

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ

КАФЕДРА КОМП'ЮТЕРНИХ НАУК

ВИПУСКНА РОБОТА

на тему:

**«Інформаційний веб-портал "Центру первинної
медико-санітарної допомоги м. Суми"»**

Завідувач

випускаючої кафедри

Довбиш А.С.

Керівник роботи

Шелехов І. В.

Студент гр. ІН-63

Халлиєв М.Г.

СУМИ 2020

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ

КАФЕДРА КОМП'ЮТЕРНИХ НАУК

Затверджую _____

Зав. кафедрою Довбиш А.С.

« _____ » _____ 2016 р.

ЗАВДАННЯ
до випускної роботи

Студента четвертого курсу групи ІН-63 спеціальності “Комп'ютерні науки”
денної форми навчання Халлиєва М.Г.

Тема: «Інтелектуальна система ідентифікації особи за підписом»

Затверджена наказом по СумДУ

№ _____ от _____ 2020 р.

Зміст пояснювальної записки: 1) аналіз проблеми. постановка задачі;
2) вибір методу розв'язання задачі; 3) програмне забезпечення.

Дата видачі завдання « _____ » _____ 2020 р.

Керівник випускної роботи _____ Шелехов І.В.

Завдання прийняв до виконання _____ Халлиєв М.Г.

РЕФЕРАТ

Записка: 56 стор., 34 рис., 1 додаток, 11 джерел.

Об'єкт дослідження – процеси проектування інформаційного веб-порталу.

Мета роботи - розробка інформаційного веб-порталу "Центру первинної медико-санітарної допомоги м. Суми".

Методи дослідження — метод проектування інформаційних систем, методи формування та оптимізації структури баз даних, методи забезпечення інформаційної безпеки веб-ресурсів.

Результат - Розроблено програмне забезпечення інформаційного веб-порталу "Центру первинної медико-санітарної допомоги м. Суми" на базі системи керування контентом CMS DataLife Engine. При цьому розглянуто питання забезпечення інформаційної безпеки веб-порталу.

ІНФОРМАЦІЙНА СИСТЕМА, БАЗА ДАНИХ, СИСТЕМА
КЕРУВАННЯ КОНТЕНТОМ, ІНФОРМАЦІЙНА БЕЗПЕКА

ЗМІСТ

ВСТУП	5
1 ІНФОРМАЦІЙНИЙ ОГЛЯД.....	6
1.1 Системи керування контентом веб-сайтів.....	6
1.2 Огляд існуючих систем керування контентом.....	7
1.3 Постановка задачі.....	12
2 ВИБІР МЕТОДІВ РІШЕННЯ ЗАДАЧІ	13
2.1 Вибір програмних засобів	13
2.2 Система управління сайтом CMS DataLife Engine	20
2.3 Розробка шаблону сайту.....	30
3 ПРОГРАМНА РЕАЛІЗАЦІЯ.....	34
3.1 Моделювання інформаційної системи.....	34
3.2 Методи захисту web– сайту	36
3.3 Безпосередня програмна реалізація	42
ВИСНОВКИ.....	48
СПИСОК ЛІТЕРАТУРИ.....	49
ДОДАТОК.....	50

ВСТУП

Web-сайт - це об'єднана під однією адресою (доменним ім'ям) сукупність документів приватної особи або організації. Web-сайти інакше називають Internet-представництвом людини чи організації. Вислів "власна сторінка в Internet", мається на увазі цілий Web-сайт або особиста сторінка в складі іншого сайту.

Раніше для створення нового веб-сайту необхідно було звертатись до команди програмістів та дизайнерів або ж самостійно вивчати основи веб-програмування. Це завдавало значних грошових та часових затрат. Саме тому популярними стали системи керування контентом (СКК), що дозволяють швидко та легко створити блог або портал новин без попереднього вивчення мов програмування. До того ж СКК дозволили зосередитися на змісті інформації, що публікується, а не її оформленні.[1]

Метою даної роботи є створення інформаційного веб-сайту «Центру первинної медико-санітарної допомоги м. Суми», який містить основну інформацію про новини мережі центру.

1 ІНФОРМАЦІЙНИЙ ОГЛЯД

1.1 Системи керування контентом веб-сайтів

Система керування контентом (*Content Management System, CMS*) - це автоматизований редакторський комплекс, що дозволяє керувати змістом і структурою Інтернет-сайту. В основі систем керування контентом закладена ідея розділення дизайну Інтернет-сайту і його контенту. [1-2]

Найпоширеніші функції систем керування контентом:

- створення контенту;
- класифікація та індексування контенту;
- перегляд контенту;
- зберігання контенту в БД;
- підтримка, актуалізація та контроль за змінами;

Складаються CMS зазвичай із двох частин:

- 1) back-end - це частина системи, відповідальна за функціональність і зберігання інформації;
- 2) front-end - це частина системи, що забезпечує інтерфейс з користувачем.

Принцип роботи всіх CMS заснований на поділі контенту (змісту) і дизайну (оформлення) сайту. Звичайно дизайн сайту змінюється рідко, тоді як зміни контенту можуть відбуватися не тільки щодня, але й навіть щогодини. Тому у своїй роботі CMS використовують так звані шаблони - спеціальні порожні заготовки сторінок, у яких дизайн сайту вже прописаний і залишилося лише наповнити їх інформацією. Користувачеві досить скористатися спеціальним редактором. Цей редактор за зовнішнім виглядом дуже схожий на звичні текстові редактори офісних додатків, тому користувачеві не особливо важко освоїти його.

Інформація зберігається в базі даних, наприклад, в MySQL або PostgreSQL і викликається з неї при завантаженні сторінок сайту.

Робота CMS не вимагає установки додаткового ПЗ, оскільки сама система знаходиться на сервері, а доступ до неї здійснюється через звичайний інтернет-браузер. Системи керування контентом підтримують різні найрозповсюдженіші браузери, такі як Internet Explorer, Mozilla Firefox, Opera і т. д.

Як і будь-який інший програмний продукт система керування може бути платною й безкоштовною, з відкритим вихідним кодом або без нього. Переваги платних систем, полягають у постійній технічній підтримці з боку виробника, у постійних оновленнях версій продукту, а також у більш високому рівні безпеки порівняно із безкоштовними продуктами. У той же час системи, розповсюджені безкоштовно, компенсують головний недолік платних CMS - їх вартість.

Доступ до вихідного коду в системах з відкритим кодом дозволяє змінювати деякі функції, розширити можливості CMS, але значно впливає на захищеність інформації веб-ресурсу.

Переваги, які надає використання CMS наступні[1-2]:

- 1) Завдяки наявності вже готових модулів, CMS дають можливість робити зміни в структурі сайтів якісно й оперативно шляхом компонування цих модулів
- 2) CMS пропонують насамперед зручний інтерфейс для редагування вмісту сайту, не вимагаючи спеціальних знань у сфері HTML-верстки і т.д.

1.2 Огляд існуючих систем керування контентом

Система керування контентом - програмне забезпечення для організації веб-сайтів чи інших інформаційних ресурсів в Інтернеті чи окремих комп'ютерних мережах.

Існують сотні, а може, навіть й тисячі доступних CMS. Завдяки їхній функціональності їх можна використовувати в різних компаніях. Незважаючи на широкий вибір інструментальних та технічних засобів, наявних в CMS, існують загальні для більшості типів систем характеристики.

Перші CMS були розроблені у великих корпораціях для організації роботи з документацією. З часом діапазон продукції розширювався і все більше інтегрувався у сучасні мережеві рішення аж до популярних веб-порталів.

Багато сучасних CMS поширюються як безкоштовні і легкі у встановленні (інсталяції) програми, які розробляються групами програмістів під ліцензією GNU/GPL.

Системи управління веб-сайтом часто розраховані на роботу у певному програмному середовищі. Наприклад, система MediaWiki, під управлінням якої працює Вікіпедія, написана мовою програмування PHP і зберігає вміст і

налаштування у базі даних типу MySQL або PostgreSQL; тому для її роботи потрібно, щоб на сервері, де вона розміщена, були встановлені веб-сервер (Apache, IIS чи інший), підтримка PHP та системи керування базами даних MySQL, а також, в разі необхідності, додаткові програми для обробки зображень чи математичних формул. Такі вимоги є досить типовими для відкритих CMS.

Розглянемо деякі з наведених системи керування контентом.[3-5]

1) Безкоштовна CMS Joomla

Переваги системи:

- можливість реалізації практично будь-якого сайту (від сайту-візитки до порталу і інтернет-магазину)
- велика кількість готових дизайнерських рішень
- відносна простота управління контентом

Недоліки системи:

- невисока стійкість до навантаження
- велика кількість готових шаблонів під Joomla спричиняє за собою появу сайтів "близнюків"
- реалізувати на сайті якісь нестандартні рішення у функціоналі або дизайні майже неможливо
- відкритий код і безкоштовність призводить до виникнення проблем у системі безпеки сайту

Приклад інтерфейсу адміністратора зображено на рисунку 1.1

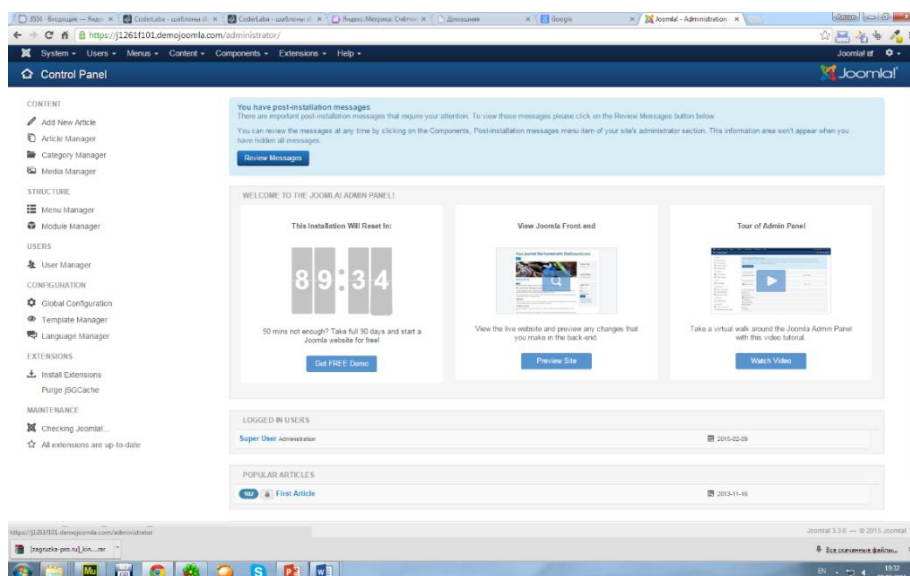


Рисунок 1.1 - Інтерфейс адміністратора системи Joomla

2) Комерційна система керування контентом NetCat

NetCat - професійна система управління сайтами, яка розробляється з 1999 року. NetCat розрахована на створення таких типів сайтів: простих «візиток» з мінімальним контентом, корпоративних представництв та інтернет-магазинів.

Переваги системи:

- керувати вмістом сайту можна як з адміністративного розділу, так і безпосередньо зі сторінок, переключившись в режим редагування;
- CMS NetCat налічує 5 редакцій, кожна з яких сформована з урахуванням необхідного функціоналу для кожного типу сайту. Одна ліцензія системи може обслуговувати необмежену кількість сайтів без будь-яких додаткових витрат, якщо дотримані відповідні пункти ліцензійної угоди.
- мобільність: адаптивний сайт, підстроюється під ширину екрана будь-якого пристрою.
- кожна версія NetCat проходить постійний аудит у фахівців з безпеки веб-додатків, що захищає ваш сайт від зломів і вірусів.

Недоліки системи:

- наймолодша редакція NetCat Standard коштує 2000 грн.
- складний інтерфейс адміністратора

Приклад інтерфейсу адміністратора зображено на рисунку 1.2

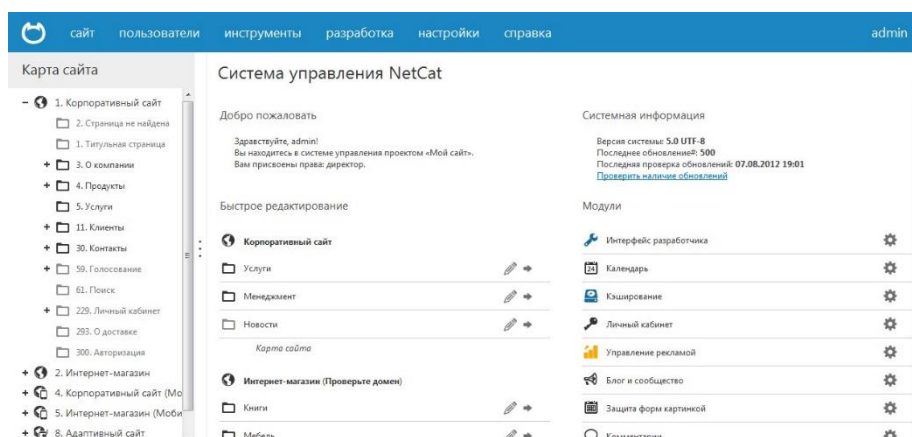


Рисунок 1.2 - Панель адміністратора CMS NetCat

3) Безкоштовна система керування контентом WordPress

Переваги системи:

- найбільш широкий набір плагінів, тим, віджетів для галерей, форумів, мультимовність, різні каталоги, магазини;
- WYSIWYG редактор полегшить життя тим, у кого є проблеми з HTML розміткою;
- адміністративна частина набагато простіше, ніж в інших CMSж

Недоліки:

- необхідно правильно кастомізувати настройки;
- складний процес установки;

Приклад інтерфейсу адміністратора зображено на рисунку 1.3

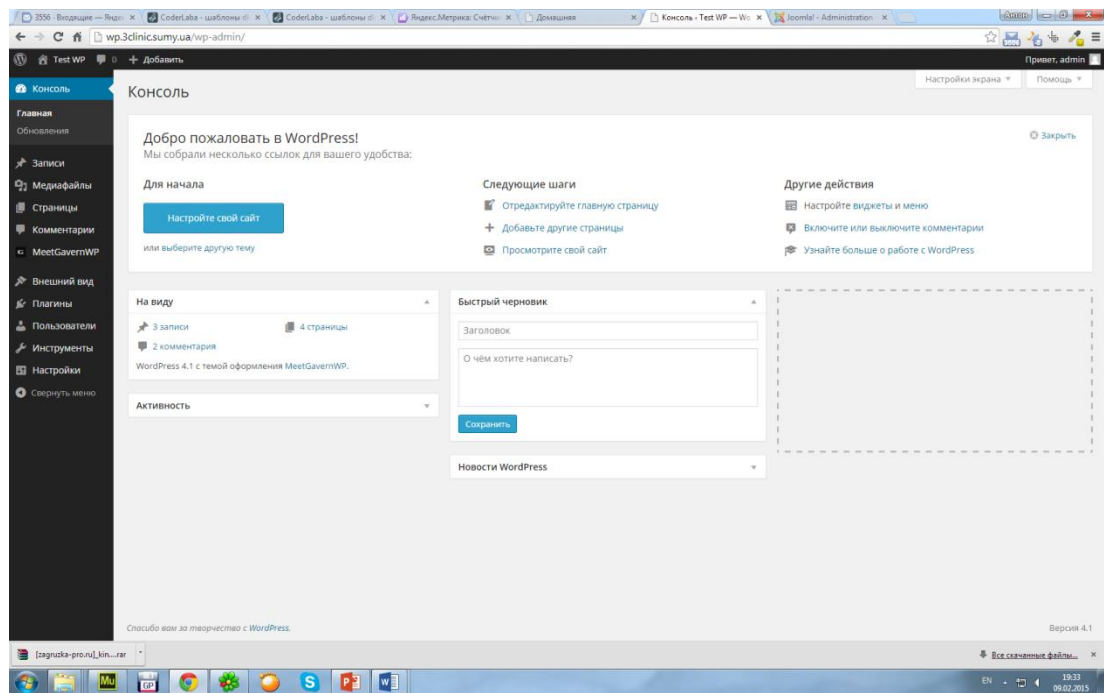


Рисунок 1.3 - Интерфейс администратора системы WordPress

4) Безкоштовна система керування контентом Drupal

Переваги системи:

- наявність хуків (hooks), які дозволяють уникнути практично будь-яких проблем в системі;
- модулі ССК і VIEWS дозволяють конструювати довільні типи даних та їх відображення;

- ця система відома своїм модулем Таксоному, який дозволяє систематизувати контент по рівнях, ознаках і категоріям;
- величезна кількість модулів;

Недоліки системи:

- складний інтерфейс адміністратора;
- drupal вимагає прогресивного технічного обладнання з високою продуктивністю.

Приклад інтерфейсу адміністратора зображено на рисунку 1.4

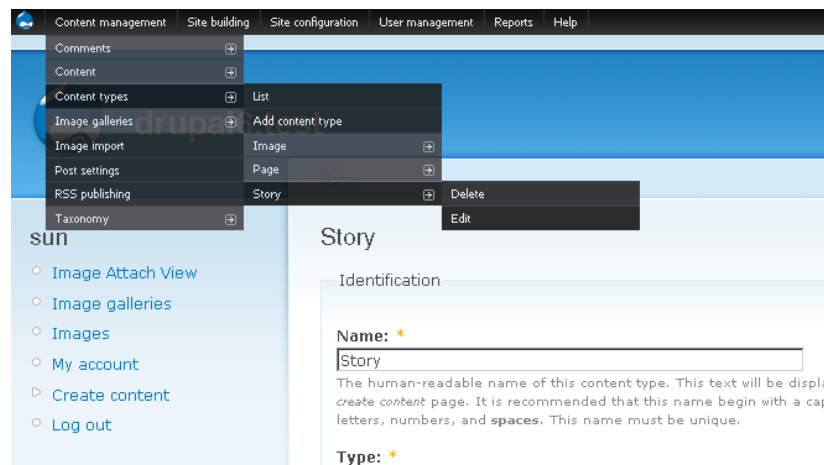


Рисунок 1.4 - Інтерфейс адміністратора системи Drupal

Готова система управління контентом характеризується високою надійністю. Розробляють програмні комплекси професіонали з усього світу. Функціонал систем постійно розширюється, додаються нові можливості для користувачів. Перед тим як випустити чергову версію якої-небудь CMS проводиться її ретельне тестування на наявність помилок. Однак, не дивлячись на це, багато початківців-розробників і навіть деякі досить авторитетні веб-студії пишуть системи управління контентом самостійно. Самописна система позбавлена непотрібного функціонала, безпечна і практично невразлива до атак. Для самописної CMS можна швидко створити додатковий нестандартний функціонал.[5-6]

Переваги самописного движка:

- більш висока швидкість роботи в порівнянні з готовою CMS;
- висока безпека (ніхто не знає «вразливі» місця системи);
- можливість швидкого додавання необхідних функцій.

Слабкі сторони:

- тривалість розробки;
- залежність від розробника. Будь-якому іншому веб-майстру буде вкрай складно або й зовсім неможливо розібратися в коді.

1.3 Постановка задачі

Web-сайт центру первинної медико-санітарної допомоги м. Суми - інформаційний сайт, який є чимось середнім між сайтом-візиткою і корпоративним сайтом. В порівнянні з першим типом має більш широкі можливості, такі як складніший і унікальний дизайн, а також більшу кількість сторінок.

Головним завданням Web-сайту є надання повної і, що важливо, регулярно оновлюваної інформації. Крім якісного текстового наповнення, Web-сайт володіє приємним для сприйняття, яскравим і привабливим дизайном.

Web-сайт центру первинної медико-санітарної допомоги м. Суми завдяки можливостям Internet повинен працювати цілодобово і дозволяти найбільш повно і оперативно інформувати користувачів сайту про додавання нової інформації.

Першим етапом створення web – сайту є проведення детального аналізу інформаційних потреб його потенційних відвідувачів.

До ключових показників якості Web – сайту центру первинної медико-санітарної допомоги м. Суми відносяться:

- кольорова гамма кольорів шаблону Web-сайту;
- простота навігації;
- стабільність інформаційних ресурсів;
- оперативність оновлення інформації;
- цілодобова робота Web-сайту.

Шаблон сайту розробити в білих тонах з використанням додаткових градієнтів (додаткових блисків) та зверстати (розставлення слів та картинок в потрібні місця) за допомогою CSS (каскадних таблиць стилів).

Простота навігації залежить від логічності схеми побудови сайту, яка зазвичай має ієрархічну структуру. Необхідно забезпечити відвідувачам можливість без зусиль рухатися від розділу до розділу, легко повертатися назад або отримати довідку. Ключове значення при цьому має розробка головної сторінки, яка містить головний перелік розділів.

2 ВИБІР МЕТОДІВ РІШЕННЯ ЗАДАЧІ

2.1 Вибір програмних засобів

Для створення WEB-сайту та для подальшого використання, перш за все, було встановлюю один із головних компонентів сайту – VertrigoServ.

OpenServ - це пакет програм, призначених для встановлення та управління повноцінного WEB-сервера Apache, як на домашньому, так і на промисловому комп'ютерному обладнанні.[7-8]

Склад пакету VertrigoServ:

- Apache 2.0.63;
- PHP 5.2.6;
- MySQL 5.0.51b;
- SQLite 3.5.9;
- Smarty 2.6.19;
- PhPMyAdmin 2.11.7;
- ZendOptimizer 3.3.3;
- SQLiteManager 1.2.0.

Apache (HTTP-сервер) - вільний WEB-сервер. Він дозволяє підключати зовнішні модулі для надання даних, використовувати СУБД для аутентифікації користувачів, модифікувати повідомлення про помилки.

PHP - скриптова мова програмування загального призначення, інтенсивно застосовується для розробки WEB-додатків. В даний час підтримується переважною більшістю хостинг-провайдерів і є одним з лідерів серед мов програмування, що застосовуються для створення динамічних WEB-сайтів.

MySQL - вільна система управління базами даних. Зазвичай MySQL використовується як сервер, до якого звертаються локальні або видалені клієнти, проте в дистрибутив входить бібліотека внутрішнього сервера, що дозволяє включати MySQL в автономні програми.

SQLite - модуль забезпечує доступ до вбудованої системи управління базами даних SQLite. SQLite - це бібліотека на мові C, що реалізує вбудований програмний комплекс управління сайтом SQL бази даних. Програми, що використовують

бібліотеку SQLite, можуть використовувати SQL бази даних без використання зовнішнього процесу реляційної системи управління базами даних.

Smarty - компільований обробник шаблонів для PHP, один з інструментів, що дозволяють відокремити прикладну логіку.

PhPMyAdmin - WEB-додаток з відкритим кодом, написаний на мові PHP і представляє собою WEB-інтерфейс для адміністрування СУБД MySQL. phpMyAdmin дозволяє через браузер здійснювати адміністрування сервера MySQL, завантажувати команди SQL і переглядати вміст таблиць і баз даних. Dodatok користується великою популярністю у WEB-розробників, оскільки дозволяє управляти СУБД MySQL без безпосереднього введення SQL команд.[9-11]

ZendOptimizer - один з найпопулярніших PHP-додатків для поліпшення продуктивності та він покращує ефективність функціонування скрипта, так як проводить проміжний PHP код через численні етапи оптимізації, які заміщують неефективні зразки коду ефективними блоками. Ці блоки працюють так само, як оригінальний код, тільки швидше.

SQLiteManager - засіб для управління базами даних SQLite2 і SQLite3. Він дозволяє управляти декількома базами даних, створювати бази, управляти опціями підключення, властивостями і опціями, маніпулювати таблицями, даними, індексами, забезпечує можливість імпортування даних з файлу, перетворення запитів MySQL, управління уявленнями, тригерами та спеціалізованими функціями, конвертувати SQLite2 в SQLite3, а також переглядати структуру бази даних і самі дані.[9-11]

Пакет VertrigoServ безкоштовний та знаходиться у вільному розповсюдженні по мережі Internet.

Для початку встановлення даного пакету, спочатку потрібно скопіювати його з офіційного сайту розробника - <http://vertrigo.sourceforge.net/>.

Після цього починаю встановлювати його, і першим кроком інсталювання даного пакету йде вибір мови:

Далі з'являється вікно привітання майстра інсталяції:

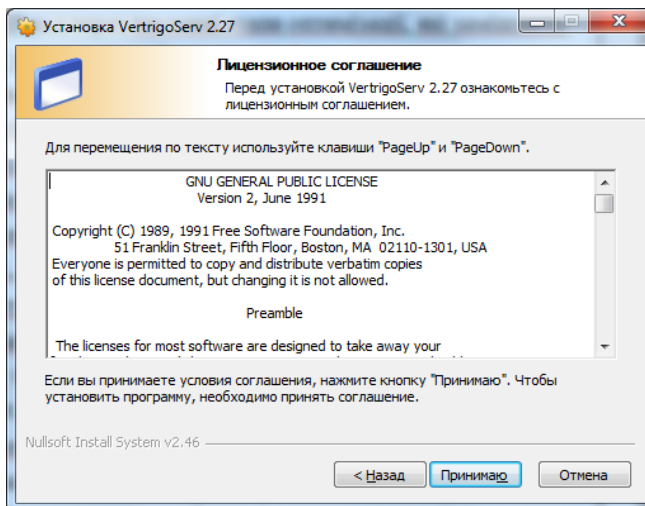


Рисунок 2.1 – Вікно ліцензійної згоди

Наступний крок інсталяції – це задання декількох питань майстром інсталювання про вибір місця для встановлення даного пакету.

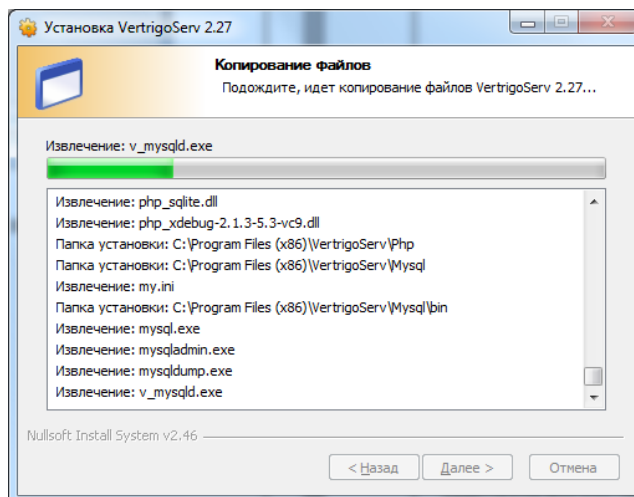


Рисунок 2.2 – Вікно інсталяції пакету програм VertrigoServ

Після вдалої інсталяції пакету VertrigoServ, бачимо вікно, в якому зображено, що пакет встановлений вдало.

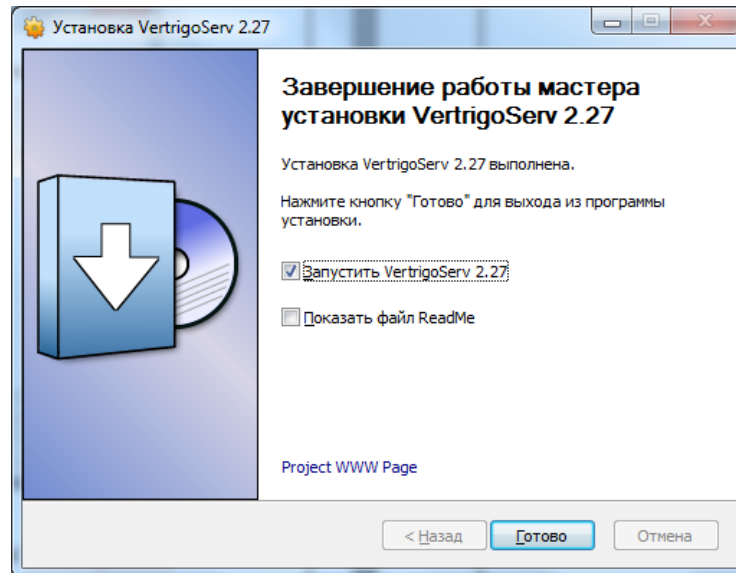


Рисунок 2.3 – Вікно вдалої інсталяції пакету програм VertrigoServ

Тепер web-сервер був успішно встановлений і повністю готовий для подальшого використання, в частині інсталяція системи управління сайтом.

Далі стасимо відмітку в полі «Запустить VertrigoServ 2.27», після натискання на кнопку «Готово» майстер завершує свою роботу і завантажує VertrigoServ на комп'ютері, про що свідчить поява наступного вікна:



Рисунок 2.4 – Вікно старту сервера VertrigoServ

Після запуску сервера VertrigoServ для перевірки на працездатність, в браузері вводимо адрес IP-адресу, за допомогою якої комп'ютер може звернутися по мережі до самого себе, незалежно від наявності у нього підключення до мережі і адреси комп'ютера в ній, тобто – <http://127.0.0.1/> або <http://localhost/>. Якщо все вдало встановлено, то відкривається сторінка з привітанням, про те, що все інстальовано вдало.

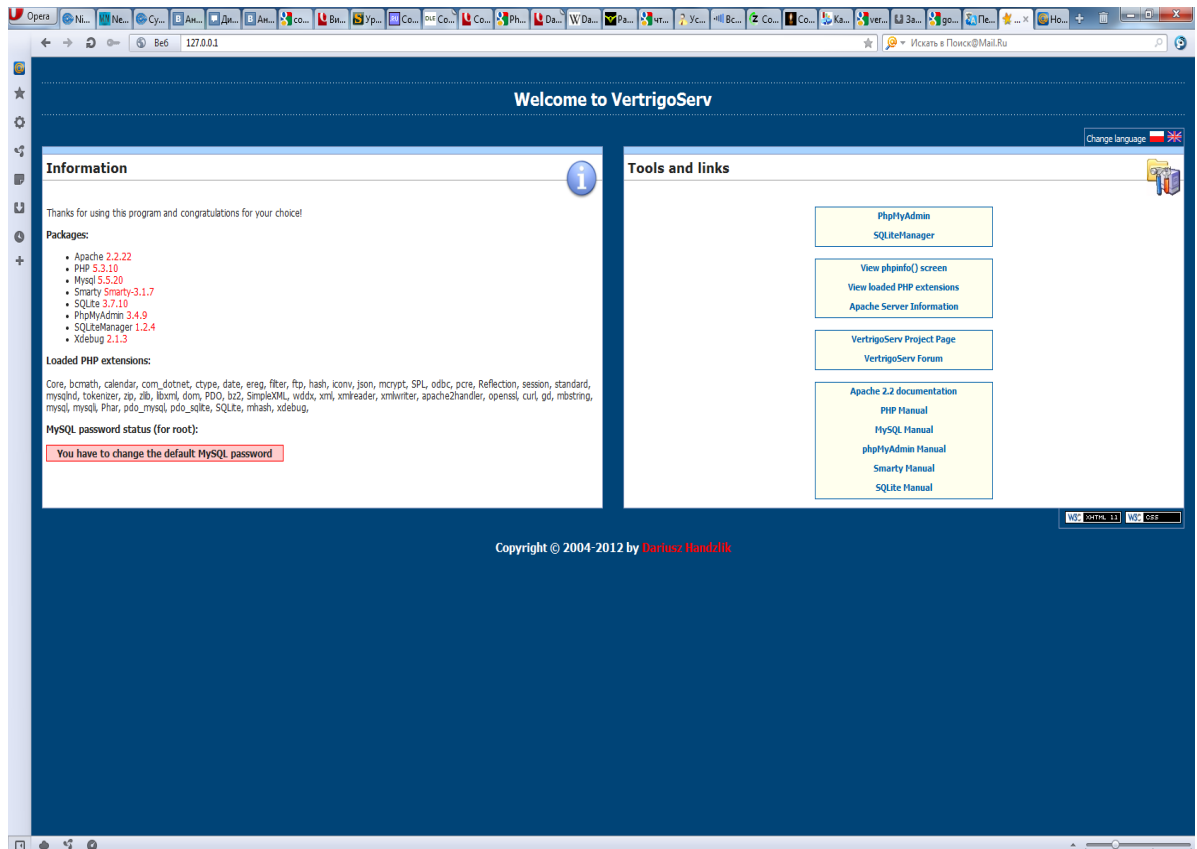


Рисунок 2.5 – Головна сторінка VertrigoServ-а

Також на цій сторінці отримуємо інформацію про склад встановленого комплексу VertrigoServ, отримуємо доступ до посилань для управління базою даних MySQL і посилання для перегляду детальної інформації про встановлений WEB-сервер Apache. Крім того, внизу, ліворуч, бачимо нагадування про те, що мені потрібно змінити встановлений за замовчуванням пароль для доступу до бази даних MySQL.

Для встановлення та зміни пароля, вибираємо розділ – PhpMyAdmin.

Після цього з'являється вікно авторизації в phpMyAdmin, де вводимо дані для входу до системи.

Стандартний логін та пароль для входу в phpMyAdmin:

- login – root;
- password – vertrigo.

Рисунок 2.6 – Форма авторизации в phpMyAdmin

Після успішної авторизації я вхожу в панель адміністрування phpMyAdmin.

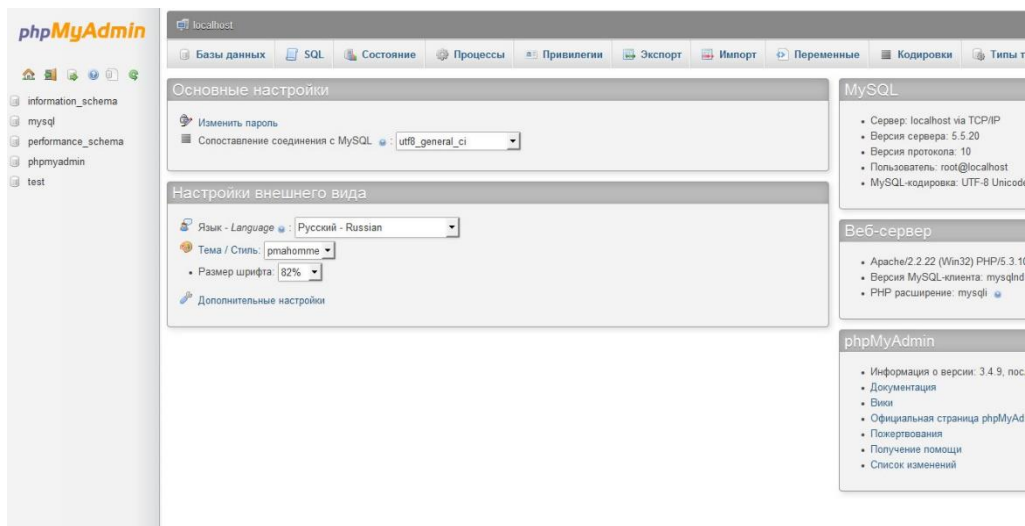


Рисунок 2.7 – Панель адміністрування phpMyAdmin

Далі для зміни стандартного пароля, натискаємо вкладку на панелі адміністрування phpMyAdmin «Привилегии».

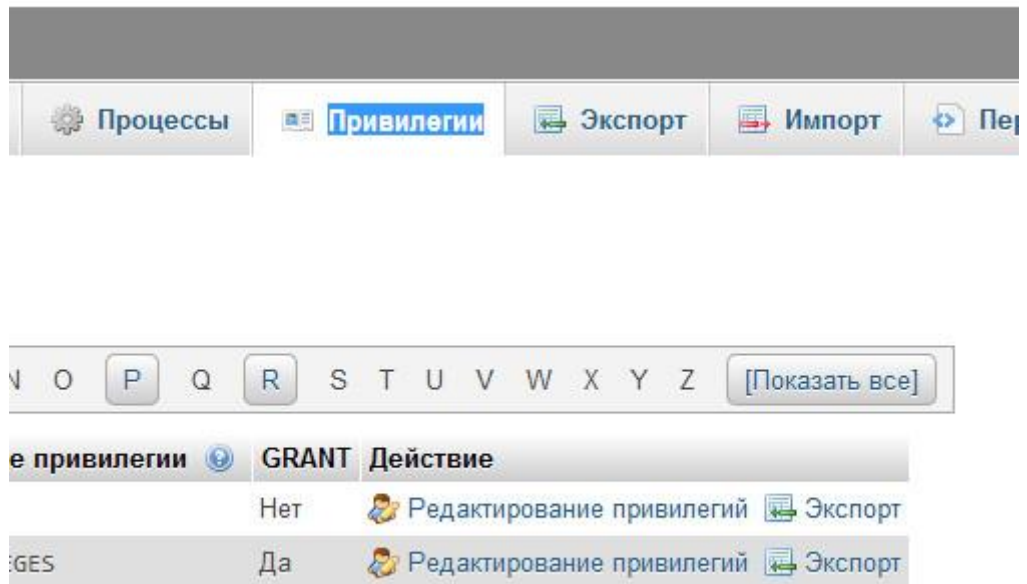


Рисунок 2.8 – Сторінка вкладки «Привилегии»

Наступним кроком для зміни стандартного паролю, натискаємо ссилку «Редактирование привилегий», де з'являється вікно зі зміною пароля.

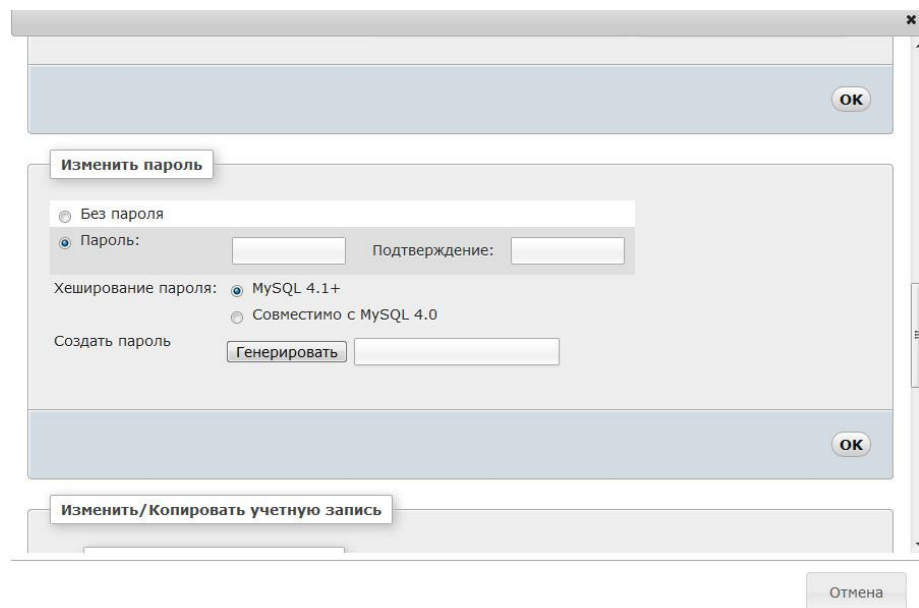


Рисунок 2.9 – Сторінка вкладки «Редактирование привилегий»

Після вдалої зміни стандартного паролю на свій, з'явилося текстове повідомлення, що пароль був успішно змінений і занесений до БД.

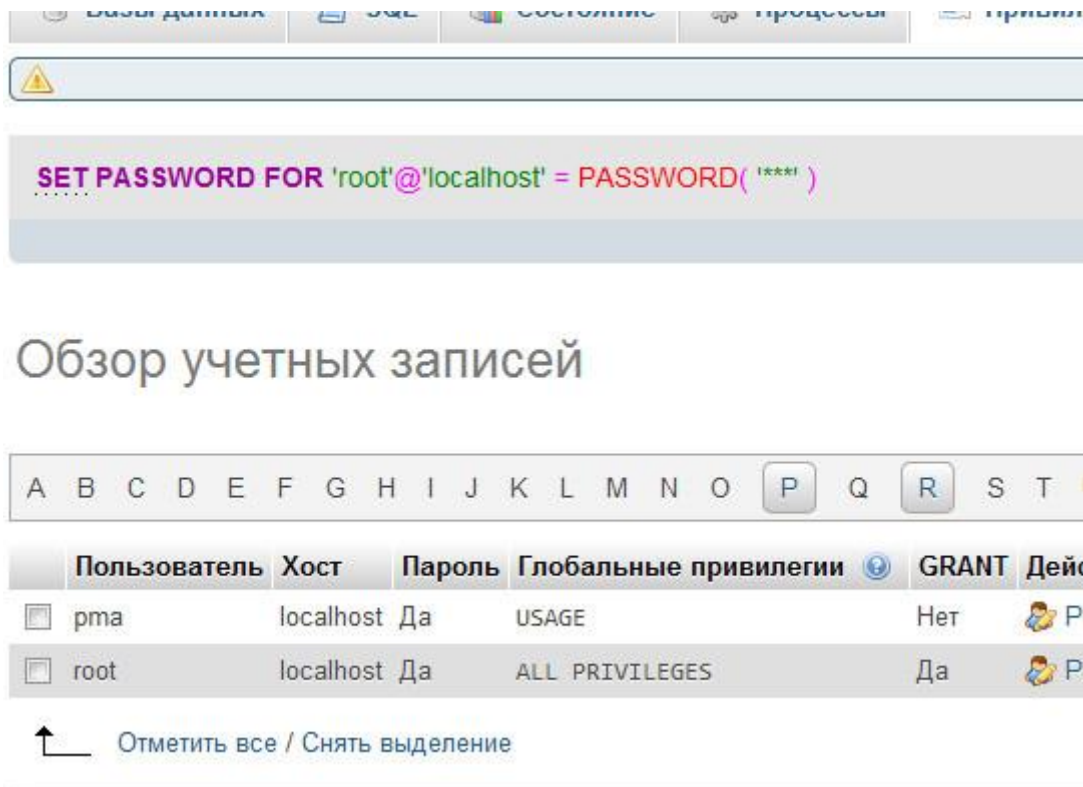


Рисунок 2.10 – Вікно вдалої зміни стандартного паролю

2.2 Система управління сайтом CMS DataLife Engine

DataLife Engine - це багатокористувацький програмний комплекс управління сайтом призначений для організації власних блогів і великих новинних порталів в мережі Internet. Завдяки потужній системі публікації, налаштування і редагування новин, програмний комплекс управління сайтом на крок вперед випереджає своїх найближчих конкурентів. А завдяки точній і продуманій структурі ядра, що зводить до мінімуму вимоги до ресурсів сервера, движок здатний з легкістю працювати з високовідвідуваними проектами, не створюючи при цьому яких-небудь особливих навантажень на сервер. Гнучкість і легкість в налаштуванні і інтеграції власного дизайну дозволить розвернути власні сайти в найбільш стислі терміни, без будь-яких додаткових матеріальних витрат. Використання технології AJAX скорочує як трафік сервера, так і трафік відвідувачів, не кажучи вже про зручність використання даної технології на сайті.



Рисунок 2.11 – Програмний комплекс управління сайтом CMS DataLife Engine

Даний програмний комплекс управління сайтом має дуже великий функціонал:

Загальні можливості:

- використовується для зберігання даних MySQL;
- підтримка розширення MySQLi;
- потужна система безпеки;
- мінімальне навантаження на базу даних (від 0 до 5 запитів);
- використання технології AJAX, що дозволяє скоротити навантаження на сервер і трафік користувачів;
- абсолютно незалежне створення дизайну сайту, скрипт не накладає ніяких обмежень на фантазію Web-майстра;
- потужна, але в той же час проста в освоєнні система шаблонів, яка дозволить створити сайт;
- спрощена система управління шаблонами;
- просунута система управління банерами (або будь-якої рекламної інформації);
- можливість створення не тільки динамічного контенту, але і статичного;
- автоматична й ручна генерація метатегів;

- система поштових шаблонів, на основі яких здійснюється генерація та відправлення повідомлень з сайту;
- можливість тимчасового блокування доступу на сайт;
- виведення новин і статей, та й взагалі всього що завгодно;
- підтримка ЛЗА (людино-зрозумілий URL) дозволяє використовувати тему статті або будь-який інший текст в URL браузера, завдяки цьому індексація сайту в пошукових системах буде якіснішою;
- можливість відключення ЛЗА;
- загальна статистика сайту;
- підтримка додаткових полів (можливість використовувати не тільки коротку і повну новину на сайті, але і є можливість для створення будь-якої кількості додаткових полів різних видів для введення інформації);
- можливість написання статей містять кілька сторінок і показ на сайті з розбивкою на сторінки;
- можливість автоматичного пошуку та демонстрації на сайті подібних за тематикою новин;
- можливість використання вкладених категорій;
- флуд контроль і численні механізми захисту від спаму;
- автоматичний фільтр слів в коментарях;
- підтримка необмеженої кількості категорій, а також підкатегорій;
- автоматичне перенесення довгих слів в коментарях;
- рейтинг новин;
- календар;
- пошук по статтях включаючи додаткові поля (підсвічування знайденого тексту);
- огляд непрочитаних новин з моменту останніх відвідин сайту;
- можливість перегляду скільки разів була прочитана новина;
- можливість додавання статей в розділ "Вибране";
- висновок сторінки використовуючи стиснення gzip, а також без нього;
- підтримка персональних повідомлень між користувачами;
- підтримка декількох мов;

- висновок популярних статей в окремому блоці;
- створення голосувань на сайті;
- можливість створення статичних сторінок безпосередньо через адмінпанель;
- можливість вибору між спрощеною реєстрацією і розширеною, при виборі розширеної реєстрації користувач буде зареєстрований тільки після активації облікового запису. Активація висилається поштою;
- підтримка необмеженої кількості груп користувачів;
- можливість обмеження доступу до різних розділів для певних груп користувачів;
- можливість завантаження і прикріплення файлів до новин;
- вбудовані засоби захисту від несанкціонованого копіювання файлів;
- RSS імпорт новин;
- RSS експорт новин;
- RSS інформери;
- багатомовна підтримка новин на сайті;
- підтримка тегів;
- підтримка єдиної реєстрації на декількох різних сайтах під керуванням DLE;
- автоматична підтримка смартфонів.

Можливості для відвідувачів сайту:

- реєстрація на сайті;
- додавання коментарів;
- виставляння рейтингу новин;
- зміна та видалення користувачами своїх власних коментарів;
- можливість додавання новин з боку відвідувачів;
- модерування новин;
- висновок різної інформації для зареєстрованих відвідувачів;
- можливість завантаження фотографії в профілі користувача;
- можливість відновлення забутого пароля;
- можливість редагування новини безпосередньо на сайті без перезавантаження сторінки з використанням технології AJAX;

- масова завантаження картинок на сервер;
- перегляд і додавання відео на сайт;
- персоналізація виведення реклами для певних груп;
- можливість завантаження великої кількості картинок і файлів в один клік;
- доступ для різних груп користувачів.

Можливості адмінцентру:

- додавання, редагування та видалення новин;
- автоматична друкарська підготовка тексту новин до публікації в Web;
- можливість розподілу доступу для конкретної новини.;
- для створення новин можливе використання WYSIWYG редактора або

BBCODES;

- антивірус перевіряє стан файлів на предмет можливого злому;
- редагування користувачів;
- можливість використовувати смайли і HTML код;
- створення необмежених груп користувачів з можливістю призначення їм різних прав доступу;
- можливість виставляння бану для відвідувачів;
- засоби для створення і редагування шаблонів безпосередньо в адмінцентру;
- можливість коригування часу;
- можливість зміни автора та дати публікації;
- автоматична публікація статті при настанні встановленого часу;
- можливість відключення календаря і архівів (для економії ресурсів);
- можливість фіксації новин (будуть завжди вгорі, незалежно від дати);
- відключення реєстрації відвідувачів;
- автоматична тимчасове припинення реєстрації при досягненні максимальної кількості зареєстрованих користувачів;
- автоматичне урізання завантажених картинок із збереженням пропорцій до вказаних розмірів;
- автоматичне видалення користувачів, які не відвідували сайт заданий час;
- накладення водяних знаків на картинки;
- можливість завантаження картинок для кожної конкретної;

- зручний менеджер завантажених картинок;
- оптимізація, ремонт, створення резервної копії і відновлення бази даних можна здійснювати безпосередньо з скрипта;
- пошук користувачів по IP адресам;
- зручне управління рекламними матеріалами;
- швидкий пошук і заміна в базі даних;
- публікація "правил" на сайті;
- створення карти сайту для Google;
- створення фільтрів для автопідстановки слів і значень.

Програмний комплекс управління сайтом DataLife Engine або скорочено DLE є платним скриптом, і коштує 520 гривень. Для придбання треба зайти на офіційний сайт виробника – <http://dle-news.ru/> та оформити замовлення на придбання даного скрипта, але можна знайти безкоштовно в мережі Internet неофіційну, тобто «пірацьку» копію. На даний момент популярною та найновішою версією програмного комплексу управління сайтом DLE, є версія 9.5.

Щоб не витратити кошти, копіюємо неліцензійну версію з сайту – <http://prowebber.ru/>. Після того, як скопіювали архів з програмним комплексом управління сайтом DLE, розархівували його в кореневу папку WEB-сервера VertrigoServ, тобто в папку WWW і потім знову вводимо в адресний рядок - <http://127.0.0.1/>. Виконавши цей крок, бачимо текстове повідомлення, про те, що програмний комплекс управління сайтом CMS DLE не інстальований та готовий до встановлення:

Перш ніж почати встановлення, створюємо базу даних, для того, щоб виконувалися певні запити при встановленні програмного комплексу управління сайтом. Для цього знову входимо в директорію PhpMyAdmin за адресою – <http://127.0.0.1/phpmyadmin/>, яку вводимо в адресному рядку.

В вікно яке з'явилося, виконуємо авторизацію, після чого входимо в директорію PhPMyAdmin для створення нової бази даних та вибираю вкладку в меню «Базы данных».

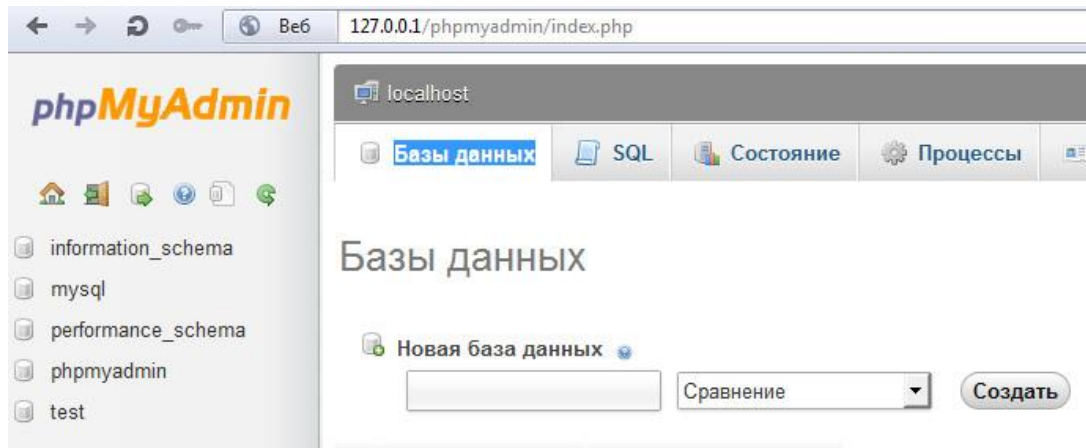


Рисунок 2.12 – Форма для створення нової бази даних

Для створення нової бази даних, в вікоце «Новая база данных», вписуємо довільну назву та натискаємо кнопку «Создать»:

Коли створили базу даних, то побачили текстове повідомлення про успішне створення.

Тепер, коли створилася база даних, переходимо до встановлення програмного комплексу управління сайтом DLE CMS, ввівши в адресному рядку адресу, яка зазначалася раніше – <http://127.0.0.1/install.php>, то бачимо вікно привітання майстра інсталяції скрипта.

Наступний крок - це натиснення кнопки «Начать установку» для початку інсталяції програмного комплексу управління сайтом CMS DLE, після чого з'являється вікно ліцензійної згоди.

Третім кроком інсталяції програмного комплексу управління сайтом CMS DLE - перевірка встановлених компонентів PHP, де перевіряються мінімальні та рекомендовані вимоги скрипта для нормальної роботи.

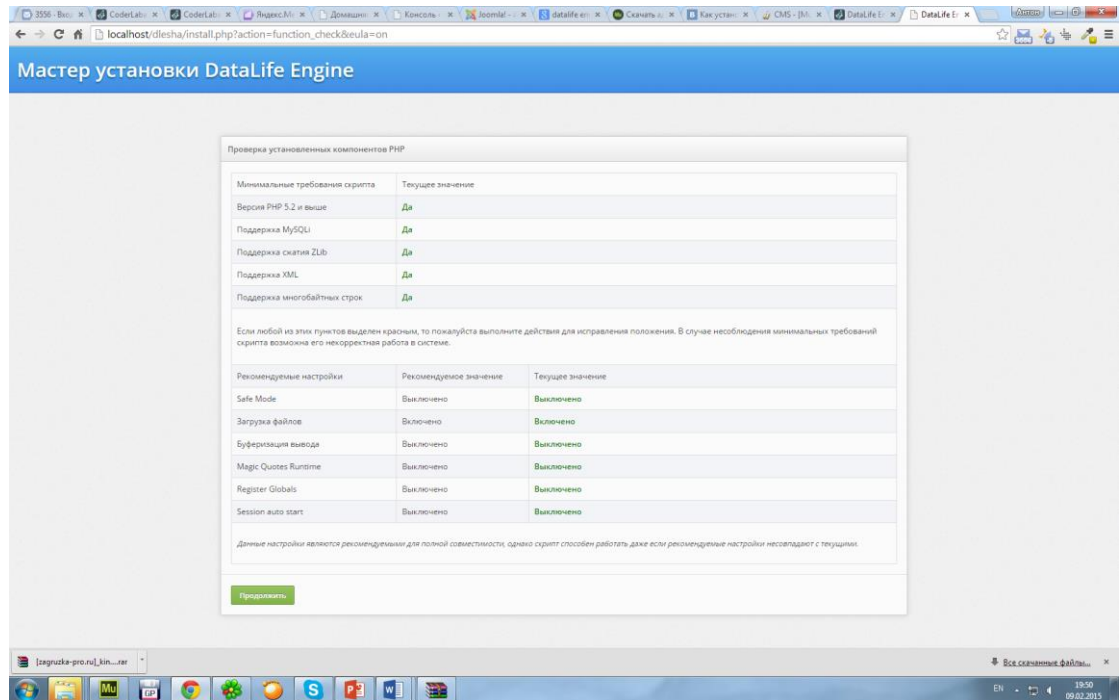


Рисунок 2.13 - Вікно перевірки встановлених компонентів PHP

Після попередніх етапів інсталяції програмного комплексу управління сайтом CMS DLE, проводиться перевірка на запис важливих файлів системи, що значно посилює надійність сайту до зловмисного втручання в файли сайту.

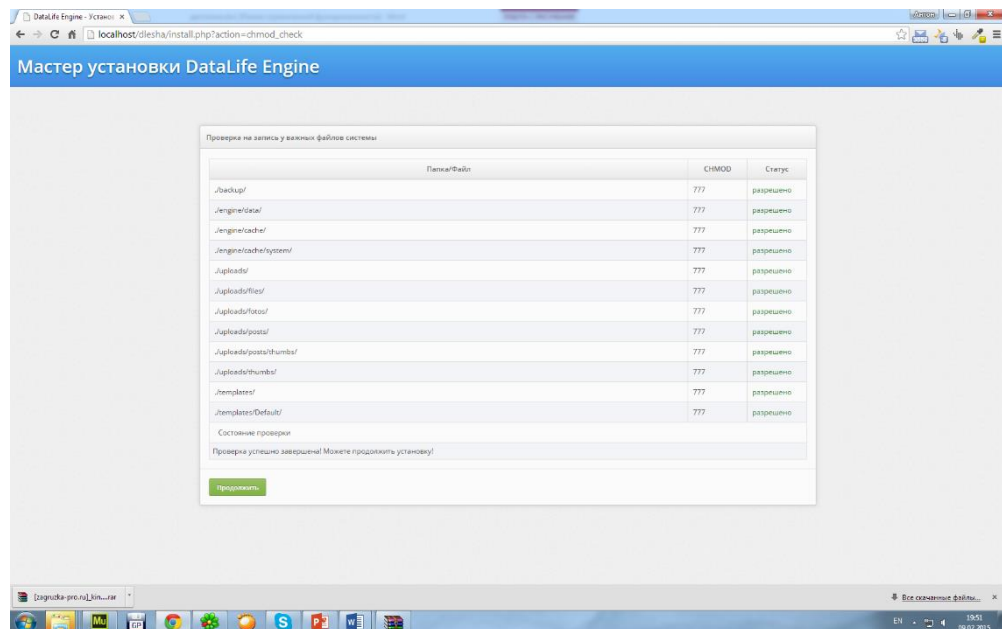


Рисунок 2.14 – Вікно перевірки на запис важливих файлів системи

Наступний етап встановлення – це головний етап, де виконується налаштування конфігурації системи, які вводимо, тобто виконується запис

головних даних для встановлення програмного комплексу управління сайтом CMS DLE.

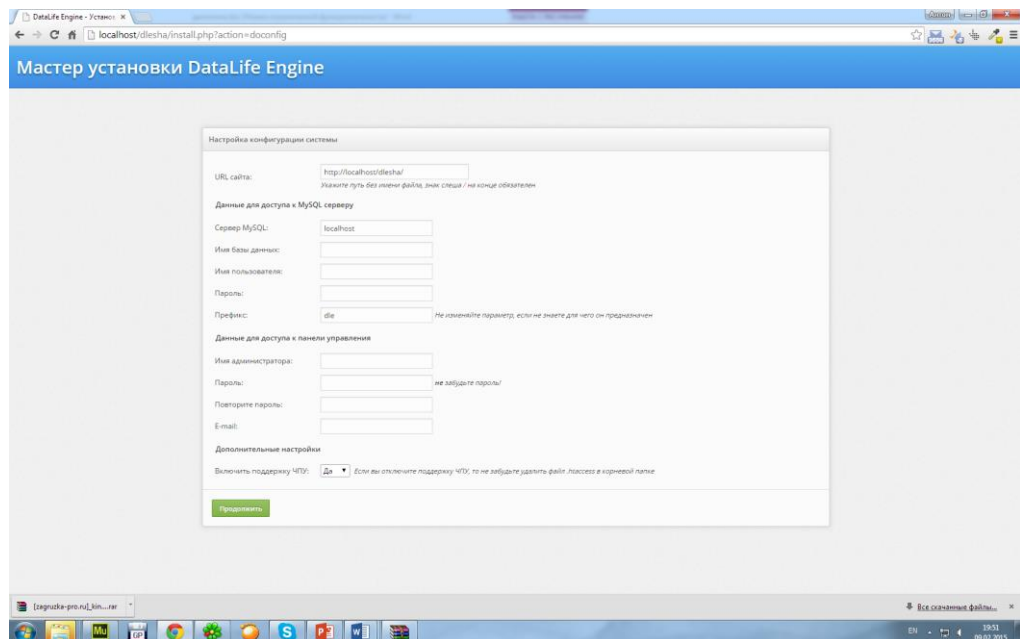


Рисунок 2.15 – Вікно налаштування конфігурації системи

У вікні налаштування конфігурації системи, знаходяться поля для заповнення даними - URL сайта.

Дані для доступу до MySQL серверу:

- сервер MySQL;
- ім'я бази даних;
- імя користувача;
- пароль;
- префікс;
- кодування для MySQL.

Дані для доступу до MySQL серверу:

- ім'я адміністратора;
- пароль;
- повторний пароль;
- e-mail;

Всі ці поля заповнюються вручну.

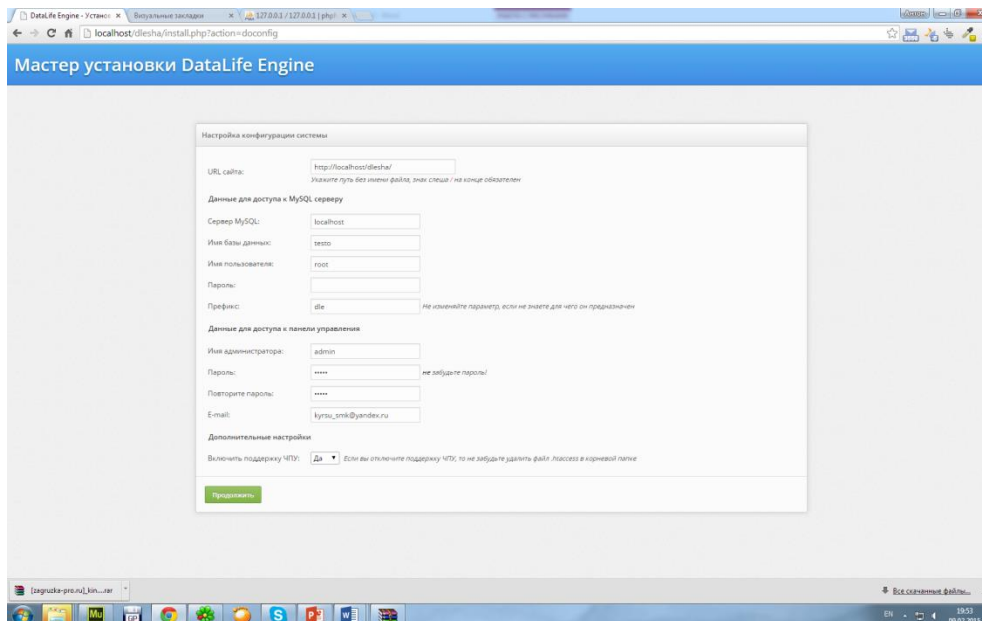


Рисунок 2.16 – Заповнене вікно конфігурації системи даними

Наступним етапом буде встановлення програмного комплексу управління сайтом CMS DLE, тобто занесення даних в базу даних, яку створено раніше. Якщо зроблено все вдало, то бачимо в вікні інсталяції, те що інсталяція движка пройшла успішно.

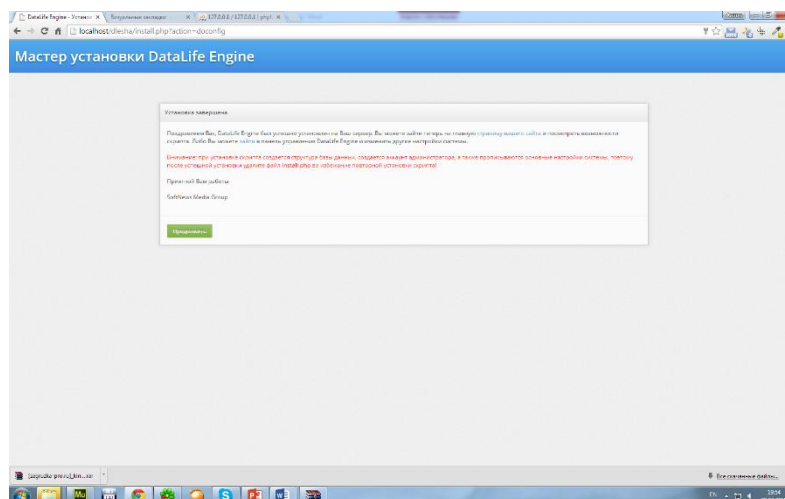


Рисунок 2.17 – Вікно успішної інсталяції програмного комплексу управління сайтом CMS DLE

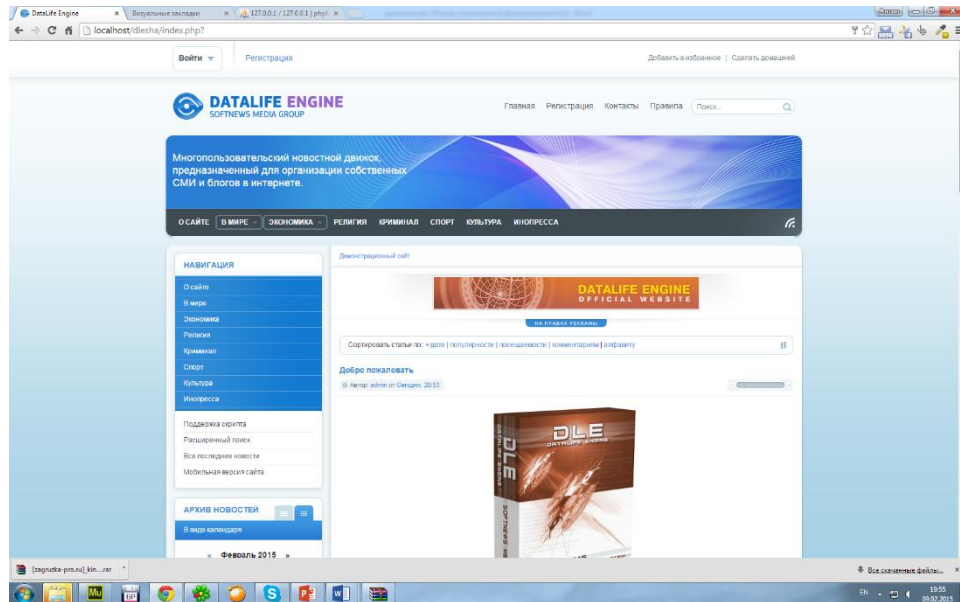


Рисунок 2.18– Головна сторінка сайту на програмному комплексі управління сайтом CMS DLE

Для надійної роботи програмного комплексу управління сайтом, з кореневої папки WWW видаляємо файл `install.php`, щоб збільшити його надійність. Щоб переконатися в тому, що сайт справді працює, вводимо в адресному рядку – `http://127.0.0.1/` та бачимо головну сторінку створеного сайту. На сайті встановлений стандартний шаблон.

2.3 Розробка шаблону сайту

Adobe Muse CC - багатофункціональний графічний редактор, розроблений і поширюваний фірмою Adobe Systems. Продукт займає лідируючі позиції ринку в області створення сайтів. Часто цю програму називають просто Muse. В даний час Photoshop доступний на платформах Mac OS X і Microsoft Windows.

За допомогою даної програми, створюємо макет шаблону, тобто графічне зображення майбутнього шаблону.

Після створення макету, програма його верстає. **CSS** - формальна мова опису зовнішнього вигляду документа, написаного з використанням мови розмітки.

Переважно використовується як засіб опису, оформлення зовнішнього вигляду WEB-сторінок, написаних за допомогою мов розмітки HTML і XHTML,

але може також застосовуватися до будь-яких XML-документами, наприклад, до SVG або XUL.

CSS використовується розробниками WEB-сторінок для задання кольорів, шрифтів, розташування окремих блоків та інших аспектів представлення зовнішнього вигляду цих WEB-сторінок. Основною метою розробки CSS було розділення опису логічної структури WEB-сторінки від опису зовнішнього вигляду цієї WEB-сторінки. Такий поділ може збільшити доступність документа, надати велику гнучкість і можливість управління його поданням, а також зменшити складність і повторюваність в структурному вмісті. Крім того, CSS дозволяє представляти один той самий документ в різних стилях або методах висновку, таких як екранне уявлення, друковане подання, читання голосом (спеціальним голосовим браузером або програмою читання з екрану).

Потім створюємо файл index та базову HTML структуру шаблону і поступово починаю її наповнювати керуючими блоками для різних частин та елементів сайту. Наступним кроком йде створення папки style, де створюємо файли engine.css та main.css. А вже після цього за допомогою даних файлу редагую зовнішній вигляд сайту, доти доки він не буде виглядати в точності і підбиваю під програмний комплекс управління сайтом CMS DLE.

В файлі engine.css, вписуємо конкретні теги та їхні значення, які входять до складу програмного комплексу управління сайтом CMS DLE

Файл main.css містить в собі теги та значення, котрі впливають на оформлення майбутнього шаблону, та закріплюють окремі картинки макету.

Впершу чергу для початку верстання, записуємо до таблиці стилів engine.css перший тег з властивостями «form».

Page - це ідентифікатор, в котрому задаються необхідні параметри для ширини сайту. В одному документі може бути визначено декілька ідентифікаторів. Формат ідентифікатора page виглядає наступним чином:

```
#page
{
    z-index: 1;
    width: 850px;
    background-image: none;
```

```
border-style: none;  
border-color: #000000;  
margin-left: auto;  
margin-right: auto;  
}
```

Також для подальшої верстки майбутнього шаблону використовуємо велику кількість тегів CSS

Після того, як закінчили верстати шаблон, поміщуємо його в папку з програмним комплексом управління сайтом CMS DLE для відображення його в браузері. Для цього входжу в директорію WEB-сервера WWW та поміщуємо папку з шаблоном в папку шаблонів програмного комплексу управління сайтом CMS DLE – templates. Потім, для відображення його на сайті входимо в адмін-центр за адресою – <http://127.0.0.1/admin.php> та вводимо свої дані.

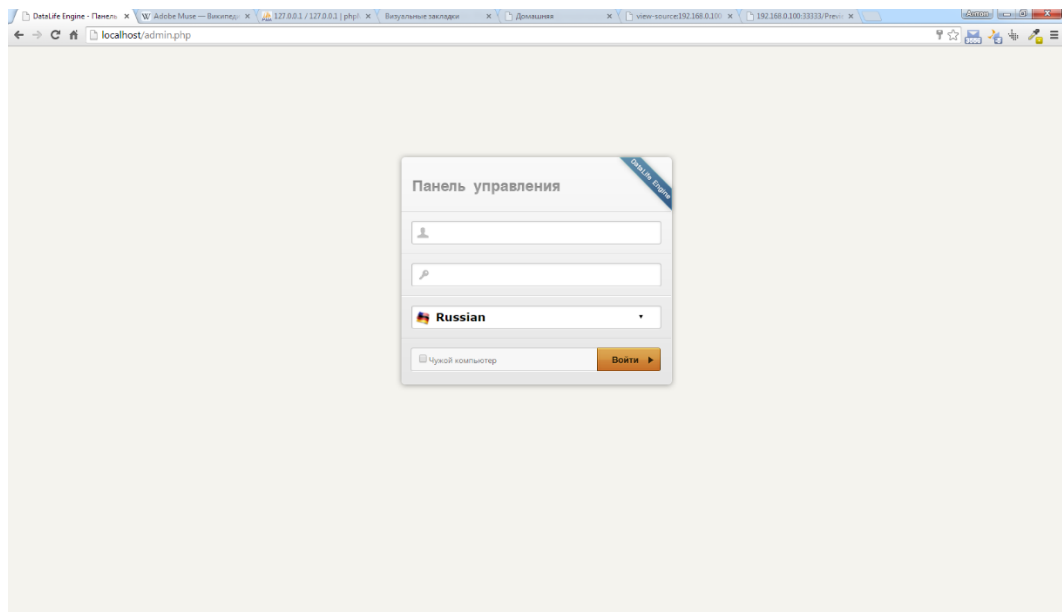


Рисунок 2.19 – Вікно авторизації в адмін-панель програмного комплексу управління сайтом CMS DLE

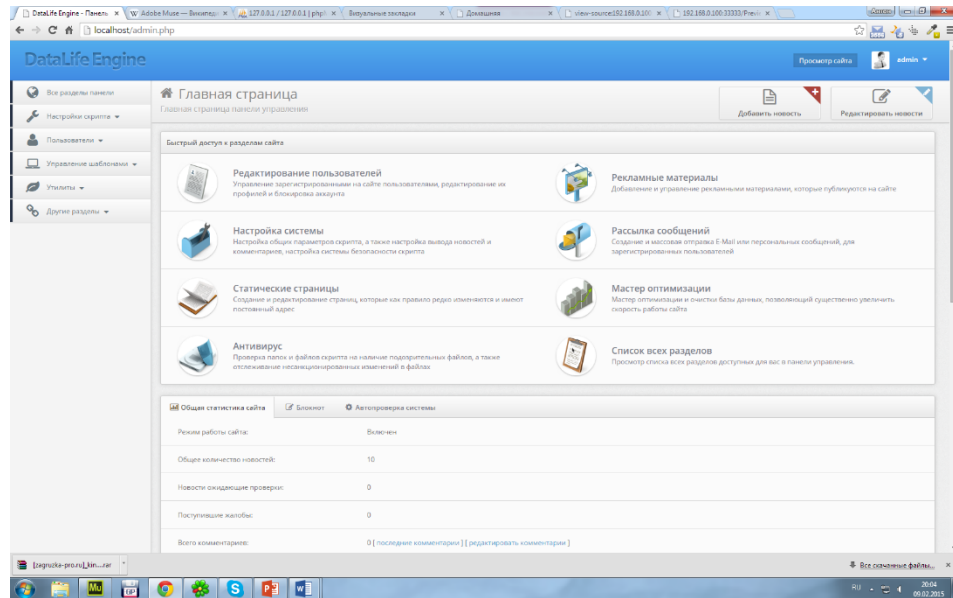


Рисунок 2.20 – Вікно вибору вкладки «Налаштування системи»

Вікно налаштувань містить в собі достатньо великий список основних функцій, який надає система управління контентом DataLife Engine.

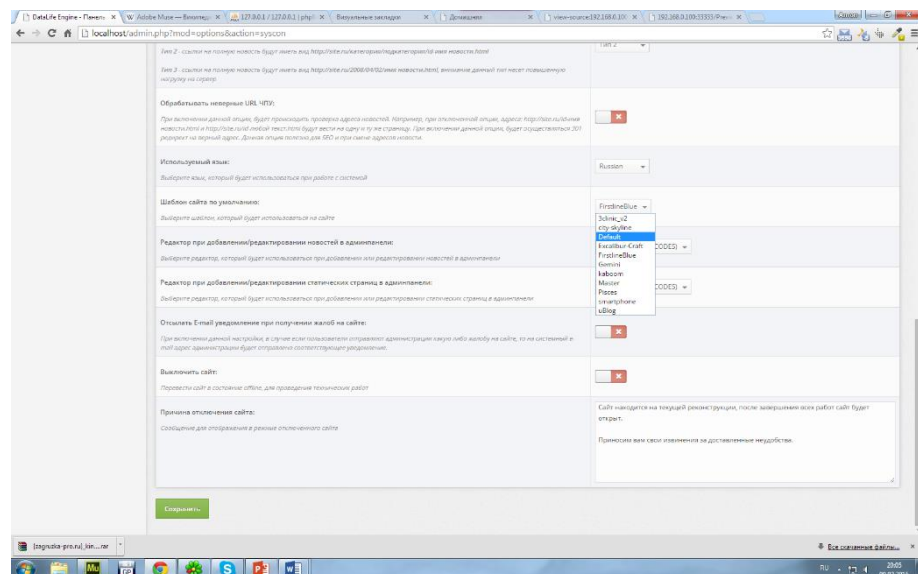


Рисунок 2.21 – Вікно вибору шаблону для сайта

Дана система управління контентом, дає можливість встановлювати додаткові шаблони, для цього треба вибрати необхідну назву шаблону і зберегти налаштування системи.

3 ПРОГРАМНА РЕАЛІЗАЦІЯ

3.1 Моделювання інформаційної системи

Темою роботи є створення Web-сайту «Центру первинної медикосанітарної допомоги №3 м. Суми», для повного створення та розміщення необхідні апаратні та програмні засоби.

Для початку створення встановлюю Web-сервер VertrigoServ, який в своєму складі має всі необхідні компоненти для подальшого створення Web-сайту.

Для того, щоб вибрати конфігурацію сервера, визначити основні характеристики його окремих елементів і потужність, необхідно чітко розуміння того, які завдання вирішуватиме даний сервер. Дійсно, адже FTP-сервер, WEB-сервер і сервер для роботи з базами даних - це зовсім різні сервери з неоднаковими ресурсами і можливостями.

Тому питання про те, який сервер вибрати, вирішується виходячи з того, які завдання він повинен вирішувати. Саме від виконуваних сервером функцій залежить те, який процесор краще всього в ньому застосувати, який обсяг оперативної пам'яті зможе забезпечити нормальну роботу, і скільки дискового простору буде достатньо для зберігання даних. Але вирішення цих питань необхідно довірити професіоналам, які зможуть визначити всі потреби і вивести формулу «ідеального» в кожному конкретному випадку сервера.

Як відомо, сервер - це той же комп'ютер, але він призначений для виконання не широкого кола завдань, а тільки одного конкретного. При цьому сервер відрізняється підвищеною надійністю апаратної та програмної частин, адже ця «машина» працює цілодобово і у край рідко вимикається.

Крім того, для роботи сервера необхідно спеціалізоване програмне забезпечення - починаючи від серверних операційних систем і закінчуючи спеціальними утилітами. Тому стає зрозумілим, що сервер ніяк не може мати таку ж вартість, як і звичайний персональний комп'ютер, а вирішення питання про те, який вибрати сервер, не повинно зводитися до пошуку найдешевшого варіанту - це може мати найбільш негативні наслідки в подальшому.

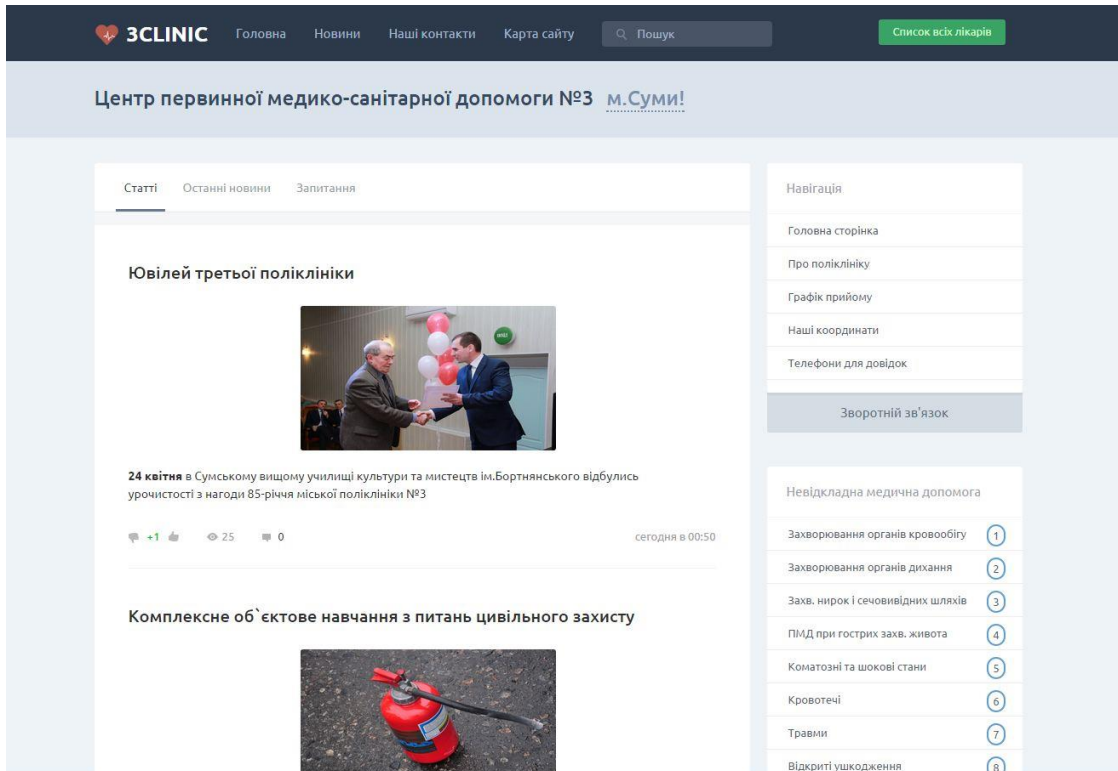


Рисунок 3.1 – Головне вікно Web-сайту

Основним етапом розробки додатку є його моделювання.

На даному етапі створення веб-сайту необхідно задати або спрогнозувати поведінку на сайті користувача. Утримувати їх на сторінках допоможе доступність меню і зрозуміла будь-якому структура сайту. Саме тому ще до початку розробки необхідно скласти бачення проекту з технічної точки зору.

Складання технічного завдання на створення сайту дозволяє отримати чіткі уявлення про те, якою буде структура, інформаційна система і функціонал сайту.

Інформаційна модель розроблюваного веб-ресурсу представлена на рисунку 3.2.



Рисунок 3.2 – Інформаційна модель веб-додатку

3.2 Методи захисту web- сайту

Після створення будь-якого сайту, головною задачею для кожного адміністратора проекту на першому місці з'являється задача захисту власного WEB-сайту. Існує декілька варіантів захисту, якими користується велика культиксть власників сайтів, це контроль зміни IP адреси. Даний метод захисту також широко розповсюджений та входить до складу програмний комплекс управління сайтом CMS DLE.

Для використання даного методу входимо в панель адміністратора за адресою – <http://127.0.0.1/admin.php> та натискаю в меню панель адміністратора посилання «Налаштування безпеки скрипту». Після цього вмикаємо метод контролю зміни IP адреси та зберігаю вибрані налаштування.

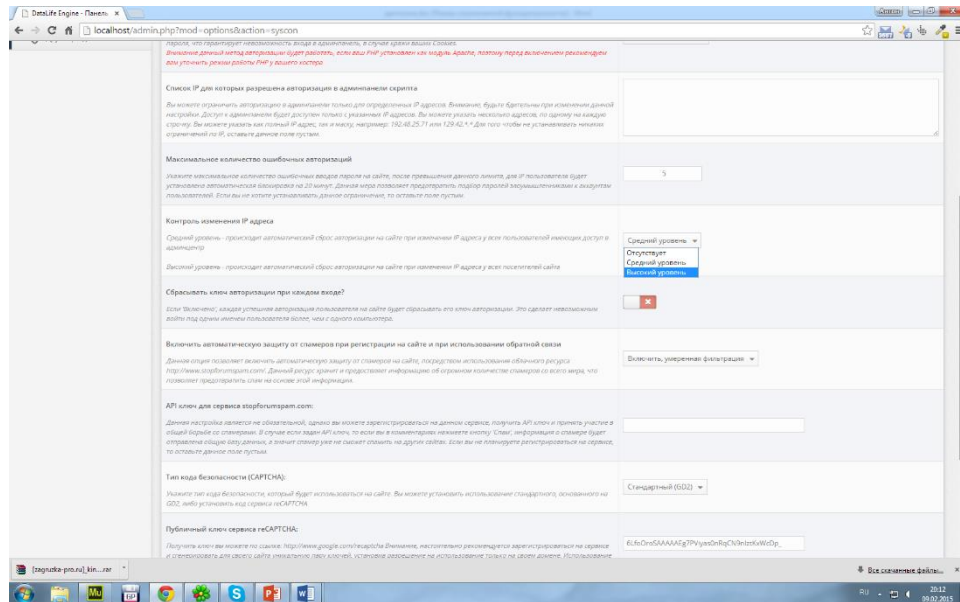


Рисунок 3.3 – Вікно встановлення методу захисту «Контроль зміни IP адреси»

Наступний з популярних методів захисту сайту - це створення захисту від небажаного вторгнення в базу даних SQL. Отже для цього створюємо спеціальну функцію:

```
function inj ($ what, $ inj, $ error) {
    if (ereg ("^[^ $ what]", $ inj)) die ($ error);
}
```

Дану функцію, викликаємо так:

```
- inj ('a-z0-9', $ var, $ error);
```

Опис функції:

- a-z0-9 - символи, які потрібно пропускати (формат регулярних виразів);
- \$ Var - змінна, яку потрібно фільтрувати;
- \$ Error - повідомлення про помилку (помилка з'явиться в разі знаходження сторонніх символів). Тобто бачимо, що дана функція просто шукає сторонні символи в рядку і у разі знаходження видає повідомлення про помилку і негайно припиняє свою роботу.

Після цього, пропонуємо перейти до теми захисту від брутфорсу (підбору паролів).

Брутфорс - метод рішення задачі шляхом перебору всіх можливих варіантів. Складність повного перебору залежить від кількості всіх можливих рішень задачі. Якщо простір рішень дуже велике, то повний перебір може не дати результатів протягом декількох років або навіть століть. Від брутфорсу непогано рятує наступний код:

```
- sleep ($sleep) ;
```

Опис даного коду - \$Sleep - кількість секунд, після яких сценарій зупиняє свою роботу.

Цей код вставляємо перед перевіркою пароля. Наприклад, двох секундна затримка користувачеві не завадить, а от брутфорс застосовувати буде безглуздо. Хіба що, можна використовувати багато комп'ютерів (ботнет) для підбору пароля. Однак і це теж можна вирішити. Наприклад, якщо користувач неправильно ввів пароль десять разів, можна заблокувати його по IP адресі, а посилання для зміни пароля відправити на e-mail користувачеві.

Як варіант - можу написати скрипт, який буде відсилати мені sms - повідомлення у разі надходження занадто великої кількості запитів до сайту, але це займе дуже великий обсяг часу.

Наступний код можна використовувати для знаходження Web-shell.

Web-shell - невелика програмка, яку хакер «заливає» на зламаний web-сервер, щоб мати можливість його віддалено вивчати і адмініструвати.

Інакше кажучи, web-shell надає web-інтерфейс для завантаження консольних команд та перегляду результатів. Web-shell створюються на різних web-мовах, таких як:

- Perl;

- PHP;

- ASP.

Ось так виглядає вміст найпростішого web-shellа на PHP:

```
{{{{{{<? php system ($ _GET ["cmd"] );?>}}}}}}
```

Проте хакери часто використовують більш просунуті shell-и з формами, кнопками. Найбільш відомими з таких web-shell є r57shell і c99.

Для захисту від таких небажаних shell-ів, напишемо невеличку за обсягом функцію:

```
function srd ($ dir, $ fn) {
    if ($ odir = opendir ($ dir)) {
        $ Cdir = '0 ';
        while (false! == ($ file = readdir ($ odir))) {
            if ($ file! = "." && $ file! = "..") {
                $ Cdir + +;
            }
        }
        closedir ($ odir); if ($ cdir! = $ fn) {echo 'Error!';}
    }
}
```

Викликати цю функцію слід так:

```
srd ('test', '15 ');
```

Опис функції:

- test - ім'я папки;
- 15 - число файлів у папці.

Коли викликаємо дану функцію, то починається підрахунок кількості файлів і папок у папці та порівняння цього числа з іншим параметром. Якщо при порівнянні числа виявилися інші параметри - скрипт видає помилку. До речі, до видачі помилки додаю, ще й функцію, яка буде відсилати sms - повідомлення адміністратору. Зрозуміло, наведений код потрібно викликати в основних скриптах сайту. Якщо є папка куди файли постійно додаються, то додаю в написану мною функцію ще й перевірку типу та розширення файлу.

Паролі користувачів краще шифрувати алгоритмом md5 і додавати до складну комбінацію символів. Так доб'ємося того, що розшифрувати пароль буде

майже неможливо, а якщо в cookies при авторизації ідентифікатора використовувати не пароль, а що-небудь ще, то потрапити в адмін-панель сайту стане ще важче.

Останнім з основних методів захисту є захист від DDos - атак.

DDos (відмова від обслуговування) - це атака на ПК або якщо більш широко брати, на обчислювальну систему, основною метою якої є вивести обчислювальну систему з ладу шляхом подачі на обчислювальну систему великої кількості помилкових ресурсоемних запитів. В результаті цього обчислювальна система «зависає» і стає недоступною.

Види DDos - атак:

- недостатня фільтрація даних може привести підвищеного споживання ресурсів, до тривалого або нескінченного циклу, виділення величезного обсягу оперативної пам'яті;

- недостатня фільтрація даних;

- атака іншого роду – подібні атаки призводять до спрацьовування системи захисту, що призводить до недоступності сервера;

- флуд - велика кількість ресурсоемних запитів, звернених до сервера.

Види боротьби з DDos - атаками:

- на рівні сервера. Сервер повинен мати віддалений ребут і висновок консолі сервера на інший ір адресу по ssh протоколу. Це дозволяє швидко перезавантажувати сервер, що буває більше ніж потрібно на самому початку DDos - атаки. Функція виведення консолі дозволить повністю виключити ssh на сервері. Це необхідно тому, що його теж дуже часто DDos-ять разом наприклад з WEB-сервером, що б ускладнити роботу адміністратора сервера або зробити сервер зовсім недоступним для адміністрації;

- на рівні мережі. Для початку блокуються все те, що може дати більше інформації. Блокується пінг і трейс. Сервер забирається під nat. Маскується його ір як тільки це можливо. Це вже дуже професійний спосіб захисту сервера шляхом ховання його ір адреси. Застосовується в багатьох платних системах захисту від ddos;

- на рівні провайдера. Через аналіз пакетів або через блокування ір адрес;

- на рівні комп'ютерного «заліза». Застосовуючи hardware рішення від провідних фірм виробників типу:

- a) Cisco;
- б) 3com;
- с) portel.

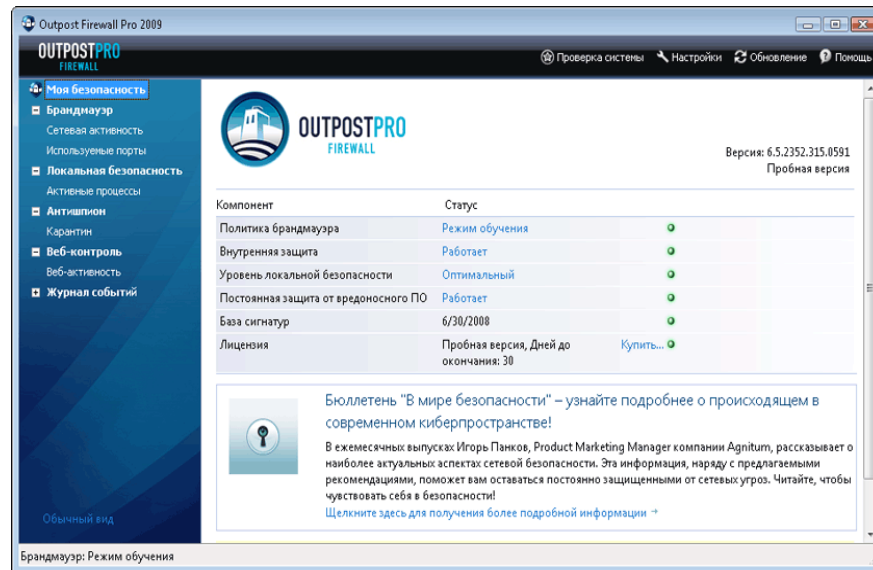
Дані рішення боротьби на апаратному рівні вимагають великих фінансових витрат.

Одним з найрозповсюджених методів захисту комп'ютера – це використання спеціальних програм Outpost Firewall PRO і Outpost Security Suite Pro, які виявляють і запобігають можливим атакам на комп'ютері.

Outpost Firewall - (персональний файрвол) програма для захисту комп'ютера від хакерських атак з Internet від російської компанії Agnitum. Крім цього, Outpost забезпечує блокування завантаження реклами і активного вмісту WEB-сторінок, тим самим - їх більш швидкого завантаження.

Можливості програми:

- фільтрація вхідних і вихідних мережевих з'єднань;
- глобальні правила для протоколів і портів;
- створення правил мережевого доступу та налаштування параметрів активного захисту програм в режимі автонавчання;
- контроль компонентів, контроль прихованих процесів і контроль пам'яті процесів дозволяють встановлювати обмеження на мережеву активність для окремих програм і процесів, визначаючи, які саме - вхідні або вихідні - з'єднання дозволені для конкретних додатків;
- візуальне оповіщення про події (наприклад, про блокування з'єднання, спробі мережевої атаки) за допомогою спливаючих вікон;
- захист від spyware, у тому числі перевірка на наявність шпигунських програм в файлах;
- наочне відображення мережевої активності;
- журнал дій програми;
- технологія SmartDecision та здійснює аналіз завантаження файлів.



Робоче вікно 3.5 - Outpost Firewall



Рисунок 3.6 - Вікно перевірки компонентів Outpost Firewall

3.3 Безпосередня програмна реалізація

Темою проекту було створення сайту «Центру первинної медико-санітарної допомоги м. Суми». Створений Web-сайт виконує функцію інформаційного порталу. Це повноцінний сайт, на якому розміщений великий перелік інформації:

- Головна сторінка;
- Про поліклініку;
- Графік прийому;
- Наші координати;

- Телефон для довідок.

Створений Web-сайт має меню, тобто навігації по сайту, і знаходиться вгорі в шаблоні.

Для всіх сторінок системи керування контентом спільним є дизайн, а також такі елементи, як логотип та «підвал».

Клієнтська та адміністративна частина вимагають також підключення файлу config.php, в якому знаходяться дані, необхідні для підключення бази даних.

```
<?PHP
//System Configurations
$config = array (
  'version_id' => "10.4",
  'home_title' => "DataLife Engine",
  'http_home_url' => "http://localhost/dlesha/",
  'charset' => "windows-1251",
  'admin_mail' => "kyrsu_smk@yandex.ru",
  'description' => "Демонстрационная страница движка
DataLife Engine",

  'keywords' => "DataLife, Engine, CMS, PHP движок",
  'date_adjust' => "Europe/Kiev",
  'site_offline' => "0",
  'allow_alt_url' => "1",
  'langs' => "Russian",
  'skin' => "Default",
  'allow_gzip' => "0",
  'allow_admin_wysiwyg' => "0",
  'allow_static_wysiwyg' => "0",
  'news_number' => "10",
  'timestamp_active' => "j-m-Y, H:i",
  'news_sort' => "date",
  'news_msort' => "DESC",
  'hide_full_link' => "0",
  'allow_site_wysiwyg' => "0",
  'allow_comments' => "1",
  'comm_nummers' => "30",
  'comm_msort' => "ASC",
  'flood_time' => "30",
  'auto_wrap' => "80",
  'timestamp_comment' => "j F Y H:i",
  'allow_comments_wysiwyg' => "0",
  'allow_registration' => "1",
  'auth_domain' => "1",
  'start_site' => "1",
  'clear_cache' => "0",
```

```
'use_admin_mail' => "0",
'allow_complaint_mail' => "0",
'spam_api_key' => "",
'create_metatags' => '1',
'admin_allowed_ip' => '',
'related_only_cats' => '0',
'allow_links' => '1',
'comments_lazyload' => '0',
'category_separator' => '/',
'speedbar_separator' => '&raquo;',
'adminlog_maxdays' => '30',
'allow_social' => '0',
'medium_image' => '450',
'login_ban_timeout' => '20',
'watermark_seite' => '4',
'auth_only_social' => '0',
'rating_type' => '0',
'allow_comments_rating' => '1',
'comments_rating_type' => '1',
'key' => '',
);
?>
```

Меню сайту - один з основних функціональних елементів, що складається з набору посилань на різні сторінки. Кожне посилання називається пунктом меню. Відвідувач використовує меню як засіб навігації.

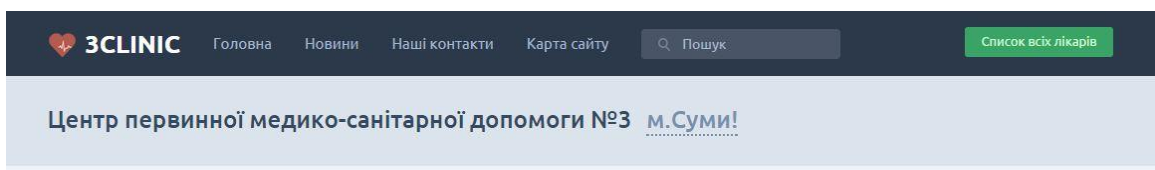


Рисунок 3.7 – Головне меню

```
<div class="header">
    <div class="center_side" id="head_center">
<a class="button fl_r master add_startup_button" href="/vs-
lkar.html">Список всіх лікарів</a>
    <a href="/" class="head_link logo "><i></i>3CLINIC</a>
```

```

<a href="/" class="head_link mobile_hide ">Головна</a>
<a href="/news/" class="head_link mobile_hide ">Новини</a>
  <a href="/nash-koordinati-ta-kontakti.html" class="head_link
mobile_hide ">Наші контакти</a>
<a href="/sitemap" class="head_link mobile_hide ">#</a>
<a href="/" class="head_link mobile_hide ">#</a>
<a href="/news/" class="head_link mobile_hide ">#</a>
  <a href="/nash-koordinati-ta-kontakti.html" class="head_link
mobile_hide ">#</a>
<a href="/sitemap" class="head_link mobile_hide ">#</a>
  <div class="head_search search_holder" data-type="global_search">
<input type="text" name="search" placeholder="Пошук" class="search"
id="head_select" autocomplete="off"> </div>
</div></div>

```

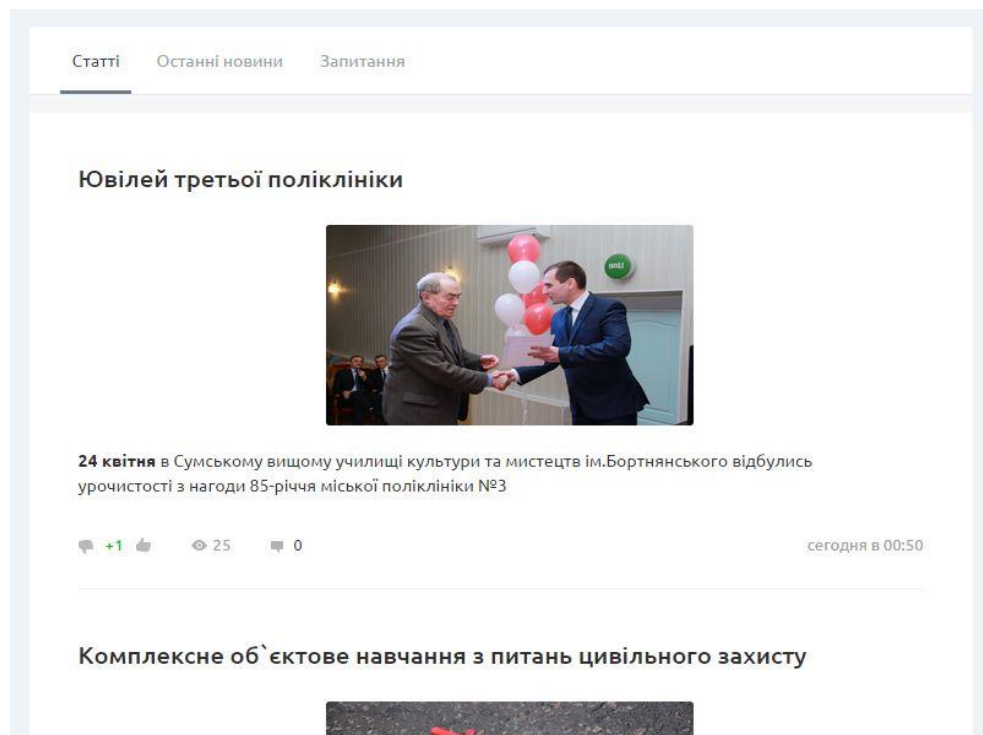


Рисунок 3.8 – Останні новини

Наступна інформативним блоком на сайті є «Невідкладна медична допомога». При переході по вкладці, видно перелік всіх актуальних послуг.

Невідкладна медична допомога	
Захворювання органів кровообігу	1
Захворювання органів дихання	2
Захв. нирок і сечовивідних шляхів	3
ПМД при гострих захв. живота	4
Коматозні та шоківі стани	5
Кровотечі	6
Травми	7
Відкриті ушкодження	8
Закриті ушкодження	9
ПМД при травмі грудної клітки	10
ПМД при травммі живота	11
Термінальні стани	12
Невідкл. стани при інф. хворобах	13
Невідкл. стани при очних хворобах	14
Стани в неврології та психіатрії	15
Перелік пункту невідкл. допомоги	16
Повний перелік	

Рисунок 3.9 – Вкладка «Невідкладна медична допомога»

Для доступу до адміністративної частини необхідно заповнити форму входу, тобто ввести логін та пароль адміністратора.

```
<?php include "templates/include/header.php" ?>
    <form action="admin.php?action=login" method="post"
style="width: 50%;">
        <input type="hidden" name="login" value="true" />
        <?php if ( isset( $results['errorMessage'] ) ) { ?>
            <div class="errorMessage"><?php echo
$results['errorMessage'] ?></div>
        <?php } ?>
        <ul>
            <li>
```

```

        <label for="username">Username</label>
        <input type="text" name="username" id="username"
placeholder="Your admin username" required autofocus maxlength="20" />
    </li>
    <li>
        <label for="password">Password</label>
        <input type="password" name="password" id="password"
placeholder="Your admin password" required maxlength="20" />
    </li>
</ul>
<div class="buttons">
    <input type="submit" name="login" value="Login" />
</div>
</form>
<?php include "templates/include/footer.php" ?>

```

Після авторизації адміністратора відкривається архів усіх опублікованих статей.

```

function listArticles() {
    $results = array();
    $data = Article::getList();
    $results['articles'] = $data['results'];
    $results['totalRows'] = $data['totalRows'];
    $results['pageTitle'] = "All Articles";
    if ( isset( $_GET['error'] ) ) {
        if ( $_GET['error'] == "articleNotFound" )
    $results['errorMessage'] = "Error: Article not found.";
    }
    if ( isset( $_GET['status'] ) ) {
        if ( $_GET['status'] == "changesSaved" )
    $results['statusMessage'] = "Your changes have been saved.";
        if ( $_GET['status'] == "articleDeleted" )
    $results['statusMessage'] = "Article deleted.";
    }
    require( TEMPLATE_PATH . "/admin/listArticles.php" );
}

```

Код програмної реалізації наведено в додатку.

ВИСНОВКИ

У даній роботі проведено детальний аналіз існуючих рішень створення сайтів, проаналізовані системи управління контентом, зроблено огляд мов програмування для виконання завдання.

На основі системи управління контентом CMS DLE був розроблений веб-сайт «Центру первинної медико-санітарної допомоги» з широким функціоналом, в тому числі модулем- форумом, RSS-стрічкою новин.

Реалізація здійснювалася з використанням сучасних інтернет-технологій HTML, CSS, PHP.

СПИСОК ЛІТЕРАТУРИ

1. Godbolt M. Frontend Architecture for Design Systems: A Modern Blueprint for Scalable and Sustainable Websites. – O'Reilly, 2016. – 198 p.
2. Terrell Bob. Creating Data-Driven Web Sites: An Introduction to HTML, CSS, PHP, and MySQL. – Momentum Press, 2019. – 140 p.
3. Adkins Heather et al. Building Secure and Reliable Systems: Best Practices for Designing, Implementing, and Maintaining Systems. – O'Reilly Media, 2020. – 558 p.
4. Abbott Martin L., Fisher Michael T. Scalability Rules: 50 Principles for Scaling Web Sites : 2nd Edition. – Addison-Wesley Professional, 2016. – 256 p.
5. Lassoff M. PHP and MySQL for Beginners. – LearnToProgram, 2014. – 518 p.
6. Brooks David. Programming in HTML and PHP . – Springer, 2017. — 293 p.
7. Pattinson Robert. Learn Basics of PHP And MySql For Web Development . – Amazon Digital Services LLC, 2018. — 558 p.
8. Ali Junade. Mastering PHP Design Patterns. – Packt Publishing, 2016. — 270 p.
9. Bierer D. PHP 7 Programming Cookbook. – Packt Publishing, 2016. — 610 p.
10. Brooks David. Programming in HTML and PHP. – Springer, 2017. — 293 p.
11. Pattinson Robert. Learn Basics of PHP And MySql For Web Development. – Amazon Digital Services LLC, 2018. — 558 p.

ДОДАТОК

```
<!DOCTYPE html>

<html class="html" lang="ru-RU">

<head>

    <script type="text/javascript">

        if(typeof Muse == "undefined") window.Muse = {};
window.Muse.assets = {"required":["jquery-1.8.3.min.js",
"museutils.js", "jquery.watch.js", "jquery.musepolyfill.bgsize.js",
"jquery.scrolleffects.js", "webpro.js", "index.css"],
"outOfDate":[]};

    </script>

    <meta http-equiv="Content-type" content="text/html; charset=UTF-
8"/>

    <meta name="generator" content="2014.2.1.284"/>

    <title>Домашняя</title>

    <!-- CSS -->

    <link rel="stylesheet" type="text/css"
href="css/site_global.css?365011649"/>

    <link rel="stylesheet" type="text/css"
href="css/index.css?3841338253" id="pagesheet"/>

    <!-- Other scripts -->

    <script type="text/javascript">

        document.documentElement.className += ' js';

    </script>

</head>

<body>
```

```

<div class="clearfix" id="page"><!-- column -->
<div class="position_content" id="page_position_content">
<div class="clearfix colelem" id="pu188"><!-- group -->
<div class="browser_width grpelem" id="u188-bw">
<div id="u188"><!-- simple frame --></div>
</div>
<!-- rasterized frame -->
<!-- rasterized frame -->
<!-- rasterized frame -->
<!-- rasterized frame -->
<!-- rasterized frame -->
</div>
<div class="clearfix colelem" id="pu137"><!-- group -->
<div class="browser_width grpelem" id="u137-bw">
<div id="u137"><!-- group -->
<div class="clearfix" id="u137_align_to_page">
<div class="clip_frame grpelem" id="u144"><!-- image -->


```

```

</div>

<div class="clearfix grpelem" id="pu153"><!-- column -->

<div class="clip_frame colelem" id="u153"><!-- image -->

</div>

<div class="clip_frame colelem" id="u162"><!-- image -->

</div>

</div>

<div class="clearfix grpelem" id="pu176-4"><!-- column -->

  <!-- rasterized
frame -->

  <!-- rasterized
frame -->

</div>

<div class="grpelem" id="u189"><!-- simple frame --></div>

<div class="clearfix grpelem" id="u172"><!-- group -->

  <!-- rasterized frame -->

</div>

</div>

</div>

```

```

</div>

<!-- rasterized
frame -->

<div class="clearfix grpelem" id="u151-4"><!-- content -->
<p>СЕТЬ ФИТНЕС&#45;ЦЕНТРОВ В УКРАИНЕ</p>
</div>

<div class="grpelem" id="u175" data-mu-ie-
matrix="progid:DXImageTransform.Microsoft.Matrix(M11=0.6947,M12=0.71
93,M21=-0.7193,M22=0.6947,SizingMethod='auto expand')" data-mu-ie-
matrix-dx="-3" data-mu-ie-matrix-dy="-3"><!-- simple frame --></div>
</div>

<div class="clearfix colelem" id="pu152"><!-- column -->
<div class="browser_width colelem" id="u152-bw">
<div id="u152"><!-- simple frame --></div>
</div>

<div class="clearfix colelem" id="pu88"><!-- group -->
<div class="browser_width grpelem" id="u88-bw">
<div class="museBGSize" id="u88"><!-- group -->
<div class="clearfix" id="u88_align_to_page">
<div class="clip_frame grpelem" id="u197"><!-- image -->

</div>
</div>
</div>
</div>
</div>

<div class="browser_width grpelem" id="u102-bw">
<div class="museBGSize" id="u102"><!-- simple frame --></div>

```

```
</div>
```

```
<div class="grpelem" id="u117"><!-- simple frame --></div>
```

```
<!-- rasterized
frame -->
```

```
<div class="grpelem" id="u126"><!-- simple frame --></div>
```

```
<!-- rasterized
frame -->
```

```
<!-- rasterized
frame -->
```

```
<div class="grpelem" id="u168"><!-- simple frame --></div>
```

```
</script>
```

```
</body>
```

```
</html>
```