

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ОДЕСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ імені І.І.МЕЧНИКОВА

(повне найменування вищого навчального закладу)

Факультет математики, фізики та інформаційних технологій

(повне найменування інституту, назва факультету (відділення))

Кафедра комп'ютерних систем та технологій

(повна назва кафедри (предметної, циклової комісії))

Дипломна робота

на здобуття ступеня вищої освіти «бакалавр»

(освітньо-кваліфікаційний рівень)

На тему: **Розробка WEB-сайту інтернет магазину комп'ютерної техніки**
(назва українською)

Development of the WEB-site of the online store of computer squipment

(назва англійською)

Виконав: студент заочної форми навчання

спеціальності 123 Комп'ютерна інженерія.
(шифр і назва напрямку підготовки, спеціальності)

Балан Василь Геннадійович

(прізвище, ім'я, по-батькові)

Керівник: Якимчук В.І., канд. фіз-мат.наук

(науковий ступінь, вчене звання, прізвище та ініціали, підпис)

Рецензент: Гоцунський В. Я доктор физ. мат наук профессор
(науковий ступінь, вчене звання, прізвище та ініціали)

Рекомендовано до захисту:
Протокол засідання кафедри
№ ____ від « ____ » ____ 2021 р.

Завідувач кафедри

Ю.О. Гунченко
(підпис) (прізвище, ініціали)

Захищено на засіданні ЕК № ____
протокол № ____ від « ____ » ____ 2021р.
Оцінка ____ / ____ / ____

(за національною шкалою, шкалою ECTS, бали)

Голова ЕК

Н.Ф. Казакова
(підпис) (прізвище, ініціали)

Одеса - 2021

АНОТАЦІЯ

В дипломній роботі «Розробка WEB-сайту інтернет магазину комп'ютерної техніки» розглянуті сучасні принципи розробки Інтернет-магазинів на прикладі магазину комп'ютерної техніки.

В роботі проаналізовано необхідність організації Інтернет-магазинів, їх переваги і недоліки, типові структури, принципи вибору програмного забезпечення.

Розглянуті принципи вибору системи управління змістом, мови програмування HTML, DHTML, XML, сервера Apache і CMS Joomla,

Детально розглянуті дизайн веб-сайта, опис створення сторінок веб-сайта і розробка сайту на основі CMS Joomla, створення меню on-line магазину і розміщення товарів в on-line магазині.

Результати роботи мають загальний і можуть бути використаними при розробці Інтернет магазинів усіх категорій.

АННОТАЦИЯ

В дипломной работе «Разработка WEB-сайта интернет магазина компьютерной техники» рассмотрены современные принципы разработки Интернет-магазинов на примере магазина компьютерной техники.

В работе проанализированы необходимость организации Интернет-магазинов, их преимущества и недостатки, типовые структуры, принципы выбора программного обеспечения.

Рассмотрены принципы выбора системы управления содержанием, языка программирования HTML, DHTML, XML, сервера Apache и CMS Joomla,

Подробно рассмотрены дизайн веб-сайта, описание создания страниц веб-сайта и разработка сайта на основе CMS Joomla, создание меню on-line магазина и размещения товаров в on-line магазине.

Результаты работы имеют общий и могут быть использованы при разработке Интернет магазинов всех категорий.

ANNOTATION

In the thesis "Development of a WEB-site for an online store of computer equipment", the modern principles of development of online stores are considered on the example of a computer equipment store.

The paper analyzes the need to organize online stores, their advantages and disadvantages, typical structures, principles of software selection.

Considered the principles of choosing a content management system, programming language HTML, DHTML, XML, Apache server and CMS Joomla,

The design of a website, a description of the creation of website pages and the development of a website based on CMS Joomla, the creation of a menu for an on-line store and placing goods in an on-line store are considered in detail.

The results of the work are general and can be used in the development of online stores of all categories.

ЗМІСТ

Вступ.....	7
1 Теоретична частина	9
1.1 Обґрунтування потреби в веб-сайті.....	9
1.2 Поняття веб-сайта	9
1.2.1 Класифікація веб-сайтів.....	10
1.3 Проектування веб-сайта.....	11
1.4 Етапи розробки веб-сайта.....	11
1.4.1 Постановка завдання при проектуванні веб- сайту	11
1.4.2 Вибір програмних засобів для розробки веб-сайта	12
1.4.3 Розробка інтерфейсу	12
1.4.4 Розробка бази даних.....	13
1.4.5 Розробка адміністративної частини сайту	13
1.5 Обґрунтування вибору системи управління вмістом	15
1.6 Засоби розробки.....	15
1.6.1 Мова розмітки гіпертекстових документів HTML	15
1.6.2 Динамічна мова розмітки гіпертекстових документів DHTML	17
1.6.3 Розширювана мова розмітки XML	18
1.6.4 Мова програмування PHP	22
1.6.5 Веб-сервер Apache.....	22
1.6.6 Система управління вмістом CMS Joomla.....	23
2. Проектна частина	25
2.1 Загальна структура сайта.....	25
2.2 Розробка інтерфейса веб-сайта	26
2.2.1 Дизайн веб-сайта	26
2.2.2 Опис створення сторінок веб-сайта.....	27
2.3 Розробка сайту на основі CMS.....	29
2.3.1 Опис встановлення на сервер CMS Joomla.....	29
2.3.2 Створення розділів сайта.....	30

2.3.3 Створення категорій сайта	33
2.3.4 Створення об'єкту вмісту	35
2.3.5 Створення меню on-line магазину	39
2.3.6 Установка шаблону сайта	42
2.3.7 Створення категорії товарів магазину.....	44
2.3.8 Розміщення товарів в on-line магазині	47
2.4 Висновки.....	48
Висновок.....	50
Використані джерела	51
2 Додатки.....	52
Додаток А.....	52
Додаток Б.....	66

ВСТУП

З появою веб-технологій комп'ютер починають використовувати абсолютно нові верстви населення Землі. Можна виділити дві найбільш характерні групи, що знаходяться на різних соціальних полюсах, які були стрімко залучені в нову технологію, можливо, навіть крім їх власного бажання. З одного боку, це були представники елітарних груп суспільства – керівники крупних організацій, президенти банків, топ-менеджери, впливові державні чиновники і так далі. З іншого боку, це були представники найширших верств населення – домогосподарки, пенсіонери, діти.

При появі технології веб, комп'ютери обернулися лицем до цих двох абсолютно протилежним категорій потенційних користувачів. Еліту об'єднувала одна межа – через високу відповідальність і практично стовідсотковій зайнятості “великі люди” ніколи не користувалися комп'ютером; типовою була ситуація, коли з комп'ютером працював секретар. У якийсь момент часу вони зрозуміли, що комп'ютер їм може бути корисний, що вони можуть результативно використовувати той невеликий час, який можна виділити на роботу за комп'ютером. Вони раптом зрозуміли, що комп'ютер – це не просто модна і дорога іграшка, але інструмент отримання актуальної інформації для бізнесу. При цьому їм не потрібно було витрачати багато часу, щоб освоїти технологію роботи з комп'ютером (в порівнянні з тим, як це було раніше).

Спектр соціальних груп, що підключаються до мережі Інтернет і що шукають інформацію в WWW, весь час розширюється за рахунок користувачів, що не відносяться до категорії фахівців в області інформаційних технологій. Це лікарі, будівельники, історики, юристи, фінансисти, спортсмени, мандрівники, священослужителі, артисти, письменники, художники. Список можна продовжувати бескінченно. Будь-

який, хто відчув корисність і незамінність мережі для своєї професійної діяльності або захоплень, приєднується до величезної армії споживачів інформації в «Всесвітній Павутині».

Веб-технологія повністю перевернула наші уявлення про роботу з інформацією, та і з комп'ютером взагалі. Виявилось, що традиційні параметри розвитку обчислювальної техніки – продуктивність, пропускну спроможність, ємкість пристроїв, що запам'ятовують, не враховували головного «вузького місця» системи – інтерфейсу з людиною. Застарілий механізм взаємодії людини з інформаційною системою стримував впровадження нових технологій і зменшував вигоду від їх застосування. І лише коли інтерфейс між людиною і комп'ютером був спрощений до природності сприйняття звичайною людиною, послідував безпрецедентний вибух інтересу до можливостей обчислювальної техніки.

З розвитком технологій гіпертекстової розмітки в Інтернеті почали з'являтися все більше сайтів, тематика яких була абсолютно різною – від сайтів крупних компаній, що оповідають про успіхи компанії і її провали, до сайтів маленьких фірм, що пропонують відвідати їх офіси в межах одного міста.

Розвиток Інтернет-технологій послужив поштовхом до появи нової вітки в Інтернеті – Інтернет-форумів. Почали з'являтися сайти, і навіть цілі портали, на яких люди зі всіх куточків планети можуть спілкуватися, отримувати відповіді на будь-які питання і, навіть, укласти ділові операції.

Також в наш час стає поширеним купівля будь-яких товарів в інтернет-магазинах. Шопінг - заняття корисне, іноді захоплююче, що приносить як позитивні емоції, так і розчарування. Уходять у минуле ті часи, коли для того, щоб стати власником бажаного предмету, потрібно було на своїх «двух» здійснювати тривалі подорожі по офлайновим торговим місцям.

Тепер в наше життя активно входять інтернет-магазини, ми все частіше робимо покупки, залишаючись на робочому місці, шляхом кількох натисків миші.

1 ТЕОРЕТИЧНА ЧАСТИНА

1.1 Обґрунтування потреби в веб-сайті

На сьогоднішній день практично кожна організація має власний веб-сайт. В умовах використання сучасних інформаційних технологій – це необхідний чинник існування, що дозволяє розширити поле рекламної діяльності і привернути тим самим додаткових клієнтів.

Створення і розробка сайтів включає:

1. Затвердження первинного технічного завдання на розробку сайту.
2. Визначення структурної схеми сайту - розташування розділів, контенту і навігації.
3. Веб-дизайн - створення графічних елементів макету сайту, стилів і елементів навігації.
4. Розробка програмного коду, модулів, бази даних і інших елементів сайту необхідних в проекті.
5. Тестування і розміщення сайту в мережі Інтернет.

1.2 Поняття веб-сайта

Інформація, доступна користувачам Internet, розташовується на комп'ютерах (веб-серверах), на яких встановлено спеціальне програмне забезпечення. Значна частина цієї інформації організована у вигляді веб-сайтів. Кожен з них має своє ім'я (адреса) в Internet.

Веб-сайт – це інформація, представлена в певному вигляді, яка розташовується на веб-сервері і має своє ім'я (адреса). Для перегляду веб-сайтів на комп'ютері користувача використовуються спеціальні програми, які називаються браузером. Залежно від того, яке ім'я (адреса) сайту ми задамо в рядку "Адреса", браузер завантажуватиме в своє вікно відповідну

інформацію.

Веб-сайт складається із зв'язаних між собою веб-сторінок. Веб-сторінка є текстовим файлом з розширенням *.htm, який містить текстову інформацію і спеціальні команди – HTML-коди, що визначають в якому вигляді ця інформація відображатиметься у вікні браузера. Вся графічна, аудіо- і відео-інформація безпосередньо в Веб-сторінку не входить і є окремими файлами з розширеннями *.gif, *.jpg (графіка), *.mid, *.mp3 (звук), *.avi (відео). У HTML-кодів сторінки містяться тільки вказівки на такі файли.

Кожна сторінка веб-сайта також має свій Internet адрес, який складається з адреси сайту і імені файлу, відповідного даній сторінці. Таким чином, веб-сайт – це інформаційний ресурс, що складається із зв'язаних між собою гіпертекстових документів (веб-сторінок), розміщений на веб-сервері і такий, що має індивідуальну адресу. Подивитися веб-сайт може будь-яка людина, що має комп'ютер, підключений до Internet.

1.2.1 Класифікація веб-сайтів

В даний час у всесвітній павутині розміщено декілька мільйонів веб-сайтів і їх число постійно росте. Це особисті сторінки, що містять інформацію про автора, його інтереси. Їх створюють для того, щоб знайти друзів по інтересах, розширити свій кругозір, свій світ.

Інформаційні сайти. До них відносяться сайти учбових закладів, співтовариств по інтересах, фірм і інше.

Сайти-портали. До них відносяться крупні веб-сайти, які пропонують велику кількість сервісів, починаючи від каталога зображень до форумів і поштових скриньок.

Інтернет-магазини. Такі сайти в наш час набирають дуже велику популярність з боку, як користувачів, так і з боку підприємців. Це веб-сайти, в яких кожна людина може купити будь-який товар не відходячи від свого місця біля комп'ютера.

Соціальні сайти. Це веб-сайти на яких кожна людина має змогу

спілкуватися зі старими друзями та знаходити нових.

1.3 Проектування веб-сайта

Проектування і розробка сайтів включає:

1. Затвердження первинного технічного завдання розробки сайту.
2. Визначення структурної схеми сайту - розташування розділів, контенту і навігації.
3. Веб-дизайн - створення графічних елементів макету сайту, стилів і елементів навігації.
4. Розробка програмного коду, модулів, бази даних і інших елементів сайту необхідних в проекті.
5. Тестування і розміщення сайту в мережі Інтернет.

1.4 Етапи розробки веб-сайта

1.4.1 Постановка завдання при проектуванні веб- сайту

Перед автором було поставлено перше завдання: розробити структурну схему проекту веб – сайту для використання в середовищі Internet. На думку керівника диплому, веб-сайт, що розробляється, повинен володіти наступними особливостями:

- гнучкістю, зручною для адміністраторів системою управління структурою;
- веб-сайт повинен підтримувати використання графічних вставок, анімації, які повинні підсилювати емоційно-ціннісний компонент змісту, формувати мотивацію;
- для користувачів повинна бути також реалізована можливість роздрукувати будь-яку сторінку веб-сайта.

Проте головним завданням проектування було створення системи управління вмістом, яка б дозволяла вносити зміни веб – сайту для людей, які

не мають навичок в розробці веб-сайтів.

1.4.2 Вибір програмних засобів для розробки веб-сайта

Враховуючи поставлені вище завдання, автор вирішив, що найбільш оптимальним засобом для розробки такого роду електронної комерції – HTML+система управління вмістом Joomla.

Joomla — написана на мові PHP і така, що використовує як сховище змісту базу даних MySQL. Joomla є, захищена ліцензією GPL. Однієї з головних особливостей Joomla є відносна простота управління при практично безмежних можливостях і гнучкості при виготовленні сайтів. [2]

Назва «Joomla» фонетично ідентично слову «Jumla», яке в перекладі з суахілі означає «всі разом» або «в цілому», що відображає підхід розробників і співтовариства до розвитку системи.

Система управління вмістом Joomla є відгалуженням широко відомої CMS Mambo. Команда незалежних розробників відокремила від проекту Mambo внаслідок незгоди в економічній політиці. І 16 вересня 2005 року в світ вийшла перша версія Joomla, що є по суті перейменованою Mambo 4.5.2.3 і що включає виправлення знайдених на той момент помилок і уязвимостей.

CMS Joomla включає різні інструменти для виготовлення веб-сайту. Важливою особливістю системи є мінімальний набір інструментів при початковій установці, який збагачується в міру необхідності. Це знижує те, що захащується адміністративна панель непотрібними елементами, а також знижує навантаження на сервер і економить місце на хостингу.

1.4.3 Розробка інтерфейсу

Титульна сторінка (головна) будь-якого сайту повинна максимально інформативно і в стислому об'ємі відображати необхідну користувачеві інформацію про сайт. На головній сторінці необхідно помістити логотип веб-сайту, основне меню сайту (для навігації по його структурі), форму

аутентифікації (входу зареєстрованих користувачів), реєстраційне посилання (реєстрація нових клієнтів).

1.4.4 Розробка бази даних

Оскільки сайт написаний за допомогою системи управління вмістом Joomla, база даних якої представлена у вигляді MYSQL, база даних сайта підтримуватиме ту ж структуру (Рис.1), вносячи до неї свої зміни.

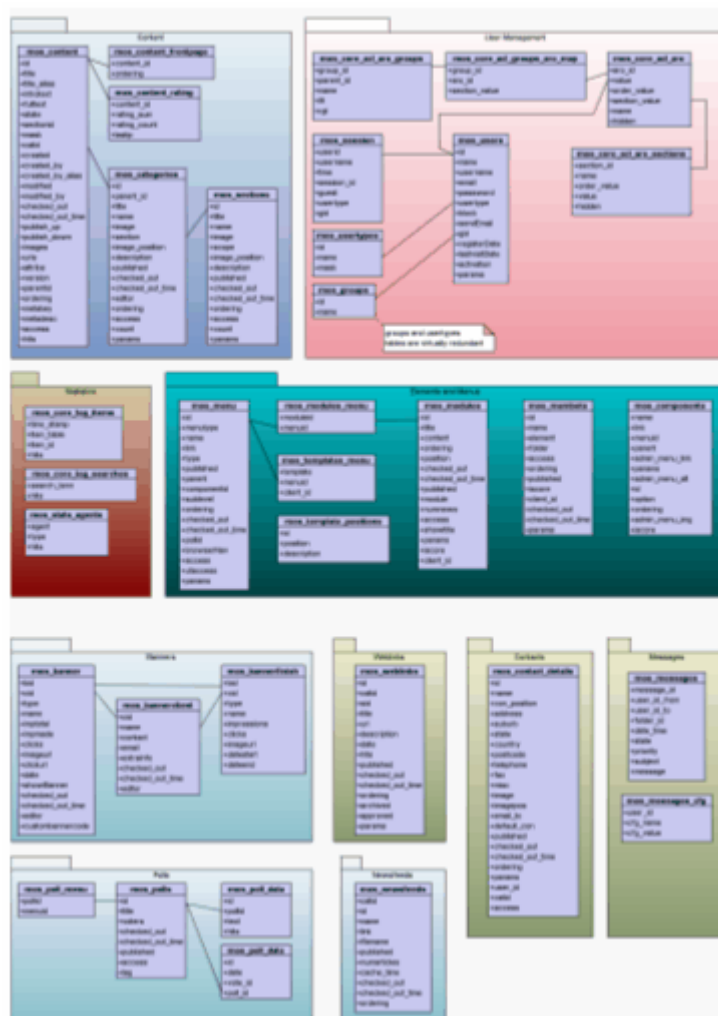


Рисунок 1– Схема бази даних сайта

1.4.5 Розробка адміністративної частини сайту

Адміністративна частина повинна включати: інформацію про нових користувачів, розділ налаштувань сайту, можливість адміністрування магазину,

можливість виведення статистики по магазину, управління обліковими записами користувачів і інші можливості.

Адміністративна частина сайту представлена у вигляді системи управління вмістом.

Система управління вмістом (англ. Content management system, CMS) — комп'ютерна програма, використовувана для управління вмістом чогонебудь (звичайний цей вміст розглядається як неструктуровані дані наочного завдання в протилежність структурованим даним, СУБД, що зазвичай знаходяться під управлінням). Зазвичай такі системи використовуються для зберігання і публікації великої кількості документів, зображень, музики або відео.

Окремим випадком такого роду систем є системи управління сайтами. Подібні CMS дозволяє управляти текстовим і графічним наповненням веб-сайту, надаючи користувачеві зручні інструменти зберігання і публікації інформації.

Зараз існує безліч готових систем управління вмістом сайту, у тому числі і безкоштовних. Їх можна розділити на три типи, за способом роботи:

Генерація сторінок за запитом. Системи такого типу працюють на основі зв'язки «Модуль редагування > База даних > Модуль уявлення». Модуль уявлення генерує сторінку із змістом при запиті на нього, на основі інформації з бази даних. Інформація в базі даних змінюється за допомогою модуля редагування. Сторінки наново створюються сервером при кожному запиті, а це створює навантаження на системні ресурси. Навантаження може бути багато разів понижена при використанні засобів кешування, які є в сучасних веб-серверах.

Генерація сторінок при редагуванні. Системи цього типу програми для редагування сторінок, які при внесенні змін до змісту сайту створюють набір статичних сторінок. При такому способі жертвується інтерактивність між відвідувачем і вмістом сайту.

Змішаний тип. Як зрозуміло з назви, поєднує в собі переваги перших

два. Може бути реалізований шляхом кешування — модуль уявлення генерує сторінку один раз, надалі вона в рази швидше підвантажується з кеша. Кеш може оновлюватися як автоматично, по закінченню деякого терміну часу або при внесенні змін до певних розділів сайту, так і вручну по команді адміністратора. Інший підхід — збереження певних інформаційних блоків на етапі редагування сайту і збірка сторінки з цих блоків при запиті відповідної сторінки користувачем.

1.5 Обґрунтування вибору системи управління вмістом

Необхідна розвинена гіпертекстова структура – HTML створений саме для цього; необхідно забезпечити гнучку структуру управління матеріалом – управляти HTML-вмістом зручно за допомогою PHP+MySQL. Саме ця зв'язка дозволяє генерувати динамічні HTML-сторінки; необхідно реалізувати варіативність представлення матеріалу – в цьому випадку використання HTML+CSS більш ніж обґрунтовано. Саме ця комбінація засобів дозволяє створювати мультимедійні документи; необхідно забезпечити підтримку внутрішнього магазину веб-сайта.

На підставі поставленого технічного завдання була вибрана система управління вмістом CMS Joomla, оскільки саме ця система відповідає всім вимогам.

1.6 Засоби розробки

1.6.1 Мова розмітки гіпертекстових документів HTML

Обмін інформацією в Інтернет здійснюється за допомогою протоколів прикладного рівня, що реалізують той або інший прикладний сервіс (пересилку файлів, гіпертекстової інформації, пошта і так далі). Одним з найбільш молодих і популярних сервісів Інтернет, розвиток якого і привело до сплеску популярності самої Інтернет, стала World Wide Web (WWW),

заснована на протоколі HTTP (Hyper Text Transfer Protocol - протокол передачі гіпертекстовій інформації). Гіпертекстові документи, представлені в WWW, мають одну принципову відмінність від традиційних гіпертекстових документів - зв'язки, в них що використовуються, не обмежені одним документом, і більш того, не обмежені одним комп'ютером. Для підготовки гіпертекстових документів використовується мова HTML (Hyper Text Markup Language – мова розмітки гіпертекстових документів), що надає широкі можливості по форматуванню і структурній розмітці документів, організації зв'язків між різними документами, засоби включення графічної і мультимедійної інформації. HTML-документи є видимими за допомогою спеціальної програми - браузера. Найбільшого поширення в даний час набули браузери Mozilla Firefox і Internet Explorer компанії Microsoft (MSIE). Реалізації Mozilla Firefox доступні практично для всіх сучасних програмних і апаратних платформ, реалізації MSIE доступні для всіх Windows платформ, Macintosh і деяких комерційних Unix-систем.

HTML-документ складається з тексту, що є змістом документа, і *тегів*, що визначають його структуру і зовнішній вигляд при відображенні браузером. Простий html-документ виглядає таким чином:

```
<html>
<head>
<title>Название</title>
</head>
<body>
<p>Тело документа
</body>
</html>
```

Як видно з прикладу, тег є ключовим словом, поміщеним в кутові дужки. Розрізняють одинарні теги, як, наприклад, `<p>`, і парні, як `<body>` `</body>`, в останньому випадку дія тега розповсюджується тільки на текст між його відкриваючою і закриваючою дужкою. Теги також можуть мати

параметри - наприклад, при описі сторінки можна задати колір фону, колір шрифту і т.д.: `<Body bgcolor="white" text="black">`.

Текст всього документа полягає в теги `<html>`, сам документ розбивається на дві частини - заголовок і тіло. Заголовок описується тегами `<head>`, в яких можуть бути включені назва документа (за допомогою тегов `<title>`) і інші параметри, що використовуються браузером при відображенні документа. Тіло документа поміщене в теги `<body>` і містить власне інформацію, яку бачить користувач. За відсутності тегів форматування весь текст виводиться у вікно браузера суцільним потоком, переклади рядків, пропуски і табуляції розглядаються як пробільні символи, декілька пробільних символів, що йдуть підряд, замінюються на один. Для форматування використовуються наступні основні теги:

`<p>` - початок нового абзацу, може мати параметр, що визначає вирівнювання:

`<p align=right>;`

`<br>` - переклад рядка в межах поточного абзацу;

`<u></u>` - виділення тексту підкресленням

Посилання на інший документ встановлюється за допомогою тега `<a href="URL">...</a>`, де URL - повна або відносна адреса документа. При цьому текст, ув'язнений в тег `<a>`, зазвичай виділяється підкресленням і кольором, і після натискання мишею по цьому посиланню браузер відкриває документ, адреса якого вказана в параметрі href. Графічні зображення вставляються в документ за допомогою тега ``.

1.6.2 Динамічна мова розмітки гіпертекстових документів DHTML

Динамічний HTML (Dynamic HTML, DHTML) не є якоюсь особливою мовою розмітки сторінок. Це всього лише термін, вживаний для позначень HTML-сторінок з динамічно змінним вмістом.

Реалізація DHTML “стоїть” на трьох “китах”: безпосередньо HTML, каскадні таблиці стилів і мови сценаріїв. Ці три компоненти DHTML зв'язано

між собою об'єктною моделлю документа (DOM, Document Object Model), що є по суті інтерфейсом прикладного програмування (API). DOM зв'язує воедино три перераховані компоненти, додаючи простому документу HTML нову якість – можливість динамічної зміни свого вмісту без перевантаження сторінки.

Об'єктна модель документа робить всі елементи сторінки програмованими об'єктами. З її допомогою через мови сценаріїв можна дістати доступ і управляти всім, що є в документі. Кожен елемент HTML доступний як індивідуальний об'єкт, а це означає, що можна змінювати значення будь-якого параметра будь-якого тега HTML-сторінки, і, як наслідок, документ дійсно стає динамічним. Будь-яка дія користувача (натискання кнопкою миші, переміщення миші у вікні браузера або натиснення клавіші клавіатури) об'єктною моделлю документа трактується як подія, яка може бути перехоплена і оброблена процедурою сценарію.

1.6.3 Розширювана мова розмітки XML

На сьогоднішній день вже всім фахівцям в області веб-технологій стало очевидне, що існуючих стандартів передачі даних по Internet недостатньо. Формат HTML, ставши свого часу проривом в області відображення вмісту вузлів Internet, вже не задовольняє всім необхідним на даний момент вимогам. Він дозволяє описати те, яким чином повинні бути відображені дані на екрані кінцевого користувача, але не надає ніяких засобів для ефективного опису передаваних даних і управління ними.

Крім того, каменем спотикання для багатьох компаній, що займаються розробкою програмного забезпечення, є необхідність сумісного використання різних компонентів, забезпечення їх взаємодії, можливості обміну даними між ними.

До недавнього часу не існувало стандарту, що надає засоби для інтелектуального пошуку інформації, обміну даними, адаптивної обробки отримуваних даних.

Вирішенням всіх описаних вище проблем став затверджений в 1998 році міжнародною організацією W3C мова XML. XML (eXtensible Markup Language) - це розширювана мова розмітки, призначена для опису в текстовій формі структурованих даних. Цей текстовий (text-based) формат, багато в чому схожий з HTML, розроблений спеціально для зберігання і передачі даних.

XML дозволяє описувати і передавати такі структуровані дані, як:

1. Окремі документи.
2. Метадані, що описують вміст якого-небудь вузла Internet.
3. Об'єкти, що містять дані і методи роботи з ними (наприклад, елементи управління ACTIVEX або об'єкти Java).
4. Окремі записи (наприклад, результати виконання запитів до баз даних)
5. Всілякі веб-посилання на інформаційні і людські ресурси Internet (адреси електронної пошти, гіпертекстові посилання і ін.)

Дані, описані на мові XML, називаються XML-документами. Мова XML легко читана і достатньо проста для розуміння. Якщо Ви були знайомі з HTML, то навчитися складати XML-документи не складе для Вас ніяких труднощів.

Початковий текст XML-документа складається з набору XML-елементів, кожен з яких містить початковий і кінцевий теги. Кожна пара тегів представляє частину даних. Тобто, як і HTML, мова XML для опису даних використовує теги. Але, на відміну від HTML, XML дозволяє використовувати необмежений набір пар тегів, кожна з яких представляє не то, як ув'язнені в неї дані повинні виглядати, а то, що вони означають.

Будь-який елемент XML-документа може мати атрибути, характеристики, що уточнюють його. Атрибут - це пара ім'я = "значення", яка задається при визначенні елементу в початковому тегу.

Принцип розширюваності мови XML полягає в можливості використання необмеженої кількості пар тегів, визначуваних творцем XML-

документа.

Принцип незалежності визначення внутрішньої структури документа від способів представлення цієї інформації полягає у відділенні даних від процесу їх обробки і відображення. Таким чином, отримані дані можна використовувати відповідно до потреб клієнта, тобто вибирати потрібне оформлення, застосовувати необхідні методи обробки.

Управляти відображенням елементів у вікні програми-клієнта (наприклад, у вікні браузера) можна за допомогою спеціальних інструкцій - стильових таблиць XSL (eXtensible Stylesheet Language). Ці таблиці XSL дозволяють визначати оформлення елементу залежно від його місцерозташування усередині документа, тобто до двох елементів з однаковою назвою можуть застосовуватися різні правила форматування. Крім того, мовою, лежачою в основі XSL, є XML, а це означає, що таблиці XSL більш універсальні, а для контролю коректності складання таких стильових таблиць можна використовувати DTD-описання або схеми даних, розглянуті нижче.

Формат XML, в порівнянні з HTML, має невеликий набір простих правил розбору, який дозволяє розбирати XML-документи, не вдаючись до яких-небудь зовнішніх описів використовуваних XML-елементів. У загальному випадку XML-документи повинні задовольняти наступним вимогам:

1. Кожен відкриваючий тег, що визначає деяку частину даних в документі, обов'язково повинен супроводжуватися таким, що закриває, тобто, на відміну від HTML, не можна опускати закриваючі теги.
2. Вкладеність тегів в XML строго контролюється, тому необхідно стежити за порядком проходження відкриваючих і закриваючих тегів.
3. У XML враховується регістр символів.
4. Вся інформація, розташована між початковим і кінцевим тегами, розглядається в XML як дані, і тому враховуються всі символи форматування (тобто пропуски, переклади рядків, табуляції не ігноруються, як в HTML).

У XML існує набір зарезервованих символів, які повинні бути задані в XML-документі тільки спеціальним чином. Багато фахівців розглядають XML як нову технологію інтеграції програмних компонентів. Основними перевагами використання XML є:

1. Інтеграція даних з різних джерел. XML можна використовувати для об'єднання різнорідних структурованих даних на середньому рівні трирівневих веб-систем, баз даних.

2. Локальна обробка даних. Отримані дані у форматі XML можна розбирати, обробляти і відображати безпосередньо на клієнтові без додаткових звернень до сервера.

3. Перегляд і маніпулювання даними в різних розрізах. Отримані дані можуть оброблятися і бути видимими клієнтом різними способами залежно від потреб кінцевого користувача.

4. Можливість часткового оновлення даних. За допомогою XML можна оновлювати тільки ту частину структурованих даних, яка була змінена, а не всю структуру цілком.

Всі ці переваги роблять XML незамінним інструментом для розробки гнучких засобів пошуку інформації в базах даних, могутніх трирівневих веб-додатків, а також додатків, що підтримують транзакції. Іншими словами, за допомогою XML можна формувати запити до баз даних різних структур, що дозволяє здійснювати пошук інформації в численних несумісних один з одним базах даних. Використання XML на середньому рівні трирівневих веб-додатків дозволяє здійснювати ефективний обмін даними між клієнтами і серверами систем електронної комерції.

Крім того, мова XML може використовуватися як засіб для опису граматики інших мов і контролю правильності складання документів.

Інструменти обробки даних, отриманих у форматі XML, можуть бути розроблені в середовищі Visual Basic, Java або C++.

1.6.4 Мова програмування PHP

Історія PHP починається осінню 1994 року. Коли Расмус Лердорф (Rasmus Lerdorf) почав працювати над тим, що згодом стало PHP, єдиною метою, яка була у нього в думках, з'ясувати, хто читає його резюме. У той час, будучи незалежним підрядчиком, Лердорф розсилав потенційним працедавцям своє міні-резюме з URL посиланням на його повну версію. Щоб стежити за відвідувачами, він створив CGI скрипт на Perl-і, який вставлявся як спеціальний тег в HTML код його сторінки, і збирав інформацію про відвідувачів. Щоб справити враження на потенційних працедавців, він дозволив будь-якому відвідувачеві сторінки проглядати збирану статистику відвідин. [1]

Він назвав цей код для збору статистики «PHP-Tools for Personal Home Page», оскільки сам використовував його на своїй персональній домашній сторінці (personal home page). Декілька чоловік поцікавилися тим, як вони могли б отримати цей інструмент, і Лердорф ухвалив рішення надати його іншим особам. «Це диво програмного забезпечення. Ви можете дати це і проте залишити це собі», - дотепно відмітив Лердорф. У той час руху Open Source не існувало. Тоді воно назвалося freeware. Близьче до кінця 1995 року Лердорф відкрив для людей перший список розсилки по PHP, щоб можна було обмінюватися ідеями, виправленнями помилок і кодом.

1.6.5 Веб-сервер Apache

Найпоширеніший веб-сервер в світі - це Apache. За даними компанії Netcraft, загальне число веб-узлів, що працюють під його управлінням, до кінця 1998 р. досягло 2 млн. (55% загального числа вузлів) і постійно росте. Для порівняння: на долю серверів Microsoft доводиться 25%, Netscape - 7%. Будучи безкоштовною відкритою програмою, призначеною для безкоштовних же Unix-систем (FREEBSD, Linux і ін.), Apache по функціональних можливостях і надійності не поступається комерційним серверам, а широкі можливості конфігурації дозволяють побудувати його для

роботи практично з будь-якою конкретною системою. Існують локалізації сервера для різних мов, у тому числі і для російської.

Історично склалося так, що російські тексти в Internet можуть бути представлені в різних кодуваннях, з яких найбільш поширені koі8-r (або просто koі8) і Windows-1251: з першою працюють більшість серверів і робочих станцій під управлінням Unix, друга є стандартною для всіх версій Windows. Оскільки кодування Windows-1251, природно, застосовується на переважній більшості клієнтських машин, частка тих, хто подорожує по російській частині WWW, використовуючи koі8, не перевищує зараз 5%. Проте в цьому кодуванні зберігаються документи на багатьох Unix-серверах, в ній найчастіше передаються поштові повідомлення і практично завжди - листи в телеконференції, з нею ж працюють багато російськомовних каналів IRC (до речі, аббревіатура KOI розшифровується як "код обміну інформацією"). Щоб вирішити проблеми, що виникають при неспівпаданні кодувань тексту на сервері і клієнтській машині, і був створений російський модуль APACHE-RUS для веб-сервера Apache.

1.6.6 Система управління вмістом CMS Joomla

Joomla – це Система управління вмістом (CMS), що створена тією ж самою командою, яка отримала безліч нагород і принесла сьогодишню популярність системі Mambo CMS. [4]

Назва Joomla є фонетичною транскрипцією слова "Jumla" з мови суахілі, що означає "всі разом" або "єдине ціле". Воно було вибране, оскільки всі команди мамберів стали одностайні в бажанні захистити інтереси творців і співтовариства, яке і є дійсною причиною успіху проекту.

Ця назва була вибрана серед тисяч запропонованих співтовариством, і на додаток пройшло серйозну перевірку професіоналами брендинга і маркетингу, які також вирішили, що Joomla - це кращий вибір.

Головна відмінність Joomla від інших систем – це прагнення розробників зберегти управління настільки простим, наскільки це можливо, в

той же час забезпечуючи максимальну функціональність. У результаті, не маючи спеціальної освіти, можна повністю контролювати свої сайти, замість того щоб платити надмірно великі суми грошей за закрите, таке, що є чистією власністю, програмне забезпечення.

Назва «Joomla» фонетично ідентично слову «Jumla», яке в перекладі з суахілі означає «всі разом» або «в цілому», що відображає підхід розробників і співтовариства до розвитку системи.

Система управління содержанием Joomla є відгалуженням широко відомої CMS Mambo. Команда незалежних розробників відокремилася від проекту Mambo унаслідок незгоди в економічній політиці. І 16 вересня 2005 року в світ вийшла перша версія Joomla, що є по суті перейменованою Mambo 4.5.2.3 і що включає виправлення знайдених на той момент помилок і уязвимостей.

CMS Joomla включає різні інструменти для виготовлення веб-сайту. Важливою особливістю системи є мінімальний набір інструментів при початковій установці, який збагачується в міру необхідності. Це знижує те, що захащується адміністративної панелі непотрібними елементами, а також знижує навантаження на сервер і економить місце на хостингу.

2 ПРОЕКТНА ЧАСТИНА

2.1 Загальна структура сайта

На Рис. 2 представлена структурна схема сайта, яка враховує всю специфіку веб-сайта, що розробляється.



Рисунок 2 – Структурна схема сайта

На головній сторінці відображається основна інформація про магазин, голосування, авторизація клієнта та додаткове меню.

Сторінка «Каталог книг» містить категорії товарів, які є в магазині.

Сторінка «Новинки» містить ті товари, які недавно поступили до продажу.

Сторінка «Оплата и доставка» містить всю необхідну покупцям інформацію про оплату товару та її доставку.

Сторінка «Новости» містить найостанніші новини магазину.

Сторінка «Полезная информация» містить цікаві публікації, які дають певну інформацію про товари та шопінг.

Сторінка «Сотрудничество» містить інформацію про компанії, з якими співпрацює магазин та її побажання для інших компаній.

При відкритті сторінки «Контакты» користувач потрапляє на сторінку з контактами. Там він може знайти всю контактну інформацію щодо магазину.

На сторінці «Пошуку» відвідувачі можуть знайти будь-яку інформацію,

яка міститься на сайті.

Сторінка «Вакансии» містить інформацію щодо вакансій в магазині

2.2 Розробка інтерфейса веб-сайта

2.2.1 Дизайн веб-сайта

Відповідно до розробленої структури була спроектована головна сторіночка сайту (Рис. 3). Вона містить всі основні структурні елементи, перехід по яких здійснюється за допомогою гіперпосилань.



Рисунок 3 – Головна сторінка сайта

Як і планувалося на етапі постановки завдання, сайт містить всі необхідні структурні і навігаційні елементи: форму пошуку, навігаційне меню (у вигляді текстових посилань у супроводі відповідних зображень).

При натисненні на посиланні «Каталог книг» виводиться інформація про категорії, які існують в магазині. Це набагато спрощує пошук книг для клієнтів. (рис. 4).



Рисесунок 4 – Каталог книг

Вид реєстраційної форми для нового користувача показаний на рис. 5.



Рисунок 5 – Реєстраційна форма

Для того щоб авторизуватися старим користувачем, їм потрібно вказати логін і пароль в формі авторизації. Вид авторизаційної форми для старих користувачів показаний на (рис. 6.)

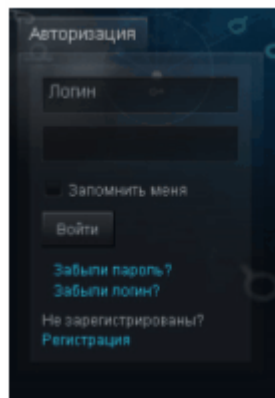


Рисунок 6 – Авторизаційна форма

2.2.2 Опис створення сторінок веб-сайта

Розробка веб-сайта велася в основному в програмном пакеті компанії Joomla 1.5. Вибір даного пакету обґрунтований тим, що його освоєння віднімає мало часу, під час розробки доступний попередній перегляд кожної окремо зробленої операції, доступна велика кількість функцій (організація інтерактивних елементів сайту, створення гіперпосилань в декілька натисків мишки і тому подібне), пакет підтримує велику кількість технологій (HTML,

PHP, ASP, Java, XML, XSLT, CSS та інші).

При створенні сайту використовувалися стандартні шаблони, проте для їх модифікації під конкретні завдання, стандартні засоби Joomla застосовувати незручно через відсутність візуального редактора. Тому для редагування шаблонів був використаний візуальний редактор Dreamweaver MX.

Розглянемо створення головної сторінки сайту (Рис. 7). Вона є основоположною для всіх інших.

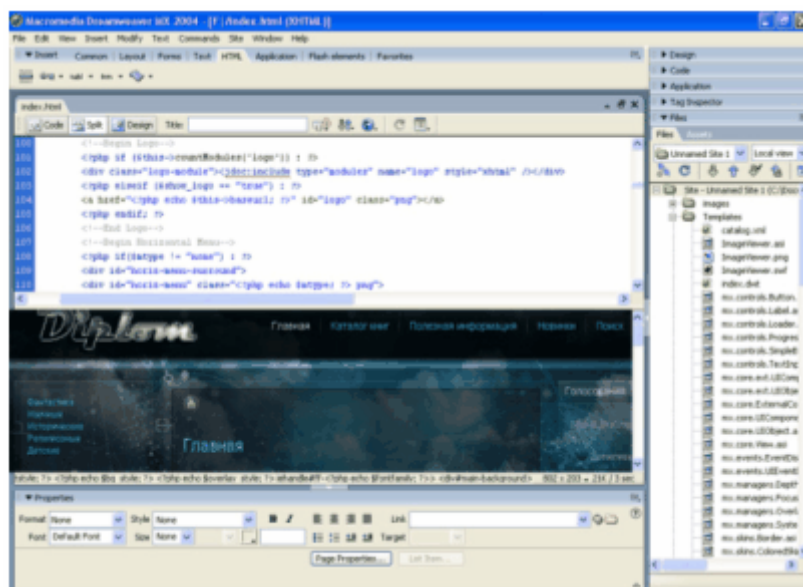


Рисунок 7 – Розробка головної сторінки сайта

Як видно з малюнка, в основі сторінки лежить контейнер. Всі елементи сторіночки складають мозаїку, яка згодом предстає на екрані користувача як єдине ціле.

Будь-яка інформація (текстова або графічна) заноситься безпосередньо в елементі контейнера. Таким чином, досягається структурованість сторіночки, що дозволяє найзручніше редагувати будь-який її елемент. Так само для зручності правки і логічного відділення одного сегменту від іншого кожному з них привласнено ім'я. Код шаблону сторінок приведений в додатку 1.

2.3 Розробка сайту на основі CMS

2.3.1 Опис встановлення на сервер CMS Joomla

Для доступу до адміністративної панелі управління Joomla використовуйте адресу Вашого веб-сайту (або повний шлях до теки, в яку встановлена Joomla) з додаванням в кінці шляху `"/administrator"`. Наприклад, якщо адреса веб-сайта `www.myweb.ua`, то доступ до панелі управління можливий за адресою `www.myweb.ua/administrator`. Після правильного введення адреси, відкриється сторінка авторизації (рис. 8)

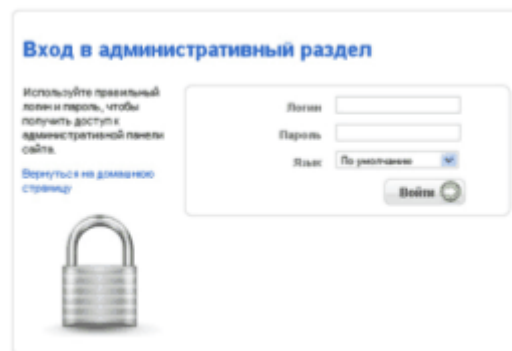


Рисунок 8 – Сторінка входу в панель управління Joomla

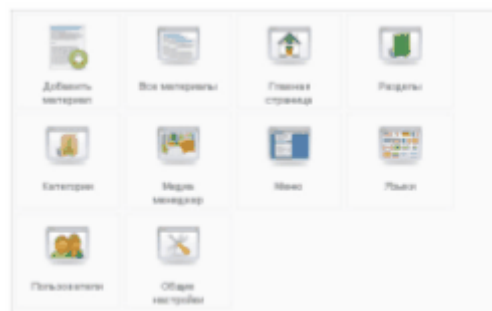


Рисунок 9 – Головна сторінка панелі управління Joomla

При правильному введенні імені користувача і пароля відкриється головна сторінка адміністрування системою, тобто буде відкрита панель управління Joomla (Рис. 9), яка надасть управління всіма функціями і можливостями Joomla.

2.3.2 Створення розділів сайту

Розділ – це головний (верхній) об'єкт в основі ієрархії структури вмісту. Пізніше в Розділ будуть додані Категорії.

Для роботи з розділами необхідно перейти на сторінку "Разделы". Можна натиснути на головній сторінці панелі кнопку "Разделы" або в основному меню вибрати "Материалы", потім пункт "Менеджер разделов" (Рис. 10).

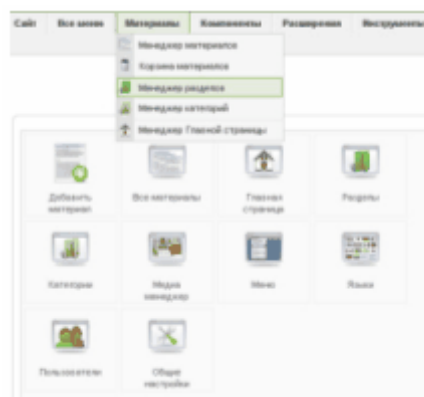


Рисунок 10 – Доступ на сторінку "Управління розділами" (Section Manager), або далі на сторінку "Менеджер категорій" (Category Manager)

Буде відкрита сторінка "Менеджер разделов".

На сторінці "Менеджер разделов", використовуючи кнопки на панелі інструментів, можна (перерахування справа наліво):

Допомогу відкрити в новому вікні сторінку опису по даному вікну ("Управління розділами"). Натисніть кнопку "Помощь".

Створити новий розділ. Натисніть кнопку "Создать".

Змінити існуючий розділ. Натисніть на назву розділу або відзначте потрібний розділ і натисніть кнопку "Изменить".

Видалити існуючий розділ. Відзначте потрібний розділ і натисніть кнопку "Удалить".

Копіювати існуючий розділ. Відзначте потрібний розділ і натисніть

кнопку "Копировать".

Приховати (зробити неопублікованим) існуючий розділ. Відзначте потрібний розділ і натисніть кнопку "Скрыть".

Допустити (Опублікувати) існуючий розділ. Відзначте потрібний розділ і натисніть кнопку "Показать".

Для створення нового розділу натисніть кнопку "Создать". Буде відкрита сторінка редагування розділу (Рисунок 11).

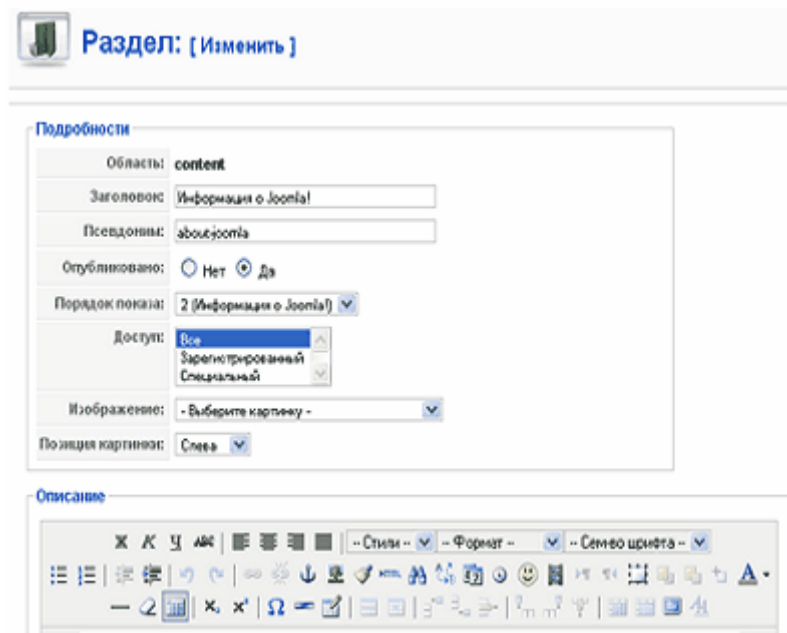


Рисунок 11 – Сторінка редагування розділу

На сторінці редагування розділу за допомогою кнопок на панелі інструментів можна (перерахування справа наліво):

Допомога – відкрити в новому вікні сторінку опису по даному вікну (редагування розділу). Натисніть кнопку "Помощь".

Застосувати (зберегти) зміни, але залишитися на сторінці редагування розділу. Натисніть кнопку "Применить".

Зберегти зміни і закрити сторінку редагування розділу і повернутися (вийти) на сторінку «Менеджер разделов» Натисніть кнопку "Сохранить".

Завантажити зображення (зберігається в теку images\stories). Натисніть

кнопку "Изображение".

На сторінці редагування розділу потрібно:

У полі "Псевдоним" ввести коротке ім'я розділу, яке використовуватиметься в меню або у випадних списках панелі управління.

У полі "Заголовок" ввести довге ім'я розділу, яке використовуватиметься при відображенні вмісту розділу на сайті, показуючи заголовок розділу.

У полі "Зображення" вибрати зображення, яке використовуватиметься при відображенні розділу в меню.

У полі "Порядок отобржения" після застосування (збереження) змін можна змінити порядок відображення розділу в списку на сторінці "Менеджер разделов".

У полі "Доступ" можна виставити права доступу на перегляд даного розділу (вмісту даного розділу):

"Все" - для всіх відвідувачів веб-сайта

"Зарагастрированный" - для зареєстрованих користувачів сайту

"Специальный" - для зареєстрованих користувачів панелі управління.

У полі "Опубликовано" - вибрати публікувати зараз чи ні.

У текстовому полі "Описание" можна написати короткий опис (анотацію) вмісту розділу.

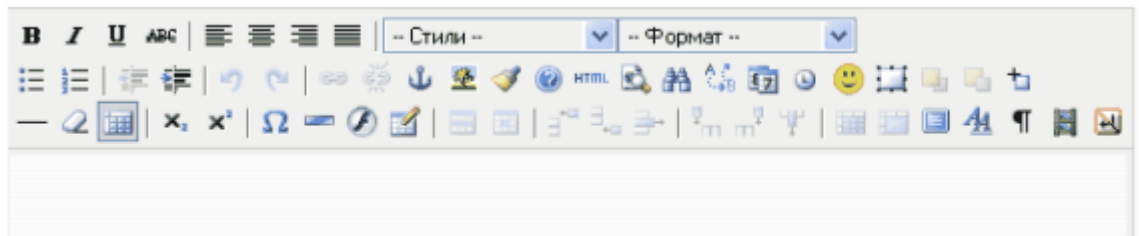


Рисунок 12 – Візуальний редактор "TINYMCE"

У полі "Описание" можна написати про зміст розділу, але пам'ятаєте, що розділ включає одну або декілька категорій, і, коли кінцевий користувач

(відвідувач Вашого веб-сайта) бачить цей опис, то воно також супроводжуватиметься (ймовірно) списком включених в даний розділ категорій.

У полі «Описание» використовується спеціальний редактор HTML, тобто не Joomla, а її розширення - мамбот. Дане розширення, мамбот, називається HTML-редактором Joomla. У системі можна використовувати будь-який з безлічі різних візуальних HTML-редакторів, але рекомендується використовувати "TINYMCE" (Рисунок. 12).

Якщо HTML-редактор не з'явився у Вашій системі, то Ви повинні встановити його і (або) включити редактор. У локалізованій збірці Joomla 1.5 редактор "TINYMCE" вже встановлений і включений. Зовнішній вигляд Вашого візуального редактора може сильно відрізнятися від того, що використовувався для написання цього керівництва.

Кнопки на панелі інструментів HTML-редактора допомагають задавати стиль (формат) тексту, що вводиться.

Коли Ви закінчите створювати або редагувати розділ, натисніть кнопку «Сохранить» для збереження результатів роботи і виходу на сторінку «Менеджер разделов» або можна натиснути кнопку «Применить» для збереження результатів, не закриваючи сторінки редагування розділу.

2.3.3 Створення категорій сайта

Категорія - це другий (після розділу) об'єкт в ієрархії структури вмісту. Створення категорії схоже на створення розділу.

Для роботи з категоріями необхідно перейти на сторінку «Менеджер категорій».

Є два способи перейти на сторінку «Менеджер категорій».

Перший спосіб: можна натиснути на головній сторінці панелі кнопку «Менеджер категорій», або в основному меню виберіть "Вміст", потім пункт «Менеджер разделов». Буде відкрита сторінка «Менеджер категорій».

Другий спосіб: у основному меню вибрати "Содержимое", потім

"Содеожимое по разделам" і далі потрібний вам розділ. І у випадному продовженні меню пункт "Добавить/изменить категории в выбранном вами разделе". Буде відкрита сторінка «Менеджер категорий» (Рисунок 13).

No	Заголовок	Опубликовано	Порядок	Доступ	Раздел	Активность	В порядке	ID
1	☐ Общие вопросы	✓	1	Все	FAQ	0	8	31
2	☐ Вопросы бывалых	✓	2	Все	FAQ	0	6	28
3	☐ Вопросы новичков	✓	3	Все	FAQ	0	3	27
4	☐ Вопросы по локализации	✓	4	Все	FAQ	0	5	32
5	☐ Проект	✓	1	Все	Информация о Joomla!	0	3	25
6	☐ SMS	✓	2	Все	Информация о Joomla!	0	5	29
7	☐ Сообщество	✓	3	Все	Информация о Joomla!	0	2	30
8	☐ Последние новости	✓	1	Все	Новости	0	5	1
9	☐ Краткие новости	✓	2	Все	Новости	0	5	3

Количество строк: 20

Рисунок 13 – Сторінка «Менеджер категорий»

При відкритті сторінки «Менеджер категорий» першим способом буде доступне фільтрація категорії по розділах. При створенні нової категорії можна буде вибрати розділ, до якого повинна увійти нова категорія. Опція фільтрації на сторінці доступна справа під панеллю інструментів.

При відкритті сторінки «Менеджер категорий» другим способом опція фільтрації буде "жорстко" прив'язана до вибраного Вами розділу. При створенні нової категорії ім'я розділу, до якого увійде нова категорії буде "жорстко" прив'язана до вибраного вами розділу.

Примітка: Щоб вибрати інший розділ, потрібно буде знову повторити спосіб другою або перший, для доступу на сторінку «Менеджер категорий».

На сторінці «Менеджер категорий», використовуючи кнопки на панелі інструментів, можна (перерахування справа наліво):

Допомога – відкрити в новому вікні сторінку опису поданому вікну («Менеджер категорий»). Натисніть кнопку «Помощь».

Створити нову категорію. Натисніть кнопку "Создать".

Змінити існуючу категорію. Натисніть на назву категорії або відзначте

потрібну категорію і натисніть кнопку «Изменить».

Видалити існуючу категорію. Відзначте потрібну категорію і натисніть кнопку «Удалить».

Копіювати існуючу категорію. Відзначте потрібну категорію і натисніть кнопку "Копировать".

Перенести – перемістити існуючу категорію і весь її вміст в інший розділ. Відзначте потрібну категорію і натисніть кнопку "Перенести".

Приховати (зробити неопублікованою) існуючу категорію. Відзначте потрібну категорію і натисніть кнопку "Скрыть".

Допустити (Опублікувати) існуючу категорію. Відзначте потрібну категорію і натисніть кнопку "Показать".

Для створення нової категорії натисніть кнопку "Создать". Буде відкрита сторінка редагування категорії.

Сторінка редагування категорії аналогічна сторінці редагування розділу (Рисунок. 13). Основна відмінність сторінок редагування "Категорії" і "Розділу" - наявність поля вибору розділу, до якого відноситиметься категорія. Створення нової категорії майже повністю аналогічно створенню нового розділу.

2.3.4 Створення об'єкту вмісту

Об'єкт вмісту – це основний вміст сайту і нижній об'єкт в ієрархії структури вмісту. Іншими словами об'єкт вмісту – це ваша стаття, яка повинна міститися в одній з ваших категорій (відповідно і розділу).

Для роботи з об'єктами вмісту необхідно перейти на сторінку "Менеджер матеріалів".

Потрібно натиснути на головній сторінці панелі кнопку "Менеджер матеріалів" або в основному меню виберіть "Матеріали", потім пункт "Менеджер матеріалів". Буде відкрита сторінка "Менеджер матеріалів", при відкритті сторінки буде доступна фільтрація об'єктів по наступних критеріях:

по розділах

по категоріях

по авторів

по заголовку, значення указується в полі "Фильтр".

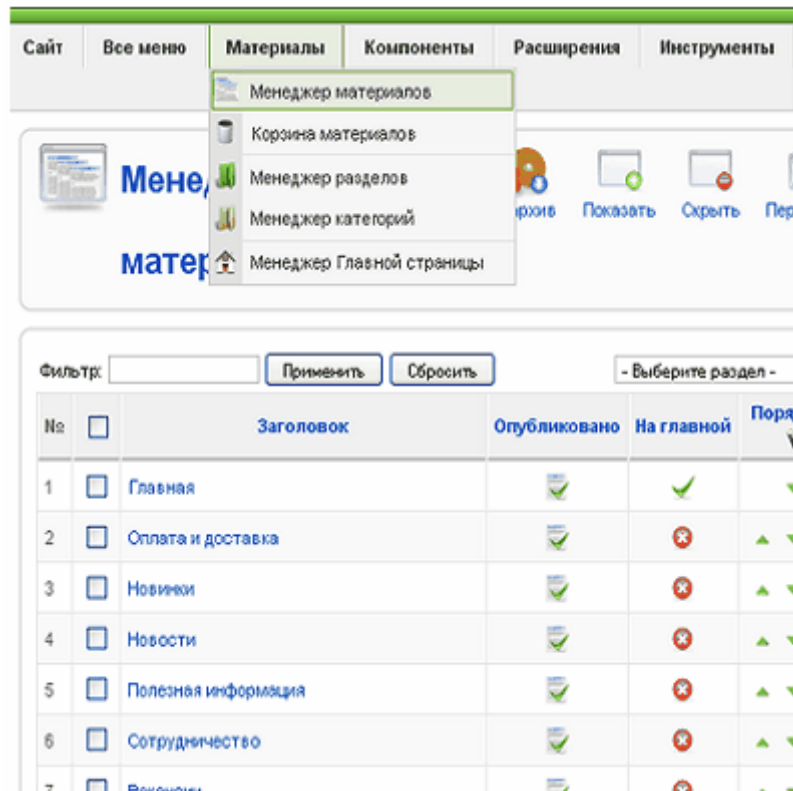


Рисунок 14 – Доступ до сторінки "Менеджер материалов"

На сторінці "Менеджер материалов", використовуючи кнопки на панелі інструментів, можна (перерахування справа наліво):

Допомога – відкрити в новому вікні сторінку опису поданому вікну ("Управління вмістом"). Натисніть кнопку «Помощь».

Створити новий об'єкт вмісту. Натисніть кнопку "Создать".

Змінити існуючий об'єкт вмісту. Натисніть на назву об'єкту або відзначте потрібний об'єкт і натисніть кнопку «Изменить».

Видалити існуючий об'єкт вмісту. Відзначте потрібний об'єкт і натисніть кнопку "В корзину".

Копіювати існуючі об'єкти вмісту. Відзначте потрібні об'єкти і

натисніть кнопку "Копировать" і потім виберіть, куди скопіювати об'єкти.

Перенести – перемістити існуючі об'єкти вмісту. Відзначте потрібні об'єкти і натисніть кнопку "Перенести" і потім виберіть, куди перенести об'єкти.

Приховати (зробити неопублікованим) існуючі об'єкти вмісту. Відзначте потрібні об'єкти і натисніть кнопку "Скрыть".

Допустити (Опублікувати) існуючі об'єкти вмісту. Відзначте потрібні об'єкти і натисніть кнопку "Показать".

Відправити до архіву існуючі об'єкти вмісту. Відзначте потрібні об'єкти і натисніть кнопку "В архив".

При створенні або редагуванні об'єкту вмісту (статті), на сторінці редагування (рисунк.15) доступні додаткові поля, необхідні для вибору того (на відміну від сторінки редагування розділу або категорії), в якому розділі (поле "Раздел:") і в якій категорії (поле "Категория:") міститиметься даний об'єкт вмісту.

Далі доступні *два* вікна HTML редактора, що дозволяють розбити зміст статті на дві частини: введення і весь вміст статті. Розділення вмісту статей дозволяє представити їх у вигляді так званого "Блога" або "Журналу", тобто списку короткого опису статей, з кнопками "Подробнее" на весь вміст статей.

Рисунок 15 – Сторінка редагування «Материал»

Блог – популярний і зручний спосіб представити інформацію або статті, які оновлюються (додаються) регулярно, наприклад новини або прес-релізи. Замість простого списку посилань до різних статей, виведення статей як "блог" дає можливість показати назву і введення, що дозволяють перейти до всього змісту статті. Таким чином, введення може бути першим параграфом всієї статті або коротким описом (резюме) статті.

Справа, після збереження, доступні додаткові параметри управління об'єктом вмісту. Остання вкладка "Связь с меню" може налаштовуватися при створенні меню.

Якщо необхідно помістити зображення в змісті, то потрібно використовувати спеціальну можливість вставки зображень. Натисніть кнопку "Вставити зображення" (кнопка розташована внизу зліва вікна HTML редактора), після цього у вміст буде доданий текст {mosimage }; при обробці цього тексту для відображення система (Joomla) визначить, що необхідно вставляти зображення, список яких розміщений на вкладці "параметри изображения". Щоб побачити зображення в тексті, натисніть кнопку "Просмотр" на панелі інструментів.

Використовувати можливість { mosimage } можна багато раз, зображення відображаються в порядку, вказаному в списку; порядок можна міняти, використовуючи кнопки "Вгору" і "Вниз".

Використання { mosimage } це приклад використання розширення Joomla – мамбота (mosimage).

Значення багатьох параметрів на вкладці "Параметры-Расширенные", можна задати за умовчанням, для того, щоб використовувалися "глобальні настройки". Задати значення "Глобальних настроек" можна в пункті меню "Менеджер материалов" меню "Материалы", у вкладці "Параметры". Дані значення можна індивідуально змінити для кожного об'єкту вмісту.

На даній вкладці можна ввести опис об'єкту і ключові слова. Якщо використовувати кнопку "Додати (Розділ, Категорія, Заголовок)", то будуть додані в полі "Ключовые слова" назви розділу, категорії і заголовок об'єкту.

2.3.5 Створення меню on-line магазину

Після створення структури вмісту, необхідно створити меню для користувачів веб-сайту. Меню – це набір кнопок, які приведуть на конкретні об'єкти, наприклад, об'єкт вмісту, список категорій, компонент і так далі.

Використовуючи розширення Joomla легко створити меню для відображення на сайті. У даному керівництві розглядається меню, побудоване на стандартному модулі (встановленому за умовчанням) `mod_mainmenu`. Даний модуль є за умовчанням головним для створення меню. Місце виведення меню на сайті визначається в шаблоні сайту. Встановлюючи додаткові розширення, можна розширити можливості відображення стандартного меню або відобразити його в іншому вигляді, наприклад у вигляді випадного каскадного меню з підміною (часто необхідно, щоб з боку користувача була включена підтримка Javascript в браузері).

Найпростіший варіант (але обмежений в можливостях) пов'язати об'єкт вмісту з пунктом меню, це використовувати на сторінці редагування об'єкту вмісту у вікні додаткових параметрів вкладку "связь с меню". У вкладці даного вікна необхідно:

1. Вибрати меню, в яке буде доданий пункт (посилання).
2. Вибрати назву пункту меню.
3. Натисніть кнопку "Связать с меню".

Примітка: На сторінці редагування розділу або категорії в додатковому вікні справа(для створюваного розділу або категорії дане вікно буде доступне після збереження – кнопка "Зберегти"). У даному вікні необхідно:

1. Вибрати меню, в яке буде доданий пункт (посилання).
2. Вибрати тип меню, в якому вигляді відображатиметься розділ або категорія.
3. Вибрати назву пункту меню.
4. Натисніть кнопку "Связать с меню".

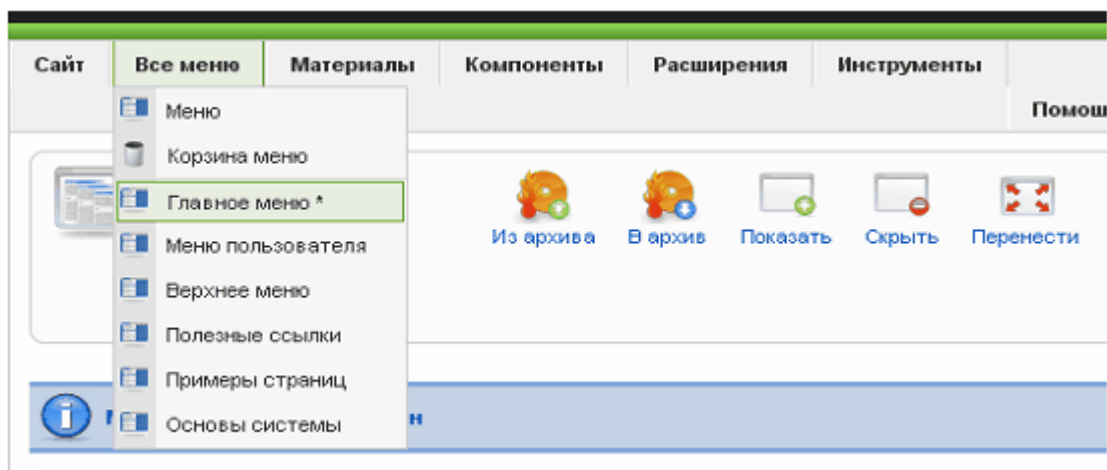


Рисунок 16 – Доступ до сторінки "Управление меню [mainmenu]"

Для работы с меню для сайта необходимо перейти на страницу "Главное меню". В основном меню выбрать "Все меню", потом пункт меню "Главное меню".

Для работы с пунктами меню певного меню (например, mainmenu) необходимо перейти на страницу "Управление меню [mainmenu]". В основном меню выбрать "Все меню", потом пункт меню "Главное меню".

На странице "Пункты меню[mainmenu]" (рисунок. 17) можно побачити список вже створених пунктів меню (за умовчанням без установки демонстраційних даних доступний тільки один пункт – "Главная").

Пункты меню: [mainmenu]

Все меню По умолчанию Показать Скрыть Перенести Копировать В корзину Изменить Создать Помощь

Фильтр: Применить Сбросить

Макс. уровень 10 - Выберите состояние -

№	Пункт меню	По умолчанию	Опубликовано	Порядок	Доступ	Системное имя	Itemid
1	Главная	★	✓	9	Все	Articles » Material	1
2	Каталог книг		✓	10	Все	VirtueMart	63
3	Полезная информация		✓	11	Все	Articles » Material	57
4	Сотрудничество		✓	1	Все	Articles » Material	58
5	Вакансии		✓	2	Все	Articles » Material	56
6	Новости		✓	3	Все	Articles » Material	59
7	Оплата и доставка		✓	4	Все	Articles » Material	60
8	Новости		✓	12	Все	Articles » Material	61
9	Поиск		✓	13	Все	Search » Поиск	62
10	Контакты		✓	14	Все	Articles » Material	55

Рисунок 17 – Сторінка "Пункты меню[mainmenu]"

Для додавання нового пункту меню необхідно перейти на сторінку "Создать". Натисніть кнопку "Создать" на сторінці "Управління Пункты меню[mainmenu]" для додавання нового пункту меню в меню mainmenu.

На сторінці "Пункт меню: [Создать]" (рисунк. 18) доступні декілька типів створюваного пункту меню. Всі типи розбиті по групах.

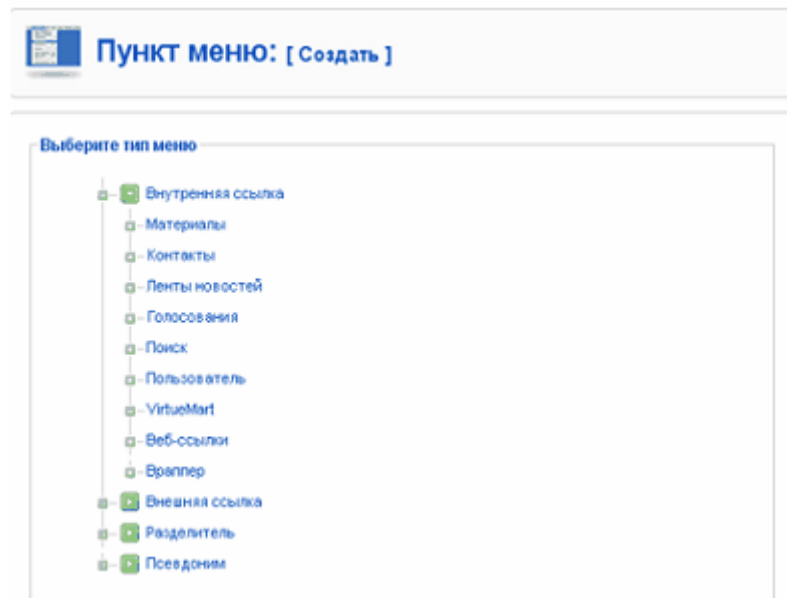


Рисунок 18 – Додавання нового пункту меню

Коли вибраний тип нового пункту меню, необхідно перейти на сторінку "Пункт меню: [Создать]". На цій сторінці необхідно вибрати і заповнити основні параметри і поля нового пункту меню. Параметри і поля відрізняються залежно від вибраного типу нового пункту меню. Натисніть на кнопку "Далі" або на вибраний тип пункту меню.

На рис. 19 розглядається приклад додавання нового пункту меню типу: "Стандартный шаблон материала". Для додавання необхідно:

1. Ввести ім'я нового пункту меню, поле "Заголовок".
2. Вибрати об'єкт вмісту (статтю), поле "Выберите материал".
3. Вибрати батьківський пункт меню, поле "Родительский элемент".
4. Вибрати порядок розташування, поле "Порядок показа".

5. Вибрати рівень доступу, поле "Доступ".

6. Вибрати опублікований пункт меню, поле «Опубліковано».

Справа можна, але не обов'язково вибрати зображення меню, поле "Картинка меню".

Рисунок 19 – Додавання нового пункта меню

У поле "Родительский элемент" перераховуються всі існуючі пункти меню, які можна вибрати як "родительский элемент" для нового пункта меню. Підтримка ієрархічної структури меню і відображення її на сайті можливі, тільки якщо в шаблоні є підтримка відображення ієрархії пунктів меню (більшість безкоштовних шаблонів не підтримують цю можливість). Для реалізації меню з підміною використовуються додаткові розширення, що дозволяють поліпшити створення і відображення структури меню на сайті.

2.3.6 Установка шаблона сайта

Якщо ви самі безпосередньо встановлюєте і настраюєте Joomla, то однією з перших дій з настройки є установка шаблону сайту для вашого проекту. Є багато вільно поширюваних шаблонів для Joomla 1.5, і установка

нового шаблону не є складною процедурою. Шаблони сайту для Joomla зазвичай розповсюджуються (так само як і розширення) у вигляді .zip файлів(або також .tar.gz файлу) .Для використання таких файлів необхідно зберегти файл шаблону і далі перейти на сторінку "Установка нового шаблону сайту" (рисунок 20).

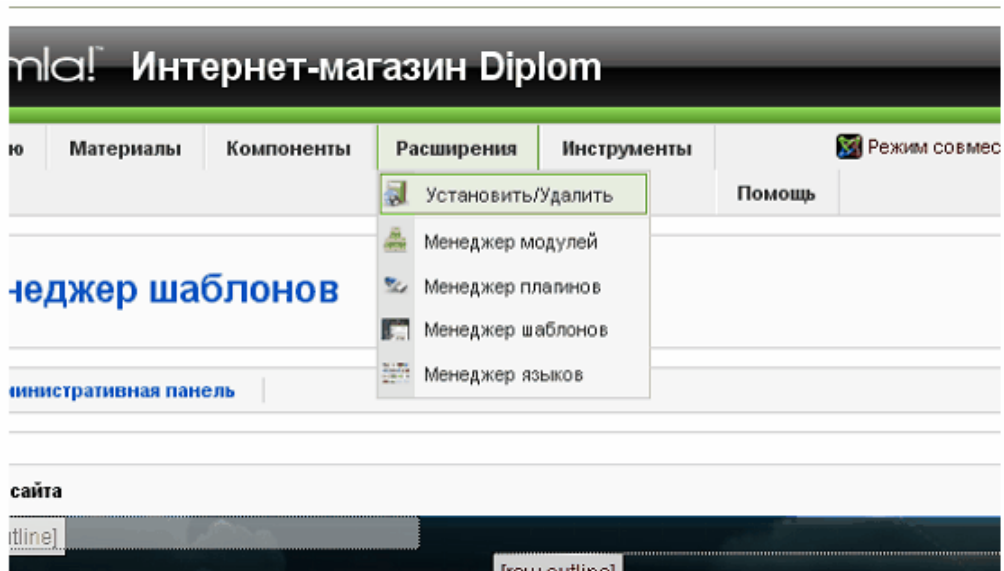


Рисунок 20 – Доступ до сторінки "Установить/Удалить"

На сторінці "Установить/Удалить" необхідно вибрати потрібний файл для установки. Натисніть кнопку "Обзор" для вибору файлу шаблону сайту і натисніть кнопку "Загрузить и установить".

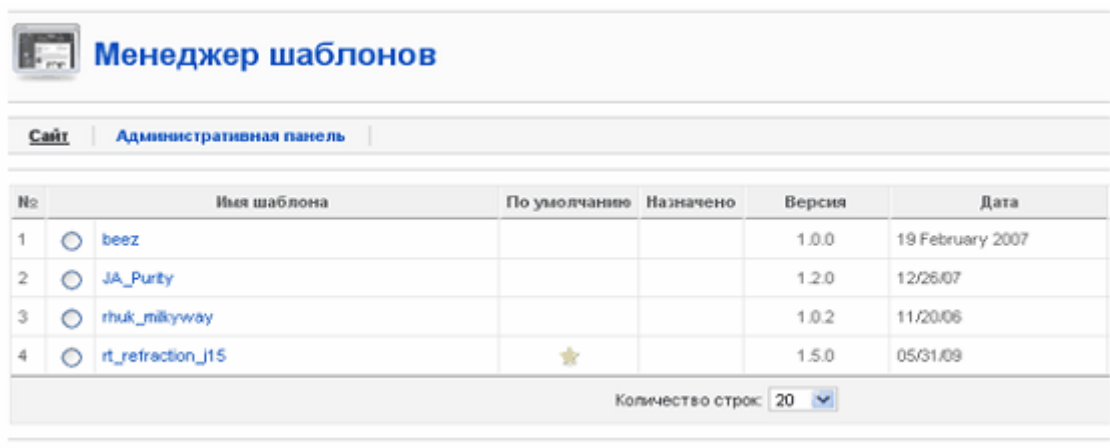


Рисунок 21 – Сторінка "Менеджер шаблонов"

Щоб почати використовувати шаблон сайту, необхідно перейти на сторінку "Менеджер шаблонів". Виберіть в головному меню "Расширения", потім пункти "Менеджер шаблонів".

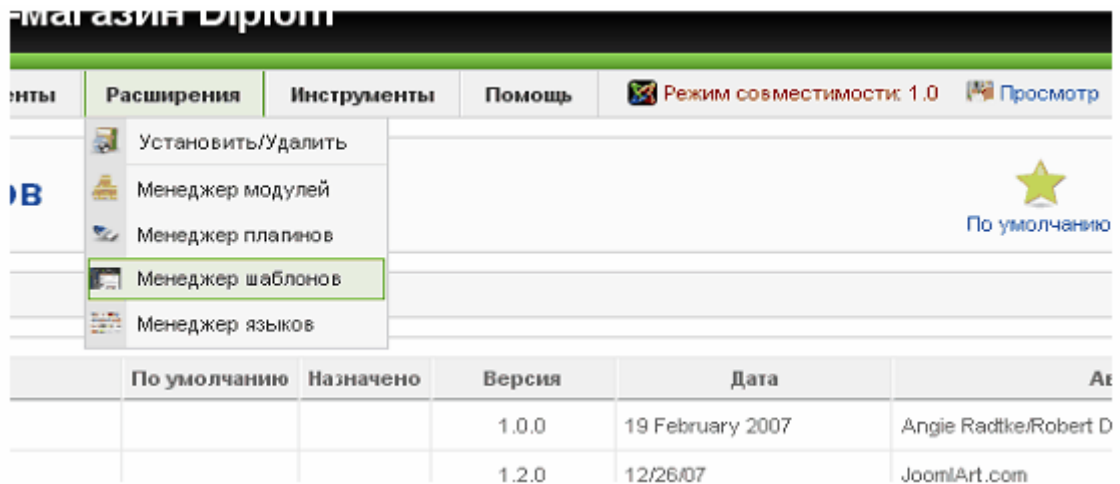


Рисунок 22 – Доступ до сторінки "Менеджер шаблонів"

На сторінці "Менеджер шаблонів" виберіть потрібний вам шаблон сайту і натисніть кнопку "По умолчанию". Шаблон сайту, вибраний за умовчанням, використовуватиметься для відображення всього сайту.

2.3.7 Створення категорії товарів магазину

Після встановлення та настройки необхідних для функціонування сайту розширень, необхідно створити категорії для товарів. Для цього ми використовуємо розширення для CMS Joomla "VirtueMart", яке розповсюджується по ліцензії GNU\GPL, тобто можна використовувати розширення безкоштовно і змінювати програмний код розширення для своїх потреб. Місце виведення категорій визначено в шаблоні сайту.

Для того, щоб створити нову категорію для товарів, потрібно в основному меню вибрати «Компоненты», потім пункт «VirtueMart» (рисунок 23). Після того, як перейшли на головну сторінку компонента «VirtueMart», потрібно вибрати «Товары», потім пункт «Добавить категорию» (рисунок 24).

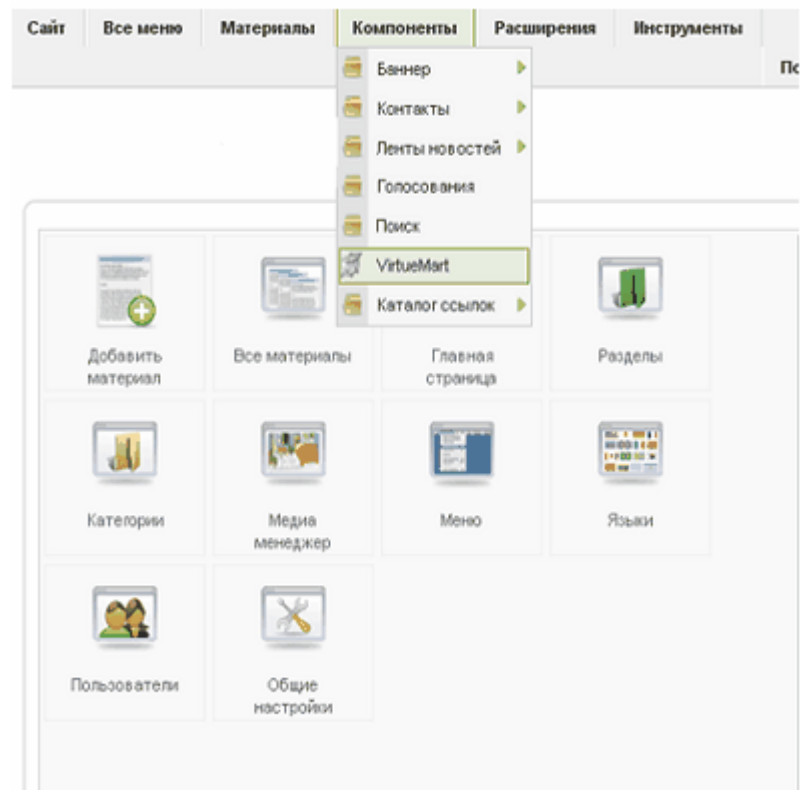


Рисунок 23 – Доступ до сторінки «VirtueMart».

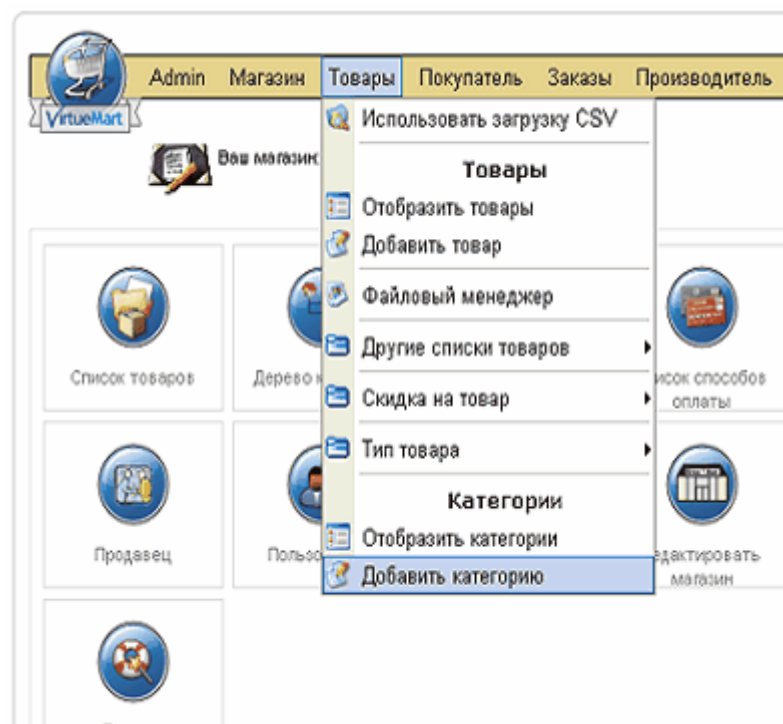


Рисунок 24 – Доступ до сторінки «Добавить категорию».

На рисунок 25 – розглядається приклад додавання нової категорії

товарів. Для додавання необхідно:

1. Ввести ім'я нової категорії в полі «Наименование категории».
2. В полі «Описание категории» ввести інформацію про категорію.
3. Вибрати чи буде це підкатегорія чи категорія в полі «Родительская категория».
4. В полі «Опубликовать?» відмітити, якщо потрібно щоб категорія була відображена на сайті.

У вкладці «Изображение» (рисунок. 26) можна загрузити зображення для категорії товарів.

Після того, як поля заповнені натискаємо «Сохранить» для зберігання введеної інформації.

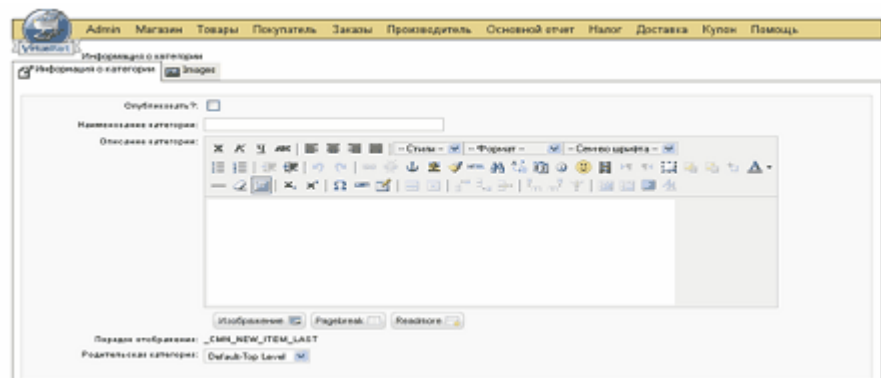


Рисунок 25 – Сторінка «Добавить категорию».

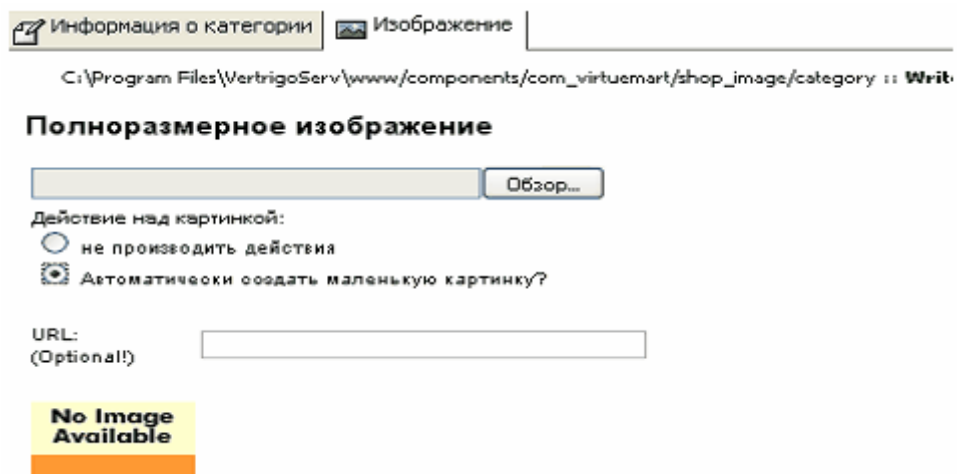


Рисунок 26 – Вкладка «Изображение».

Для того щоб передивитися категорії товарів, потрібно потрібно вибрати «Товары», потім пункт «Отобразить категорию». На сторінці «Дерево категорий» будуть відображені всі категорії товарів. Щоб категорія не відображалась в магазині, потрібно відзначити певну категорію та натиснути «Скрыть». Для того щоб видалити категорію, потрібно вибрати категорію та натиснути «Удалить».

2.3.8 Розміщення товарів в on-line магазині

Після створення категорій товарів, потрібно розміщати товари в on-line магазині. Для того, щоб додати новий товар в магазин, потрібно в основному меню вибрати «Компоненты», потім пункт «VireMart» (рисунок. 23). Після того, як перейшли на головну сторінку компонента «VirtueMart», потрібно вибрати «Товары», потім пункт «Добавить товар».

На рисунок. 27 розглядається приклад додавання нового товару. Для додавання необхідно:

1. В полі «Артикул» потрібно ввести артикул нового товару.
2. Ввести назву товару в поле «Название».
3. Вибрати категорію товару із списку поля «Категории».
4. В полі «Цена товара» вказати ціну даного товару.
5. Ввести коротке описання товару в поле «Краткое описание».
5. Ввести повне описання товару в поле «Описание товара».

Рисунок 27 – Сторінка «Новый товар».

На сторінці «Новый товар» окрім основної вкладки «Информация о товаре» існує 4 додаткові вкладки:

1. Вкладка «Состояние товара» - в цій вкладці можна додати інформацію про кількість товару, коли буде в наявності, спеціальні умови.
2. Вкладка «Габариты товара и вес» - в цій вкладці можна ввести габаритні параметри товару.
3. Вкладка «Изображения товара» - в цій вкладці можна завантажити зображення товару.
4. Вкладка «Сопутствующие товары» - в цій вкладці можна вибрати товари, які супутні даному товару.

2.4 Висновки

Отже, на підставі поставленого технічного завдання була вибрана система управління вмістом CMS Joomla, оскільки саме ця система відповідає всім вимогам технічного завдання.

Для розробки дизайну була вибрана програма Macromedia Dreamweaver MX 2004, оскільки дана програма дозволяє візуально споглядати за створенням дизайну для сайту. На головній сторінці було розміщено логотип веб-сайту, основне меню сайту (для навігації по його структурі), форму аутентифікації (входу зареєстрованих користувачів), реєстраційне посилання (реєстрація нових клієнтів).

Згідно технічного завдання було створено сайт інтернет-магазин на базі CMS Joomla. Був створений дизайн сайту в стилі шаблону, який використовується в системі управління вмістом Joomla. На головній сторінці розміщено логотип веб-сайту, основне меню сайту (для навігації по його структурі), форму аутентифікації (входу зареєстрованих користувачів), реєстраційне посилання (реєстрація нових клієнтів). Також сайт має початкові категорії товарів та початкові товари для відображення працездатності інтернет-магазину. Окрім каталогу товарів сайт має

інформативні сторінки, які мають інформативний вміст для користувачів і корисну інформацію для клієнтів інтернет-магазину.

ВИСНОВОК

В ході виконання випускної кваліфікаційної роботи був отриманий повнофункціональний веб-сайт, повністю готовий до застосування. Даний сайт орієнтований для широкого спектру застосування в on-line торгівлі. З його допомогою користувачі зможуть отримувати необхідну інформацію про товар, а також можливість купівлі товару. При розміщенні його в глобальній мережі географія розповсюдження зростає до масштабів всього світу.

При розробці веб-сайта були проаналізовані сучасні веб-технології, що дозволяють створювати інтерактивні веб-сторінки. Найбільш відповідними для виконання поставленого завдання виявилися Macromedia Dreamweaver MX 2004 і Joomla 1.5.

Розроблений сайт задовольняє всім вимогам, поставленим на етапі постановки завдання. При розробці веб-сайта були використані готові модулі аутентифікації, голосування, on-line магазин і здійснення пошуку по сайту. Дані модулі були допрацьовані з урахуванням специфіки веб-сайта і успішно упроваджені в його структуру.

Як подальше вдосконалення веб-сайта представляється можливим розробка модулів доступу. Так само можливе доопрацювання інтерфейсу сайту з метою подальшого підвищення його інформативності, привабливості і зручності.

ВИКОРИСТАНІ ДЖЕРЕЛА

1. «Веб Database Application with PHP and MySQL», 2nd Edition By David Lane, Hugh E. Williams. © O'Reilly, May 2004. ISBN: 0-596-00543-1.
2. CMS List. Огляд cms. Сайт про системи управління сайтом.
<http://www.cmslist.ru>
3. Joomla CMS по-російськи. <http://joomlaportal.ru>
4. Матеріал з Вікіпедії — вільній енциклопедії про системи управління сайтом. <http://ru.wikipedia.org/wiki/CMS>
5. Content management system <http://www.brutto.ru/informacija/uznat-bolshe/content-management-system>
6. Joomla-инструменты. <http://docs.joom.ru/!/;C651=00:Categories>
7. CMS огляд: CMS, движок сайта, система управления сайтом, mambo, php nuke, netcat, phpbb, invision power board, vbulletin. <http://cmsobzor.ru/news.php>
8. «Dreamweaver MX 2004 для "чайников".» Уорнер, Джанни, Гарднер, Сюзанна.
Пер. с англ. — М. : Издательский дом "Вильямс", 2004. — 352 с.
9. «PHP, MySQL и Dreamweaver MX 2004. Разработка интерактивных Web-сайтов.» Дронов В. А. — СПб.: БХВ-Петербург, 2005. — 448 с : ил.
9. 10. «PHP and MySQL Web Development (4th Edition)», Luke Welling, Laura Thomson 848 стр., с ил.; ISBN 978-5-8459-1574-0, 978-0-672-32916-6.
- 10.11. «Joomla! 1.5. Руководство пользователя», Колисниченко Д.Н., с ил.; ISBN 978-5-8459-1509-2; 2009, 224 с.
12. «Движок для вашего сайта. CMS Joomla!. Slaed, PHP-Nuke». Колисниченко Д., СПб.: БХВ-Петербург, 368 с.,
13. «Joomla! Практическое руководство», Норт Б., 448 с., СПб: Символ-Плюс.
14. «Самоучитель Joomla!», Дэн Рамел, БХВ-Петербург, 2008 г., 448 с.
15. «Создание веб-сайтов с помощью Joomla! 1.5». Хаген Граф, Издательский дом "Вильямс", 304 с., 2009.

ДОДАТКИ

Додаток А

Код шаблону сайта

```
<?php
// no direct access
defined( '_JEXEC' ) or die( 'Restricted index access' );
define( 'YOURBASEPATH', dirname(__FILE__) );
require( YOURBASEPATH.DS."styles.php");
require( YOURBASEPATH.DS."rt_styleswitcher.php");
JHTML::_( 'behavior.mootools' );

global $template_real_width, $leftcolumn_width, $rightcolumn_width,
$style;

global $js_compatibility, $menu_rows_per_column, $menu_columns,
$menu_multicollevel;

global $overlay_style, $bg_style, $showcase_title, $showcase_text,
$showcase_link, $body_text, $body_link, $primary_color;

$live_site = $mainframe->getCfg('live_site');
$template_path= $this->baseurl . '/templates/' . $this->template;
$preset_style = $this->params->get("presetStyle", "style3");
$overlay_style = $this->params->get("overlayStyle", "dark");
$bg_style = $this->params->get("bgStyle", "bgstyle3");
$showcase_title = $this->params->get("showcaseTitle", "#ffffff");
$showcase_text = $this->params->get("showcaseText", "#ffdd");
$showcase_link = $this->params->get("showcaseLink", "#21bddf");
$body_text = $this->params->get("bodyText", "#999999");
$body_link = $this->params->get("bodyLink", "#21bddf");
$frontpage_component = $this->params->get("enableFrontpage", "show");
```

```

$enable_ie6warn    =    ($this->params->get("enableIe6warn",    0)    ==
0)?"false":"true";

$font_family = $this->params->get("fontFamily", "refraction");

$enable_fontspans    =    ($this->params->get("enableFontspans",    1)    ==
0)?"false":"true";

$enable_inputstyle=    ($this->params->get("enableInputstyle",    1)    ==
0)?"false":"true";

$inputs_exclusion =                                $this->params->get("inputsExclusion",
"".content_vote");

$enable_featured_effects=    ($this->params->get("featEffects",    1)    ==
0)?"false":"true";

$panel_hooks=    $this->params->get("panelHooks",    ".showcase-panel-
trigger, a[href*=showcasemoduletrigger]");

$panel_fixheight= $this->params->get("panelFixHeight", 1);
$panel_height= $this->params->get("panelHeight", "337");
$panel_opacity= $this->params->get("panelOpacity", "0.9");
$panel_totop= $this->params->get("panelToTop", 1);
$panel_clickclose = $this->params->get("panelClickClose", 1);
$panel_closebutton= $this->params->get("panelCloseButton", 0);
$template_width = $this->params->get("templateWidth", "959");
$leftcolumn_width= $this->params->get("leftcolumnWidth", "210");
$rightcolumn_width= $this->params->get("rightcolumnWidth", "210");
$leftinset_width= $this->params->get("leftinsetWidth", "180");
$rightinset_width= $this->params->get("rightinsetWidth", "180");
$splitmenu_col= $this->params->get("splitmenuCol", "rightcol");
$menu_name = $this->params->get("menuName", "mainmenu");
$menu_type = $this->params->get("menuType", "moomenu");
$menu_rows_per_column = $this->params->get("menuRowsPerColumn");
$menu_columns = $this->params->get("menuColumns");
$menu_multicollevel = $this->params->get("menuMultiColLevel", 1);

```

```

$default_font = $this->params->get("defaultFont", "default");
$show_logo = ($this->params->get("showLogo", 1) == 0)?"false":"true";
$show_topbutton = ($this->params->get("showTopbutton", 1) ==
0)?"false":"true";
$show_copyright = ($this->params->get("showCopyright", 1) ==
0)?"false":"true";
$js_compatibility = ($this->params->get("jsCompatibility", 0) ==
0)?"false":"true";
// moomenu options
$moo_bgiframe = ($this->params->get("moo_bgiframe","0") ==
0)?"false":"true";
$moo_delay = $this->params->get("moo_delay", "500");
$moo_duration = $this->params->get("moo_duration", "600");
$moo_fps = $this->params->get("moo_fps", "200");
$moo_transition = $this->params->get("moo_transition", "Sine.easeOut");
$moo_bg_enabled = ($this->params->get("moo_bg_enabled","1") ==
0)?"false":"true";
$moo_bg_over_duration = $this->params->get("moo_bg_over_duration",
"500");
$moo_bg_over_transition= $this->params->get("moo_bg_over_transition",
"Expo.easeOut");
$moo_bg_out_duration= $this->params->get("moo_bg_out_duration",
"600");
$moo_bg_out_transition = $this->params->get("moo_bg_out_transition",
"Sine.easeOut");
$moo_sub_enabled= ($this->params->get("moo_sub_enabled","1") ==
0)?"false":"true";
$moo_sub_opacity= $this->params->get("moo_sub_opacity","0.95");
$moo_sub_over_duration= $this->params->get("moo_sub_over_duration",
"50");

```

```

        $moo_sub_over_transition=                                $this->params-
>get("moo_sub_over_transition", "Expo.easeOut");
        $moo_sub_out_duration =      $this->params->get("moo_sub_out_duration",
"600");
        $moo_sub_out_transition=    $this->params->get("moo_sub_out_transition",
"Sine.easeIn");
        $moo_sub_offsets_top= $this->params->get("moo_sub_offsets_top", "0");
        $moo_sub_offsets_right =      $this->params->get("moo_sub_offsets_right",
"1");
        $moo_sub_offsets_bottom=                                $this->params-
>get("moo_sub_offsets_bottom", "0");
        $moo_sub_offsets_left= $this->params->get("moo_sub_offsets_left", "1");
        require(YOURBASEPATH . "/rt_styleloader.php");?>
        <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
        <html   xmlns="http://www.w3.org/1999/xhtml"   xml:lang="<?php   echo
$this->language; ?>" lang="<?php echo $this->language; ?>" >
        <head>
        <jdoc:include type="head" />
        <?php
        require(YOURBASEPATH . DS . "rt_utils.php");
        require(YOURBASEPATH . DS . "rt_head_includes.php");
        ?>
        </head>
        <body id="ff-<?php echo $fontfamily; ?>" class="<?php echo $fontstyle; ?>
<?php echo $style; ?> <?php echo $bg_style; ?> <?php echo $overlay_style; ?>
iehandle">
        <div id="main-background">
        <!--Begin Header-->
        <?php   if ($show_logo == "true" or $mtype != "none" or $this-

```

```

>countModules('logo')) : ?>
    <div id="header-overlay" class="png">
    <div class="wrapper">
    <!--Begin Logo-->
    <?php if ($this->countModules('logo')) : ?>
    <div class="logo-module"><jdoc:include type="modules" name="logo"
style="xhtml" /></div>
    <?php elseif ($show_logo == "true") : ?>
    <a href="<?php echo $this->baseurl; ?>" id="logo" class="png"></a>
    <?php endif; ?>
    <!--End Logo-->
    <!--Begin Horizontal Menu-->
    <?php if($mtype != "none") : ?>
    <div id="horiz-menu-surround">
    <div id="horiz-menu" class="<?php echo $mtype; ?> png">
    <?php if($mtype != "module") : ?>
    <?php echo $stopnav; ?>
    <?php else: ?>
    <jdoc:include type="modules" name="toolbar" style="none" />
    <?php endif; ?>
    </div>
    </div>
    <?php endif; ?>
    <!--End Horizontal Menu-->
    </div>
    </div>
    <?php endif; ?>
    <!--End Header-->
    <!--Begin Showcase-->
    <?php $mClasses = modulesClasses('case5'); if ($this-

```

```

>countModules('showcase') or $this->countModules('showcase2') or $this-
>countModules('showcase3')) : ?>
    <div id="showcase-section" class="png">
    <div class="wrapper">
    <div id="showmodules" class="spacer"<?php echo $showmod_width; ?>">
    <?php if ($this->countModules('showcase')) : ?>
    <div class="block <?php echo $mClasses['showcase'][0]; ?>">
    <jdoc:include type="modules" name="showcase" style="main" />
    </div>
    <?php endif; ?>
    <?php if ($this->countModules('showcase2')) : ?>
    <div class="block <?php echo $mClasses['showcase2'][0]; ?>">
    <jdoc:include type="modules" name="showcase2" style="main" />
    </div>
    <?php endif; ?>
    <?php if ($this->countModules('showcase3')) : ?>
    <div class="block <?php echo $mClasses['showcase3'][0]; ?>">
    <jdoc:include type="modules" name="showcase3" style="main" />
    </div>
    <?php endif; ?>
    </div>
    </div>
    </div>
    <?php endif; ?>
    <?php if ($this->countModules('showcase-panel')): ?>
    <div id="showcase-panel">
    <div class="wrapper">
    <div id="showcase-panel-close">
    <div          class="readon-wrap1"><div          class="readon1-l"></div><a
class="readon-main" href="#"><span class="readon1-m"><span class="readon1-

```

```

r">Close</span></span></a></div>
    </div>
    <jdoc:include          type="modules"          name="showcase-panel"
style="showcasepanel" />
    </div>
</div>
<?php endif;?>
<!--End Showcase-->
<!--Begin Main Body-->
<div id="mainbody-overlay" class="png">
<div id="mainbody-overlay2" class="png">
<div class="wrapper">
<div id="main-body">
<div id="main-body-surround">
<!--Begin Main Content Block-->
<div id="main-content" class="<?php echo $col_mode; ?>">
<div class="colmask leftmenu">
<div class="colmid">
    <div class="colright">
        <!--Begin col1wrap -->
        <div class="col1wrap">
            <div class="col1pad">
                <div class="col1">
                    <div id="maincol2">
                        <div class="maincol2-padding">
                            <?php if ($this->countModules('newsflash')) : ?>
                            <div id="newsflash-bar" class="png">
                                <jdoc:include type="modules" name="newsflash" style="newsflash" />
                            </div> <?php endif; ?><?php $mClasses = modulesClasses('case0'); if ($this-
>countModules('feature') or $this->countModules('feature2') or $this-

```

```

>countModules('feature3')) : ?>
    <div id="featuremodules" class="spacer<?php echo $featuremod_width; ?>
png"> <?php if ($this->countModules('feature')) : ?>
    <div class="block <?php echo $mClasses['feature'][0]; ?> png">
    <jdoc:include type="modules" name="feature" style="feature" /> </div>
    <?php endif; ?> <?php if ($this->countModules('feature2')) : ?>
    <div class="block <?php echo $mClasses['feature2'][0]; ?> png">
    <jdoc:include type="modules" name="feature2" style="feature" /> </div>
    <?php endif; ?>
    <?php if ($this->countModules('feature3')) : ?>
    <div class="block <?php echo $mClasses['feature3'][0]; ?> png">
    <jdoc:include type="modules" name="feature3" style="feature" />
    </div>
    <?php endif; ?>
    </div>
    <?php endif; ?>
    <div id="maincontent-tm" class="png"><div id="maincontent-tl"
class="png"><div id="maincontent-tr" class="png"></div></div></div>
    <div id="maincontent-m" class="png"><div id="maincontent-l"
class="png"><div id="maincontent-r" class="png"><div id="maincontent-
surround" class="png"><?php if ($this->countModules('breadcrumb')) : ?>
    <div id="breadcrumbs">
    <a href="<?php echo $this->baseurl; ?>" id="breadcrumbs-home"
class="png"></a> <jdoc:include type="modules" name="breadcrumb"
style="none" /> </div> <?php endif; ?> <?php $mClasses =
modulesClasses('case1'); if ($this->countModules('user1') or $this-
>countModules('user2') or $this->countModules('user3')) : ?> <div
id="mainmodules" class="spacer<?php echo $mainmod_width; ?>">
    <?php if ($this->countModules('user1')) : ?>
    <div class="block <?php echo $mClasses['user1'][0]; ?>">

```

```

<jdoc:include type="modules" name="user1" style="main" /> </div> <?php
endif; ?> <?php if ($this->countModules('user2')) : ?> <div class="block <?php
echo $mClasses['user2'][0]; ?>"> <jdoc:include type="modules" name="user2"
style="main" /></div>
    <?php endif; ?>
    <?php if ($this->countModules('user3')) : ?>
    <div class="block <?php echo $mClasses['user3'][0]; ?>">
    <jdoc:include type="modules" name="user3" style="main" />
    </div>
    <?php endif; ?> </div> <?php endif; ?>
    <div class="bodycontent">
    <?php if ($this->countModules('inset2') and !$editmode) : ?>
    <div id="inset-block-right"><div class="right-padding">
    <jdoc:include type="modules" name="inset2" style="main" />
    </div></div>
    <?php endif; ?>
    <?php if ($this->countModules('inset') and !$editmode) : ?> <div id="inset-
block-left"><div class="left-padding">
    <jdoc:include type="modules" name="inset" style="main" />
    </div></div>
    <?php endif; ?>
    <div id="maincontent-block">
    <jdoc:include type="message" />
    <?php if (!$frontpage_component == 'hide' and JRequest::getVar('view') ==
'frontpage')): ?>
    <jdoc:include type="component" />
    <?php endif; ?> </div>
    </div><div class="clr"></div>
    <?php $mClasses = modulesClasses('case2'); if ($this-
>countModules('user4') or $this->countModules('user5') or $this-

```

```
>countModules('user6')) : ?>
```

```
<div id="mainmodules2" class="spacer<?php echo $mainmod2_width; ?>">
<?php if ($this->countModules('user4')) : ?> <div class="block <?php echo
$mClasses['user4'][0]; ?>"> <jdoc:include type="modules" name="user4"
style="main" />
```

```
</div>
```

```
<?php endif; ?>
```

```
<?php if ($this->countModules('user5')) : ?>
```

```
<div class="block <?php echo $mClasses['user5'][0]; ?>">
```

```
<jdoc:include type="modules" name="user5" style="main" />
```

```
</div>
```

```
<?php endif; ?> <?php if ($this->countModules('user6')) : ?>
```

```
<div class="block <?php echo $mClasses['user6'][0]; ?>">
```

```
<jdoc:include type="modules" name="user6" style="main" />
```

```
</div>
```

```
<?php endif; ?>
```

```
</div>
```

```
<?php endif; ?>
```

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

```
<div id="maincontent-bm" class="png"><div id="maincontent-bl"
class="png"><div id="maincontent-br" class="png"></div></div></div> </div>
</div>
```

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

```
</div> <!--End col1wrap -->
```

```
<!--Begin col2 -->
```

```
<?php if ($leftcolumn_width != 0) : ?>
```

```
<div class="col2">
```

```
<div id="leftcol">
```

```
<div id="leftcol-padding">
```

```
<?php if ($subnav and $splitmenu_col=="leftcol") : ?> <div
```

```

class="sidenav-block"> <?php echo $subnav; ?> </div>

    <?php endif; ?> <jdoc:include type="modules" name="left" style="sidebar"
/> <?php if (!isset($active)) :?><jdoc:include type="modules" name="inactive"
style="sidebar" /> <?php endif; ?>

    </div>

    </div>

    </div>

    <?php endif; ?>

    <!--End col2 -->

    <!--Begin col3 -->

    <?php if ($rightcolumn_width != 0) : ?>

        <div class="col3">

            <div id="rightcol">

                <div id="rightcol-padding">

                    <?php if ($subnav and $splitmenu_col=="rightcol") : ?> <div
class="sidenav-block"> <?php echo $subnav; ?> </div> <?php endif; ?>
<jdoc:include type="modules" name="right" style="sidebar" /></div>

                    </div> </div>

                    <?php endif; ?>

                <!--End col3-->

            </div>

        </div>

    </div>

    </div>

    <!--End Main Content Block-->

</div>

</div>

</div>

<!--Begin Bottom Main Modules-->

<?php    $mClasses    =    modulesClasses('case3');    if    ($this-

```

```

>countModules('user7')    or    $this->countModules('user8')    or    $this-
>countModules('user9')) : ?><div class="wrapper">
    <div id="bottom-main">
        <div id="mainmodules3" class="spacer<?php echo $mainmod3_width; ?>">
            <?php if ($this->countModules('user7')) : ?>
                <div class="block <?php echo $mClasses['user7'][0]; ?>">
                    <jdoc:include type="modules" name="user7" style="sidebar" />
                </div>
            <?php endif; ?>
            <?php if ($this->countModules('user8')) : ?>
                <div class="block <?php echo $mClasses['user8'][0]; ?>">
                    <jdoc:include type="modules" name="user8" style="sidebar" />
                </div>
            <?php endif; ?>
            <?php if ($this->countModules('user9')) : ?>
                <div class="block <?php echo $mClasses['user9'][0]; ?>">
                    <jdoc:include type="modules" name="user9" style="sidebar" />
                </div>
            <?php endif; ?>
        </div>
    </div>
    </div>
    </div>
    <?php endif; ?>
    <!--End Bottom Main Modules-->
</div></div>
</div>
<!--End Main Body-->
<!--Begin Bottom Section-->
    <?php    $mClasses    =    modulesClasses('case4');    if    ($this-
>countModules('bottom')    or    $this->countModules('bottom2')    or    $this-

```

```

>countModules('bottom3')) : ?>
    <div class="wrapper">
        <div id="bottom-border-top" class="png"></div>
        <div id="bottom" class="png"><div id="bottom2" class="png">
            <div id="bottom3" class="png"><div id="bottom4" class="png">
                <div id="mainmodules4" class="spacer"<?php echo $mainmod4_width; ?>">
                    <?php if ($this->countModules('bottom')) : ?>
                        <div class="block <?php echo $mClasses['bottom'][0]; ?>">
                            <jdoc:include type="modules" name="bottom" style="bottom" />
                        </div>
                    <?php endif; ?>
                <?php if ($this->countModules('bottom2')) : ?>
                    <div class="block <?php echo $mClasses['bottom2'][0]; ?>">
                        <jdoc:include type="modules" name="bottom2" style="bottom" />
                    </div>
                <?php endif; ?>
                <?php if ($this->countModules('bottom3')) : ?>
                    <div class="block <?php echo $mClasses['bottom3'][0]; ?>">
                        <jdoc:include type="modules" name="bottom3" style="bottom" />
                    </div>
                <?php endif; ?>
            </div>
        </div></div></div></div>
        <div id="bottom-border-bottom" class="png"></div>
    </div>
    <?php endif; ?>
    <!--End Bottom Section-->
    <!--Begin Footer-->
    <div id="footer">
    <div class="wrapper">

```

```

<?php if ($show_copyright == "true") : ?>
<div class="copyright-block">
  <a      href="http://www.rockettheme.com/"      title="<?php      echo
JText::_('ROCKETTHEME_JTC'); ?>" id="rocket" class="png"></a>
  <div id="copyright">
    &copy; <?php echo JText::_('COPYRIGHT'); ?>
  </div>
</div>
<?php else: ?>
<div class="footer-mod">
<jdoc:include type="modules" name="footer" style="xhtml" />
</div>
<?php endif; ?>
<?php if ($show_topbutton == "true") : ?>
<div id="top-button"><a href="#" id="top-scroll" class="top-button-desc">
<?php echo JText::_('TOP'); ?></a></div>
<?php endif; ?>
</div>
</div>
<!--End Footer-->
<?php if ($this->countModules('debug')) : ?>
<div id="debug-mod">
<jdoc:include type="modules" name="debug" style="none" />
</div>
<?php endif; ?>
</body>
</html>

```

Структурна схема веб- сайту



Головна сторінка Веб- сайту



Настройки Веб-сайта

