

Міністерство освіти і науки, молоді та спорту України
Міжнародний економіко-гуманітарний університет
імені академіка Степана Дем'янчука
Факультет кібернетики

Кафедра математичного моделювання

ДИСЕРТАЦІЯ МАГІСТРА З ІНФОРМАТИКИ

на тему:

**Розробка веб-сайту для підтримки роздрібних
торгівельних підприємств
(на прикладі приватного підприємця С.Онищука)**

Робота допущена до захисту
Протокол № 6 від “ 14” лютого 2012 р.
Завідувач кафедри
математичного моделювання
д. ф.-м. н., професор Джунь Й.В.

(підпис)

Студента магістратури
факультету кібернетики
групи 01КІН-М
спеціальності 8.080201
„Інформатика”
Онищука Сергія
Вікторовича

Науковий керівник:
к. т. н., доцент
Літнарів Р.М.

(підпис)

Рівне – 2012

ЗМІСТ

ВСТУП	4
РОЗДІЛ 1. ТЕОРЕТИЧНА ЧАСТИНА	
1.1 Поняття веб-сайту	6
1.1.1 Класифікація веб-сайтів	7
1.1.1.1 Сайт-візитка	7
1.1.1.2 Корпоративний сайт	7
1.1.1.3 Промо-сайт	8
1.1.1.4 Сайт-вітрина	8
1.1.1.5 Інтернет – магазин	9
1.1.2 Обґрунтування потреби у веб-сайті	10
1.1.3 Проектування веб-сайту	10
1.2 Програмні засоби	10
1.2.1 Web-сервери	11
1.2.2 Web-сервер Apache	12
1.2.3 PHP-гіпертекстовий препроцесор	13
1.2.4 MySQL проти PostgreSQL	14
1.2.5 Протокол http	18
1.3 Електронна комерція	22
1.3.1 Поняття торгівлі в Інтернеті	23
1.3.2 Особливості електронної торгівлі	24
РОЗДІЛ 2. ХАРАКТЕРИСТИКА ОБ'ЄКТУ ПРОЕКТУВАННЯ	
2.1 Опис об'єкту проектування	27
2.2 Актуальність проекту	28
2.3 Відвідування веб сайту та SEO оптимізація	29

РОЗДІЛ 3. СИСТЕМНИЙ АНАЛІЗ

3.1 Завдання проектного рішення.....	30
3.2 Дерево проблем.....	30
3.3 Обмеження доступу до захищених розділів сайту.....	32
3.4 Алгоритм генерації пароля.....	34
3.5 Інформаційне наповнення сайту.....	35
3.6 ІЛ 4. ОПИС ПРОГРАМИ	
4.1 Постановка задачі.....	38
4.2 Модель системи.....	42
4.3 Структура системи.....	43
4.4 Опис алгоритму.....	44
4.5 Блок-схема алгоритму.....	47
4.6 Опис тестової програми.....	53
4.7 Опис інтерфейсу програми.....	53
4.8 Результати роботи програми.....	56
4.9 Організація бази даних.....	65
4.10 Реалізація роботи з базою даних.....	66
РОЗДІЛ 5. ЕКОНОМІЧНА ЧАСТИНА	
5.1 Економічна характеристика проектного рішення.....	68
5.2 Висновок.....	69
ВИСНОВКИ.....	70
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ.....	71
ДОДАТОК Окремі тексти програм.....	74

ВСТУП

Досить часто в останні роки ми чуємо слово "веб-сайт". Сучасний світ зробив крок у бік цифрових технологій і крок цей подібний до стрибка в безодню. Ми дуже звикли до користування мережею Інтернет - зручного і що головне, миттєвого доступу до необхідної нам інформації. Наше життя буквально просочене інформацією і ми звикли одержувати її й ділитися нею у найкоротші строки.

Сам по собі Інтернет не є чимось новим та незвичним. Навіть в Україні, де ця мережа розвивалась із помітним запізненням, сьогодні ледь не у кожній домівці люди користуються мережею, черпаючи інформацію з різноманітних сайтів. Основною перевагою та перспективою цієї технології є те, що онлайн технологія може бути поєднана з оффлайн справами (покупками, роботою, відпочинком, тощо), а отже нашим життям, роблячи його ще комфортнішим та залишаючи більше часу для себе. Адже, для прикладу, пошук необхідного товару, консультація та перегляд відгуків значно економить наш час на подорожі по багатьом магазинам міста. Також мережа Інтернет пропонує значні переваги і для самих магазинів та підприємців, які створюють персональні веб-сайти, збільшуючи тим самим кількість потенційних клієнтів та партнерів, а також економлячи кошти на оренду торгових приміщень та рекламі.

Метою дисертаційної роботи є комплексний аналіз існуючих програмних систем, підходів, технічних засобів та розробка практичної методики для створення веб-сайтів по підтримці роздрібних торговельних підприємств.

Необхідно:

- ✓ проаналізувати основні програмні підходи по створенню веб-сайтів;
- ✓ проаналізувати сучасні технічні засоби по розміщенню веб-сайту в мережі;
- ✓ підібрати оптимальне середовище для надійного збереження даних;

- ✓ врахувати оптимізацію щодо швидкості та надійності роботи веб-сайту;
- ✓ захистити конфіденційні данні від несанкціонованого втручання, редагування чи копіювання;
- ✓ розробити архітектуру системи та структуру бази даних придатну для розширення та модифікації в майбутньому;
- ✓ створити зручний графічний інтерфейс користувача.

На сьогоднішній час, **тема дипломного проекту є досить актуальною** в Україні. Стрімкий ріст користувачів Інтернету та перехід комерції в онлайн-торгівлю є головною причиною популярності та доцільності створення веб-сайтів по представленню послуг та продажу товарів в мережі. Сфера застосування проектів цієї теми є досить широкою. Досить швидко та не виходячи із своєї домівки можна знайти необхідний товар і придбати його дешевше аніж в простому магазині, адже в цьому випадку в ціну товару не закладена оренда торговельного приміщення. Наразі, ця ніша в сфері послуг ще не є достатньо заповненою та розвинутою з точки зору якості, безпеки та зручності користування, а тому потребує вдосконалення, розвитку та іноваційних рішень.

Наукова новизна . В процесі виконання магістерської дисертації розроблено програмне рішення для підтримки роздрібних торговельних підприємств. Вибір предметної сфери зумовлений актуальністю веб-технологій, сферою їх застосування та стрімким розвитком Інтернет представництв оффлайн бізнесу.

Практична значимість і реалізація. Дана система розроблялася на веб-орієнтовній мові програмування PHP та СУБД MySQL. Розроблений нами проект є завершеним продуктом і може успішно функціонувати в мережі Internet надаючи перелічені вище переваги для роздрібних торговельних підприємств.

1. ТЕОРЕТИЧНА ЧАСТИНА

1.1 Поняття веб-сайту

Веб-сайт (англ. Website) — сукупність веб-сторінок, доступних у мережі Інтернет, які об'єднані як за змістом, так і навігаційно. Фізично сайт може розміщуватися як на одному, так і на декількох серверах.

Веб-сторінка є текстовим файлом з розширенням *.htm чи *.html, який містить текстову інформацію і спеціальні команди – HTML-теги, що визначають в якому вигляді ця інформація відобразатиметься 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), *.swf (Flash-аплікація). У HTML-кодї сторінки містяться тільки вказівники на такі файли.

Окрім файлів з розширенням *.htm та *.html, веб-сторінки можуть бути створенні за допомогою різноманітних мов програмування. Так для прикладу, в данному дипломному проєкті веб-сайт буде створено на основі мови програмування PHP. Для роботи веб-сайту, розробленому на PHP, сервер повинен бути обладнаний додатковим програмним забезпеченням, який дозволяє розуміти та компілювати PHP код, перетворюючи його у відповідний HTML-код, який вже в свою чергу сприймається браузером.

Кожна сторінка веб-сайту також має свій Internet адрес (URL), який складається з доменного імені, імені файлу та при потребі GET параметрів. 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. Переглянути веб-сайт можна практично з будь-якого комп'ютера, підключений до мережі Internet. [1,7]

Останім часом виробники мобільних телефонів також орієнтуються на користувачів Інтернету обладнуючи моделі телефонів браузерами для перегляду веб-сайтів. Щоправда перегляд сайтів на маленьких дисплеях телефонів є дещо

не зручним і саме тому, для деяких веб-сайтів додатково створюють вап версії сайтів, які спеціально оптимізовані під їх зручне користування з мобільних телефонів.

1.1.1 Класифікація веб-сайтів

На даний момент у всесвітній павутині розміщено декілька мільйонів веб-сайтів і їх число постійно росте. Усі вони поділяються на різноманітні категорії серед яких можна перелічити наступні: [15,16]

1.1.1.1 Сайт-візитка

Саме той набір інформації, поширення якої в Інтернеті, безсумнівно, корисно як починаючим, так і акулам бізнесу. Це своєрідний довідник про фірму, який містить всі необхідні контактні дані та інформацію про діяльність компанії. Це повинна бути коротка, чітка і найважливіша інформація про вас, вашу сферу діяльності, інтереси. Не варто перевантажувати ваш сайт-візитку інформацією, яка можливо не знадобиться потенційному клієнтові або партнерові.

Сайт-візитка – це:

- зручний засіб для сканування ґрунту ринку;
- не дорого;
- швидко;
- вигідний для організацій, у яких бізнес здійснюється не в Інтернеті.

1.1.1.2 Корпоративний сайт

Більш серйозним виглядом представництва компанії в Інтернеті є корпоративні сайти. Це обов'язковий інструмент успішного бізнесу будь-якої компанії. У першу чергу, сайти подібного роду допомагають формувати позитивний імідж компанії, залучають нових замовників, які можуть не знати про існування вашої фірми на ринку, і відповідно, розширюють ринок збуту компанії. За допомогою корпоративного сайту клієнти та замовники отримують інформацію про ціни на товари і послуги в зручному вигляді. Значно збільшити

прибуток ви зможете шляхом надання можливості робити замовлення вашим клієнтам прямо в Інтернеті, на вашій сторінці. [17]

Отже, корпоративний сайт – це:

- доказ серйозності і перспективності Вашого бізнесу;
- гідне представництво Вашої компанії на ринку;
- найдешевший і наймасовіший вид реклами Вашого бізнесу;
- зручно і комерційно виправдано.

1.1.1.3 Промо-сайт

Прекрасним рекламним інструментом може служити промо-сайт. Особливо, якщо Ви плануєте провести рекламну компанію окремого товару або послуги (культурного заходу. ..). Структура промо-сайту сама по собі повинна виконувати функцію рекламного проспекту або буклету, і привертати увагу клієнтів. Також на такому сайті можна розмістити ту інформацію, яка не поміститься на сторінці журналу, в радіоповідомленні, телевізійному ролику. Обсяг промо-сайту - до 10 сторінок, на яких можна розмістити дані про компанію, напрямок діяльності, просування послуги, контактні дані. Окремо повідомляється інформація про акцію, із зазначенням термінів і місця проведення, умов участі, новин та ін.

Промо-сайт – це:

- використання можливостей інтернету для впливу на цільову аудиторію;
- донесення до потенційних клієнтів максимум інформації про товар і послугу;
- потужний й ефективний інструмент реклами компанії та її продуктів.

1.1.1.4 Сайт-вітрина

Бюджетним варіантом реклами вашої компанії є сайт-вітрина. Якщо ви хочете продавати в Інтернеті один конкретний товар або групу товарів, то цей тип сайтів ідеально підходить для вас. На таких сторінках знаходиться вся інформація, необхідна для того, щоб клієнт побачив, зацікавився і купив даний

товар, не відволікаючись на новини та посилання. Це структурований каталог продукції вашої компанії. Але цей каталог організований таким чином (фото, опис, ціна), щоб клієнт зробив як можна менше кліків, для того, щоб знайти потрібний товар. Головне призначення сайту-вітрини - продавати. [17]

Сайт-вітрина – це:

- максимальний прибуток при мінімальному вкладенні сил та засобів;
- стабільний дохід;
- позитивний імідж компанії.

1.1.1.5 Інтернет-магазин

На відміну від сайту-вітрини, де клієнт може побачити наявність всіх товарів, за допомогою інтернет-магазину клієнт може ще й зробити замовлення, вибрати варіант розрахунку, спосіб отримання замовлення та одержати рахунок на оплату. Такий сайт має бути зручним і функціональним, щоб відвідувач міг легко знайти те, що йому потрібно, відправити товар у кошик та оформити покупку в кілька кліків. Інтернет-магазин істотно зменшить ваші витрати, так, як не треба орендувати приміщення для магазину, не треба платити заробітну плату співробітникам. Серед інших переваг: величезна аудиторія, можливість розміщувати необмежений асортимент товарів, своєчасно реагувати на зміни на ринку, враховувати потреби клієнтів та інші.

- Інтернет - магазин – це:
- нові можливості для бізнесу;
- розширення ринку збуту;
- залучення широкої аудиторії клієнтів;
- скорочення витрат.

Такі сайти в наш час набирають дуже велику популярність, як з боку користувачів, так і з боку підприємців. Це веб-сайти, в яких кожна людина може купити будь-який товар не відходячи від свого комп'ютера. [16,17]

1.1.2 Обґрунтування потреби у 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. Для прикладу значна кількість потенційних клієнтів з більшою долею ймовірності відвідає веб-сайт, щоб ознайомитись з діяльністю компанії, аніж дзвонитиме за номером телефону опублікованому цією компанією в рекламній брошурі. Ти більше для того, щоб розмістити на рекламних носіях адресу веб-сайту компанії достатньо лише однієї стрічки, а об'єм поданої інформації на сторінках самого веб-сайту не обмежений і завжди актуальний .[1,2,7]

1.1.3 Проектування веб-сайту

Проектування і розробка сайтів включає наступні необхідні пункти:

- Затвердження первинного технічного завдання розробки сайту.
- Визначення структурної схеми сайту - розташування розділів, контенту і навігації.
- Веб-дизайн - створення графічних елементів макету сайту, стилів і елементів навігації.
- Розробка програмного коду, модулів, бази даних і інших елементів сайту необхідних в проекті.
- Тестування і розміщення сайту в мережі Інтернет. [10]

1.2 Програмні засоби

Для створення веб-сайту використовуються різноманітні технології і мови програмування. Практично жоден сучасний проект не обходиться без програмних засобів.

Отже, щоб почати працювати над нашим власним сайтом, нам знадобиться декілька продуктів, що наведені нижче.

1.2.1 Web-сервери

До основних функцій Web-серверів відноситься збереження і пересилка документів у форматі HTML (та в інших форматах) по протоколу HTTP, забезпечення роботи з інформацією, що міститься в базах даних, обробка запитів користувачів за допомогою спеціально розроблених програм.

Клієнтське ж програмне забезпечення (браузер) посилає запити користувача та отримує від сервера необхідну інформацію, яку відображає у своєму вікні. (рис.1.1)

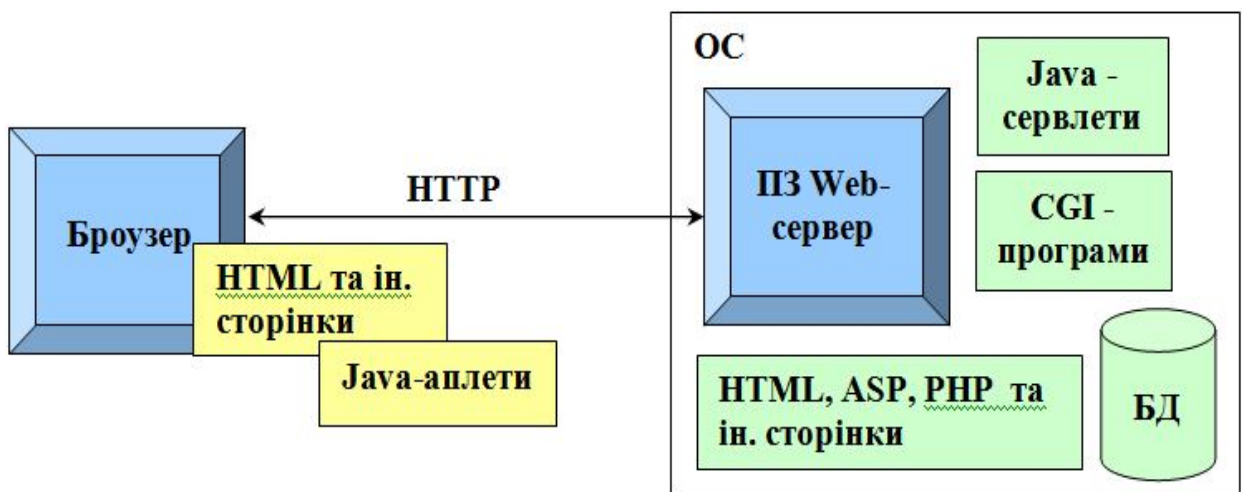


Рис. 1.1. ПЗ Web-сервера

Нинішнє програмне забезпечення Web-серверів відрізняється великими потужностями, низькими цінами, усі зростаючим числом додаткових інструментів і постійно розширюються можливостями.

При виборі серверного програмного забезпечення необхідно насамперед вирішити, яка буде використовуватися операційна система. Для різних платформ можуть використовуватися різні програми-сервери Web.

Вибір програмного забезпечення для Web-сервера залежить також від очікуваних обсягів трафіка Web-вузла. Так, для невеликих обсягів трафіка підходять недорогі продукти на зразок безкоштовного пакета Personal Web

Server (Microsoft), для середніх корпоративних і загальнодоступних вузлів можуть використовуватися WebSite Professional (O'Reilly and Associates) чи FastTrack Server (Netscape Communications). Для вузлів з великим трафіком необхідно використовувати такі продукти, як Apache, Enterprise Server (Netscape), Internet Information Server (Microsoft) і Web Server (Novell).

Компанія Netcraft (<http://www.netcraft.com/Survey/>) опублікувала статистичні дані по використовуваному програмному забезпеченню Web-серверів і їхніх операційних систем. Було обстежено 35,543,105 сайтів. Лідерів серед програмного забезпечення для web-серверів (за результатами цього дослідження) подано в таблиці 1.1.

Таблиця 1.1. Проценти використання web-серверів

Web-сервер	Відсоток використання
Apache	62.02%
Microsoft	27.58%
Zeus	2.12%
iPlanet	1.35

1.2.2 Web-сервер Apache

Apache - це потужний і зручний Web-сервер, сумісний зі стандартом HTTP/1.1. Якщо ви створюєте свій динамічний сайт із використанням мов програмування Perl або PHP, а може просто пишете сайт із використанням директив SSI, то без нього вам не обійтися. Звичайно, якщо ви працюєте із простими сторінками, які написані на HTML, то сервер Вам поки не потрібний, але це тільки поки. Web-сервер Apache називають самим головним скарбом руху "Open source". Його можна одержати зовсім безкоштовно. Він має відмінні робочі характеристики й тому використається більш широко, ніж всі інші Web-сервери разом узяті. У даний момент більше 65 відсотків всіх Web-вузлів у світі створені з використанням сервера Apache. Сервер Apache має ще одну перевагу:

він простий настільки, що будь-який досить грамотний користувач може опанувати ним. [5 - 6]

Переваги Web-сервера Apache:

- Модульність структури;
- Відкрита архітектура (можна скачати як вихідний код, так й відкомпільований варіант);
- Працездатність під декількома платформами:
 - Unix
 - Linux
 - Windows
 - Netware
- Безкоштовне поширення.

1.2.3 PHP-гіпертекстовий препроцесор

PHP - це мова програмування для динамічної генерації web-сторінок за допомогою скриптів, що запускаються на web-сервері. Ви створюєте сторінку за допомогою php та html. Коли відвідувач сайту відкриває сторінку, сервер виконує включені в html-код оператори php і посилає результат браузеру відвідувача, точно так само, як це робиться за допомогою asp або coldfusion. Однак, на відміну від asp чи coldfusion, php є продуктом з відкритим вихідним кодом (open source) і незалежним від платформи. php працює на windows і багатьох версіях unix. Він може бути запущений як модуль в apache або програма, що виконується через cgi. У випадку запуску у вигляді модуля apache, php працює особливо легко й швидко. У цьому випадку відсутні накладні витрати, пов'язані зі створенням процесів, тому результат видається швидко, і не потрібно набувати mod perl для зменшення витрати пам'яті сервера. [7]

Крім різних операцій із змістом Ваших сторінок, за допомогою php, Ви можете формувати http-заголовки, установити cookie, управляти аутентифікацією і перенаправляти користувача на іншу сторінку. Php надає

чудові можливості по доступі до баз даних (і odbc), а також інтеграцію з різними зовнішніми бібліотеками. Це дозволяє Вам робити все - від генерації pdf-документів до граматичного розбору в xml. [3,4]

Оператори php вставляються прямо у Ваші web-сторінки так, що немає необхідності в спеціальних середовищах розробки (ide). Ви починаєте блок php-коду тегом `<?php` і закінчуєте його тегом `?>`. (Ви можете сконфігурувати php так, що будуть використатися теги в стилі asp: `<% %>` або теги виду `<script language="php"></script>`.) Усе, що перебуває між цими тегами буде інтерпретуватися як php-код.

Синтаксис мови php подібний до синтаксису perl. Ви не повинні повідомляти змінні до їхнього використання. Легко створюються масиви й хеші (асоціативні масиви). В 5-й версії PHP повноцінно підтримує об'єктно-орієнтоване програмування, забезпечуючи зручний спосіб організації коду.

Хоча PHP швидше всього працює в тому випадку, коли він вбудований в apache, на web-сайті php є інструкції по його встановленні на Microsoft та Netscape Enterprise Server. PHP є безкоштовним і його можна скачати з офіційного web-сайту (www.php.net). Там же знаходиться докладне керівництво, у якому описані всі функції й особливості php. [7- 10]

1.2.4 MySQL проти PostgreSQL

Дуже часто запитують, "Чому ви надаєте перевагу, PostgreSQL чи MySQL?". Відповідь завжди та сама: "Це питання переваги". Ви можете задати те ж саме питання безлічі інших розробників, і їхні відповіді будуть досить різні. Ось порівняння баз даних MySQL й PostgreSQL, запропоноване не заради висловлення думки, а заради того, щоб допомогти іншим прийняти власне рішення:

Обидвом системам є що запропонувати в питаннях стабільності, гнучкості й продуктивності. MySQL має особливості, які у PostgreSQL є недолікам, і

навпаки. Первинне завдання - допомогти вирішити, яка із цих двох баз даних буде використатися у ваших власних розробках. [7, 8]

Перш, ніж почати порівняння цих баз даних, потрібно прояснити, що будуться дотримуватися інсталяцій за замовчуванням. MySQL має багато різних типів таблиць, які підтримують транзакції й зовнішні ключі. Однак, деякі з конфігурацій цих типів таблиць досить складні. Не багато розробників мережі або програмісти використовують додаткові типи таблиць, що існують в MySQL. А тепер давайте зрівняємо ці два продукти.

Список особливостей і можливостей

У таблиці 1.2 наведене порівняння найбільш вживаних особливостей і можливостей баз даних MySQL й PostgreSQL. Таблиця 1.2 - це не вичерпний список особливостей, типів даних або проблем продуктивності, що стосується цих двох систем баз даних. Вона лише дає деяке подання про те, що кожна з них може запропонувати. З таблиці ми бачимо, що PostgreSQL пропонує повні особливості й можливості традиційних додатків баз даних, у той час як MySQL зосереджує на більше швидкому виконанні (роботі) для веб додатків. Розвиток індустрії "open source" принесе більшу кількість особливостей і можливостей у наступних версіях обох баз даних. [8, 10-11,18]

Таблиця 1.2. Порівняння MySQL та PostgreSQL

Особливості	PostgreSQL	MySQL
ANSI SQL сумісність	Близька до стандарту ANSI SQL	Треба деяким стандартам ANSI SQL
Швидкість роботи	Повільніше	Швидше
Вкладені селекти	Так	Немає
Транзакції	Так	Так, однак повинен використатися тип таблиці InnoDB
Відповідь бази даних	Так	Так
Підтримка зовнішніх ключів	Так	Немає
Подання	Так	Немає

Збережені процедури	Так	Немає
Тригери	Так	Немає
Unions	Так	Немає
Повні Joins	Так	Немає
Обмежники цілісності	Так	Немає
Підтримка Windows	Так	Так
Вакуум (очищення)	Так	Немає
ODBC	Так	Так
JDBC	Так	Так
Різні типи таблиць	Немає	Так

Коли використати MySQL?

В яких випадках краще використовувати MySQL, ніж PostgreSQL? Якщо потрібно створити веб додаток, і головне, це продуктивність і швидкість - MySQL буде кращим вибором, тому що вона швидка й розроблена для того, щоб добре працювати з веб серверами. Однак, якщо потрібно створити інший додаток, що вимагає виконання транзакцій і наявності зовнішніх ключів, кращим вибором стане PostgreSQL.

Навіть при тому, що MySQL не повністю сумісне з ANSI SQL стандартом, потрібно згадати, що, у той час як PostgreSQL ближче до ANSI SQL стандарту, MySQL ближче до ODBC стандарту. [12,18]

Деякі плюси використання MySQL:

- MySQL відносно швидше PostgreSQL.
- Дизайн і планування бази даних трохи простіше.
- Можна створити простий веб сайт із використанням бази.
- Відповіді на запити MySQL були добре протесторованні.
- Немає потреби використати методи очищення (вакуум).

Коли використати PostgreSQL?

Не багато веб-розробників використовують у своїй роботі PostgreSQL, тому що вважають, що додаткові особливості й можливості знижують продуктивність і швидкість роботи. Однак, PostgreSQL має багато переваг над MySQL. Наприклад, деякі з особливостей, які часто використовуються - зовнішні ключі, тригери й подання. Вони дозволяють приховувати складність бази даних від додатку, у такий спосіб уникаючи створення складних команд SQL. Існує чимало розробників, які обирають багаті функціональні можливості SQL команд PostgreSQL. Одне з найбільш відчутних розходжень між MySQL й PostgreSQL - неможливість створення вкладених підзапитів (селектів) в MySQL. PostgreSQL відповідає багатьма SQL стандартам ANSI, у такий спосіб дозволяючи створення складних команд SQL. [12]

Кілька причин використати PostgreSQL:

- Складний дизайн бази даних.
- Складні набори правил.
- Використання процедурних мов на сервері.
- Транзакції
- Використання збережених процедур.
- Використання географічних даних.
- R-Trees (наприклад, використання індексів).

Висновок

Ви повинні будете вибрати, зваживши всі плюси та мінуси, яка база даних є "створеною" для вашого додатка або сайту. А може бути й таке, що ви захочете використати обидві бази (бувають і такі випадки). Одна база не обов'язково краща за іншу, і кожна з них займає свою відповідну нішу у світі баз даних з відкритим вихідним кодом.[7-11]

1.2.5 Протокол НТТР

Протокол НТТР (Hypertext Transfer Protocol, протокол передачі гіпертекстів) - це протокол прикладного рівня для розподілених гіпертекстових інформаційних систем. Крім передачі гіпертекстів він може застосовуватися й в інших областях, таких, як сервери імен і розподілені системи керування об'єктами, але для наших цілей важливо те, що на цьому протоколі з моменту своєї появи в 1990 р. і донині базується World Wide Web.

НТТР є протоколом типу запит/відгук. Клієнт посилає серверу запит, що складається з типу запиту, URI і версії протоколу, за якими слідує повідомлення, що містить модифікатори запиту, клієнтську інформацію й можливо тіло запиту. Сервер відповідає рядком стану, що містить версію протоколу й код стану, за якими слідує повідомлення, що містить серверну інформацію, метаінформацію й можливо тіло повідомлення. [5]

Повідомлення НТТР

Повідомлення НТТР складаються із запитів клієнта до сервера й відгуків сервера до клієнта. Обидва типи повідомлень складаються з:

- стартового рядка, що визначає тип повідомлення;
- нуля або більше полів заголовка (коротко названих просто заголовки);
- порожнього рядка, що містить тільки символи CR LF, що вказує на кінець заголовків;
- необов'язкового тіла повідомлення. [9]

Кожне з полів заголовків має формат:

ім'я_поля ":" значення_поля

Імена полів не залежать від регістра. Порядок, у якому передаються поля заголовка, неважливий, хоча звичайно першими йдуть загальні поля заголовка, потім заголовки запиту або відгуку, а потім заголовки тіла.

HTTP допускає об'єднання декількох полів заголовка, що мають одне ім'я. Таке поле заголовка складається з імені поля, двокрапки й набору значень, розділених комами, тобто:

ім'я_поля ":" значення1 "," значення2

Якщо тіло повідомлення закодоване з міркувань безпеки, то частина полів заголовка може бути причеплена до нього, тобто додана в кінець тіла повідомлення й також закодована. [6]

Типи запитів

Рядок запиту має вигляд

метод " " URI_запиту " " версія_HTTP CR LF

Метод - це назва типу запиту, URI_запиту - це або URI запитуваного ресурсу, або символ "*" у тих випадках, коли запит не пов'язаний з конкретним ресурсом, а версія_HTTP виглядає так:

"HTTP/" цифра "." цифра

Типові приклади рядків запиту:

GET http://www.mysite.com/index.html HTTP/1.1

OPTIONS * HTTP/1.1

Запит OPTIONS

Метод OPTIONS являє собою запит інформації про умови комунікації, доступних у ланцюжку запит/відгук для даного URI запиту. Він дозволяє клієнтові визначити опції й/або вимоги, пов'язані з даним ресурсом, не виконуючи ніяких операцій з ресурсом. [13]

Якщо URI запиту задано символом "*", то запит ставиться до сервера, а не до конкретного ресурсу, і дозволяє запросити можливості сервера.

Запит GET

Метод GET використовується для запиту конкретного ресурсу. Зокрема, він генерується щораз, коли ми клацаємо у вікні оглядача на гіперпосилання.

Цей метод є умовним, якщо в його заголовку задані поля If-Modified-Since, If-Unmodified-Since, If-Match, If-None-Match або If-Range. Умовний метод відрізняється тим, що запитаний ресурс буде повернутий тільки в тому випадку, коли він задовольняє заданій умові.

Запит HEAD

Метод HEAD повністю аналогічний методу GET, але по ньому сервер повертає тільки метаінформацію, пов'язану із запитаним документом, а не повністю весь документ. Звичайно цей метод використовується клієнтами, які підтримують кешування ресурсів, для перевірки того, чи варто оновити інформацію в кеші й перезавантажити документ.

Запит POST

Метод POST використовується в наступних випадках:

- створення анотацій до наявних ресурсів;
- посилка повідомлення на дошку оголошень, у групу новин, список поштового розсилання й т.п.;
- передача блоку даних, наприклад, результату заповнення форми, програмі обробки даних;
- додавання записів у базу даних.

Суть даного методу полягає в тому, що тіло запиту передається серверу, і той інтерпретує його залежно від URI запиту (звичайно це адреса програми, що повинна обробити дані, що втримуються в тілі запиту). Наприклад, якщо URI запиту містить ім'я CGI-сценарію, то цей сценарій буде запущений на сервері і як параметр йому буде передане тіло запиту POST. [2]

Запит PUT

Метод PUT використовується для створення нового ресурсу із заданим у запиті URI. Якщо ресурс із таким URI уже існує, то тіло запиту розглядається як його оновлена версія.

Запит DELETE

Метод DELETE використовується для видалення ресурсу, що має заданий URI.

Запит TRACE

Метод TRACE використовується з метою налагодження й діагностики. Він складається в посилці повідомлення по заданому ланцюжку й у прийомі зворотнього повідомлення.

Запит CONNECT

Ім'я цього методу зарезервовано для прокси-серверів, які можуть динамічно перемикатися в тунельний режим. [6]

Формат відгуку

Після одержання запиту і його інтерпретації сервер відправляє клієнтові відгук наступного формату:

рядок_стану

заголовки

CR LF

тіло_повідомлення

де рядок_стану має вигляд

версія_HTTP " " код_стану " " опис CR LF

Тут версія HTTP має той же формат, що й у запиті, коди стану описані нижче, а опис - це рядок, що містить короткий текстовий опис коду стану.

Типовий приклад відгуку, що повертає запитаний HTML-документ:

HTTP/1.1 200 OK

Date: Wednesday, 02-Feb-00 23:04:12 GMT

Server: NCSA/1.3
MIME-version: 1.0
Last-modified: Monday, 15-Nov-99 23:33:16 GMT
Content-type: text/html
Content-length: 2345
порожній рядок
<HTML><HEAD>...</HTML>

Коди стану

Код стану - це ціле число із трьох цифр, що вказує на результат виконання запиту. Перша цифра цього числа визначає клас відгуку:

1. Інформаційні повідомлення (запит отриманий, іде його обробка).
2. Успіх (запит був отриманий, зрозумілий і зрозумілий сервером).
3. Перенапрямок (для завершення запиту необхідні подальші дії оглядача).
4. Помилка клієнта (запит невірно складений або не може бути виконаний).
5. Помилка сервера (сервер не зміг виконати, очевидно, вірний запит).

Дві цифри, що залишилися визначають його конкретне значення в рамках даного класу. [5-6]

1.3 Електронна комерція

Електронна комерція - така форма поставки продукції, при якій вибір і замовлення товарів здійснюється через комп'ютерні мережі, а розрахунки між покупцем і постачальником здійснюються з використанням електронних документів й/або засобів платежу. При цьому, в якості покупців товарів (або послуг) можуть виступати як приватні особи, так й організації.

Глобальна мережа Internet зробила електронну комерцію доступною для фірм будь-якого масштабу. Якщо раніше організація електронного обміну даними вимагала помітних вкладень у комунікаційну інфраструктуру й була по плечу лише великим компаніям, то використання Internet дозволяє сьогодні вступити в ряди "електронних торгівців" і невеликим фірмам. Електронна

вітрина в World Wide Web дає будь-якій компанії можливість залучати клієнтів із усього світу. Подібний on-line бізнес формує новий канал для збуту - "віртуальний", що майже не вимагає матеріальних вкладень. Якщо інформація, послуги або продукція (наприклад, програмне забезпечення) можуть бути поставлені через Web, то весь процес продажу (включаючи оплату) може відбуватися в on-line режимі. Під визначення електронної комерції підпадають не тільки системи, орієнтовані на Internet, але також й "електронні магазини", що використовують інші комунікаційні середовища - BBS, VAN і т.д. У той же час процедури продажів, ініційованих інформацією з WWW, але, що використовують для обміну даними факс, телефон й ін., можуть бути лише частково віднесені до класу електронної комерції. Відзначимо також, що, незважаючи на те, що WWW є технологічною базою електронної комерції, у ряді систем використовуються й інші комунікаційні можливості. Так, запити до продавця для уточнення параметрів товару або для оформлення замовлення можуть бути послані й через електронну пошту. На сьогоднішній день домінуючим платіжним засобом при on-line покупках є кредитні картки. Однак на сцену виходять і нові платіжні інструменти: смарт-картки, цифрові гроші (digital cash), мікроплатежі й електронні чеки. [19-21]

1.3.1 Поняття торгівлі в Інтернеті

Торгівля в Інтернеті як поняття має багато авторських та кілька офіційних визначень. Якщо узагальнити офіційні визначення (європейської комісії, організації економічної кооперації та розвитку, світової організації торгівлі СОТ), то сутність електронної торгівлі полягає в організації процесу товарно-грошового обміну у формі купівлі-продажу на базі електронних технологій. Більш широко трактується це поняття Комісією ООН з міжнародного торгового права (UNCITRAL): електронна торгівля – це організація і технологія купівлі-продажу товарів, послуг електронним способом з використанням

телекомунікаційних мереж та електронних фінансово-економічних інструментів.

Отже, обидва визначення містять купівлю-продаж, що здійснюється електронними засобами, причому у віртуальному режимі.

Електронна торгівля, на відміну від електронної комерції, охоплює не весь процес товарно-грошового обміну, а тільки ту його частину, яка безпосередньо пов'язана з купівлею-продажем. Таке твердження належить до електронної роздрібно торгівлі. Що стосується електронної оптової торгівлі, то діапазон функцій тут значно ширше. Але в цілому електронна комерція поняття складніше, ніж електронна торгівля. Остання є частиною електронної комерції.

Будучи динамічною моделлю ведення бізнесу, електронна роздрібно торгівля розвивається не тільки швидкими темпами, але й широкомасштабно охоплює дедалі більше нових операцій та процесів: [22,24]

- установлення контакту (не фізичного) між «покупцем» і «продавцем»;
- обмін комерційною інформацією;
- повний цикл інформаційного супроводу покупця;
- пошук, демонстрація, вибір, консультація, конкурентоспроможність товару;
- продаж товарів, у тому числі електронних продуктів, надання послуг;
- взаєморозрахунки (у тому числі з використанням електронного переказу грошей, кредитних карток, електронних грошей);
- управління процесом доставки товарів безпосередньо покупцю або за вказаною ним адресою;
- післяпродажне обслуговування.

1.3.2 Особливості електронної торгівлі

Залежно від формату електронної торгівлі наведені функції можуть бути розширені або звужені. Отже, за змістом своєї діяльності електронна торгівля

суттєво відрізняється від реальної роздрібної торгівлі. Її притаманні характерні риси, особливості. Серед них можна виділити головні:

Віртуальність – брак особистого контакту між фізичними особами – суб'єктами процесу купівлі-продажу, тобто електронна роздрібна торгівля здійснюється в режимі on-line. Реально ж роздрібна торгівля здійснюється в режимі off-line, де безпосередній фізичний контакт обов'язковий.

Інтерактивність – адекватне інформаційне забезпечення покупця (споживача) його запиту у вигляді інтерфейсу, тобто німого діалогу.

Глобальність – брак часових, просторових, адміністративних, соціально-демографічних, асортиментно-товарних меж.

Динамічність – спроможність on-line торгівлі до моментальних змін і адаптації до нових умов.

Ефективність – спроможність забезпечити прибуток, інші економічні вигоди, а також соціальний ефект.

Отже, електронну торгівлю відрізняють особливості і характерні риси, яких бракує в реальних традиційних формах торгівлі. [23]

Характерним для електронної торгівлі є те, що ця форма має значні переваги навіть перед прогресивними формами реальної (off-line) торгівлі. Щоправда, їй притаманні також і вади. Переваги і вади можуть бути поділені між основними суб'єктами процесу купівлі-продажу таким способом. (табл.1.3)

Таблиця 1.3 Переваги і вади електронної роздрібної торгівлі

Основні переваги	Вади
Для торговельної компанії	Для торговельної компанії
1. Розширення торгівлі, вільний рух товару, послуги без обмежень і бар'єрів	1. Складність реалізації технічних, технологічних проектів, відсутність кадрів
2. Безперервність роботи торговельного об'єкта. Максимальна інтенсивність торговельного	2. Несумісність платіжних систем різних країн

обслуговування	
3. Зменшення витрат обігу, а також утримання інфраструктури	3. Брак правових механізмів
4. Вбудовані засоби реклами, маркетингу, інформації.	4. Можливий несанкціонований доступ до товару та грошових засобів, конфіденційної інформації
5. Прискорення розрахунків	
6. Відсутність обмежень комерційного бізнесу	
Для покупця	Для покупця
Доступність до товару: територіальна, демографічна, часова	1. Невпевненість, скутість при здійсненні дорогих покупок
2. Економія часу і грошей	2. Перебої з доставкою товару
3. Зручність і високий рівень торговельного обслуговування	3. Складність процесу повернення та обміну товару
4. Доставка товару за конкретною адресою	
5. Морально-психологічна перевага при відвіданні сайту, ознайомлення з товаром, його характеристиками	

2. ХАРАКТЕРИСТИКА ОБ'ЄКТУ ПРОЕКТУВАННЯ

2.1 Опис об'єкту проектування

Результатом проектної рішення повинна бути розроблена система, яка б на своєму прикладі, допомогла дослідити процес створення сучасного веб-сайту для підтримки роздрібних торговельних підприємств.

Для цього необхідно:

- проаналізувати основні програмні підходи по створенню веб-сайтів;
- проаналізувати сучасні технічні засоби по розміщенню веб-сайту в мережі;
- підібрати оптимальне середовище для надійного збереження даних;
- врахувати оптимізацію щодо швидкості та надійності роботи веб-сайту;
- захистити конфіденційні данні від не санкціонованого втручання, редагування чи копіювання;
- розробити архітектуру системи та структуру бази даних придатну для розширення та модифікації в майбутньому;
- створити зручний графійний інтерфейс користувача.

Для оптимізації процесу розробки веб-сайтів слід використовувати технологію фреймворк (Framework), яка за допомогою готового набору класів та методів дозволить швидко розробити необхідний функціонал веб-сайту. У випадку данного дипломного проекту веб-сайт необхідно забезпечити додатковою системою по управлінню контентом (CMS). Оскільки специфіка роботи торговельних підприємств є досить динамічною і потребує постійного оновлення асортименту продукції чи послуг неодмінно виникне потреба зміни інформаційного наповнення веб-сторінок сайту. Зазвичай власники чи працівники торговельних підприємств не мають ні навиків програмування ні мінімального набору знань для редагування HTML сторінок. Саме тому, CMS дозволить легко змінювати інформацію щодо послуг, які надає підприємство. Також адміністратор такого сайту буде мати зручну можливість оновлювати

каталоги продукції, розміщувати прайс-листи, а також різноманітну документацію для своїх потенційних чи діючих партнерів.

В свою чергу можливість до такої гнучкої зміни інформаційного наповнення сайту підіймає питання про безпеку інформації і захист від не санкціонованого доступу до неї. Для цього також існують різноманітні підходи та методи, які будуть розглянуті в межах даного дипломного проекту.

Оскільки програмна частина проекту буде мати досить складну структуру, варто звернути увагу на її швидкодію при великих навантаженнях. Тобто у випадку великої кількості відвідувачів, які інтенсивно користуються веб-сайтом (переглядають сторінки, завантажують документи з серверу, спілкуються на форумі) не коректно розроблена система може працювати не стабільно, створювати додаткове навантаження на сервер, тим самим вивевши його з ладу, або не доцільно використовувати ресурси бази даних, тим самим все більше та більше уповільнюючи її роботу з ростом об'єму збереженої інформації.

Окрім цього особливу увагу потрібно приділити питанню розробки графічного інтерфейсу користувача. Він повинен бути чітким та зрозумілим. Подача інформації повина бути здійснена в такий спосіб, щоб відвідувач не витрачаючи зайвого часу та уваги почерпнув усю необхідну та найважливішу інформацію щодо підприємства.

2.2 Актуальність проекту

В актуальності та доцільності даного проекту немає особливої потреби переконувати будь-кого, оскільки усім відомі такі переваги як:

- зручний доступ до поданої інформації про підприємство;
- актуальне інформування усіх зацікавлених про новини підприємства;
- автоматизація взаємодії між підприємством та партнерами;
- додатковий спосіб реклами;
- додатковий (якщо існує магазин) чи основний метод продажу товару;
- економія коштів на рекламі та оренди торгівельних площадок;

- збільшення числа клієнтів.

Тим не менше в сучасному Інтернеті велика кількість веб-сайтів не лише не приносить вигоду від вищеперечислених переваг, а ще й завдає збитки репутації підприємства від хибного враження про підприємство в лиці не грамотно розробленого сайту. Тобто основним питанням цього завдання є максимальне використання можливостей для досягнення поставленої цілі перед розробленим веб-сайтом.

2.3 Відвідування веб сайту та SEO оптимізація

Для кожного веб сайту в Інтернеті, важливим фактором є відвідування ресурсу цілеспрямованою аудиторією. Мається на увазі люди, які дійсно цікавляться товаром підприємства, а не ті, що зайшли помилково із посилання в спамі. Для того, щоб залучити саме таку аудиторію проводяться різноманітні рекламні заходи, а також SEO оптимізація веб сайту. Сама оптимізація спрямована на те, щоб подана на сайті інформація, була представлена за певними правилами. Дотримання цих правил забезпечує правильну індексацію сайту пошуковими роботами, що в свою чергу гарантує розміщення сайту на перших позиціях пошукових систем за тими ключовими словами, на які акцентувалася увага при оптимізації. Таким чином, чим вищі позиції займатиме сайт в пошукових системах, тим більше відвідувачів потраплятиме на нього.

3. СИСТЕМНИЙ АНАЛІЗ

3.1 Завдання проектного рішення

Розроблений сайт виконує роль інформаційного ресурсу для роботи в межах поточних об'ємів, а також розширення бізнесу. Тобто:

- 1) Потенційні покупці можуть почерпнути усю необхідну інформацію про асортимент товарів, а потенційні партнери вичерпну інформацію про саме підприємство та про умови співпраці;
- 2) Постійні клієнти окрім доступу до роздрібних цін можуть отримати прайс-листи із знижками, а діючі партнери мають змогу моніторингу продажу їхнього товару чи завантаження підготовлених для них звітів.

3.2 Дерево проблем

Для професійного вирішення поставленої задачі, слід заздалегідь проаналізувати усі деталі об'єкту проектування. Ця процедура виконується перед основною стадією проектування та реалізації алгоритму системи. Детальний аналіз проекту вимагає обов'язкового виконання послідовних кроків, представлених у дереві проблем. (рис. 3.1)

Оскільки головною метою проекту є висока відвідуваність сайту з інтенсивним завантаженням прайс-листів та іншої документації з сайту, то можливі великі навантаження на сервер. Це може призвести до припинення працездатності сервера. Для запобігання цього потрібно коректно підібрати програмне та апаратне забезпечення сервера, а також використовувати оптимізацію програмних алгоритмів функціонування сайту.

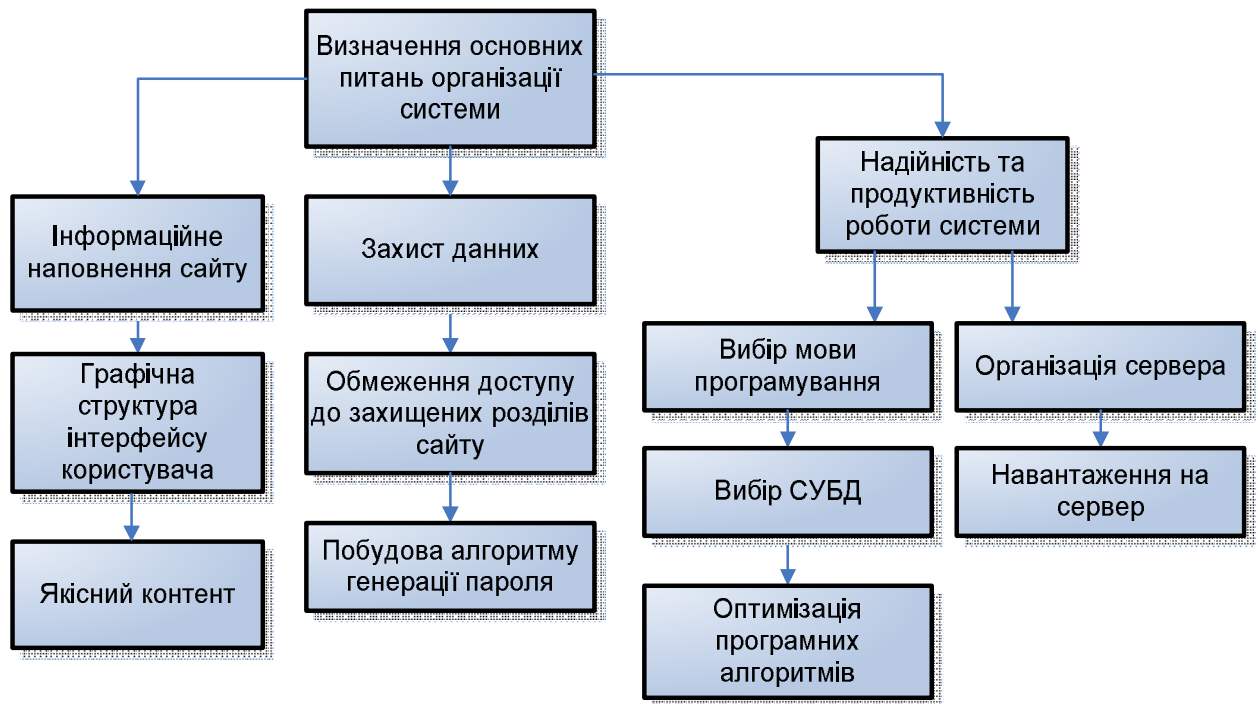


Рис. 3.1. Дерево проблем

Оскільки в базі даних буде зберігатись така конфіденційна інформація, як гуртові прайс-листи, звітна документація для партнерів, примірники угод та контрактів для потенційних партнерів, то особливу увагу потрібно звернути на захист проекту від несанкціонованого доступу до адміністраторської частини та захищеного розділу завантаження документів. Для цього організуємо захист до відповідних розділів з допомогою входу вказуючи секретний пароль, а також щоденну автоматичну зміну пароля на документацію.

Не менш важливим є питання якісного інформаційного наповнення сайту. Навігація меню повинна бути зрозумілою та логічною. Контент – чітко, стисло та доступно подавати інформацію відвідувачу. Також можуть бути розроблені різноманітні інтерактивні віджети, які б робили сайт більш динамічним і забезпечували відвідувача додатковою не менш важливою інформацією.

Вище перелічені потенційні недоліки в роботі сервісу повинні бути усунені та ретельно протестовані на завершальній стадії розробки програмного

продукту. Будь-які помилки в роботі сервісу можуть негативно відбитись на репутації та довірі до підприємства користувачами, що в свою чергу призведе до зменшення прибутків і марну витрату коштів на розробку веб-проекту.

3.3 Обмеження доступу до захищених розділів сайту

В загальному весь сайт поділяється на front-end та back-end. Front-end — це частина програмної системи, яка безпосередньо взаємодіє з користувачем. Ця частина не є захищеною і дозволяє будьякому відвідувачу переглядати публічні дані. На відміну від front-end у back-end частині знаходиться доступ до конфіденційних даних, а також засоби для їх редагування, додавання, видалення, тощо. Саме тому, back-end частина є захищеною і не доступною для пересічного відвідувача сайту. Специфіка даного сайту полягає в тому, що така доволі класична структура front-end та back-end розділів відміна від більшості сайтів тим, що в межах front-end частини у нас існує також захищений розділ для завантаження різноманітних документів. Тобто, не лише адміністратори сайту, а й інший тип користувачів (партнери, клієнти, тощо.) мають доступ до частини конфіденційних даних.

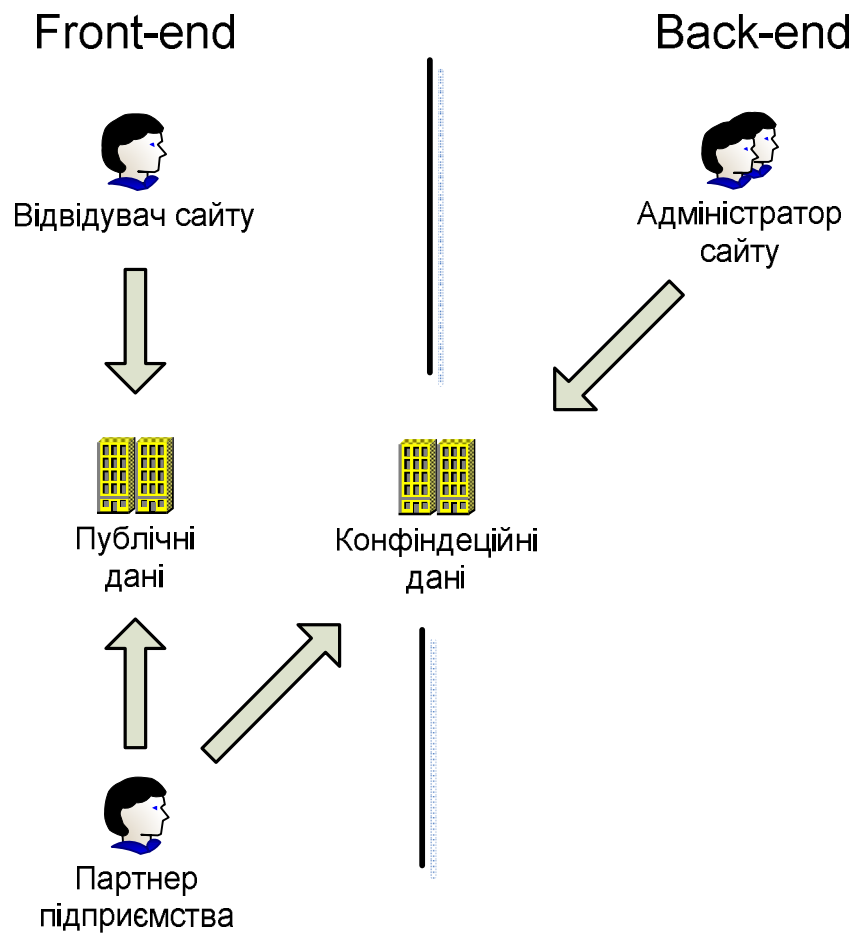


Рис. 3.2. Структура front-end та back-end розділів

Програмно, такий підхід до захисту даних реалізований наступним чином:

1. Перш ніж доступитись до захищеного розділу сайту, користувач повинен, бути авторизованим.
2. Усі дані про авторизацію користувача зберігаються в сесії і їх час життя діє до закриття браузера.
3. Коли користувач намагається доступитись до будь-якого захищеного розділу, відбувається процес авторизації:
 - Система перевіряє дані в сесії і якщо там міститься інформація про авторизованого користувача, він вважається авторизованим. В цьому випадку користувач отримує доступ до захищеного розділу.

- Якщо сесія не містить інформацію про авторизованого користувача, то він перенаправляється не в форму входу.
4. Форма входу складається з двох полів: логін та пароль.
 5. Коли користувач вводить свій логін та пароль в ці поля, відбувається процес ідентифікації:
 - Система звіряє логін та пароль введені користувачем із логіном та паролем, які містяться в системі.
 - Якщо ці данні співпадають, то користувач вважається авторизованим і в сесії зберігаються відповідні данні про успішну ідентифікацію.
 - Якщо ж ці дані не співпадають, то користувачу відображається відповідне повідомлення про не успішну ідентифікацію. В цьому випадку він повинен ще раз спробувати ввести правильні логін та пароль в формі входу.

3.4 Алгоритм генерації пароля

В даному випадку застосований досить простий але ефективний алгоритм генерації пароля.

1. Для того, щоб пароль змінювався кожного дня, він генерується на основі поточної дати.
`mktime(0,0,0,date("m"),date("d"),date("Y"))`
2. Також використовуватиметься ідентифікатор категорії для того, щоб паролі були різні для кожної окремої категорії документів.
`$val->id`
3. Отримане значення кодуємо використовуючи функцію md5()
`md5(mktime(0,0,0,date("m"),date("d"),date("Y")).$val->id)`
4. Отримане значення (32 символи) вкорочуємо для того, щоб пароль був зручним для користування. Для цього використовуємо наступну функцію:
`substr()`

Кінцевий PHP код для генерації паролів виглядає наступним чином:

```
substr(md5(mktime(0,0,0,date("m"),date("d"),date("Y")).$val->id), -10);
```

Перевагами данного алгоритму є:

- пароль автоматично змінюється кожного дня;
- не потребує втручання адміністратора;
- не потребує додаткових програмних можливостей сервера (таких як крони);

Серед недоліків:

- досить легкий алгоритм, який не використовує ключа для кодування, що збільшує ймовірність бути взломаним;

Оскільки рівень конфіденційності документів, які будуть захищені цим алгоритмом не є надто високим, ми можемо застосувати його в даному випадку.

3.5 Інформаційне наповнення сайту

Як вже йшлося попередньо, інформаційному наповненню сайту слід приділити чи не найбільшу увагу. Адже це лице підприємства, адже саме для подачі інформації власне й розробляється сайт. Отже на рисунку 3.3 можна побачити наступні інформаційні елементи:

1 – віджет, який відображає комплектуючі в довільному порядку і пропонує переглянути детальну сторінку комплектуючих. Цей блок є досить корисним, адже переглядаючи велику кількість сторінок, користувачу пропонується багато різних комплектуючих, серед яких може бути цікаві йому. Це підвищує ймовірність купівлі того, чи іншого товару у підприємства;

2 – шапка сайту, яка відображає назву сайту, та власне не чому цей сайт присвячений. Шапка виконана в традиційному вигляду і містить назву, слоган

та основну картинку (в нашому випадку фото підприємця). Дана шапка з першого погляду дає зрозуміти з яким саме сайтом відвідувач має справу;

3 – основне меню сайту пропонує перелік розділів які можна переглянути. Чіткий поділ сайту на розділи гарантує те що відвідувач швидко знайде необхідну йому інформацію;

4 – блок завантаження прайс-листа дозволяє без особливих труднощай знайти на сайті асортимент продукції та ціни на них. Розмістивши цей блок на кожній сторінці ми значно підвищуємо ймовірність того, що відвідувачі знайдуть за переглянуть прайс-лист;

5 – частина в якій відображається основний контент сторінок;

6 – контактні дані представлені на кожній сторінці забезпечують підвищення кількості звернень клієнтів по телефону чи електроній пошті;

7 – форма пошуку дозволяє здійснити пошук по усім сторінкам сайту.

Як бачимо на сторінці розміщено досить багато інформаційних елементів, кожен з яких виконує своє безпосереднє завдання, тобто подає відвідувачу якісну інформацію про підприємство і спонукає цікавитись більше.

1 Комплектуючі
Монітори
30" HP ZR30x (VM617A4)
(8840 грн,00грн.)
[більше >>>](#)

2 **Онишук Сергій Вікторович**
приватний підприємець

3 ГОЛОВНА 0.1
НАША КОМАНДА 0.2
ПРАЙСИ 0.3
КОМПЛЕКТУЮЧІ 0.4
ФОРУМ 0.5
КОНТАКТИ 0.6

4 Завантажити прайс-лист
комплектуючих

5 **Комп'ютерна компанія "CompHard"**
Наше підприємство перебуває на ринку поставок комп'ютерних комплектуючих, витратних матеріалів, комп'ютерної та офісної техніки починаючи з 2001 року. За даний проміжок часу накопичений багатий досвід спілкування з клієнтами, налагоджені тісні і довірчі відносини з провідними компаніями і їхніми представниками, до числа яких входять: Samsung, LG, Prestigio, Intel, AMD, Canon, Hewlett Packard, Epson, F & D, PowerCom, Genius, Katun, Static Control, Integral та інші.

Силами нашого підприємства виконуються комплексні роботи з постачання повного спектра офісної техніки для підприємств малого та великого бізнесу. В даний час ми є дилерами й офіційними представниками наступних всесвітньо відомих виробників:

- Intel
- Microsoft
- DLink
- Prestigio

Починаючи з 2004 року, асортимент наших послуг на ринку розширений відкриттям роздрібного магазину з продажу комп'ютерної та офісної техніки.

Гідний асортимент комплектуючих і широкий вибір витратних матеріалів доповнюється конкурентоспроможними цінами та консультаціями кваліфікованих фахівців.

6 **КОНТАКТИ**
Для того, щоб задати запитання, Вам необхідно скористатись **контактною формою**, або доступними контактами:

0682270101
0672280203
info@Sergey.com
ICQ: 673947564

7 **ПОШУК**

ПАНЕЛЬ АДМІНІСТРУВАННЯ
2012 Онишук Сергій Вікторович - приватний підприємець.

Рис. 3.3. Розміщення інформаційних елементів на сторінці.

4. ОПИС ПРОГРАМИ

4.1 Постановка задачі

За умовою теми дипломного проекту, приклад програмної реалізації повинен наглядно продемонструвати підтримку роздрібних торговельних підприємств веб-сайтом. Для контрольного прикладу було вирішено обрати розроблений мною веб-сайт, концепція якого, заключається у публікації важливої інформації про підприємство в мережі Інтернет, а також доступ потенційних покупців до прайс-листів, а партнерів до важливих документів, звітів, контрактів, тощо. На основі цього проекту ми можемо дізнатись про те, які переваги надає веб-сайт для підприємця.

Одним із важливих компонентів сайту є можливість завантажувати необхідні документи в back-end частині, тим самим пропонуючи ці документи для скачування партнерами із захищених розділів front-end частини.



Рис.4.1. Блок-схема алгоритму завантаження/скачування документів

Функціональність цього компоненту можна розділити на 6 умовних частин:

1. Створення адміністратором каталогу документів захищеного паролем

В захищеній back-end частині сайту адміністратор має можливість створювати каталоги для розміщення в них документів. Створивши каталог, система автоматично відображає пароль, який необхідний для завантаження документів з цього каталогу. Пароль автоматично змінюється кожного дня і попередньо згенерований пароль не може бути застосованим в інші дні.

2. Додавання адміністратором документу в каталог

В іншому back-end розділі сайту знаходиться список усіх завантажених документів з прив'язкою до каталогу. Тут адміністратор має можливість завантажити новий документ в один із створених каталогів. Після того, як каталог доданий в системі, адміністратор повідомляє про це партнера і передає йому пароль, який повинен бути використаний для доступу до скачування документу.

3. Перегляд каталогу документів партнером

Користувач має змогу поверхнево ознайомитись з переліком документів, їх назвами та коротким описом. Ця інформація є публічною і доступною для пересічного відвідувача веб-сайту. Кожен документ має посилання «Завантажити», яке використовується для переходу на процес завантаження.

4. Вибір документа та перехід на процес завантаження

Коли користувач знайшов необхідний йому документ і натиснув посилання «Завантажити», він переходить на сторінку з формою входу в захищений розділ. Форма містить два поля: логін та пароль, де поле логін завжди повинно мати значення «download», а поле пароль, надане адміністратором значення пароля, яке змінюється щоденно.

5. Процес авторизації партнера в захищений розділ

Коли користувач в формі входу вводить логін (download) та пароль, розпочинається процес ідентифікації користувача, під час, якого звіряються логін та пароль із значеннями, які існують в системі. Якщо ці данні співпадають, то система зберігає в сесії інформацію про вдалу ідентифікацію користувача. Якщо користувач був залогований раніше і намагається увійти в захищений розділ знову, йому не потрібно повторно вводити логін/пароль, оскільки він вже помічений як залогований.

6. Завантаження документу

Коли користувач допущений до захищеного розділу, він переходить на сторінку завантаження документа, де знаходиться коротка інформація про документ, а також, пряме посилання на файл документу. Скориставшись цим посиланням, можна скачати документ.

4.2 Модель системи

Зображена на рис. 4.2 модель системи наглядно показує, яким чином відбувається взаємодія основних складових системи від програмних та апаратних засобів до кінцевих користувачів та адміністратора сайту.

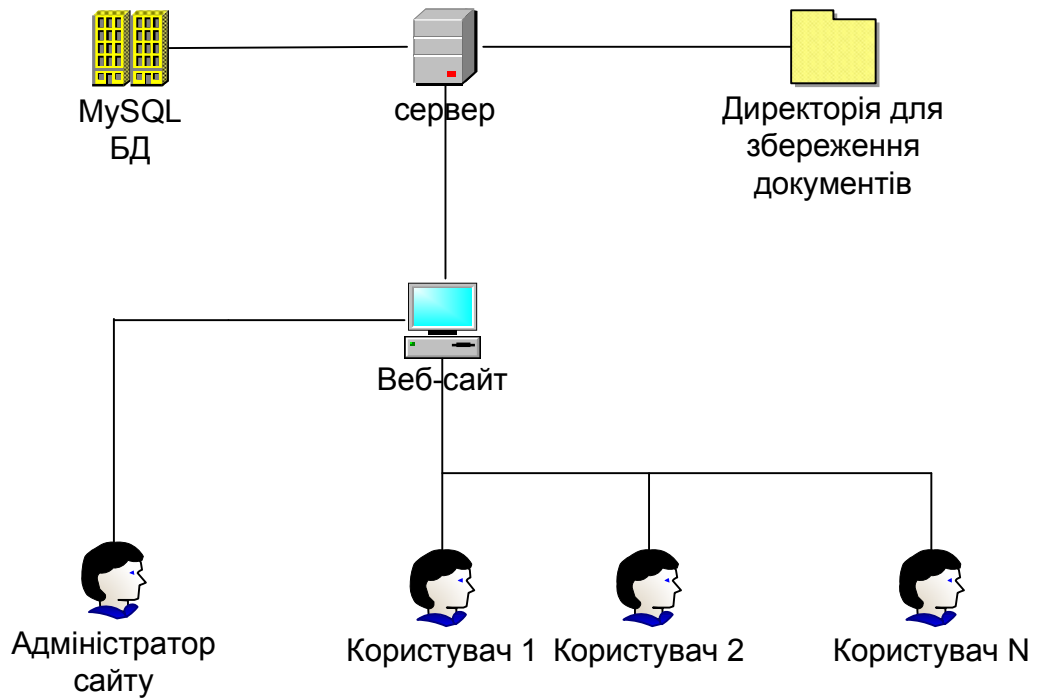


Рис.4.2. Модель системи

4.3 Структура системи

Структура системи, яка відповідає за функціонування веб-сайту в Інтернеті, тобто перегляд користувачами, та доступ до захищених розділів партнерами зображений на рис. 4.3. Для функціонування системи необхідні такі частини як база даних, базові класи, константи, модулі (перегляд контенту сторінок, завантаження документів, форум, пошук по сайту, тощо) та темплейти (забезпечують графічний інтерфейс системи).

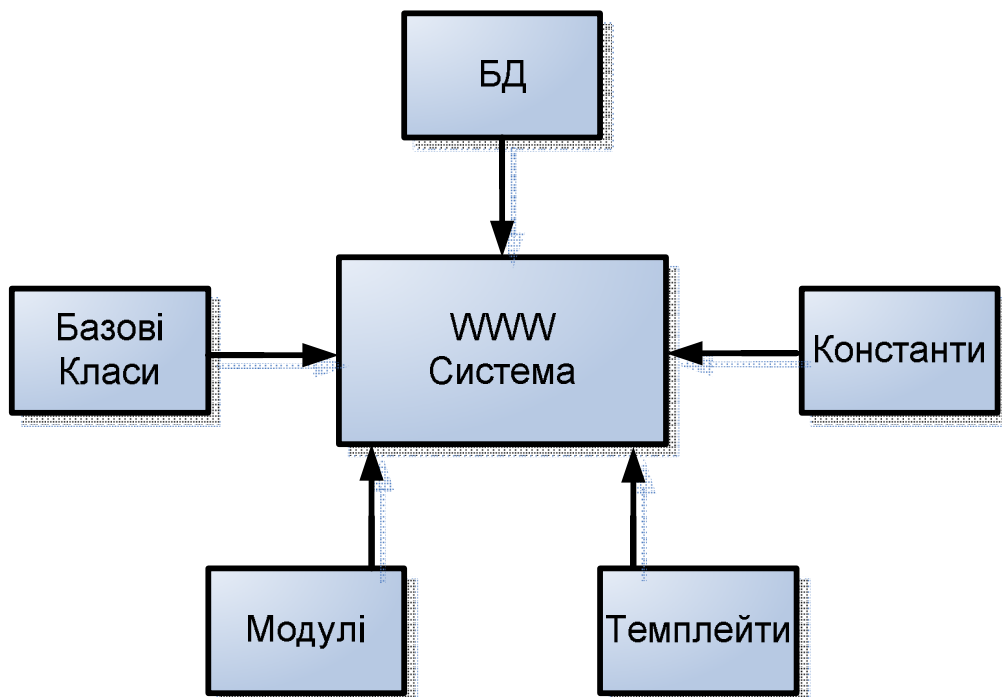


Рис.4.3. Структура системи

4.4 Опис алгоритму

Описаний алгоритм реалізовує процес завантаження/скачування документів партнерами в захищеному розділі веб-сайту.

Алгоритм складається з наступних кроків:

Створення адміністратором каталогу документів захищеного паролем

Крок 1. Відкриття адміністратором сторінки захищеного розділу в back-end частині веб-сайту.

Крок 2. Система перевіряє чи адміністратор авторизований. Тобто сесія повинна містити відповідні данні про авторизацію адміністратора. Якщо адміністратор авторизований, виконується Крок 5, якщо ні Крок 3.

Крок 3. Перехід на форму входу в захищений розділ, де адміністратор вводить логін/пароль.

Крок 4. Система звіряє введені логін/пароль з логіном/паролем, які знаходяться в системі і якщо вони співпадають виконується Крок 5, якщо ні адміністратор отримує відповідне повідомлення і процес припиняється.

Крок 5. Адміністратор переходить у розділ перегляду списку каталогів.

Крок 6. Адміністратор переходить на форму створення нового каталогу і заповнює усі обов'язкові поля.

Крок 7. Під час створення нового каталогу, система перевіряє унікальність назви каталогу. Система не може містити два каталоги з однаковою назвою. Якщо, під час перевірки було знайдено вже існуючий каталог з вказаною назвою адміністратор повертається до Кроку 6, щоб змінити назву. Якщо співпадіння не було знайдене виконується Крок 8.

Крок 8. Адміністратор переходить на список каталогів, де він може побачити новостворений каталог, а також згенерований пароль, який необхідний для скачування документів з цього каталогу.

Блок-схема алгоритму наведена на рис.4.4.

Додавання адміністратором документу в каталог

Крок 9. Адміністратор переходить до розділу сайту де знаходиться список усіх завантажених документів з прив'язкою до каталогу.

Крок 10. Адміністратор переходить на форму створення нового документу і заповнює усі обов'язкові поля додавши документ з свого локального комп'ютера чи мережі.

Крок 11. Під час створення нового документу, система перевіряє тип завантаженого файлу. З ціллю підвищення безпеки дозволяється завантажувати лише текстові документи, адже такі файли як .exe, .bat, .sh, тощо можуть бути використані з негативними наслідками для працездатності веб-сайту. Якщо файл дозволеного типу, виконується Крок 12, якщо ні, система відображає відповідне повідомлення і процес припиняється.

Крок 12. Фізично файл зберігається в спеціальній директорії на сервері, а в БД прописується шлях до цього файлу.

Крок 13. Адміністратор переходить на список документів, де він може побачити новостворений документ.

Блок-схема алгоритму наведена на рис.4.5.

Перегляд каталогу документів партнером

Крок 14. Партнер отримує від адміністратора інформацію про створення документу, а також пароль для розділу в якому він знаходиться.

Крок 15. Партнер переглядає вміст каталогів та завантажені в систему документи.

Блок-схема алгоритму наведена на рис.4.6.

Вибір документа та перехід на процес завантаження

Крок 16. Знайшовши необхідний документ, партнер переходить на його завантаження, натиснувши на посилання «Завантажити»

Крок 17. Система перевіряє чи партнер є авторизованим в захищеному розділі і якщо ні, то перенаправляє на сторінку із формою входу, якщо так, виконується Крок 21.

Крок 18. Партнер вводить пароль розділу і намагається увійти в захищений каталог для того, щоб завантажити документ.

Блок-схема алгоритму наведена на рис.4.7.

Процес авторизації партнера в захищений розділ

Крок 19. Система звіряє введені партнером логін/пароль з логіном/паролем, які знаходяться в системі і якщо вони співпадають, то виконується Крок 20, якщо ні, то виводиться відповідне повідомлення і процес зупиняється.

Крок 20. Система вписує в сесію данні про те, що партнер авторизований в захищений розділ.

Блок-схема алгоритму наведена на рис.4.8.

Завантаження документу

Крок 21. Партнеру відображається сторінка, на якій знаходиться коротка інформація про документ. Також, ця сторінка містить пряме посилання на файл документу.

Крок 22. Скориставшись цим посиланням браузер відображає вікно з пропозицією відкрити, або зберегти документ.

Крок 23. Вибравши одну з запропонованих дій, партнер отримує документ.

Блок-схема алгоритму наведена на рис.4.9.

4.5 Блок-схема алгоритму

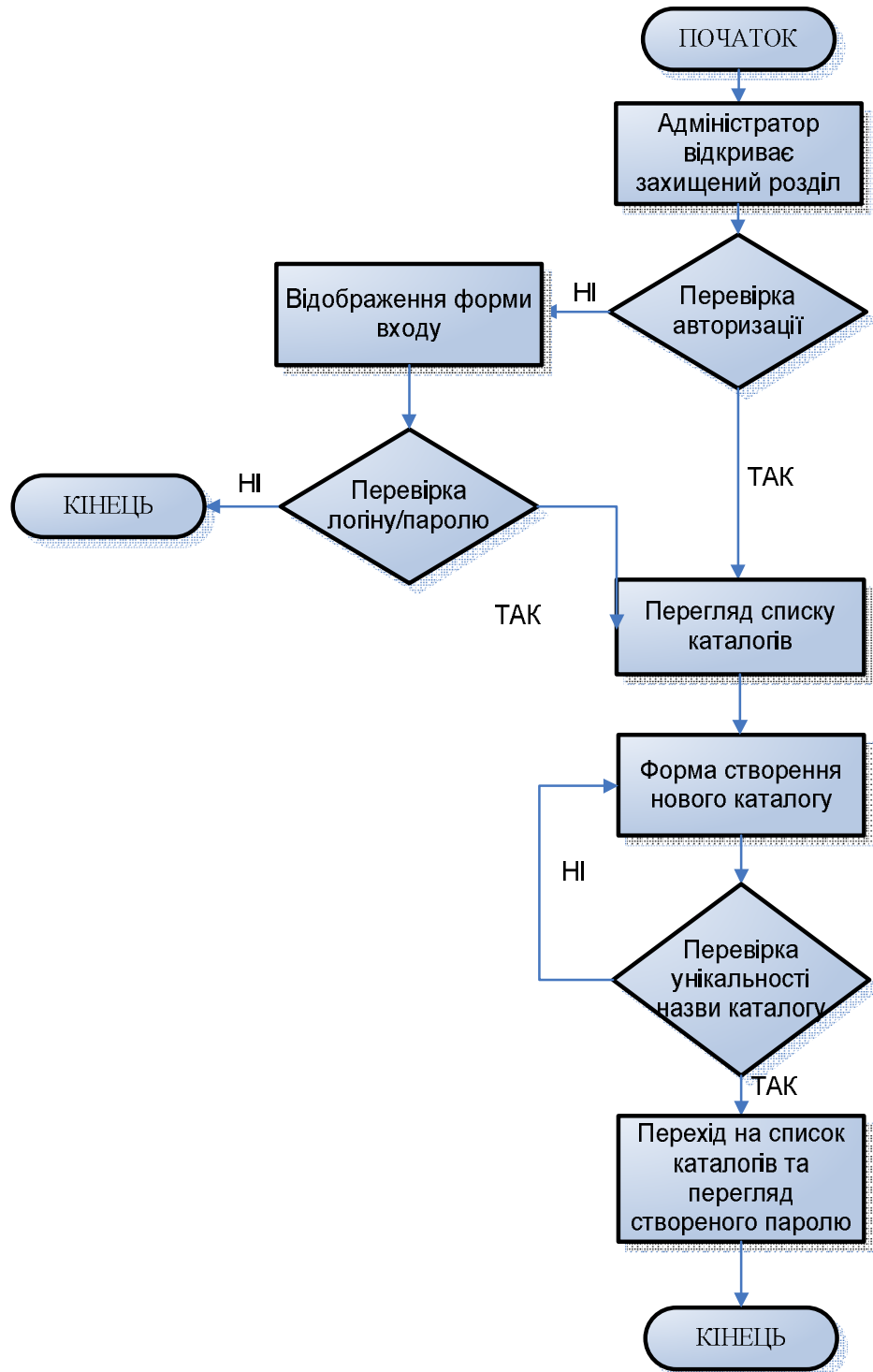


Рис.4.4. Блок-схема алгоритму створення адміністратором каталогу документів захищеного паролем

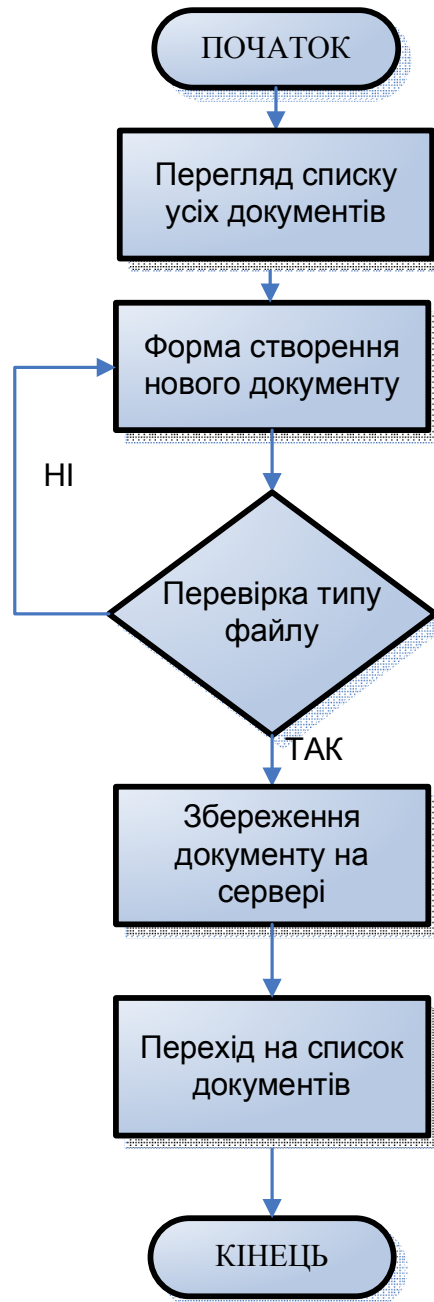


Рис.4.5. Блок-схема алгоритму додавання адміністратором документу в каталог



Рис.4.6. Блок-схема алгоритму перегляду каталогу документів партнером



Рис.4.7. Блок-схема алгоритму вибору документу та переходу на процес завантаження

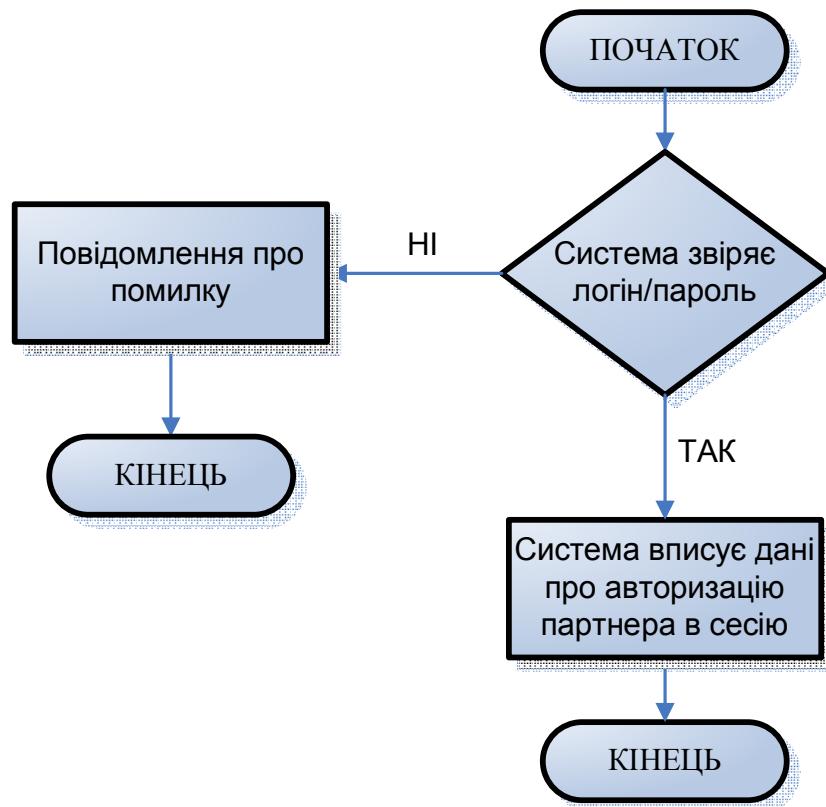


Рис.4.8. Блок-схема алгоритму процесу авторизації партнера в захищений розділ

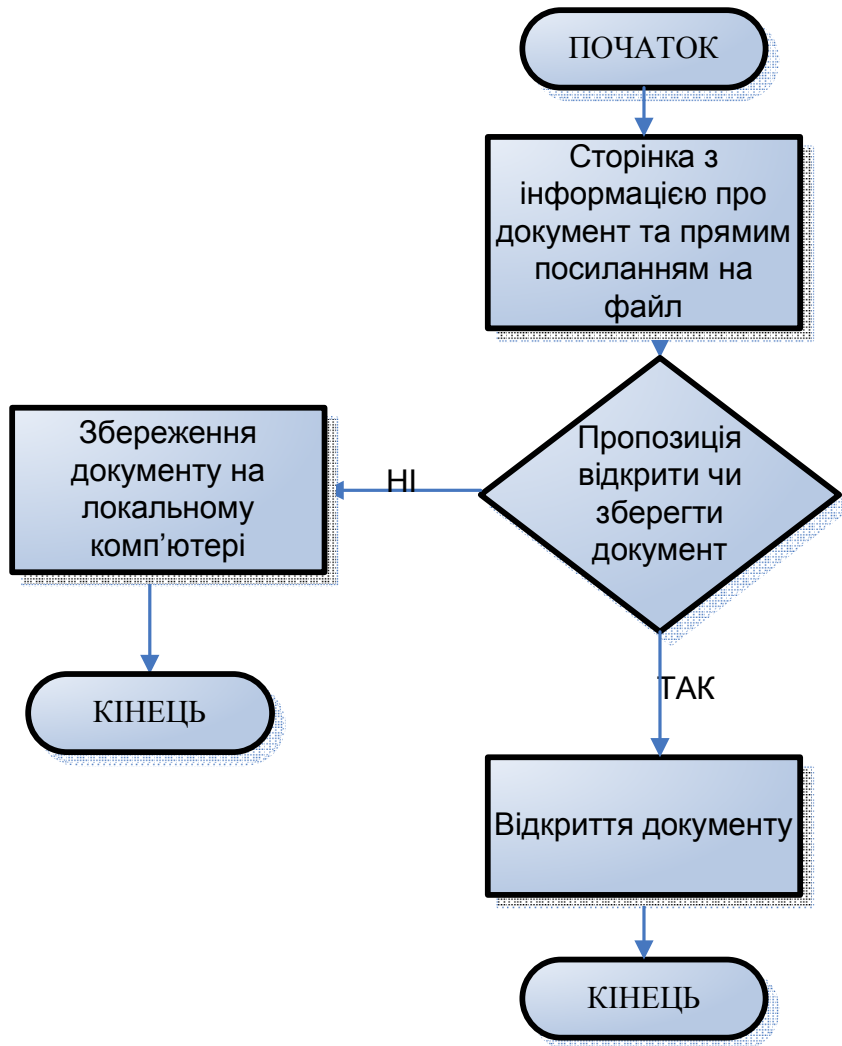


Рис.4.9. Блок-схема алгоритму завантаження документа

4.6 Опис тестової програми

Програма написана на мові PHP. В якості СУБД було обрано MySQL. Програмний продукт працює на сервері Apache.

Проект можна розділити на дві логічні частини. Скрипти, які відповідають за логіку захисту конфіденційних даних (на цьому я акцентуватиму більшу увагу) та скрипти, які забезпечують графічний інтерфейс для роботи користувачів із сайтом.

Для роботи веб-сайту використовується набір базових класів. Розглянемо кожен із них:

TBasic.class.php – базовий клас, який містить додаткові функції для роботи системи.

TDataBase.class.php – клас, який забезпечує роботу проекту з СУБД MySQL. Містить функції підключення до БД, вибірки з БД, виконання sql запитів.

TTemplates.class.php – робота з темплейтами проекту.

TUsers.class.php – робота з користувачами, авторизацію, багаторівневим доступом до адміністративної панелі керування проектом.

TInit.class.php – клас, який відповідає на ініціалізацію початкових даних та сесії.

4.7 Опис інтерфейсу програми

Розроблений інтерфейс веб-сайту націлений на зручність користування відвідувачами. На сайті витриманий строгий, чіткий та зрозумілий стиль дизайну із використанням легких для сприйняття кольорів. Вигляд сайту відіграє велику роль на привернення уваги відвідувачів. Їх перше враження про підприємство складається саме на основі вигляду сайту.

Форум, який знаходиться на веб-сайті є дуже корисним як для відвідувачів сайту, так і для самого підприємства. Для прикладу, відвідувачі

мають реальну можливість обговорити питання, які їх цікавлять безпосередньо на сайті, на якому планують купувати комплектуючі. Тобто, користувачі можуть задавати питання у подібних клієнтів, які вже мають досвід купівель в цього підприємства. Окрім цього в форумі може брати участь адміністратор сайту і якимось впливати на рішення потенційних клієнтів. З іншого боку, форум це чудовий спосіб зробити свій веб-сайт більш значущим в мережі і підвищити відвідуваність потенційними клієнтами. Адже не потрібно витратити час на створення унікального, корисного контенту. Цей контент створений відвідувачами, він буде проіндексований пошуковими системами і збільшуватиме кількість переходів з пошуковиків.

ГОЛОВНА 01
НАША КОМАНДА 02
ПРАЙСИ 03
КОМПЛЕКТУЮЧІ 04
ФОРУМ 05
КОНТАКТИ 06

Завантажити прайс-лист комплектуючих

ФОРУМ

Якість та ціна комплектуючих

Пропонується обговорити в цьому розділі питання якості комплектуючих, які пропонуються нашим підприємством. Також порівняти ціни з аналогічними підприємствами.

[Повернутись назад...](#)

Андрій 28/03/12 19:12

Мені якість дуже подобається, вже не в перше купую тут комплектуючі і ще не було нарікань на придбаній товар.

Записи 1 - 1 з 1

● *обов'язкове поле*

Додати коментар

● **Ім'я :**

● **Коментар :**

ПОШУК

Рис. 4.10. Форум

Контактна сторінка є однією із загально прийнятих сторінок більшості сайтів. Ця сторінка також чи аби як впливає на рейтинг підприємства та довіру до нього. Якщо потенційний клієнт не знайде на сторінці контактів телефону (який, до речі, робочий і на нього можна додзвонитись) чи адреси офісу, то довіра до такого підприємства буде мізерною. Адже не кожен захоче мати справу з не відомо якою конторою, яка навіть не має свого офісу чи телефону. Також на цій сторінці знаходиться контактна форма, яка дозволяє без зайвих перешкод зв'язатись з адміністрацією сайту.

ГОЛОВНА 01
НАША КОМАНДА 02
ПРАЙСИ 03
КОМПЛЕКТУЮЧІ 04
ФОРУМ 05
КОНТАКТИ 06

Завантажити прайс-лист комплектуючих

КОНТАКТИ

Для того, щоб задати запитання, Вам необхідно скористатись контактною формою, або наступними контактами:

0682270101
0672280203
info@Sergey.com
ICQ: 673947564

ПОШУК

КОНТАКТИ

Онищук Сергій Вікторович
0682270101
0672280203
info@Sergey.com
ICQ: 673947564

80600, м.Броди
вул. Золота 1/34

КОНТАКТНА ФОРМА

Контактна інформація

Ім'я ●
Прізвище ●
Телефон ●
Адреса
Індекс
E-Mail ●

Ваше повідомлення

Рис. 4.11. Контактна сторінка

На кожній сторінці сайту знаходиться блок, який пропонує користувачу завантажити роздрібний прайс-лист. Цей прайс-лист знаходиться в доступному без пароля розділі і є доступним усім відвідувачам сайту.



Рис. 4.12. Завантаження прайс-листа

4.8 Результати роботи програми

Для наглядності роботи компоненти завантаження документів в back-end частині та скачування партнерами із захищених розділів front-end частини продемонструємо роботу системи в дії.

Для початку адміністратор повинен увійти в захищений розділ back-end частини скориставшись формою входу:

Комплектуючі

Ноутбуки
Dell Inspiron N5110 Модель:
(210-357878iu) (5900
грн,00грн.)

[більше >>>](#)

Онищук Сергій Вікторович
приватний підприємець

ЛОГІН

● обов'язкове поле

Авторизація

● Логін : admin

● Пароль : ●

Вхід

2012 Онищук Сергій Вікторович - приватний підприємець.

Рис. 4.13. Форма входу в back-end частину

Після вдалого входу в захищений розділ, адміністратор створює новий каталог для документів заповнюючи поле назви каталогу.

Комплектуючі

Зовнішні HDD
Samsung STORY Station
1.5TB 3.5 USB2.0 (HX-
DU015EB/A62) (1160
грн.00грн.)

[більше >>>](#)

Матеріали

● обов'язкове поле

Додати новий розділ

● Розділ :

ПАНЕЛЬ АДМІНІСТРУВАННЯ
2012 Онищук Сергій Вікторович - приватний підприємець.

Рис. 4.14. Створення нового каталогу документів

Пізніше адміністратор додає відповідний документ у створений каталог.
Кілька документів можуть бути доданими в один каталог.

Комплектуючі

Монітори
30" HP ZR30w (VM617A4)
(8840 грн,00грн.)
[більше >>>](#)

Онишук Сергій Вікторович
приватний підприємець

ПРАЙСИ

обов'язкове поле

Додати новий документ

Розділ :

Заголовок :

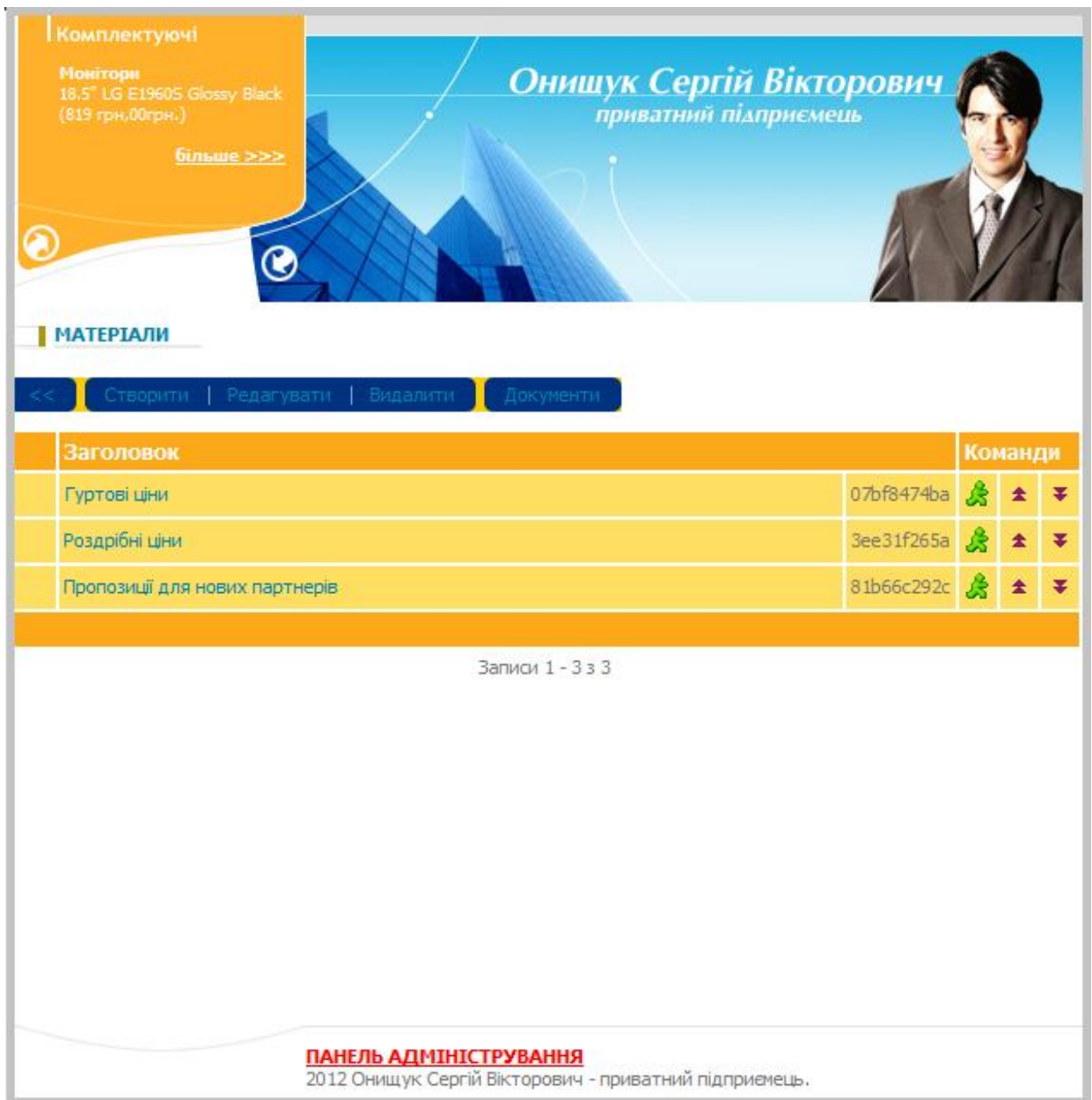
Опис :

Документ :

ПАНЕЛЬ АДМІНІСТРУВАННЯ
2012 Онишук Сергій Вікторович - приватний підприємець.

Рис. 4.15. Додавання нового документу

Після цього кроку, документ є доданим і адміністратор може передати пароль доступу своєму партнеру. Усі паролі є доступними для адміністратора в back-end частині сайту. В нашому випадку пароль: 81b66c292c



Комплектуючі

Монітори
18.5" LG E1960S Glossy Black
(819 грн,00грн.)
[більше >>>](#)

Онишук Сергій Вікторович
приватний підприємець

МАТЕРІАЛИ

<< | Створити | Редагувати | Видалити | Документи

Заголовок		Команди
Гуртові ціни	07bf8474ba	
Роздрібні ціни	3ee31f265a	
Пропозиції для нових партнерів	81b66c292c	

Записи 1 - 3 з 3

ПАНЕЛЬ АДМІНІСТРУВАННЯ
2012 Онишук Сергій Вікторович - приватний підприємець.

Рис. 4.16. Перелік паролів каталогів

Як відомо автоматична генерація паролів прив'язана до дати. Отже ми можемо наглядно перевірити динамічну зміну паролів кожного дня змінивши дату на сервері.

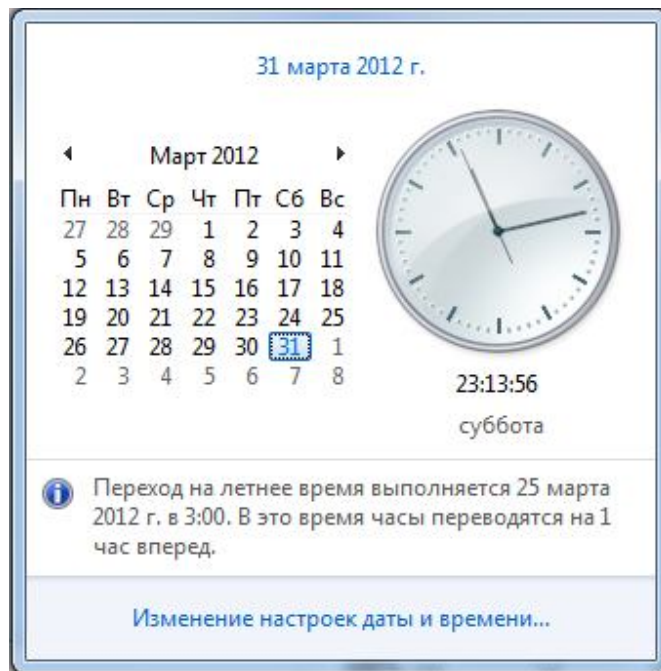


Рис. 4.17. Зміна дати на сервері

Після зміни дати, пароль нового розділу змінюється з 81b66c292c на 11e5e0b1ab

Комплектуючі

Інтернет-планшети
Apple iPad 2 Wi-Fi 16Gb
Black (5482 грн,00грн.)
[більше >>>](#)

Онищук Сергій Вікторович
приватний підприємець

МАТЕРІАЛИ

<< | Створити | Редагувати | Видалити | Документи

Заголовок	Команди
Гуртові ціни	53836b58a3
Роздрібні ціни	d984d9bb48
Пропозиції для нових партнерів	11e5e0b1ab

Записи 1 - 3 з 3

ПАНЕЛЬ АДМІНІСТРУВАННЯ
2012 Онищук Сергій Вікторович - приватний підприємець.

Рис. 4.18. Автоматична зміна паролів в системі

Тепер потенційний партнер, якому адміністратор передав пароль, може перейти на сторінку скачування цього документу. Для цього йому потрібно скористатись формою входу і ввести там отриманий пароль.

The screenshot displays a website interface. On the left, there is a navigation menu with the following items: ГОЛОВНА (01), НАША КОМАНДА (02), ПРАЙСИ (03), КОМПЛЕКТУЮЧІ (04), ФОРУМ (05), and КОНТАКТИ (06). Below the menu is a link to download a price list: 'Завантажити прайс-лист комплектуючих'. The 'ПРАЙСИ' section is currently active. The main content area features a login form titled 'Авторизація' with fields for 'Логін' (containing 'download') and 'Пароль'. A 'Вхід' button is located below the password field. A note indicates that the asterisk (*) denotes a required field. The top right of the page shows a profile picture and the name 'Онишук Сергій Вікторович, приватний підприємець'. The top left corner contains information about computer components: 'Комплектуючі: Оперативна пам'ять для ноутбуків Samsung SO-DIMM DDR3 4Gb 1333MHz (M471B5273DH0-CN900) (199 грн.00грн.)'.

Рис. 4.19. Форма входу для скачування документу

Якщо введений партнером пароль є дійсним, то він переходить на сторінку де знаходиться пряме посилання до файлу документу. Скориставшись посиланням «Завантажити», браузер пропонує партнеру відкрити чи зберегти цей документ у себе на комп'ютері.

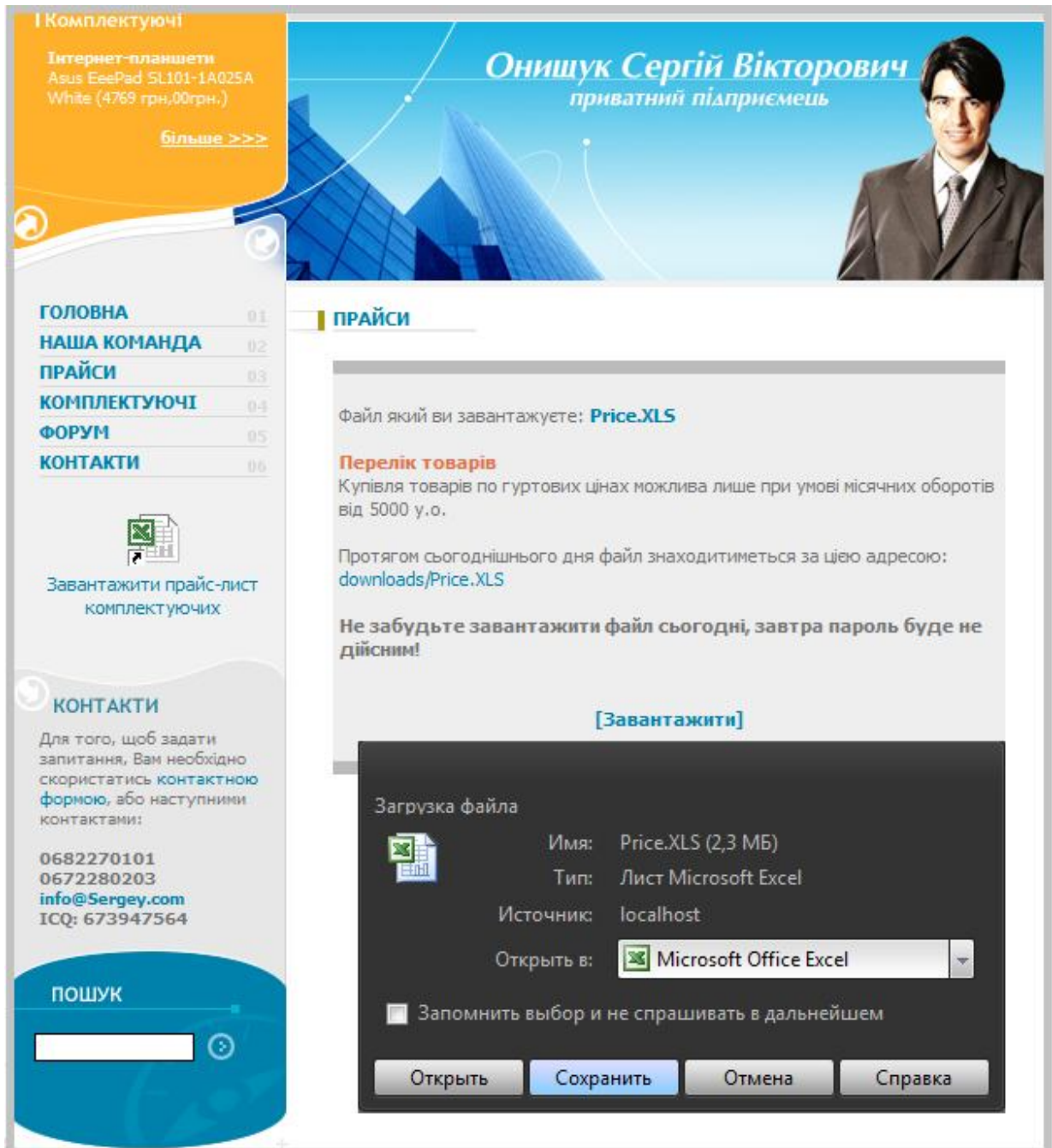


Рис. 4.20. Скачування документу

Кожна таблиця відповідає за збереження певного виду даних :

- t_config – таблиця для збереження різноманітних налаштувань системи;
- t_contacts – контактні дані підприємства;
- t_docs – інформація про завантажені документи, які можна скачати з захищеного розділу;
- t_docs_categ – перелік створених каталогів для документів;
- t_forum – таблиця, яка містить відповіді на теми форуму;
- t_forum_subjects – теми форуму;
- t_hardware – перелік комп’ютерних комплектуючих;
- t_hardware_categ – категорії для комп’ютерних комплектуючих;
- t_profile – інформаційні розділи, які відображаються на головній сторінці веб-сайту;
- t_users – користувачі адміністраторської частини системи;
- t_wife – інформаційні розділи, які відображаються в розділі «Наша Команда».

Створення відповідних зв’язків дає змогу організувати основні складові БД системи. Структура БД є простою і універсальною. Під час її проектування не було використано спеціальних можливостей СУБД MySQL. Це дасть змогу в майбутньому легко налаштувати систему для використання з іншою СУБД.

4.10 Реалізація роботи з базою даних

Сервер постійно працює з БД. Щоб організувати ефективну й надійну роботу сервера з БД був використаний клас TDataBase. Цей клас дає змогу працювати з набором функцій, які реалізують під’єднання до БД, виконання sql запитів та повернення результатів отриманих при виконанні запитів. Мета зібрання усіх функцій для роботи з БД в одному класі полягає в наступному: коли уся система взаємодіє з базою даних через один клас, то для того, щоб перевести систему з MySQL на postgresSQL необхідно лише замінити кілька

функцій класу TDataBase. В нашому випадку, для повноцінної роботи з БД використовуються такі функції:

- openConnect() – виконує з'єднання з базою даних;
- execQuery() – виконує заданий sql запит;
- getRowNums() – повертає число отриманих рядків;
- getFeachArray() – повертає результат виконання sql запиту у вигляді масиву;
- loadObjects() – повертає результат виконання sql запиту у вигляді об'єкту;
- getTableValue() – викорує вибірку значення заданого поля з заданої таблиці;
- getMultilangTableValue() – викорує вибірку значення заданого поля з заданої таблиці у випадку бігітомовності сайту;
- randVal() – вибірка значення довільним чином;
- status() – зміна статусу контент елемента;
- deleteImage() – видалення інформації про завантажену картинку з БД;
- move() – зміна порядку відображення контент елемента;
- hardware() - вибірка інформації про комплектуючі з БД.

Цього не великого набору функцій достатньо для вибору з БД довільної кількості даних та їх опрацювання на сервері.

5. ЕКОНОМІЧНА ЧАСТИНА

5.1 Економічна характеристика проектного рішення.

Метою дипломного проекту є реалізація веб-сайту, який наведе наглядний приклад оптимізації роботи роздрібного торгівельного підприємства, публікацію асортименту товарів та обмін документації з партнерами. Розроблений проект являє собою веб-сайт орієнтований на підтримку існуючих об'ємів продаж, а також залучення нових покупців та партнерів.

Кінцевим результатом роботи веб-сайту є надання необхідної інформації про підприємство та асортимент товарів, який воно пропонує покупцям.

Розроблювана система створюється для широкого кола відвідувачів – користувачів комп'ютерної техніки. Веб-сайт буде розміщеним в глобальній мережі Інтернет і надаватиме можливість пошуку необхідної інформації щодо комплектуючих та їх цін, обміну знаннями на форумі, завантаженню гуртових прайс-листів та різноманітної документації для потенційних та діючих партнерів.

Проте, оскільки дана система може бути застосована для розробки аналогічних веб-сайтів для безлічі підприємств подібної біяльності України, то розроблена система має надзвичайно широкий спектр застосування.

Проект має абсолютний ринковий попит в Уанеті, оскільки на сьогодні в Україні (а отже і на Західній Україні) є дуже багато підприємств, які рано чи пізно будуть змушені створити своє представництво в мережі Інтернет крокуючи в ногу з часом.

Економічний ефект від впровадження розробки напряму залежить як від внутрішніх бізнес процесів самого підприємства так і від грамотної подачі інформації на сторінках сайту. Від розробленого проекту очікується фінансовий прибуток за рахунок підвищення кількості клієнтів та партнерів, а отже збільшення об'ємів продаж комп'ютерної техніки.

Реклама є важливим чинником залучення користувачів на будь-який веб-ресурс, особливо, якщо цей ресурс молодий і ще не встиг себе зарекомендувати. Враховуючи, що даний веб-сайт буде в першу чергу орієнтований на коло користувачів комп'ютерної техніки, важливим є здійснення наступних заходів:

- обмін посиланнями з існуючими ресурсами комп'ютерної тематики;
- замовлення реклами у Google – програма AdWords;
- розміщення дописів та коментарів з прихованою рекламою на тематичних форумах і блогах;
- роздрук та розповсюдження поліграфічної рекламної продукції (календариків, плакатів, буклетів тощо) з рекламою веб-сайту.
- оптимізація веб ресурсу для кращої індексації пошуковими роботами.

5.2 Висновок

Як вже було зазначено, рівень технічної та ринкової цінності програмного рішення є достатньо високим. Проте, впровадження даного проекту пов'язано з ризиком, оскільки наперед неможливо оцінити рентабельність проекту. Разом з тим, велика кількість успішних стартапів, що працюють в подібному напрямку, дозволяють оптимістично оцінити перспективи впровадження цього програмного продукту.

ВИСНОВКИ

В процесі виконання магістерської дисертації розроблено програмне рішення для підтримки роздрібних торговельних підприємств. Вибір предметної сфери зумовлений актуальністю веб-технологій, сферою їх застосування та стрімким розвитком Інтернет представництв оффлайн бізнесу.

В роботі проведено аналітичний огляд літературних джерел, досліджені основні засоби реалізації даної системи, що існують на сьогодні в області веб-дизайні. Проаналізовано відомі структури та функціональні можливості таких систем, перспективи їх розвитку. Розглянута проблематика можливих порушень в роботі системи та прийняття відповідних заходів по забезпеченню вирішення цих питань.

На основі опрацьованих даних наведена модель та структура розробленої системи.

Розроблено зручний графічний інтерфейс системи, який забезпечує комфортну роботу з веб-сайтом як для пересічних відвідувачів, потенційних чи існуючих партнерів так і для адміністратора сайту.

Розроблена підсистема автоматичної генерації паролів, яка забезпечує щоденну зміну паролів, окремо для кожного захищеного розділу документації. Це забезпечує систему більшою автоматизацією, та не потребує додаткового персоналу по її обслуговуванню.

Дана система розроблялася на веб-орієнтовній мові програмування PHP та СУБД MySQL. Розроблений нами проект є завершеним продуктом і може успішно функціонувати в мережі Internet надаючи перелічені вище переваги для роздрібних торговельних підприємств.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Ньюкомер Э. Веб-сервисы. Для профессионалов: Пер. с англ. - С.-Петербург: Питер, 2003. - 256 с.
2. Хабибулин И.Ш. Самоучитель XML: Пер. с англ. - С.-Петербург: ВHV-СПб, 2003. - 336 с.
3. Бин Д. XML для проектировщиков: Пер. с англ. - М.:Символ-Плюс, 2004. - 256 с.
4. Мартин Д., Бирбек М., Лозген Б., Пиннок Д., Ливингстон С. XML для профессионалов: Пер. с англ. - С.-Петербург: Лори, 2001. - 900 с.
5. Боуэн Р., Лиска А. Apache. Настольная книга администратора: Пер. с англ. - М.:ДиаСофт ЮП, 2002. - 384 с.
6. Айлебрехт Л. Web-сервер Apache: Пер. с англ. - М.: Новое знание, 2002. – 592 с.
7. Веллинг Л. Разработка Web-приложений с помощью PHP и MySQL 3-е издание: Пер. с англ. - М.: Вильямс, 2005. - 910 с.
8. Дюбуа П. MySQL. Сборник рецептов: Пер. с англ. - М.:Символ-Плюс, 2004. - 1056 с.
9. Аргерих К., Чой В., Коггсхол Д., Эгервари К., Сколло К. Профессиональное PHP программирование. 2-е издание: Пер. с англ. - М.:Символ-Плюс, 2003. - 1048 с.
10. Гешвинде Э., Шенинг Г. Разработка WEB- приложений на PHP и PostgreSQL: Руководство разработчика и администратора: Пер. с англ. - М.:ДиаСофт ЮП, 2002. - 608 с.
11. Уорсли Д., Дрейк Д. PostgreSQL. Для профессионалов: Пер. с англ. - С.-Петербург: Питер, 2002. - 496 с.

12. Стоунз Р., Мэттью Н. PostgreSQL. Основы: Пер. с англ. - М.:Символ-Плюс, 2002. - 640 с.
13. Макконнелл С. Совершенный код: Практическое руководство по разработке программного обеспечения: Пер. с англ. - С.-Петербург: Питер, 2005. - 896 с.
14. Цвики Э., Купер С., Чапмен Б. Создание защиты в Интернете (2 издание): Пер. с англ. - М.:Символ-Плюс, 2002. - 928 с.
15. <http://www.2day.com.ua/> - Дизайн студія
16. http://www.webi.com.ua/make_ru/ - Создание сайтов
17. <http://scbali.com/> - Компанія “Спеціалізований центр БАЛІ”
18. PHP и MySQL. Создание интернет-магазина: Кристиан Дари, Эмилиан Баланеску — Санкт-Петербург, Вильямс, 2010 г.- 640 с.
19. Козье Д. Электронная коммерция: Пер. с англ. - Москва: Издательско-торговый дом "Русская редакция". 1999. - 288 с.: ил.
20. Кобелев О.А. Электронная коммерция: учеб. пособие / С.В. Пирогов (ред.). — 3-е изд., перераб. и доп. — М. : Дашков и Ко, 2008. — 683с
21. Гаврилов Л. П. Электронная коммерция. Учебное пособие по выполнению практических работ Издательство: Солон-Пресс, 2006 г. 112 с.
22. Балабанов И. П. Интерактивный бизнес / Балабанов И. П. - СПб.: Питер, 2001. - 128 с
23. Апопій В. В. Інтернет-торгівля: проблеми і перспективи розвитку / Апопій В. В. // Регіональна економіка. - 2003. - № 1. - с. 25.
24. Бруннер М. Принципы электронного бизнеса / Бруннер М. - М.: Мир электронной коммерции, 2000. - 453 с.
25. Скотт Хокинс. Администрирование веб-сервера Apache и руководство по электронной коммерции = Apache Web Server Administration and e-Commerce Handbook. — М.: Вильямс, 2001. — 336 с.

26. Томсон Л. Разработка Web-приложений на PHP и MySQL. К:Диасофт, 2001. —672 с.
27. Гарсиа-Молина Г., Ульман Дж., Уидом Дж. Системы баз данных. Полный курс. — М.: Вильямс, 2003. — 1088 с.
28. Кузнецов С. Д. Основы баз данных. — 2-е изд. — М.: Интернет-Университет Информационных Технологий; БИНОМ. Лаборатория знаний, 2007. — 484 с.
29. Гончаров А. Ю. Web-дизайн: HTML, JavaScript и CSS. Карманный справочник.. — "КУДИЦ-ПРЕСС", 2007. —320 с.
30. Якоб Нильсен Веб-дизайн. — СПб: Символ-Плюс, 2003. — 512 с .
31. Яковлев Алексей Александрович. Раскрутка и продвижение сайтов: основы, секреты, трюки.. — СПб.: БХВ-Петербург, 2007. — 336 с

ДОДАТОК

Окремі тексти програм

Програмний код файлу TUsers.class.php, який відповідає за ідентифікацію та авторизацію партнерів та адміністратора:

```

<?
class TUsers extends TBasic {

var $neededAccessLevel=2;                // default required access level

var $superAdmin_login = ADMIN_LOGIN;     // username of superadmin
var $superAdmin_password = ADMIN_PASSWORD; // password of superadmin

// constructor
    function TUsers() {
        $this->dataBase = new TDataBase();
        $this->templates = new TTemplates(); // create object to work with presentation
layer
        // initialize path to templates of competitions module
        $this->templates->template_path = $GLOBALS['dir']."modules/";
        return true;
    }

/*
 *   Function which creates login form
 *
 * INPUT:
 *
 *       none
 *
 * OUTPUT:
 *
 *       content:      html code of login form
 */

    function printLoginForm($login=NULL){

```

```

$this->templates->template_path = $this->templ_dir_path;
$tpl = $this->templates->tplRead("login_form.html"); // retrieve HTML code
from template file

// Array with keywords which is used in the template and corresponding values for
replacement
$arr = array(
    '_obligatory' => _EDIT_FORM_OBLIGATORY,
    '_form_title' => _FORM_TITLE,
    '_login' => _FORM_TITLE_LOGIN,
    '_pass' => _FORM_TITLE_PASS,
);
$tpl = $this->templates->tplReplace($tpl,$arr);

// Array with keywords which is used in the template and corresponding values for
replacement
$form = array (
    'formname' => 'form',
    'target' => "",
    'method' => 'post',
    'name' => 'form1',
    'enctype' => "",

    'cmd' => array (
        'type' => 'hidden',
        'value' => 'try_login',
    ),

    'login' => array (
        'type' => 'text',
        'value' => $login,
        'special' => ' size=20 ',
    ),
);

```

```

        'pass' => array (
                                'type' => 'password',
                                'value' => "",
                                'special' => ' size=20 ',
                                ),
        'submit' => array (
                                'type' => 'submit',
                                'name' => 'sub',
                                'value' => 'Вхід',
                                ),
    );

    // produce final HTML of form after replacements
    $tpl = $this->templates->tplPrintForm($tpl,$form);
    $content = $this->templates->tplPrint($tpl);

    return $content;
}

/*
 *   Function which tries to login admin if his login/pass/access levels is correct
 *
 * INPUT:
 *
 *     none
 *
 * OUTPUT:
 *
 *     status
 */
function tryLogin(){
    //   verify required fields
    if ( $_POST['login'] == "" AND $_POST['pass'] == "" )
        $this->addError(_PLEASE_ENTER_U_AND_P, $this->error_vars);
}

```

```

if ( $_POST['login'] == "" AND $_POST['pass'] != "" )
    $this->addError(_PLEASE_ENTER_U, $this->error_vars);
if ( $_POST['pass'] == "" AND $_POST['login'] != "" )
    $this->addError(_PLEASE_ENTER_P, $this->error_vars);
if ($this->isError())
    return false;

// search specified username and password in the database
$sql = "SELECT
        *
        FROM
        t_users
        WHERE
        login = '".$_POST['login']."' AND
        password = '".$_POST['pass']."' AND
        type <= '".$_this->neededAccessLevel.'" AND
        status = '1'
        ";
$DBres = $this->dataBase->execQuery($sql);
$DBrowNums = $this->dataBase->getRowNums($DBres);

if($DBrowNums>0){
    foreach($this->dataBase->loadObjects($DBres) as $val){
        $DBid = $val->id;
        $DBtype = $val->type;
    }

    $this->setUser(true, $DBid, $DBtype);    // login admin
}
else{
    // search entered username and password in config file
if ( ( $_POST['login'] == $this->superAdmin_login )

```

```

        && ( $_POST['pass'] == $this->superAdmin_password ) )
            $this->setUser(true, 0, 2); // login superadmin
        else
            $this->addError(_WRONG_USERNAME_OR_PASS, $this-
>error_vars); // display message about incorrect username/password
        }
        return !$this->isError();
    }

/*
 * Function which register user in session
 *
 * INPUT:
 *     val : true - login user; false - logout user;
 *     id  : id of user (if it is superadmin then id = 0)
 *
 * type: access level of user
 *
 *         1 - superadmin
 *         2 - admin
 *         3 - member
 *
 * OUTPUT:
 *     true
 *
 */

function setUser($val, $id = 0, $type = 99){
    if ($val == true ){
        $_SESSION['user_logged'] = "1";
        $_SESSION['user_id'] = $id;
        $_SESSION['user_type'] = $type;
    }else{
        $_SESSION['user_logged'] = "0";
        $_SESSION['user_id'] = "0";
    }
}

```

```
        $_SESSION['user_type'] = "99";
    }
    return true;
}

/*
 * Function which return id of logged user
 *
 * INPUT:
 *     none
 *
 * OUTPUT:
 *     id      :      id of logged user
 *
 */

function userId(){
    if($_SESSION['user_id'])
        return $_SESSION['user_id'];
    else
        return 0;
}

/*
 * Function to logout user
 *
 * INPUT:
 *     none
 *
 * OUTPUT:
 *     true
 *
 */
```

```

function tryLogout(){
    $this->setUser(false);
    return true;
}

/*
 * Function which print out a welcome message if user is logged
 *
 * INPUT:
 *     none
 *
 * OUTPUT:
 *     true
 */

function print_login_ok(){
    $adminName = $this->dataBase->getTableValue("t_users", "login",
$_SESSION['user_id']);
    if(!$adminName)$adminName = ADMIN_LOGIN;
    $this->message_vars = array($adminName);
    $this->addMessage(_SUCCESS_LOGGED, $this->message_vars); // message
about member is successfully logged
    return true;
}

/*
 * Function which return true if user is logged or false if not logged
 *
 * INPUT:
 *     none
 *
 * OUTPUT:

```



```
*      status
*
*/

function isUserLogged(){
    if($_SESSION['user_logged'])
        return true;
    return false;
}

/*
*      Function which return access level of logged user
*
*      INPUT:
*          none
*
*      OUTPUT:
*          access level
*
*/

function userAccessLevel(){
    if($_SESSION['user_logged'])
        return $_SESSION['user_type'];
    else
        return 99;
}

/*
*      Function which return access status of logged user
*
*      INPUT:
*          none
*
```

```

* OUTPUT:
*     true - access level is ok
*     false - user has no access rights for this action
*
*/

function userAccess(){

    if(!$this->isUserLogged() || $this->userAccessLevel() > $this->neededAccessLevel){
        return false;
    }else{
        return true;
    }

    return true;
}

/*
* Function which redirect user into homepage page if he has no access rights
*
* INPUT:
*     level : required access level
*
* OUTPUT:
*     none
*
*/

function accessForUser($level){
    $this->neededAccessLevel=$level;

    // if user dont have needed access, system logout him and reditedcted on
homepage
    if(!$this->userAccess()){

```

```

        $this->tryLogout();
        echo "<SCRIPT>location.href='index.php';</SCRIPT>";
        die();
    }
}
}
?>

```

Файл /languages/ua/main.php, який містить усі системні фрази і може бути використаний у випадку багатомовності веб-сайту:

```

<?
# admin/index.php
define("_DONT_WORK_HERE","Цей модуль не працює в %s");
define("_LINK_CSPANEL","Адміністративна Панель");
define("_PAGE_TITLE_2","Адміністративна Панель");

    // navigation
define("_PREV","Назад");
define("_NEXT","Вперед");
define("_RESULTS","Записи");
define("_RESULTS_DE","з");

define("_PLEASE_1_MIN","Будь-ласка виберіть хочаб один розділ!");
define("_PLEASE_1_ONLY","Можна вибрати лише один розділ!");

#     ./classes/TDataBase.class.php
define("_NO_MYSQL","База даних не знайдена");
define("_BASE_ERROR","Помилка бази даних");

#     ./classes/TTemplates.class.php

```

```
define("_PLEASE_SPECIFY_1","Будь-ласка заповніть");
define("_PLEASE_SPECIFY_2","поле");
define("_ARE_YOU_SURE_DELETE_ITEM","Ви впевнені в тому, що хочите видалити цей розділ(и)?");
define("_ARE_YOU_SURE_DELETE_FORUM","Ви впевнені в тому, що хочите видалити цей запис?");

define("_BUT_TITLE_EDIT","Редагувати");
define("_BUT_TITLE_DELETE","Видалити");
define("_BUT_TITLE_NEW","Створити");

define("_ADD_NEW","Створити");
define("_ADD_EDIT","Редагувати");
define("_ADD_DELETE","Видалити");

define("_BUT_TITLE_BACK","Назад >>>");
define("_BUT_TITLE_STATUS","Видимий:Прихований");
define("_BUT_TITLE_MOVE","Перемістити Вгору:Перемістити Вниз");

# ./classes/TUsers.class.php
define("_WRONG_USERNAME_OR_PASS","Ви ввели не вірний логін чи пароль!");
define("_PLEASE_ENTER_U_AND_P","Будь-ласка введіть коректний логін та пароль");
define("_PLEASE_ENTER_U","Будь-ласка введіть коректний логін");
define("_PLEASE_ENTER_P","Будь-ласка введіть коректний пароль");
define("_SUCCESS_LOGGED","Ви залоговані як %s");
define("_SUPER_ADMIN","Супер Адмін");
define("_FORM_TITLE","Авторизація");
define("_EDIT_FORM_OBLIGATORY","обов'язкове поле");
define("_REQUIRED","обов'язкове поле");

define("_FORM_TITLE_LOGIN","Логін");
define("_FORM_TITLE_PASS","Пароль");
```

```
#      ./inc/run_site.php
define("_NO_MODULE", "Не знайдений модуль \"\%s\");
define("_NO_LANGUAGE", "Не знайдена мова \"\%s\");

#      ./classes/TTemplates.class.php
define("_EMPTY_TEMPLATE", "Template is empty : \%s");
define("_NO_TEMPLATE", "There is no such template : \%s");
define("_NO_CODE_WITH_NAME", "There is no code with name \"\%s\");
define("_IS_NOT_AN_ARRAY", "Argument \%s in function `tplReplace` is not an array!!!");
define("_NO_LOOP_WITH_NAME", "There is no item (loop) in template with name \"\%s\");
define("_LOOP_IS_NOT_AN_ARRAY", "Argument \%s in function `tplReplaceLoop` is not an
array!!!");
define("_KEY_NOT_PROCESSED", "Template key \"\%s` is not processed!!!");
define("_LOOP_NOT_PROCESSED", "Template item (loop) \"\%s` is not processed!!!");
define("_PART_OF_CODE_NOT_PROCESSED", "Template part of code \"\%s` is not
processed!!!");
define("_FORM_ENTRY_NOT_PROCESSED", "Template form entry \"\%s` is not processed!!!");

#      page titles
define("_PAGETITLE_LOGIN", "Логін");
define("_PAGETITLE_WELCOME_ADMIN", "Панель Адміністрування Сайту");

#      lists
define("_NO_RESULTS", "Результатів не знайдено.");

#      try delete
define("_ERROR_CANT_DEL", "This topic cannot be deleted because it is used in other items.
Please delete related items first in order to delete this topic:<br>\%s");

define("_FORM_BUT_SUBMIT", "Дальше");
```

```
define("_FORM_BUT_CANCEL","Відмінити");

define("_BACKEND","backend");
define("_FRONTEND","frontend");

define("_LIST_TITLE_TITLE","Заголовок");
define("_LIST_TITLE_COMMANDS","Команди");

define("_GIF_IMAGE","GIF");
define("_JPG_IMAGE","JPG");
define("_PDF_IMAGE","PDF");
define("_DOC_IMAGE","DOC");
define("_RTF_IMAGE","RTF");
define("_MOVIE_IMAGE","MOVIE");
define("_NO_IMAGE","Немає файлу");

define("_ARE_YOU_SURE_IMG","Ви впевнені в тому, що хочите видалити цю картинку?");
define("_ARE_YOU_SURE_DOC","Ви впевнені в тому, що хочите видалити цей документ?");

define("_IMG_UPLOAD_FILED_MAX_LENGTH","Uploading failed. You have tried to upload image with width larger then %s pixels. Please scale your image and try again.");
define("_IMG_UPLOAD_FILED_MAX_LENGTH2","Uploading failed. Size of image must be 136x91 pixels. Please scale your image and try again.");
define("_IMG_UPLOAD_FILED_FORMATS","Uploading failed. You have tried to upload image with wrong file format. Please use images of only gif and jpg formats.");
define("_DOC_UPLOAD_FILED_FORMATS","Uploading failed. You have tried to upload document with wrong file format. Please use documents of only pdf, doc and rtf formats.");
define("_MOVIE_UPLOAD_FILED_FORMATS","Uploading failed. You have tried to upload movie with wrong file format.");
define("_SWF_UPLOAD_FILED_FORMATS","Uploading failed. You have tried to upload Flash with wrong file format.");
define("_IMG_UPLOAD_FILED","Uploading failed. Please try again.");
```

```

define("_ADD_DELETE_IMG","Видалити картинку");
define("_ADD_DELETE_DOCUMENT","Видалити документ");

define("_FORM_SENDER_IP","User IP");
define("_YOUR_CONTACT_INFO","Контактна інформація");
define("_YOUR_FIRSTNAME","Ім'я");
define("_YOUR_LASTNAME","Прізвище");
define("_YOUR_PHONE","Телефон");
define("_YOUR_EMAIL","E-Mail");
define("_YOUR_ADDRESS","Адреса");
define("_YOUR_ADDRESS1","Адреса");
define("_YOUR_ADDRESS2","Індекс");
define("_YOUR_PROBLEM","Ваше повідомлення");
define("_FORM_BUT_RESET","Очистити");
define("_THANK_YOU_FOR_EMAIL","Дякуєма за ваше повідомлення.");

?>

```

Деякі функції класу TDataBase:

```

// open connect with DB
function openConnect(){
    $dbh = mysql_connect($this->DBserver,$this->DBuser,$this->DBpass) or die
("mysql not found");
    mysql_select_db($this->DBdbname,$dbh) or die ("base error");

    return true;
}

/*
* exec sql query
*

```

```
*      sql : sql query
*
*/

function execQuery($sql){
if($res = mysql_query($sql))
    return $res;
else
    return false;
}

/*
* return Array of needed row
*
*      DBres : resource
*
*/

function getFeachArray($DBres){
    $row_array = mysql_fetch_array($DBres);

    return $row_array;
}

/*
* return object after exec sql query
*
*      DBres : resource
*
*/

function loadObjects($DBres){
    $array = array();
    while ($row = mysql_fetch_object($DBres)){
        $array[] = $row;
    }
}
```



```

    }
    return $array;
}

```

Конфігураційний файл системи config.php:

```

<?
// admin constants
define("ADMIN_LOGIN","admin");           // username of administrator to login
define("ADMIN_PASSWORD","1");           // password of administrator to login

define("SITE_DOMAIN","");

// admin e-mail
define("ADMIN_EMAIL","info@zuki.com");    // email of website admin
define("MAIL_FROM","Онищук Сергій Вікторович"); // defines FROM field of email

// maximum items in list per page
define("MAX_ITEMS_IN_LIST_PER_PAGE", 10); // number of items in backend lists
define("MAX_ITEMS_IN_FORUM_SUBJECT", 20);
define("MAX_ITEMS_IN_FORUM", 30);

// default parameters of the site
define("DEFAULT_LANG","ua");              // Default language of the website
define("DEFAULT_MODULE","profile");       // Default module of the website
define("DEFAULT_TEMPLATE","templ-1.html"); // Default module of
the website
define("USE_MOD_REWRITE","0");            // 1 - use search engine friendly
URLs; 0 - do not use;
define("RESIZE_IMAGES_WIDTH_FOR_PAGES", "135"); // maximum size of images which
upload into site
define("PRICE_ON", "грн.");

```

```
$DOC_FILE_TYPES = array(".pdf",".doc",".rtf",".zip",".rar",".xls");

$SITE_LANGS = array("ua");    // array of languages on site

//database constants on the local machine
define("DB_SERVER","localhost");
define("DB_DATABASE_NAME","dyplom");
define("DB_USER","root");
define("DB_PASSWORD","");
?>
```

Рецензія

на дипломну роботу студента Онищука Сергія Вікторовича
 групи 01КІН-М спеціальності «Інформатика»
 на тему Розробка веб-сайту для підтримки роздрібних торгівельних підприємств (на прикладі приватного підприємця С.Онищука)

Стисла характеристика розділів роботи В роботі розглядаються теоретичні поняття, програмні засоби, характеристика та опис об'єкту проектування, системний аналіз, опис програми, економічна частина. У висновках представлений підсумок роботи.

Пропозиції, внесені студентом, рівень їх наукового обґрунтування та ефективність на основі проведеного дослідження створено веб-сайт для підтримки роздрібних торгівельних підприємств.

Практичне значення роботи Робота має важливе практичне значення, вибір предметної сфери зумовлений актуальністю веб-технологій, сферою їх застосування та стрімким розвитком Інтернет представництв оффлайн бізнесу.

Якість оформлення роботи робота оформлена у відповідності до вимог, які ставляться до робіт такого виду.

Недоліки в роботі У роботі присутні граматичні помилки (хоч і в незначній кількості).

Загальний висновок студент виявив достатній рівень знань та практичних навичок при проектуванні та створенні веб-сайту.

Оцінка дипломної роботи Відмінно (90 балів –А за шкалою ECST).

Рецензент Панченко Ігор Михайлович,
 кандидат фізико-математичних наук, доцент,
 завідувач кафедрою математичних дисциплін
 та інформаційних систем Рівненської філії
 Європейського університету

“_3_” квітня__ 2012р.

(Підпис рецензента)

Місце печатки

Підпис засвідчую
 Начальниквідділукадрів

(Підпис)

Відгук

наукового керівника *к. т. н., доцента Літнарівича Руслана Миколайовича* на дисертацію магістра з інформатики Онищука Сергія Вікторовича групи 01КІН-М спеціальності «Інформатика» на тему Розробка веб-сайту для підтримки роздрібних торговельних підприємств (на прикладі приватного підприємця С.Онищука)

Актуальність теми На сьогоднішній день практично кожна організація має власний веб-сайт. В умовах використання сучасних інформаційних технологій – це необхідний чинник існування.

Мета дослідження Метою даної роботи є грамотне проектування і розробка веб-сайту.

Об'єкт дослідження Об'єктом дослідження є розробка програмного рішення для підтримки роздрібних торговельних підприємств.

Коротка характеристика роботи В роботі розглядаються теоретичні поняття, програмні засоби, характеристика та опис об'єкту проектування, системний аналіз, опис програми, економічна частина. У висновках представлений підсумок роботи.

Практичне значення роботи Робота має важливе практичне значення, вибір предметної сфери зумовлений актуальністю веб-технологій, сферою їх застосування та стрімким розвитком Інтернет представництв оффлайн бізнесу.

Реалізація результатів дослідження В результаті проведених досліджень було створено веб-сайт для підтримки роздрібних торговельних підприємств.

Зауваження та недоліки У роботі присутні граматичні помилки (хоч і в незначній кількості).

Висновки та оцінка дисертація заслуговує оцінки «Відмінно» , а її автор-присвоєння академічного ступеня магістра з інформатики.

Науковий керівник *к. т. н., доцент Літнарівич Руслан Миколайович*

“ 28 ” 02 2012р.