значним недоліком є складність налаштування та підтримки інфраструктури. Використання фреймворків передбачає самостійне налаштування серверів, баз даних та інших компонентів, що вимагає додаткових ресурсів. Наприклад, на відміну від CMS, де серверна частина зазвичай налаштована автоматично або пропонується хостинг-провайдерами, у випадку з фреймворками розробники мають самі вирішувати питання з налаштуванням серверів, встановленням необхідних бібліотек та оптимізацією продуктивності. Це може вимагати додаткових зусиль та часу на підтримку серверної інфраструктури. Також не можна не згадати про технічну боргову залежність. Оскільки фреймворки розвиваються й оновлюються, існує ризик, що певні версії фреймворків чи бібліотек можуть застаріти, що вимагатиме регулярного оновлення та адаптації коду. Це може призвести до виникнення додаткових витрат на підтримку проекту та його сумісність із новими версіями технологій.

В результаті, використання фреймворків навряд чи може бути ефективним для невеликих проектів. Вибір фреймворку може бути надмірним рішенням, яке призведе до ускладнення процесу розробки. У таких випадках більш прості

рішення, такі як CMS або навіть статичні сайти, можуть бути більш доцільними як з точки зору ресурсів, так і часу на розробку.

Використання статичних сайтів (HTML/CSS/JS)

Статичні сайти, побудовані на основі таких технологій, як HTML, CSS, і JavaScript, представляють собою один із найстаріших і найпростіших підходів до веброзробки. Статичні сайти відрізняються тим, що їх вміст є фіксованим і не змінюється в залежності від дій користувача або зовнішніх факторів. Вони не потребують серверної обробки для генерації сторінок на запит, оскільки кожна сторінка існує як окремий файл на сервері. Такий підхід може бути привабливим для простих проектів, зокрема для невеликих інформаційних сайтів, але він також має значні обмеження, особливо коли йдеться про управління динамічними даними, що є критичним для сайту кафедри.

Однією з головних переваг статичних сайтів є їх швидкість. Оскільки сторінки не генеруються динамічно, вони завантажуються швидше, оскільки сервер просто надсилає готовий HTML-документ користувачеві без необхідності обробки запитів до бази даних або виконання серверного коду. Це також робить статичні сайти менш залежними від продуктивності серверного обладнання. Для кафедри це може бути привабливо у випадку, якщо планується простий сайт-візитівка з мінімальною кількістю інформації та без інтерактивних компонентів. Також важливою перевагою є простота впровадження та підтримки: статичні сайти можна розробляти без складних фреймворків чи платформ, використовуючи лише базові вебтехнології. Це робить їх легко зрозумілими та доступними навіть для початківців, які мають базові знання з HTML, CSS і JavaScript. Крім того, такі сайти не потребують регулярних оновлень серверних компонентів, плагінів чи баз даних, як у випадку з CMS або фреймворками, що знижує загальні витрати на їх підтримку. Ще однією перевагою статичних сайтів є їхня безпека. Оскільки вони не мають серверної логіки і баз даних, статичні сайти практично не схильні до таких поширених загроз, як SQL-ін'єкції або міжсайтові скриптові атаки (XSS).

Вони також менше піддаються ризику хакерських атак, оскільки немає серверних компонентів, що можуть бути вразливими до експлойтів.

Однак, незважаючи на переваги, використання статичних сайтів має значні недоліки, особливо коли йдеться про динамічні дані та складні функції. Основний недолік полягає в тому, що статичні сайти не можуть генерувати або оновлювати контент у режимі реального часу. У випадку кафедри, де необхідно постійно оновлювати інформацію про викладачів, навчальні програми, розклад, а також забезпечувати інтерактивні функції, такі як керування навантаженням викладачів або відображення персоналізованого контенту для різних груп користувачів, статичний сайт є неефективним рішенням. Кожного разу, коли виникає потреба змінити контент на статичному сайті, розробник повинен вручну редагувати HTML-файли, що є трудомістким і незручним процесом. Це особливо ускладнює роботу з великими сайтами, де кожна зміна вимагає редагування кількох сторінок або навіть усієї структури сайту. Крім того, відсутність серверної логіки означає, що статичні сайти не можуть взаємодіяти з базами даних або обробляти складні запити, такі як реєстрація користувачів, обробка форм або управління навчальними матеріалами. Ще одним істотним недоліком є відсутність можливості персоналізації контенту. На відміну від динамічних сайтів, які можуть змінювати вміст залежно від дій користувачів або їхніх уподобань, статичні сайти завжди показують однакову інформацію всім відвідувачам. Для кафедри, де може виникнути потреба в наданні різного доступу до певного контенту (наприклад, для викладачів, студентів або адміністрації), такий підхід є непридатним. Окрім цього, масштабованість статичних сайтів є обмеженою. Хоча вони можуть легко працювати для невеликих проектів або сайтів з малою кількістю сторінок, у випадку значного збільшення кількості контенту або інтеграції нових функцій підтримка такого сайту може стати надзвичайно складною. Кожна нова сторінка додається вручну, що може призвести до плутанини і збільшення часу на управління сайтом. Якщо кафедра планує регулярно розширювати свій сайт, наприклад, додаючи нові розділи для наукових публікацій або інтерактивні

інструменти для управління курсами, статичний сайт не зможе ефективно задовольнити ці вимоги. Також хоча й JavaScript може додати деяку динаміку до статичних сайтів, така динаміка є обмеженою у порівнянні з тим, що може запропонувати повноцінний серверний бекенд. Наприклад, створення інтерактивних панелей адміністрування або динамічних таблиць для розподілу викладацького навантаження на статичному сайті є значно складнішим і менш ефективним. Для кафедри, яка потребує постійної взаємодії з контентом і оновлення даних, це може стати серйозною перешкодою.

Підсумовуючи, використання статичних сайтів (HTML/CSS/JS) може бути ефективним для простих, невеликих проектів з обмеженими вимогами до динаміки та інтерактивності. Вони відрізняються високою швидкістю завантаження, безпекою та простотою впровадження. Однак для сайтів, які потребують постійного оновлення, роботи з динамічними даними або інтерактивними функціями, статичні сайти є надзвичайно обмеженими. У випадку кафедри інформаційних технологій, де необхідно забезпечити динамічний контент і взаємодію з різними категоріями користувачів, статичний сайт є непридатним рішенням через свою неспроможність ефективно працювати з динамічними даними та масштабуватися відповідно до зростання потреб кафедри.

Обґрунтування використання CMS для розробки сайту

Вибір системи управління контентом (CMS) для створення сайту кафедри інформаційних технологій був зумовлений низкою факторів, що роблять CMS більш придатними для цього проекту порівняно з фреймворками або статичними сайтами. У контексті вимог до електронної системи управління діяльністю кафедри, CMS надає переваги у простоті розробки, гнучкості, швидкості впровадження та можливості масштабування, які неможливо досягти за допомогою інших підходів.

CMS, такі як WordPress, забезпечують більш швидке та просте впровадження порівняно з фреймворками та статичними сайтами. Оскільки основна частина системи вже розроблена, це дозволяє значно скоротити час на початкову розробку

сайту. Використання фреймворків, таких як Django або Laravel, потребувало б значного обсягу ручного кодування для побудови базової структури сайту, що затягує процес розробки. Статичні сайти (HTML/CSS/JS), своєю чергою, є ще більш трудомісткими, оскільки для кожної сторінки потрібно писати окремий код без можливості використання інтегрованих механізмів для динамічного управління контентом. Сайт для кафедри інформаційних технологій не має потреби в складній архітектурі, яку надають фреймворки, але потребує швидкого та ефективного способу створення сайту з базовою функціональністю. CMS дає змогу впровадити рішення значно швидше та без зайвих витрат часу на розробку з нуля. Також один з ключових аспектів у виборі CMS — це зручність управління контентом. CMS, такі як WordPress, спрощують створення, редагування і публікацію контенту для користувачів. Кафедра може делегувати управління сайтом персоналу, який не є розробниками, завдяки інтуїтивному інтерфейсу, що дозволяє легко додавати нові сторінки, оновлювати інформацію або змінювати налаштування. Фреймворки і статичні сайти потребують більше технічних знань і втручання в код, що ускладнює їх використання для користувачів. Фреймворки, як правило, націлені на створення складних рішень з великою кількістю користувацької логіки, де управління контентом реалізується через спеціально розроблені панелі адміністрування, що потребує більше ресурсів для налаштування. Статичні сайти, своєю чергою, не передбачають інтегрованого інтерфейсу для управління контентом і вимагають внесення змін безпосередньо до коду, що є значною перешкодою для щоденного використання. CMS, зокрема WordPress, має величезний набір плагінів, що дозволяють легко додати новий функціонал без необхідності писати код. Це робить CMS ідеальним вибором, оскільки плагіни можуть покрити більшість вимог до сайту — від управління персоналом до розподілу навантаження. Для фреймворків аналогічні функції потрібно розробляти вручну, що потребує значного часу і ресурсів. Статичні сайти взагалі не мають можливості інтеграції таких динамічних елементів без використання додаткових технологій, таких як JavaScript або серверні мови

програмування. Один з важливих факторів для вибору CMS є можливість легкого масштабування системи в майбутньому. CMS дозволяють легко розширювати сайт, додаючи нові функції або розділи. Наприклад, якщо кафедра потребуватиме додавання нових модулів, таких як система управління курсами чи онлайн-тестування, це можна буде зробити без необхідності переписувати всю систему.

Огляд популярних CMS

При розробці електронної системи управління діяльністю кафедри важливим є вибір системи управління контентом (CMS), яка забезпечить необхідну функціональність, зручність використання та масштабованість. Розглянемо кілька популярних CMS

WordPress є найпопулярнішою системою управління контентом у світі, використовується для створення різноманітних сайтів – від блогів до великих корпоративних порталів. Основними причинами популярності WordPress є його простота у використанні та велика кількість доступних плагінів і тем. Інтерфейс WordPress інтуїтивно зрозумілий, що дозволяє навіть новачкам швидко освоїти систему та почати створювати контент. Велика спільнота користувачів і розробників забезпечує широкий доступ до ресурсів, документації, форумів та груп підтримки, що робить вирішення проблем та пошук інформації легким і швидким процесом. WordPress регулярно оновлюється, що гарантує високу безпеку та стабільність платформи. Система підтримує сучасні стандарти безпеки, а численні плагіни дозволяють додатково посилити захист сайту. Важливою перевагою WordPress є його масштабованість – платформа здатна підтримувати як невеликі сайти, так і великі портали з високим трафіком.

Joomla є другою за популярністю CMS, відомою своєю гнучкістю та широкими можливостями налаштування. Вона підходить для створення складних веб-сайтів та веб-додатків завдяки своїй архітектурі, яка дозволяє легко розширювати функціональність. Joomla підтримує багато користувачів і надає розширені можливості управління доступом, що робить її зручною для сайтів з великим обсягом контенту та численними авторами. Однак використання Joomla

може вимагати більше технічних знань порівняно з WordPress, що може бути перешкодою для новачків. Спільнота Joomla також є досить активною, хоча і меншою за спільноту WordPress. Це означає, що доступ до ресурсів і підтримки може бути трохи обмеженим. Joomla надає високий рівень безпеки, але, як і будь-яка інша CMS, потребує регулярного оновлення та підтримки для забезпечення стабільної роботи.

Drupal є потужною і гнучкою CMS, яка часто використовується для створення великих та складних веб-сайтів з високими вимогами до безпеки та продуктивності. Drupal відомий своєю модульною структурою, що дозволяє розробникам створювати індивідуальні рішення для будь-яких потреб. Ця система особливо підходить для урядових і корпоративних сайтів, а також для сайтів з великою кількістю інтерактивних елементів. Основним недоліком Drupal є його складність: система має круту криву навчання і вимагає значних технічних знань для налаштування та підтримки. Незважаючи на це, Drupal пропонує безліч функцій прямо "з коробки", що робить його потужним інструментом для досвідчених розробників. Безпека є одним із ключових аспектів Drupal, і система регулярно отримує оновлення для захисту від нових загроз. Спільнота Drupal менша, але дуже активна і професійна, що забезпечує високий рівень підтримки та розвитку платформи.

Magento спеціалізується на створенні електронних магазинів і є однією з провідних платформ для e-commerce. Magento надає потужні інструменти для управління товарами, замовленнями та клієнтами, що робить її ідеальною для великих і складних інтернет-магазинів. Вона підтримує численні валюти та мови, що дозволяє вести бізнес на глобальному рівні. Однак, як і у випадку з Drupal, Magento має високу складність у налаштуванні та використанні, що вимагає значних технічних знань і досвіду. Magento відзначається своєю гнучкістю та можливістю розширення, що дозволяє створювати індивідуальні рішення для будь-яких бізнес-потреб. Система забезпечує високий рівень безпеки та продуктивності, що є критичним для електронної комерції. Спільнота Magento є досить великою і

включає як розробників, так і бізнес-користувачів, що сприяє активному розвитку та підтримці платформи.

Обґрунтування вибору CMS WordPress

WordPress варто обрати для створення сайту з кількох причин, які роблять його ідеальною платформою для широкого спектра веб-проектів. Перш за все, це найпопулярніша система управління контентом у світі, що використовується на понад 40% всіх веб-сайтів. Така популярність забезпечує постійну підтримку та розвиток платформи, що є важливим фактором для будь-якого веб-проекту. Величезна спільнота користувачів і розробників означає, що для WordPress існує безліч ресурсів, включаючи документацію, форуми, відеоуроки та курси, що робить навчання і вирішення проблем простішим і швидшим.

Однією з ключових переваг WordPress є його простота у використанні. Інтерфейс платформи інтуїтивно зрозумілий, що дозволяє навіть новачкам швидко освоїти основні функції і почати створювати контент без глибоких технічних знань. Це особливо важливо для невеликих організацій та індивідуальних підприємців, які не мають можливості наймати професійних веб-розробників. Вбудований редактор блоків, відомий як Gutenberg, дозволяє легко створювати і редагувати сторінки за допомогою перетягування блоків, що робить процес розробки ще більш доступним.

WordPress також відзначається своєю гнучкістю та масштабованістю. Платформа підтримує великий вибір плагінів і тем, які дозволяють додавати нові функції і змінювати дизайн сайту без необхідності писати код. Існує понад 58,000 безкоштовних плагінів у офіційному репозиторії, що дозволяє знайти рішення для будь-яких потреб - від SEO оптимізації до інтеграції з соціальними мережами та електронною комерцією. Крім того, існує безліч преміум плагінів і тем, які надають розширені можливості та професійний дизайн. Така гнучкість робить WordPress підходящим для різних типів сайтів, включаючи блоги, бізнес-сайти, портфоліо, новинні портали і навіть інтернет-магазини.

Безпека є ще одним важливим аспектом, який варто враховувати при виборі CMS, і WordPress не є винятком. Платформа регулярно оновлюється для забезпечення захисту від нових загроз, а велика спільнота розробників постійно працює над виявленням і виправленням вразливостей. Додатково, існує багато плагінів безпеки, які дозволяють посилити захист сайту, забезпечуючи багаторівневу аутентифікацію, сканування на наявність шкідливого ПЗ та інші функції безпеки. Завдяки цьому WordPress є надійним вибором для створення безпечних веб-сайтів.

Ще одна перевага WordPress – це його SEO-дружелюбність. Вбудовані інструменти SEO дозволяють оптимізувати контент для пошукових систем без особливих зусиль. Крім того, існують потужні плагіни, такі як Yoast SEO, які надають розширені можливості для оптимізації, допомагаючи підвищити видимість сайту в пошукових системах. Це особливо важливо для бізнесів, які хочуть залучити більше трафіку і збільшити свою онлайн-присутність.

WordPress також добре підходить для мультимовних сайтів, що дозволяє залучити аудиторію з різних країн. За допомогою плагінів, таких як WPML або Polylang, можна легко створити багатомовний сайт, забезпечуючи користувачам можливість вибору мови. Це важлива функція для міжнародних організацій, які прагнуть розширити свою аудиторію і зробити свій контент доступним для людей з різних культур і регіонів.

Ще однією перевагою є інтеграція з соціальними мережами. WordPress дозволяє легко ділитися контентом через соціальні платформи, що сприяє збільшенню охоплення та залученню нових відвідувачів. Багато тем та плагінів мають вбудовані інструменти для інтеграції соціальних медіа, що дозволяє автоматично публікувати нові статті або ділитися оновленнями з підписниками. З точки зору продуктивності та масштабованості, WordPress здатний обробляти великі обсяги трафіку і підтримувати великі сайти. За правильного налаштування та використання оптимізаційних плагінів, таких як кешування сторінок та оптимізація бази даних, можна забезпечити швидку роботу сайту навіть при

великому навантаженні. Це робить WordPress ідеальним вибором для сайтів, які планують рости і розширюватися з часом. Нарешті, економічна ефективність WordPress є важливим фактором для багатьох користувачів. Платформа є безкоштовною та з відкритим вихідним кодом, що дозволяє уникнути витрат на ліцензії. Велика кількість безкоштовних плагінів і тем також сприяє зниженню витрат на розробку та підтримку сайту. Це робить WordPress доступним для малого бізнесу, некомерційних організацій та окремих користувачів, які прагнуть створити професійний сайт без значних інвестицій.

РОЗДІЛ 2

ВИМОГИ ДО ВЕБ-САЙТУ

2.1 Опис серверного середовища та хостингу для WordPress

Одним із ключових етапів підготовки до впровадження електронної системи управління діяльністю кафедри на базі WordPress є вибір серверного середовища та хостингу. Оскільки WordPress є популярною системою управління контентом, вона підтримує широкий спектр серверних платформ. У нашому випадку було обрано Linux як основну операційну систему для серверу, що зумовлено низкою важливих факторів, серед яких стабільність, безпека та оптимізація під CMS WordPress.

При виборі хостинг-платформи для розгортання WordPress розглянемо кілька типів хостингу:

Спільний хостинг - найдоступніший варіант, де один сервер використовується для хостингу кількох сайтів одночасно. Спільний хостинг зручний з точки зору ціни, проте він може мати недоліки у вигляді обмеженої продуктивності, оскільки ресурси сервера розподіляються між багатьма користувачами.

VPS (віртуальний приватний сервер) - більш потужний варіант, який дозволяє отримати віртуальний сервер з виділеними ресурсами. VPS забезпечує більше контролю над серверним середовищем і дозволяє налаштовувати операційну систему та програмне забезпечення відповідно до потреб.

Виділений сервер надає повний доступ до всіх ресурсів сервера, що робить його ідеальним для великих та ресурсоємних проектів. Проте цей варіант є надто дорогим і складним для підтримки, тому для потреб сайту він є надмірним.

Останнім часом набирає популярність хмарний хостинг, який дозволяє динамічно керувати ресурсами і забезпечувати гнучкість у використанні потужностей сервера залежно від поточних потреб. Це може бути привабливим варіантом для подальшого розвитку сайту.

Окрім вибору операційної системи Linux, для забезпечення роботи WordPress існує рекомендовані наступні компоненти програмного забезпечення:

Apache або Nginx. Обидва ці веб-сервери є найбільш поширеними для роботи з WordPress. Apache зазвичай вибирають через його широке розповсюдження і підтримку. Nginx, зі свого боку, забезпечує більшу продуктивність при високих навантаженнях. Кращим вибором буде поєднання цих двох веб-серверів для роботи у зв'язці nginx+Apache. Веб-сервер nginx оброблює статичний контент, в той час як Apache – динамічний.

СУБД MySQL або MariaDB. Для зберігання даних WordPress зазвичай використовує MySQL або її форк MariaDB. Обидві бази даних забезпечують надійну роботу і легко інтегруються з WordPress.

Оскільки WordPress написаний на PHP, тому необхідно забезпечити підтримку останньої стабільної версії PHP на сервері. Використання сучасної версії PHP підвищує продуктивність сайту та забезпечує сумісність із новими функціями WordPress.

Також під час налаштування серверного середовища для WordPress слід також врахувати аспекти, описані далі.

Для забезпечення безпеки користувачів сайту кафедри важливо налаштувати SSL-сертифікати, які шифрують трафік між сайтом і користувачами. Це підвищує довіру до сайту та гарантує конфіденційність переданих даних. Оптимальним варіантом буде використання безкоштовного SSL сертифіката від Let's Encrypt, Zero SSL чи GTS. Також для підвищення продуктивності сайту необхідно налаштувати механізми кешування. Це дозволяє зменшити навантаження на сервер і покращити швидкість завантаження сторінок. Зазвичай для цього використовується або вебсервер Litespeed у поєднанні з плагіном LiteSpeedCache або сторонні плагіни (W3C Cache тощо) або ж об'єктне кешування (Redis, Memcached). Щоб захистити дані сайту від втрат, важливо налаштувати регулярне автоматичне створення резервних копій або обрати хостинг, де є вже кешування. Це дозволить швидко відновити сайт у разі технічних збоїв чи атак.

2.2 Забезпечення безпеки веб-сайту

Забезпечення безпеки сайту на платформі WordPress є критичним завданням, оскільки WordPress є однією з найпопулярніших CMS у світі, в результаті чого і часто стає мішенню для хакерських атак. Для розуміння, як краще захищати сайт, необхідно ознайомитися з основними типами атак і методами їх запобігання:

Brute Force Attacks. Цей тип атак є одним із найбільш поширених на WordPress сайти. Його мета – підібрати логін і пароль для доступу до адміністративної панелі сайту. Атаки грубою силою автоматизовані і можуть використовувати тисячі варіантів для підбору правильних авторизаційних даних. Оскільки WordPress за замовчуванням використовує сторінку для входу за адресою `/wp-admin` або `/wp-login.php`, ці сторінки часто стають об'єктом атак. Шляхи запобігання: налаштування плагінів, які блокують IP-адресу після кількох невдалих спроб входу (наприклад, плагін `Limit Login Attempts Reloaded`), використання двофакторної аутентифікації (2FA), зміна стандартної адреси сторінки входу на адміністративну панель зменшує кількість спроб несанкціонованого доступу.

SQL Injection. Атаки типу SQL Injection полягають у тому, що зловмисник вставляє шкідливий SQL-код у поля для введення на сайті (наприклад, у форми для пошуку або реєстрації). Якщо сайт не належним чином перевіряє введені дані, цей код може бути виконаний на сервері, що дасть хакеру доступ до бази даних сайту. Через SQL-ін'єкції зловмисники можуть отримати конфіденційну інформацію або навіть повний контроль над сайтом. Шляхи запобігання: WordPress, у поєднанні з базою даних MySQL/MariaDB, дозволяє використовувати підготовлені запити, які перевіряють та очищають дані перед виконанням SQL-запитів. XSS-атаки спрямовані на вставку шкідливого JavaScript-коду в веб-сторінки, що відображаються користувачам. Цей код може виконувати різноманітні дії: викрадення облікових даних, перенаправлення на шкідливі сайти або зміну вмісту сторінок. Зазвичай XSS-атаки використовують недостатньо захищені форми або

інші елементи введення даних. Шляхи запобігання: плагіни, як-от Wordfence або Sucuri Security, можуть допомогти у виявленні та блокуванні спроб XSS-атак.

Також слід виконати загальні заходи з підвищення безпеки: виконати налаштування SSL-сертифікатів – це забезпечує шифрування даних, що передаються між сайтом і його користувачами, реалізувати створення резервних копій сайту, щоб у разі атаки або збою можна було швидко відновити роботу сайту. Це можна налаштувати через спеціальні плагіни (UpdraftPlus, BackupBuddy тощо). Окрім захисту входу до панелі адміністратора, варто обмежити доступ до панелі лише з певних IP-адрес, що значно ускладнить доступ для сторонніх осіб, виконати встановлення плагінів для моніторингу діяльності на сайті дозволить відстежувати підозрілу активність. Логи можуть бути корисними для виявлення та аналізу атак.

2.3 Оптимізація продуктивності сайту

Продуктивність сайту є критично важливим фактором для забезпечення позитивного користувацького досвіду та підтримки ефективної роботи ресурсу. Сайт має бути не тільки функціональним, але й швидким. Оптимізація продуктивності безпосередньо впливає на задоволеність користувачів, позиції сайту в пошукових системах (SEO), а також на зниження навантаження на сервер. Для цього необхідно враховувати кілька ключових аспектів: використання кешування, оптимізацію зображень та інших елементів сайту, а також моніторинг продуктивності через метрики швидкодії.

Оптимізація зображень і мультимедійних файлів. Медіаконтент, в тому числі зображення, є одним із найбільших за розміром елементів на веб-сторінці, тому їх оптимізація суттєво впливає на швидкість завантаження сайту. Для зменшення часу завантаження можна виконати стиснення зображень без значних втрат якості у сучасний формат WebP за допомогою плагінів (наприклад, WP Smush, ShortPixel). Також можна впровадити відкладене завантаження зображень (lazy loading), тобто зображення завантажуються лише тоді, коли користувач прокручує сторінку до них. Це зменшує початковий час завантаження сторінки.

Оптимізація коду сайту. Стиснення та мінімізація коду (HTML, CSS, JavaScript) можуть суттєво прискорити завантаження сторінок. Мінімізація CSS і JavaScript-файлів виконується шляхом видалення зайвих пробілів, коментарів та інших незначних елементів.

Використання CDN (Content Delivery Network) – це дозволить прискорити завантаження сайту для користувачів по всьому світу, розміщуючи копії контенту на різних серверах у різних географічних регіонах. Це зменшує час відповіді сервера, оскільки користувач підключається до найближчого сервера. Можна використовувати безкоштовний тарифний план від компанії CloudFlare для реалізації CDN.

Кешування є одним із найефективніших методів оптимізації продуктивності веб-сайтів. Воно дозволяє зберігати готові дані, щоб при наступному зверненні не генерувати їх наново. У контексті WordPress можна виділити кілька видів кешування:

1. Об'єктне кешування зберігає результати запитів до бази даних або об'єктів PHP у пам'яті, що дозволяє значно прискорити завантаження сторінок. Використовується, наприклад, плагіном W3 Total Cache. Це особливо корисно для динамічних сайтів, де багато даних отримується з бази даних при кожному запиті.

2. Кешування браузера дозволяє зберігати статичні ресурси (зображення, CSS, JavaScript) у браузері користувача на певний період часу, щоб уникнути повторного завантаження цих файлів при наступних візитах. Це значно знижує навантаження на веб-сервер та покращує швидкодію для повторних відвідувань.

3. LiteSpeed Cache – це потужне рішення для кешування, яке забезпечує високу швидкість завантаження сторінок завдяки кешуванню на серверному рівні. LiteSpeed Cache інтегрується з веб-сервером LiteSpeed і дозволяє кешувати як статичні сторінки, так і динамічні запити. Переваги LiteSpeed Cache включають мінімізацію запитів до бази даних, зниження навантаження на сервер та збільшення швидкості відображення контенту користувачам.

Для вимірювання і моніторингу продуктивності сайту необхідно використовувати спеціальні інструменти та метрики. Основні метрики, які допомагають оцінити швидкодію сайту:

1. TTFB вимірює час, який проходить від моменту надсилання запиту користувачем до моменту отримання першого байта відповіді від сервера. Чим менше цей показник, тим швидше сервер відповідає на запити.

2. LCP визначає час, необхідний для завантаження найбільшого видимого елемента на сторінці (зображення, тексту тощо). Це важлива метрика, оскільки користувачі зазвичай оцінюють швидкість сайту за тим, як швидко вони можуть побачити основний контент.

3. FID вимірює час між взаємодією користувача з сайтом (наприклад, натискання кнопки) і моментом, коли браузер починає обробляти цей запит. Оптимізація FID дозволяє покращити інтерактивність сайту.

4. CLS вимірює стабільність верстки сайту під час завантаження. Якщо елементи сторінки пересуваються або змінюють свої позиції під час завантаження, це може викликати незручності для користувачів. Низьке значення CLS вказує на стабільний інтерфейс.

Для моніторингу та аналізу продуктивності сайту можна використовувати такі інструменти:

1. Google PageSpeed Insights – аналізує сторінку та надає рекомендації для оптимізації продуктивності.

2. GTmetrix – показує метрики швидкодії, включаючи час завантаження сторінки та загальну кількість запитів.

3. Pingdom Tools – інструмент для вимірювання часу завантаження сторінки та надання рекомендацій щодо покращення.

Отже, оптимізація продуктивності сайту є важливою частиною забезпечення ефективної роботи електронної системи управління діяльністю кафедри. Використання кешування (об'єктного, браузерного, LiteSpeed), оптимізація зображень та кодів, а також впровадження CDN дозволяють зменшити час завантаження сторінок і підвищити загальну швидкість сайту. Постійний моніторинг продуктивності за допомогою інструментів та метрик забезпечує своєчасне виявлення та усунення потенційних проблем.

РОЗДІЛ 3

ПРАКТИЧНА РЕАЛІЗАЦІЯ ЕЛЕКТРОННОЇ СИСТЕМИ УПРАВЛІННЯ ДІЯЛЬНІСТЮ КАФЕДРИ

3.1 Огляд серверного оточення

У даному підрозділі буде розглянуто серверне середовище, яке використовується для роботи веб-сайту. Серверне оточення є основою для стабільної, безпечної та швидкої роботи веб-ресурсу. Детально проаналізуємо кожен компонент цього середовища.

Операційна система

На сервері встановлена операційна система CloudLinux 8.10. CloudLinux (рис. 1) — це комерційна операційна система, заснована на Red Hat Enterprise Linux (RHEL), що орієнтована на хостинг-сервери. До переваг цієї ОС можна віднести:

- Ізоляція користувачів (LVE): кожен акаунт хостингу працює у власному середовищі (контейнері). Це ізолює користувачів один від одного, забезпечуючи захист від впливу ресурсомістких або шкідливих процесів.
- Розподіл ресурсів: CloudLinux дозволяє адміністраторам обмежувати ресурси для кожного користувача (процесор, RAM, I/O). Це забезпечує стабільну роботу навіть при високому навантаженні.
- Enhanced Security: ОС використовує модуль CageFS, який надає користувачам доступ лише до їхніх файлів і захищає сервер від атак типу "перехрестного читання файлів".
- Оптимізація продуктивності: CloudLinux має вбудовані інструменти для моніторингу та оптимізації роботи серверів, що робить його ідеальним вибором для хостинг-провайдерів.
- Підтримка довготривалих оновлень: версія 8 базується на останніх стабільних розробках Red Hat Enterprise Linux, що гарантує сучасність та безпеку середовища.

```

sh-4.4# cat /etc/os-release
NAME="CloudLinux"
VERSION="8.10 (Vladimir Aksyonov)"
ID="cloudlinux"
ID_LIKE="rhel fedora centos"
VERSION_ID="8.10"
PLATFORM_ID="platform:el8"
PRETTY_NAME="CloudLinux 8.10 (Vladimir Aksyonov)"
ANSI_COLOR="0;31"
CPE_NAME="cpe:/o:cloudlinux:cloudlinux:8.10:GA:server"
HOME_URL="https://www.cloudlinux.com/"
BUG_REPORT_URL="https://www.cloudlinux.com/support"
VARIANT_ID="cloudlinux"
sh-4.4#

```

Рис. 3.1. Метадані опису ОС у терміналі

Панель керування

Також на сервері інстальована панель керування WHM/cPanel. Це одна з найпопулярніших панелей управління для серверів, що забезпечує інтуїтивний інтерфейс для адміністрування серверів та сайтів. WHM надає можливість створювати та налаштовувати облікові записи cPanel для окремих користувачів. WHM інтегрує різноманітні засоби захисту, включаючи брандмауери, функції блокування IP, та підтримку SSL-сертифікатів. Вбудовані функції резервного копіювання дозволяють автоматизувати процес створення копій даних для зменшення ризику втрати інформації. Можливість призначати різні ліміти на дисковий простір, пропускну здатність і CPU для кожного облікового запису.

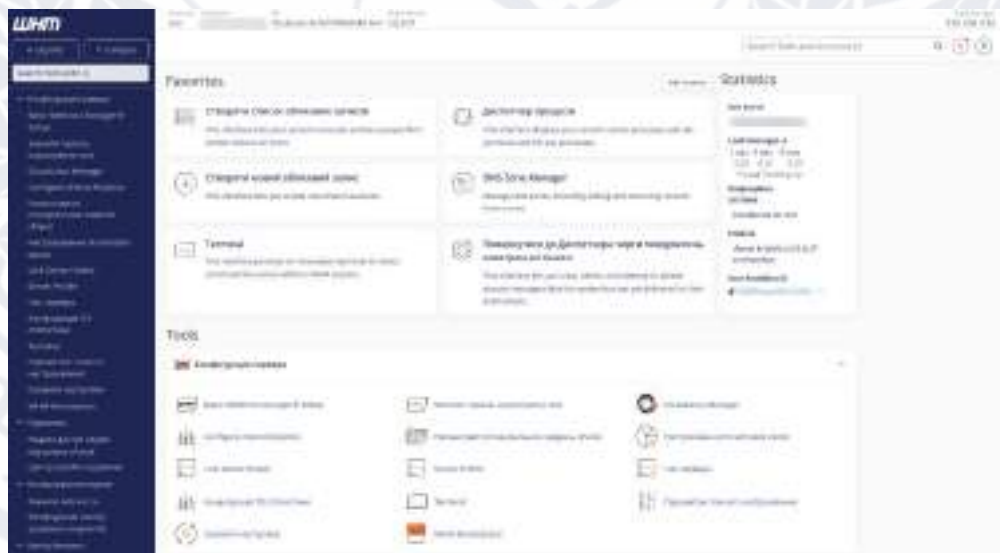


Рис. 3.2. Головна сторінка панелі керування WHM

cPanel дозволяє користувачам легко керувати своїми сайтами, базами даних, електронною поштою та доменами, а також дозволяє користувачам завантажувати, редагувати та організовувати файли сайту без використання FTP-клієнта. Панель cPanel має автоматизовані інструменти для інсталяції та управління CMS WordPress.

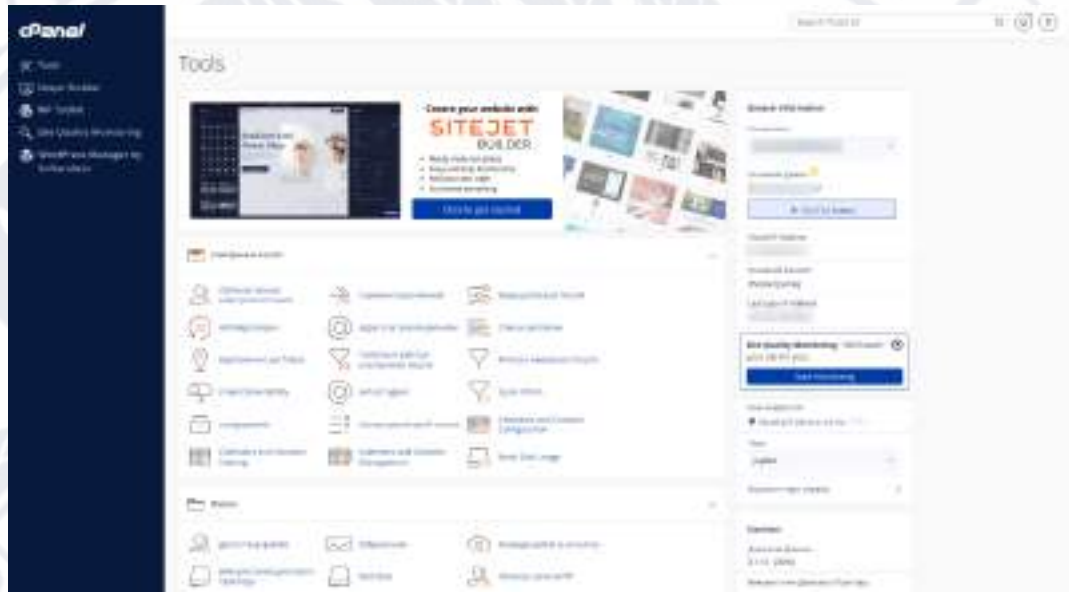


Рис. 3.3. Головна сторінка панелі керування cPanel

Веб-сервер

Серверне оточення використовує комбінацію nginx та Apache для оптимізації продуктивності та функціональності. Apache є одним із найпопулярніших веб-серверів, відомим своєю гнучкістю та багатством модулів. Apache підтримує розширення, такі як `mod_rewrite` для обробки URL, `mod_ssl` для SSL-сертифікатів та `mod_security` для захисту від атак. Висока сумісність із різними CMS (зокрема й WordPress) та численними мовами програмування. Користувачі можуть налаштовувати конфігурацію серверів через файли `.htaccess` для кожного веб-серверу. nginx використовується як зворотній проксі-сервер для покращення продуктивності, оброблюючи статичний контент. nginx значно швидше обробляє

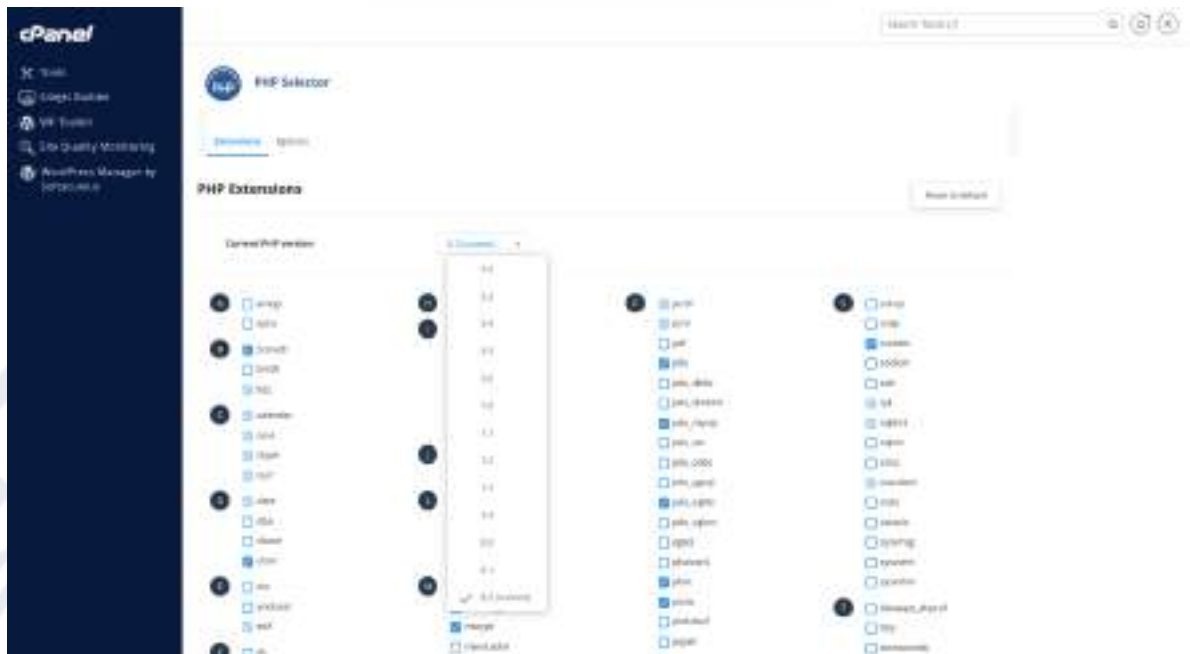


Рис. 3.5. Меню налаштування версії PHP та розширень PHP у панелі керування cPanel

Система управління базами даних (СУБД)

На сервері встановлена система управління базами даних MariaDB 10.11. Це сучасна, швидка та масштабована СУБД, створена як відкритий форк MySQL. Версія MariaDB 10.11 пропонує значні поліпшення у продуктивності, безпеці та функціональності, що робить її ідеальним вибором для веб-додатків, таких як WordPress, Joomla, або інші CMS.

Ця СУБД забезпечує підвищену продуктивність за рахунок нових алгоритмів обробки запитів, покращеного кешування та підтримки сучасних апаратних ресурсів. Це дозволяє обробляти великі обсяги даних з мінімальною затримкою. MariaDB є 100% сумісною з MySQL у плані SQL-запитів, структур таблиць і API. Це спрощує перехід від MySQL до MariaDB без змін у додатках, у той же час вона пропонує покращення, яких немає в MySQL, наприклад, додаткові механізми зберігання та розширені функції SQL. MariaDB підтримує декілька механізмів зберігання (storage engines), серед яких в основному використовують два варіанти: InnoDB та Aria - це основний механізм, оптимізований для транзакцій та високої

цілісності даних та високопродуктивний механізм для читання-запису, створений для роботи з великими таблицями, відповідно.

Обрана СУБД також дозволяє організовувати кластерну роботу для високої доступності, як от мультисайтова реплікація, яка підтримує розподілені системи з географічно рознесеними вузлами, покращені механізми шифрування даних у транспорті (SSL/TLS) та на диску, гранулярна система доступу до бази даних із можливістю налаштування прав для кожного користувача або ролі, захист від SQL-ін'єкцій через покращення у обробці запитів.

Завдяки підтримці кластеризації за допомогою Galera Cluster MariaDB забезпечує безперебійну роботу баз даних. Усі вузли в кластері синхронізуються в реальному часі, що гарантує відсутність втрати даних, а інтелектуальне балансування навантаження між кількома серверами забезпечує підвищення продуктивності.

Висновок щодо серверного оточення

Комбінація CloudLinux, WHM/cPanel, nginx/Apache та мультиверсійності PHP забезпечує надійне, продуктивне та безпечне серверне середовище. Такий підхід дозволяє з легкістю адаптувати сервер під різні типи веб-додатків, знижуючи ризики відмов і підвищуючи продуктивність ресурсів.

3.2 Встановлення та налаштування CMS WordPress

Завдяки автоматизованим інструментам, таким як WordPress Manager by Softaculous у cPanel, процес встановлення WordPress стає простим, швидким та зручним навіть для користувачів без технічного досвіду. У цьому підрозділі буде детально описано кожен етап встановлення CMS WordPress через інструмент Softaculous, доступний у панелі cPanel.

Перш за все, доступ до WordPress Manager by Softaculous можна отримати через інтерфейс cPanel, який зазвичай використовується для управління обліковими записами хостингу. Щоб почати процес встановлення, користувачеві потрібно увійти до cPanel, використовуючи облікові дані, надані хостинг-провайдером. Після входу на головній сторінці cPanel необхідно знайти розділ «Softaculous Apps

усі останні оновлення безпеки та функціональні вдосконалення. Softaculous автоматично завантажує і встановлює обрану версію WordPress з офіційного репозиторію, що гарантує її оригінальність і безпеку.



Рис. 3.7. Базові налаштування сайту - протокол, домен, директорія, версія

Наступним кроком є налаштування облікового запису адміністратора. У цьому розділі користувачеві потрібно вказати ім'я користувача, пароль і електронну пошту адміністратора. Важливо обрати надійний пароль, оскільки він є основним засобом захисту панелі управління WordPress від несанкціонованого доступу. Також бажано вказаний логін, відмінний від admin для ускладнення підбору авторизаційних даних злоумисниками. Електронна пошта адміністратора використовується для відновлення доступу у разі втрати пароля, тому слід переконатися, що вона є актуальною.



Рис. 3.8. Налаштування облікового запису адміністратора

Додатково можна налаштувати назву сайту та короткий опис, які відображатимуться на головній сторінці WordPress за замовчуванням. Ці налаштування можна змінити пізніше через адміністративну панель WordPress, якщо користувач поки не визначився з остаточним варіантом назви чи опису.

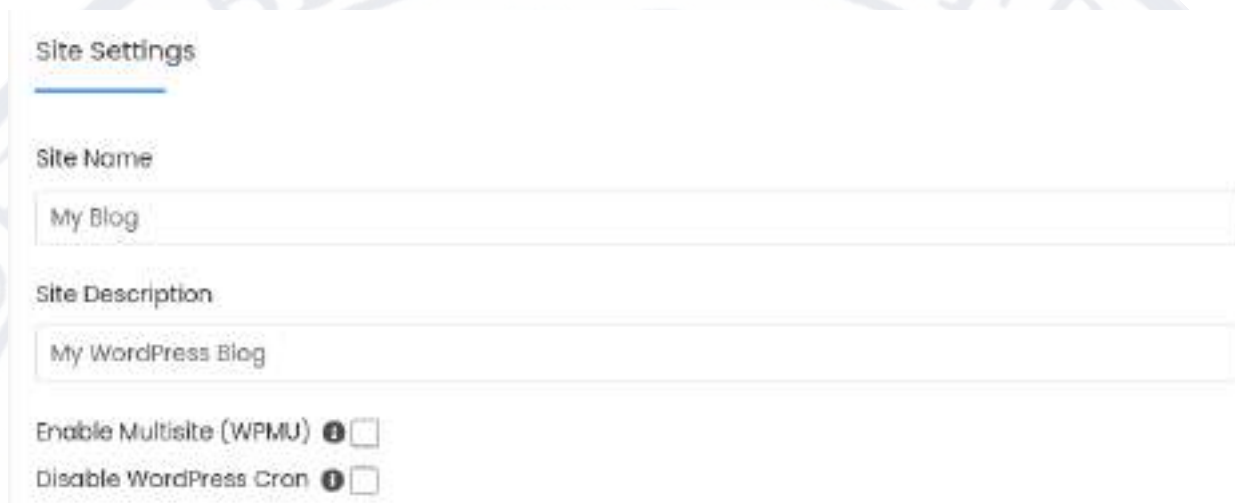
The image shows a screenshot of the 'Site Settings' section in the WordPress admin interface. It features two text input fields: 'Site Name' with the value 'My Blog' and 'Site Description' with the value 'My WordPress Blog'. Below these are two checkboxes: 'Enable Multisite (WPMU)' and 'Disable WordPress Cron', both of which are currently unchecked. The background of the page shows a faint watermark of the Ukrainian coat of arms.

Рис. 3.9. Налаштування опису веб-сайту

Після заповнення форми надається можливість обрати мову інтерфейсу WordPress. У списку доступно багато мов, включаючи українську, що дозволяє користувачам працювати в середовищі з рідною мовою. Вибір мови не впливає на функціональність CMS і може бути змінений у майбутньому.

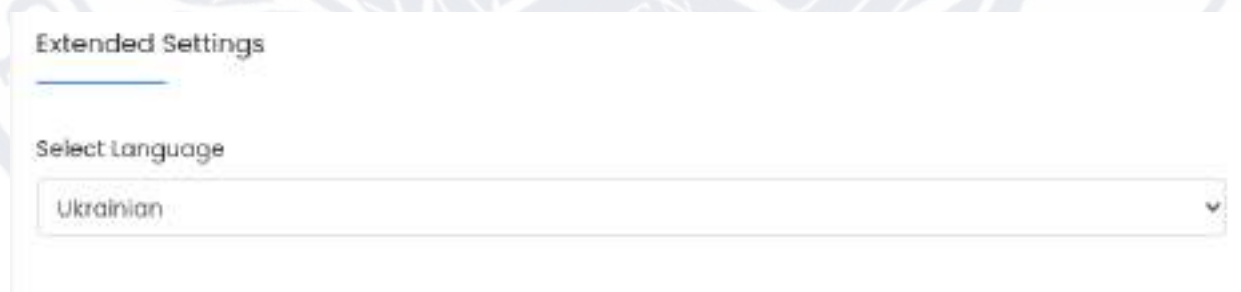
The image shows a screenshot of the 'Extended Settings' section in the WordPress admin interface. It features a 'Select Language' dropdown menu with 'Ukrainian' selected. The background of the page shows a faint watermark of the Ukrainian coat of arms.

Рис. 3.10. Вибір мови для нової інсталяції CMS WordPress

На цьому ж етапі можна активувати додаткові параметри, такі як встановлення базових плагінів для захисту сайту або оптимізації продуктивності. Наприклад, Softaculous часто пропонує встановити плагіни для захисту від спаму або для створення резервних копій даних. Ці плагіни можна встановити відразу або залишити цю задачу на пізніше.

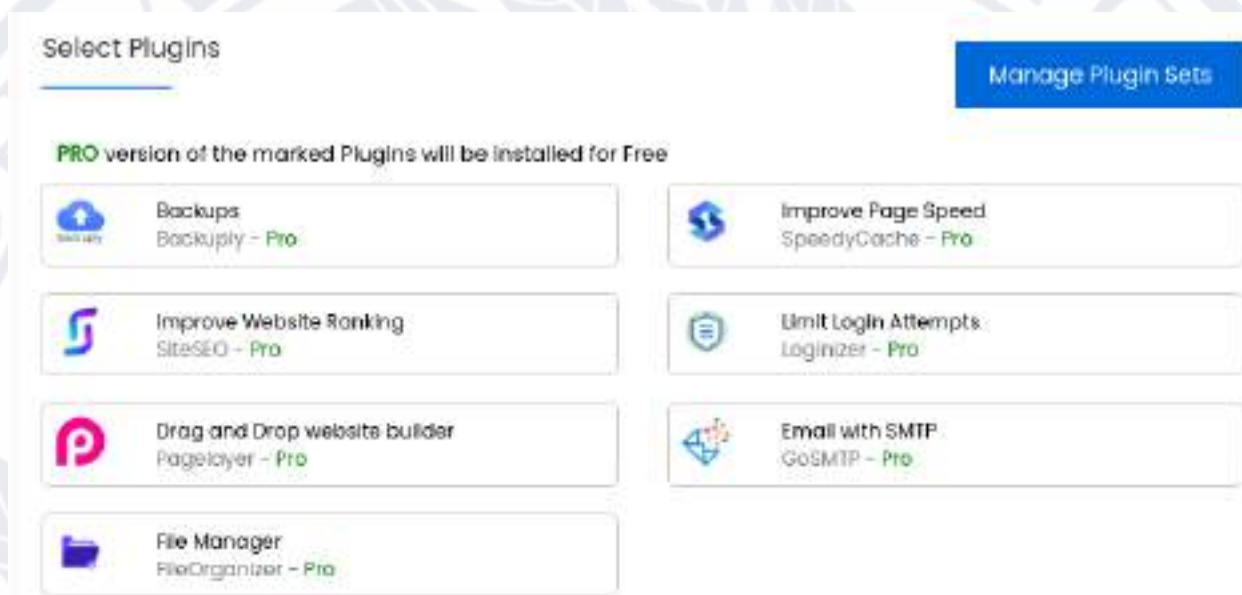


Рис. 3.11. Пропоновані плагіни при інсталяції WP

Завершивши налаштування, користувач натискає кнопку «Install». Softaculous запускає автоматичний процес встановлення WordPress, який займає лише кілька хвилин. У цей час інструмент створює базу даних для сайту, завантажує файли CMS на сервер і виконує всі необхідні налаштування. Після завершення процесу користувач отримує повідомлення про успішну інсталяцію, а також посилання на адміністративну панель сайту і сам сайт.

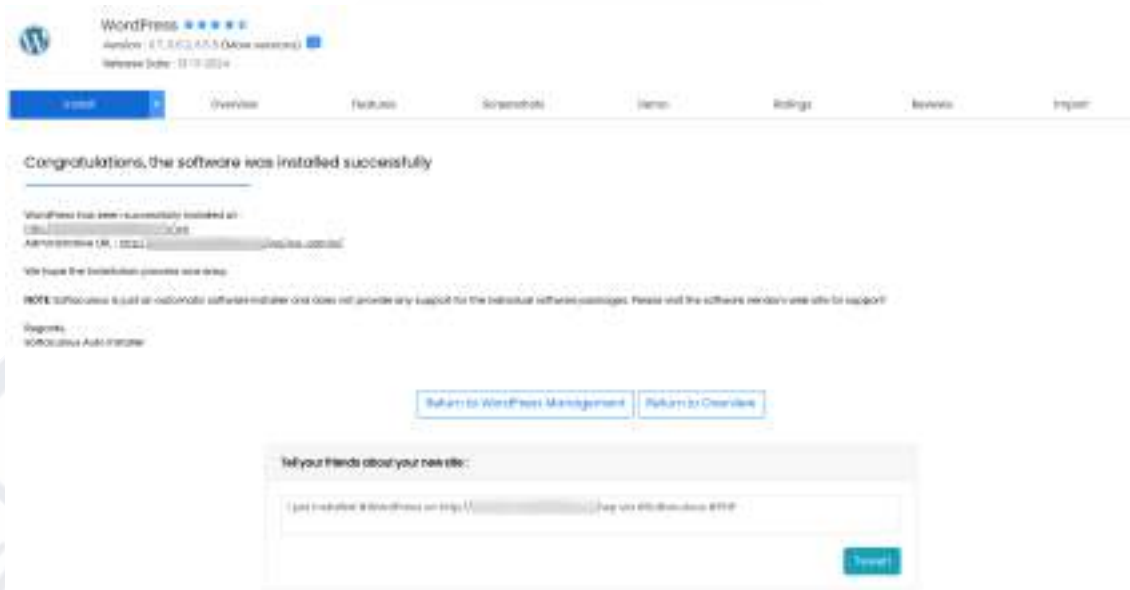


Рис. 3.12. Повідомлення про успішне встановлення CMS WordPress

Адміністративна панель WordPress доступна за URL адресою виду `example.com/wp-admin`. Використовуючи облікові дані адміністратора, створені під час встановлення, користувач може увійти до панелі управління для подальшого налаштування, вибору теми оформлення, встановлення плагінів та додавання контенту.

В результаті, встановлення WordPress через WordPress Manager by Softaculous є простим і швидким процесом, який автоматизує більшість технічних аспектів. Це дозволяє зосередитися на налаштуванні та розвитку сайту, замість витрачання часу на ручне завантаження файлів або конфігурацію баз даних. Інтуїтивно зрозумілий інтерфейс Softaculous значно спрощує процес для початківців і робить його максимально комфортним.

Проектування структури веб-сайту

У рамках цього етапу були виконані основні налаштування системи, обрана тема оформлення, а також створені і налаштовані ключові сторінки сайту.

Після встановлення WordPress першим кроком було налаштування основних параметрів сайту. Це важливий етап, адже правильна конфігурація базових

налаштувань створює фундамент для стабільної роботи сайту. У розділі «Налаштування» адміністративної панелі WordPress були виконані наступні дії:

- Встановлено формат дати як «День.Місяць.Рік» для зручності відображення в українському регіоні. Формат часу обрано у вигляді 24-годинного циклу (наприклад, 14:00), що є більш звичним для цільової аудиторії.
- Налаштовано часовий пояс відповідно до місцевого часу (UTC+2, для Києва). Це забезпечує правильне відображення часу публікацій і планування матеріалів.
- У полі «Короткий опис» було введено текст, який відображає мету і тематику сайту. Цей опис також використовується пошуковими системами для кращого ранжування.

Мова сайту  Українська

Часовий пояс UTC+2

Виберіть місто в тому ж часовому поясі або UTC зрушення щодо часу UTC.

Час UTC: 19.11.2024 20:06

Місцевий час: 19.11.2024 22:06

Формат дати

☐ 19.11.2024 d.m.Y

☐ 2024-11-19 Y-m-d

☐ 11/19/2024 m/d/Y

☒ 19/11/2024 d/m/Y

☐ Користувачський d/m/Y

Попередній перегляд: 19/11/2024

Формат часу

☒ 22:06 H:i

☐ 10:06 PM g:i A

☐ Користувачський H:i

Попередній перегляд: 22:06

[Документація процесу форматування дати і часу](#)

Тиждень починається з Понеділок

Рис. 3.13. Налаштування формату дати, часу й часового поясу

- Додано іконку у форматі PNG розміром 512x512 пікселів. Ця іконка використовується у вкладках браузера та закладках, створюючи візуальну ідентичність сайту.

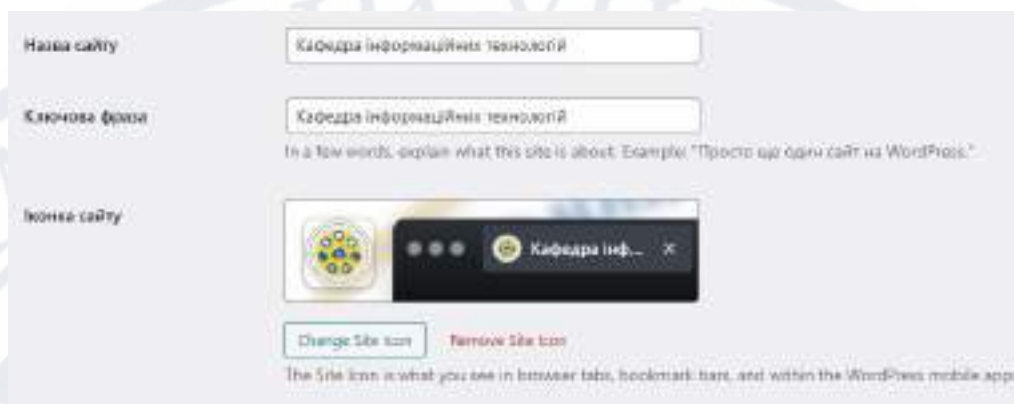


Рис. 3.14. Налаштування іконки, назви сайту

Тема сайту

Одним із ключових моментів у розробці сайту було обрання візуальної теми. У каталозі тем WordPress була знайдена та встановлена тема Kentwood, яка підходить для створення сучасного, естетично привабливого веб-сайту кафедри інформаційних технологій. Ця тема вирізняється адаптивним дизайном, широкими можливостями налаштувань і підтримкою сучасних стандартів веб-розробки.

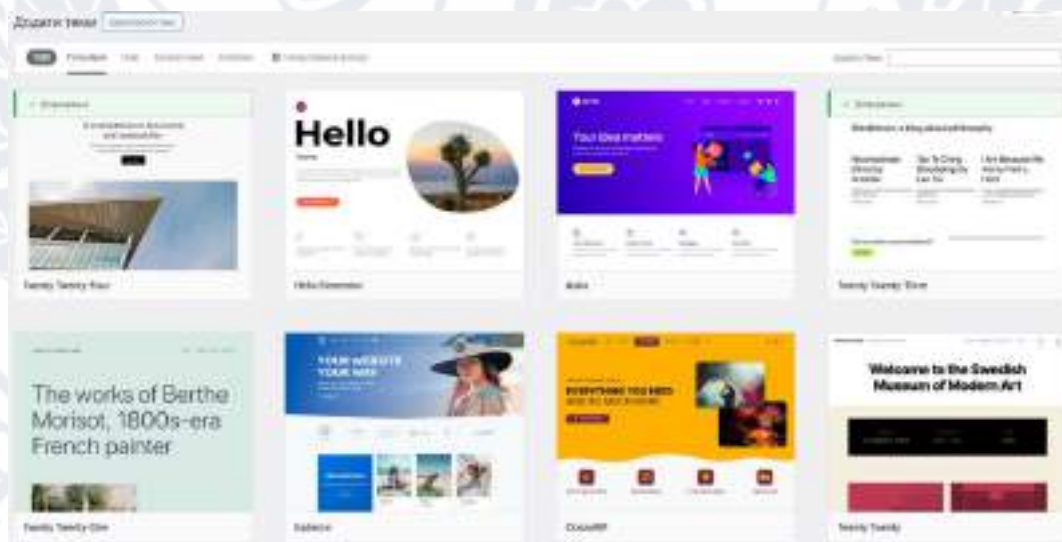


Рис. 3.15. Темі у магазині WordPress

Після активації теми було виконано її базове налаштування. У розділі «Зовнішній вигляд» адміністративної панелі WordPress налаштування проводилися таким чином:

Хедер – ц верхня частина сайту, вона була налаштована як горизонтальне меню. До меню додані основні сторінки сайту, такі як «Головна», «Про нас», «Послуги» та «Контакти». Також у хедері передбачено місце для логотипу та пошукового рядка. Додатково налаштовано фіксацію меню при прокручуванні сторінки, щоб забезпечити користувачам легкий доступ до навігації.



Рис. 3.16. Хедер сайту кафедри

Футер – це нижня частина сайту, що була налаштована для відображення контактної інформації, посилань на соціальні мережі та копірайту. Окрім цього, додано посилання на геолокацію головного корпусу ДонНУ.

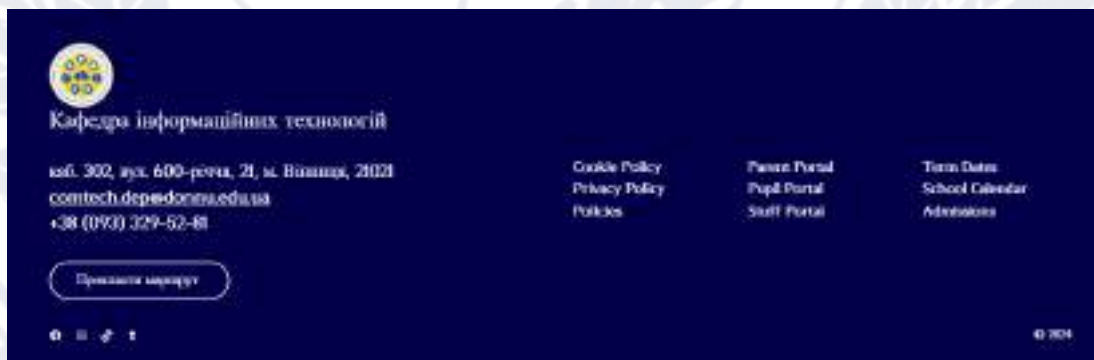


Рис. 3.17. Футер сайту

Було змінено основну кольорову схему теми, щоб вона відповідала фірмовому брендбуку ДонНУ.



Рис. 3.18. Налаштування кольорової схеми теми

Встановлена тема Kentwood пропонує шаблони для створення сторінок, які спрощують роботу над структурою сайту. Кожна сторінка була створена на основі запропонованих шаблонів і адаптована до потреб проекту.

Основна сторінка сайту виконана за принципом односторінкового дизайну з вертикальною структурою. Вона містить слайдер із зображеннями, інформаційні блоки про кафедру.



Рис. 3.19. Елемент головної сторінки сайту

Контент для кожної сторінки був адаптований під формат, запропонований шаблонами. Важливим аспектом було дотримання єдиного стилю: однакові шрифти, розміри заголовків і кольорові акценти. Окрім текстової інформації, на сайті використані якісні зображення, які підкреслюють професіоналізм компанії та додають естетичної привабливості.

3.3 Покращення швидкодії веб-сайту

Оптимізація веб-сайту є ключовим етапом у розробці, що значно впливає на швидкість завантаження, продуктивність і користувацький досвід. Висока швидкість роботи сайту сприяє зменшенню показника відмов, підвищенню рівня задоволеності користувачів і покращенню ранжування в пошукових системах. У цьому підрозділі буде розглянуто проведені оптимізаційні заходи, результати тестування продуктивності та висновки щодо досягнутого ефекту.

Результати початкового тестування

Перед проведенням оптимізацій веб-сайт був протестований за допомогою інструменту Google PageSpeed Insights. Початкові показники продемонстрували, що сайт працює із задовільною швидкістю, але все ще мав значний потенціал для вдосконалення.

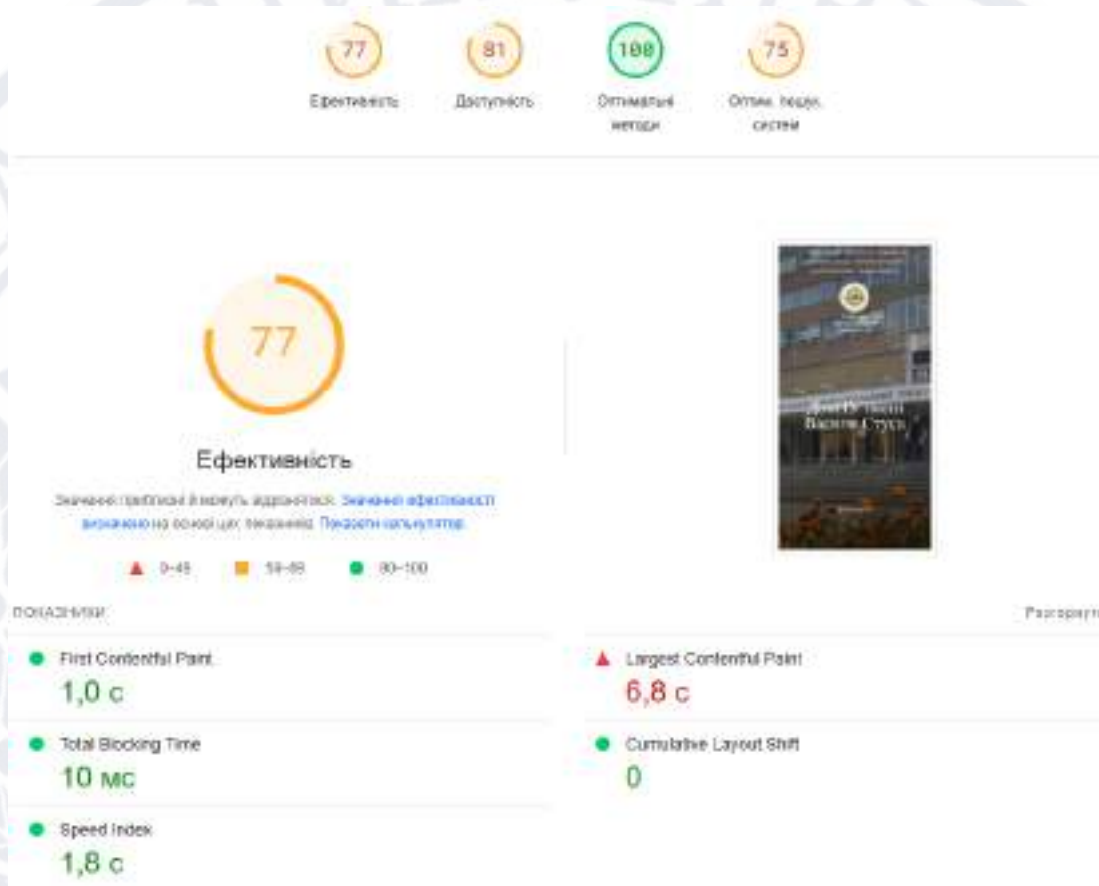


Рис. 3.20. Початкове тестування продуктивності

Опис показників:

Ефективність: 77 (середній рівень).

First Contentful Paint (FCP): 1,0 c – перший елемент контенту завантажувався швидко, що позитивно впливає на перше враження користувачів.

Largest Contentful Paint (LCP): 6,8 c – занадто тривалий час завантаження найбільшого елементу контенту, що є критичним для оцінки продуктивності.

Total Blocking Time (TBT): 10 mc – дуже короткий час блокування, що свідчить про відсутність проблем із JavaScript.

Cumulative Layout Shift (CLS): 0 – стабільність макета ідеальна, без зміщення елементів під час завантаження.

Speed Index (SI): 1,8 с – загальний індекс швидкості досить хороший, але можна покращити.

Основна проблема полягала у високому значенні LCP (Largest Contentful Paint), яке свідчило про довге завантаження найбільш помітного елементу сторінки, найчастіше – великих зображень.

Проведені заходи з оптимізації

Для вирішення проблем і підвищення ефективності сайту було виконано кілька кроків.

1. Оптимізація зображень із плагіном "Image Converter for WebP".

Зображення часто є основною причиною уповільнення завантаження сторінок. Для зменшення розміру файлів без втрати якості всі наявні зображення були конвертовані у формат WebP. Цей формат має значно кращу компресію, порівняно з JPEG чи PNG, що дозволяє зменшити розмір файлів на 25-34% без помітного погіршення якості.

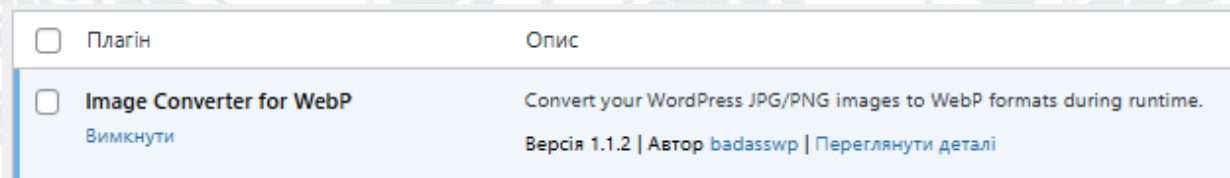


Рис. 3.21. «Image Converter for WebP» у списку плагінів

Крім того, було налаштовано автоматичну конвертацію всіх майбутніх зображень, що завантажуватимуться на сайт. Це забезпечує постійну підтримку високого рівня оптимізації у довгостроковій перспективі.

2. Кешування за допомогою плагіна "W3 Total Cache".

Встановлення та налаштування плагіна "W3 Total Cache" значно покращило швидкість завантаження сторінок. Було активовано такі функції:

- Браузерне кешування: збереження статичних ресурсів на пристрої користувача для зменшення навантаження на сервер при повторних відвідуваннях сайту.
- Стиснення HTML, CSS і JavaScript: зменшення розміру цих файлів для швидшої передачі даних.
- Кешування сторінок: збереження готових HTML-сторінок, щоб уникнути повторної генерації для кожного користувача.



Рис. 3.22. Дашборд W3 Total Cache

В результаті ці заходи значно зменшили час завантаження найважливіших елементів сторінок, оптимізували взаємодію сервер-клієнт і підвищили продуктивність.

Результати повторного тестування

Після впровадження оптимізацій сайт був повторно протестований у Google PageSpeed Insights, і результати виявилися вражаючими.

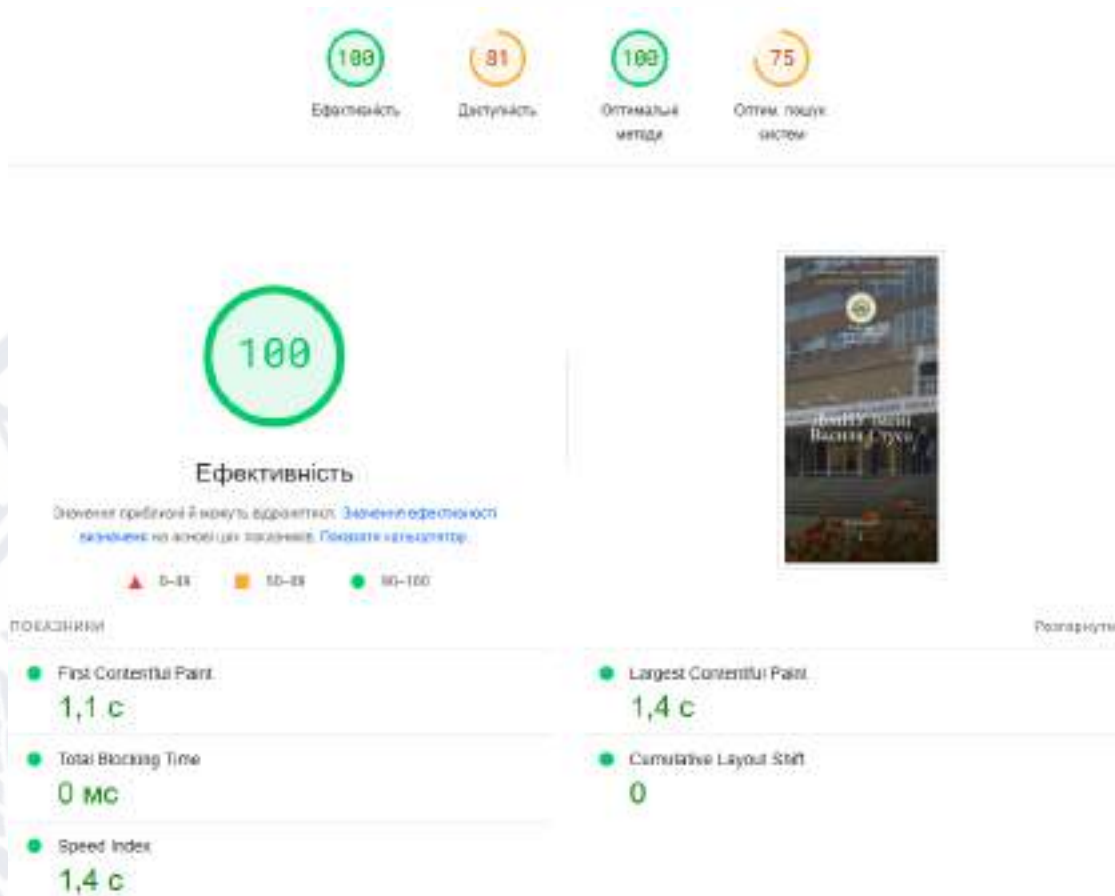


Рис. 3.23. Повторне тестування продуктивності

Ефективність: 100 (ідеальний результат).

First Contentful Paint (FCP): 1,1 с – незначне збільшення порівняно з попереднім тестом, яке не впливає на сприйняття швидкості.

Largest Contentful Paint (LCP): 1,4 с – значне покращення, що свідчить про швидке завантаження великих елементів, таких як зображення.

Total Blocking Time (TBT): 0 мс – ідеальний показник, що свідчить про повну відсутність затримок через виконання JavaScript.

Cumulative Layout Shift (CLS): 0 – стабільність макета залишилася ідеальною.

Speed Index (SI): 1,4 с – зменшення індексу швидкості на 0,4 с, що позитивно впливає на користувацький досвід.

Результати оптимізації демонструють, що навіть початково добре налаштований сайт може бути значно покращений шляхом використання сучасних

технологій та інструментів. Конвертація зображень у WebP дозволила суттєво зменшити розмір файлів, а кешування знизило навантаження на сервер і прискорило завантаження сторінок.

Досягнутий рівень оптимізації (100 балів) є показником того, що сайт відповідає найкращим стандартам швидкості й оптимізації. Це позитивно вплине на користувацький досвід, сприятиме кращому ранжуванню в пошукових системах і зменшенню навантаження на сервер. У підсумку, проведені заходи забезпечили якісний результат, створивши стабільну основу для майбутньої роботи сайту.

3.4 Налаштування безпеки веб-сайту

У сучасному інтернет-середовищі кіберзагрози постійно зростають, тому забезпечення безпеки є пріоритетом для кожного веб-ресурсу. Для захисту сайту, створеного на WordPress, було впроваджено кілька важливих заходів, які включають використання спеціалізованих плагінів, налаштування серверних функцій та шифрування даних.

Wordfence

Одним із основних інструментів для захисту сайту став плагін "Wordfence Security". Цей плагін є комплексним рішенням для забезпечення безпеки WordPress-сайтів. Він забезпечує багаторівневий захист, включаючи файрвол, моніторинг трафіку, аналіз шкідливого коду та запобігання несанкціонованому доступу. Після встановлення плагіна було активовано його файрвол, який функціонує як бар'єр між сервером і потенційними загрозами. Він блокує спроби злому, DDoS-атаки, сканування уразливостей та інші шкідливі дії.



WAF ModSecurity



ModSecurity – це потужний веб-аплікаційний файрвол (WAF), який використовується для захисту веб-серверів. Його основне завдання – аналізувати HTTP-запити, що надходять на сервер, і блокувати шкідливі запити ще до їх виконання. У cPanel було активовано ModSecurity, який працює у тандемі з веб-серверами Apache та Nginx. Завдяки використанню правил OWASP (Open Web Application Security Project), цей модуль здатний виявляти й блокувати широкий спектр атак, таких як SQL-ін'єкції, XSS (міжсайтове скриптування) та інші загрози.

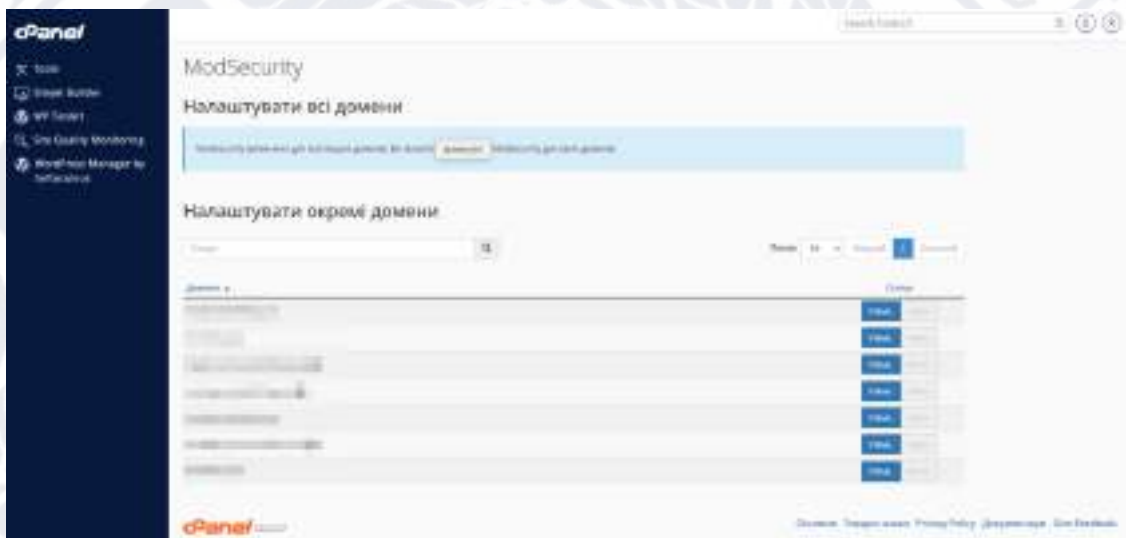


Рис. 3.26. Налаштування «ModSec» у панелі керування cPanel

Одним із найбільш поширених методів атак на WordPress-сайти є спроби підбору пароля (brute force attacks). Для ускладнення таких атак було змінено стандартний логін адміністратора з "admin" на унікальний, який не можна легко вгадати. Це, в парі з надійним паролем, сильно ускладнює підбір авторизаційних даних.

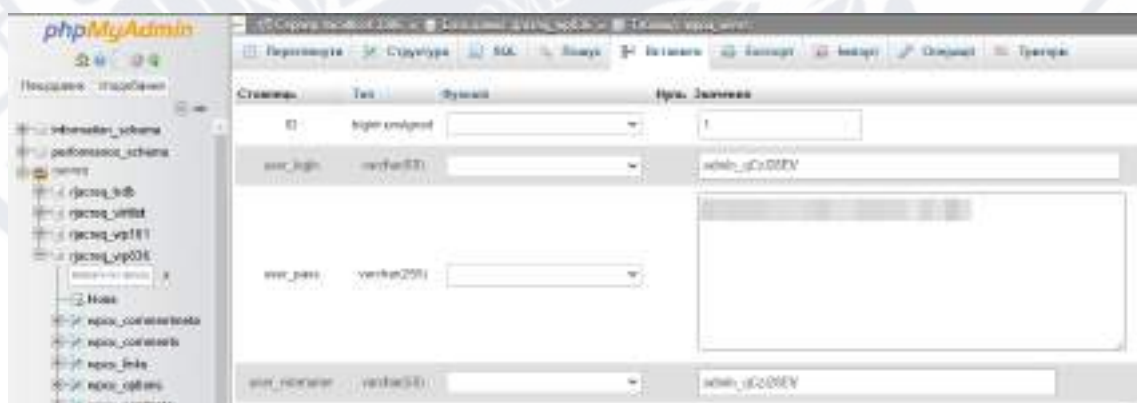


Рис. 3.27. Редагування БД веб-сайту у phpMyAdmin

Hide Login

Додатково з метою захисту адміністративної панелі було встановлено плагін "WPS Hide Login". За допомогою цього плагіна було змінено стандартне посилання на сторінку входу в панель адміністрування WordPress (як правило, це /wp-admin або /wp-login.php) на нестандартне, яке відоме лише адміністраторам. Це унеможлиблює доступ до панелі для автоматизованих ботів, що намагаються здійснити атаки.



Рис. 3.28. Налаштування плагіна «WPS Hide Login»

SSL сертифікат

Для шифрування переданих даних між сервером і користувачами було налаштовано SSL-сертифікат від Let's Encrypt у панелі керування cPanel. SSL (Secure Sockets Layer) забезпечує шифрування всього трафіку між браузером користувача та сервером, що захищає від перехоплення чутливих даних, таких як паролі або платіжна інформація. Використання SSL не тільки підвищує безпеку, але й позитивно впливає на ранжування сайту в пошукових системах, оскільки Google надає перевагу сайтам із HTTPS. Процес активації SSL у cPanel був автоматизованим: сертифікат від Let's Encrypt було встановлено через відповідний інструмент.

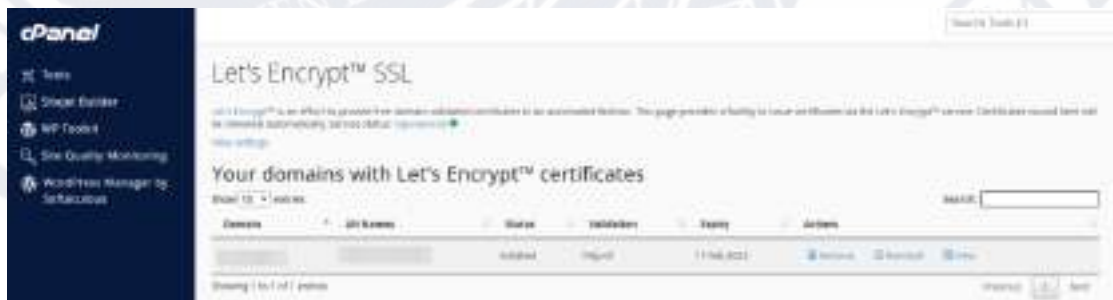


Рис. 3.29. Керування Let's Encrypt SSL сертифікатами у cPanel

Після активації виконано перенаправлення всього трафіку з HTTP на HTTPS, що гарантує використання захищеного з'єднання для всіх сторінок сайту, додавши наступне правило у конфігураційний файл .htaccess:



```
Terminal
This interface provides command line access to your account on the server.
rjacreq
RewriteEngine On
RewriteCond %{HTTPS} !=1
RewriteCond %{HTTP:X-Forwarded-Proto} !https
RewriteRule ^(.*)$ https://%{HTTP_HOST}%{REQUEST_URI} [L,R=301]
```

Рис. 3.30. Перегляд файла .htaccess у текстовому редакторі Vim

ВИСНОВКИ

У рамках виконання кваліфікаційної (магістерської) роботи було проведено комплексне дослідження та реалізовано практичну розробку електронної системи управління діяльністю кафедри. Основною метою було створення сучасного, функціонального, безпечного та зручного інструменту для автоматизації адміністративних процесів і підвищення ефективності роботи кафедри.

На першому етапі роботи було визначено актуальність теми та сформульовано основні вимоги до майбутньої системи. Зокрема, акцент зроблено на адаптивності дизайну, зручності використання, високій продуктивності та можливостях забезпечення безпеки даних. Детально проаналізовано специфіку діяльності кафедри, що дозволило врахувати потреби ключових груп користувачів: викладачів та студентів.

Далі було проведено порівняння методів розробки вебсайтів. Вивчено різні методи розробки, зокрема й популярні системи управління контентом (CMS), що дозволило оцінити їх переваги та обмеження. Особливу увагу було приділено таким критеріям, як простота впровадження, гнучкість у налаштуванні, можливість інтеграції з іншими освітніми платформами та довгострокова підтримка.

Обґрунтованим вибором для реалізації системи стала CMS WordPress. Ця платформа була обрана завдяки її численним перевагам, включаючи широкий вибір тем і плагінів, активну спільноту підтримки, доступність як для досвідчених розробників, так і для звичайних користувачів. WordPress також пропонує можливість налаштування сайту відповідно до специфічних потреб кафедри, забезпечуючи зручний інтерфейс та інструменти для створення інтерактивного контенту.

Практичний етап роботи включав встановлення WordPress, огляд серверного середовища, забезпечення безпеки вебсайту та його оптимізацію для швидкої роботи. У системі було реалізовано адаптивний дизайн, який гарантує коректне відображення вебсайту на різних пристроях, включаючи смартфони, планшети та

комп'ютери, впроваджено функціонал для зручної взаємодії користувачів із системою: розділи вебсайту у хедері й футері, дані для зворотного зв'язку, інформаційні сторінки тощо.

Розроблений вебсайт сприяє автоматизації багатьох рутинних процесів, як от розподіл завдань, обмін інформацією, зберігання даних та їх обробка. Система також забезпечує покращення комунікації між викладачами та студентами, що є важливим для ефективного функціонування певного підрозділу освітнього закладу.

У результаті виконаної роботи створено сучасний вебсайт, який повністю відповідає вимогам електронної системи управління діяльністю кафедри. Проект готовий до впровадження в реальну експлуатацію і може бути адаптований для потреб інших структурних підрозділів закладу освіти. Надалі є можливість розширення функціоналу системи та інтеграції з іншими цифровими інструментами, що сприятиме подальшій цифровізації освітнього процесу та вдосконаленню його організації.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Онлайн ресурс «About WordPress». URL: <https://uk.wordpress.org/about/>
2. Онлайн ресурс «Що таке Joomla!». URL: <https://byshep.com/statti/shho-take-cms-joomla/>
3. Онлайн ресурс «Drupal як Система Управління Сайтом (CMS)». URL: https://www.drupal.org/uk/docs/user_guide/uk/understanding-drupal.html
4. Онлайн ресурс «Що таке Magento?». URL: <https://smile-ukraine.com/ua/magento/introduction>
5. Онлайн ресурси «Переваги та недоліки CMS WordPress». URL: <https://esfirum.com/blog/pros-and-cons-cms-wordpress/>
6. Пономаренко В., Журавльова І., Латишева І. Інформаційні системи в управлінні персоналом : навч. посіб. Харків, 2008, 337 с.
7. Безус А., Безус П. Шевчун. М. (2022). Особливості впровадження інформаційних технологій в сучасних умовах. «Ефективна економіка», №4. doi: 10.32702/2307-2105-2022.4.76
8. Співаковський А. Особливості автоматизованих систем управління вищими навчальними закладами. Вісник Харківського національного університету. 2004 р. №629. Серія «Математичне моделювання. Інформаційні технології. Автоматизовані системи управління», випуск 3. С. 86-99.
9. Воротнікова З. (2019). Застосування сучасних інформаційних технологій у освітньому процесі Europejskie Studia Humanistyczne: Państwo i Społeczeństwo, 3(I). doi: 10.38014/ehs-ss.2019.3-I.04
10. Гринчак О., Моцик Р. (2021). Веб-технології та дизайн. Вісник Хмельницького національного університету, №1, 2021 (293). doi: 10.31891/2307-5732-2021-293-1-22-26

11. Бондаренко О., Ушкаленко І. Безпека web-додатків: актуальні проблеми та їх аналіз. Формування ринкової економіки в Україні. 2017. Вип. 38. С. 28-36.
12. Войтенко О. Проектування інтерактивних медіа систем. Їх розвиток та застосування у сучасному світі. Житомирський державний університет ім. І. Франка, 2012 р.
13. Інститут цифровізації освіти НАПН України. Автоматизовані системи управління для розбудови сучасного інформаційноосвітнього середовища. № 5, 2023
14. Авраменко В., Авраменко А. Проектування інформаційних систем: навчальний посібник. Черкаський національний університет імені Богдана Хмельницького, 2017 р.
15. Клушин Ю., Захарчин Ю. Підвищення швидкості роботи веб-додатків. Комп'ютерні системи та мережі, 2020р., Vol 2. No. 1. doi: 10.23939/csn2020.01.033
16. What is Web Application Security?. Cloudflare. URL: <https://www.cloudflare.com/learning/security/what-isweb-application-security/>
17. Є. Максименко, В. М. Желіховський. – К. : КНТ, 2006. – 280 с.
18. Денисюк В.О. Захист інформації у локальних мережах / В.О.Денисюк, В.В.Письменний // Матеріали Всеукраїнської студентської конференції «Кібернетичне управління економічними об'єктами» (20 квітня 2017 р.). - Вінниця: ВНАУ, 2017. – С.55 -56.
19. Терещенко, В. В. Аналіз сучасних методик пошукової оптимізації (SEO). Вісник Кременчуцького національного університету імені Михайла Остроградського 6 (1) (2015): 48-54.
20. Доліновський Р. Методи захисту веб-застосунку від XSS і CSRF вразливостей. Тернопіль, 2023, 75 с.
21. Стиснення Gzip. URL: <https://www.8host.com/blog/povyshenie-proizvoditelnosti-sajta-s-pomoshhyu-gzip-inginx/>

22. How to Read a Waterfall Chart. URL: <https://gtmetrix.com/blog/how-to-read-a-waterfall-chart-for-beginners/>
23. Time To Last Byte URL: [https://www.dareboost.com/en/doc/website-speed-test/metrics/time-to-lastbytettlb#:~:text=The%20Time%20to%20Last%20Byte,to%20First%20Byte%20\(TTFB\)](https://www.dareboost.com/en/doc/website-speed-test/metrics/time-to-lastbytettlb#:~:text=The%20Time%20to%20Last%20Byte,to%20First%20Byte%20(TTFB))
24. Респонсивний дизайн: як створити сайт, який коректно відображатиметься на різних пристроях та екранах. URL: <https://konotop.city/articles/272786/responsivnij-dizajn-yak-stvoriti-sajt-yakijkorektno-na-riznih-pristroyah-ta-ekrana>
25. PageSpeed URL: <https://developers.google/speed/pagespeed/>
26. W3C. (n.d.). Web Performance Working Group URL: <https://www.w3.org/webperf/>
27. Що таке SEO? Значення, приклади та як оптимізувати свій сайт. URL: <https://www.semrush.com/blog/what-is-seo/>
28. WordPress Codex. Official WordPress Documentation. URL: <https://codex.wordpress.org>
29. MySQL Documentation. The official guide for MySQL users. URL: <https://dev.mysql.com/doc/>
30. Apache HTTP Server Documentation. URL: <https://httpd.apache.org/docs/>.
31. Nginx Documentation. URL: <https://nginx.org/en/docs/>.

ДЕКЛАРАЦІЯ

про дотримання академічної доброчесності

Я, _____

Повністю вказується ПІБ та статус (освітня (освітньо-наукова) програма – для здобувачів вищої освіти, назва кваліфікаційної роботи)

що нижче підписалась/підписався, розуміючи та підтримуючи загальновизнані засади справедливості, доброчесності та законності,

ЗОБОВ'ЯЗУЮСЬ:

дотримуватися принципів та правил академічної доброчесності, що визначені законодавством України, локальними нормативними актами Донецького національного університету імені Василя Стуса, положеннями, правилами, умовами, визначеними іншими суб'єктами, та не допускати їх порушення.

ПІДТВЕРДЖУЮ:

що мені відомі положення статті 42 Закону України «Про освіту»;

що у даній роботі не представляла/представляв чийсь роботи повністю або частково як свої власні. Там, де я скористалася/скористався працею інших, я зробила/зробив відповідні посилання на джерела інформації;

що дана робота не передавалась іншим особам і подається вперше, не порушує авторських та суміжних прав закріплених статтями 21-25 Закону України «Про авторське право та суміжні права», а дані та інформація не отримувались в недозволений спосіб.

УСВІДОМЛЮЮ:

що ця робота може бути перевірена університетом на плагіат або інші порушення академічної доброчесності, в тому числі з використанням спеціалізованих сервісів;

що у разі порушення академічної доброчесності, до мене можуть бути застосовані процедури, передбачені законодавством України та Кодексом академічної доброчесності та корпоративної етики Донецького національного університету імені Василя Стуса, іншими локальними нормативними актами університету, та я можу бути притягнута/притягнутий до академічної відповідальності.

(дата)

(підпис)

