

Галицький фаховий коледж імені В'ячеслава Чорновола
відділення комп'ютерних та видавничих технологій
циклова комісія інформатики та комп'ютерних дисциплін

ДОПУСТИТИ ДО ЗАХИСТУ

Завідувач відділенням
комп'ютерних та видавничих
технологій

Чубей О.О. /_____/

підпис

«___» _____ 2022 р.

ПОЯСНЮВАЛЬНА ЗАПИСКА

до дипломного проекту
освітньо-кваліфікаційного рівня «молодший спеціаліст»
зі спеціальності 122 «Комп'ютерні науки»
на тему: «Сайт-візитка міста Збараж»

Студент групи КН-41 Грецькіх В.О

(підпис)

Керівник проекту Кульчинська Н.З

(підпис)

Консультанти:

з техніко-економічного
обґрунтування

Меленчук Л.І.

(підпис)

нормоконтролер

Гавришків Н.Г.

(підпис)

Тернопіль – 2022

Галицький фаховий коледж імені В'ячеслава Чорновола
відділення комп'ютерних та видавничих технологій
циклова комісія інформатики та комп'ютерних дисциплін

ЗАТВЕРДЖУЮ

Завідувач відділенням
комп'ютерних та видавничих
технологій

Чубей О.О. / _____ /

підпис

«__» _____ 2022 р.

ЗАВДАННЯ

на дипломне проектування

на здобуття освітньо-кваліфікаційного рівня «молодший спеціаліст»
студенту _____

(прізвище, ім'я та по-батькові студента)

1. Тема проекту _____

затверджена наказом по коледжу від “___” _____ 2021 р., № _____

2. Термін здачі студентом завершеного проекту “___” _____ 2022 р.

3. Вихідні дані до проекту

4. Перелік питань, які повинні бути розроблені в проекті: _____

а) основна частина _____

б) техніко-економічного обґрунтування _____

5. Перелік графічного матеріалу _____

6. Консультанти проекту: _____

Розділ	Консультанти	Підпис, дата	
		Завдання видано	Завдання прийнято
з техніко- економічного обґрунтування	_____ (вчена ступень, звання, _____ П.І.Б. консультанта)		

КАЛЕНДАРНИЙ ПЛАН
дипломного проектування

№ п/п	Найменування етапу	Терміни	
		початку	завершення
1.	Вибір теми, ознайомлення з вимогами до дипломного проектування.	20.11.21 р.	06.12.21 р.
2.	Огляд типових рішень та написання відповідного розділу ПЗ	06.12.21 р.	26.01.22 р.
3.	Дослідження технологій реалізації та написання відповідного розділу ПЗ	26.01.22 р.	14.02.22 р.
4.	Розробка функціональних вимог до проекту та робота над структурою програмного продукту. Написання відповідного розділу ПЗ	14.02.22 р.	02.03.22 р.
5.	Встановлення на налаштування середовища реалізації та написання відповідного розділу ПЗ	02.03.22 р.	16.03.22 р.
6.	Проектування програмного засобу (функціоналу, інтерфейсу, бази даних продукту) та написання відповідного розділу ПЗ	16.03.22 р.	17.04.22 р.
7.	Реалізація та налаштування програмного засобу та написання відповідного розділу ПЗ	17.04.22 р.	03.05.22 р.
8.	Доопрацювання модулів	05.05.22 р.	18.05.22 р.
9.	Тестування на налагодження програмного продукту та написання відповідного розділу ПЗ	18.05.22 р.	01.06.22 р.
10.	Опрацювання економічного розділу дипломного проекту та оформлення спеціального розділу	20.05.22 р.	05.06.22 р.
11.	Робота над оформленням пояснювальної записки	05.06.22 р.	12.06.22 р.
12.	Попередній захист дипломного проекту, доопрацювання	12.06.22 р.	
13.	Підготовка до захисту дипломного проекту	15.06.22 р.	22.06.22 р.
14.	Захист дипломного проекту	22.06.22 р.	26.06.22 р.

7. Дата видачі завдання ” ____ ” _____ 2021 р.

Керівник _____ / _____

Завдання прийняв до виконання _____ / _____

Реферат

Розробка Web-сайту м. Збаража з використанням PHP. Дипломний проєкт. Грецьких Володимир Олексійович. Галицький фаховий коледж імені В'ячеслава Чорновола, відділення комп'ютерних та видавничих технологій. Спеціальність 122 «Комп'ютерні науки». Сторінок – 78, рисунків – 44.

Об'єкт дослідження – розробка Web-сайту. Метою проєкту є реалізація Web-сайту м. Збараж з використанням динамічного генерування сторінок за допомогою PHP.

Важливими фактором розвитку туристичної галузі є потреба в рекламі міста Збараж в мережі Інтернет.

Використання веб-сайту м. Збаража дасть можливість проводити рекламну політику економічного та туристичного потенціалу міста.

В проєкті використано мову HTML (Hyper Text Markup Language, мова гіпертекстової розмітки), PHP (PHP: Hypertext Preprocessor) та базу даних MySQL. Розроблено алгоритми роботи стрічки новин, гостьової книги, пошуку по сайту.

Результатом розробки є завершений веб-сайт м. Збараж, який готовий до розміщення на веб-хостингу, тобто готовий до використання.

Ключові слова, ІНТЕРНЕТ, WEB-САЙТ, HTML, PHP, MYSQL, БАЗА ДАНИХ.

Abstract

Development of the Web-site of Zbarazh using PHP. Degree project. Gretsikh Volodymyr Oleksiiiovych. Vyacheslav Chornovil Galician College, Department of Computer and Publishing Technologies. Specialty 122 "Computer Science". Pages - 78, pictures - 44.

An important factor in the development of the tourism industry is the need to advertise the city of Zbarazh on the Internet.

The use of the Zbarazh website will provide an opportunity to pursue an advertising policy of economic and tourist potential of the city.

The project uses HTML (Hyper Text Markup Language), PHP (PHP: Hypertext Preprocessor) and MySQL database. Algorithms for the operation of the news feed, guest book, site search have been developed.

The result of the development is a completed website of Zbarazh, which is ready to be hosted on the web hosting, ie ready for use.

Keywords: INTERNET, WEB SITE, HTML, PHP, MYSQL, DATABASE

ЗМІСТ

Вступ	7
1 Аналіз існуючих рішень і постановка завдань	9
1.1 Дослідження об'єкта інформації	9
1.2 Обґрунтування доцільності створення системи	12
1.3 Огляд існуючих рішень	13
1.4 Аналіз засобів реалізації	15
1.5 Постановка задачі	16
1.6 Аналіз засобів реалізації	17
2 Проектування веб-сайту міста Збараж	24
2.1 Формулювання вимог до системи	24
2.2 Проектування структури веб-сайту м. Збараж	24
2.3 Проектування бази даних	28
3 Реалізація та тестування веб-сайту.....	31
3.1 Вибір засобів реалізації	31
3.2 Встановлення веб-сервера	32
3.3 Реалізація інтерфейсу	37
3.4 Реалізація продукту	38
3.5 Тестування роботи веб-сайту м. Збараж	57
4 Техніко - економічне обґрунтування	67
4.1 Розрахунок норм часу на виконання розробки Web-сайту	67
4.2 Розрахунок витрат на проведення розробки Web-сайту	68
4.3 Розрахунок ціни розробки Web-сайту	74
4.4 Оцінка економічної ефективності розробки програмного продукту	74
Висновок	76
Перелік посилань	78

					ДП.КН 22.459.13.000 ПЗ		
Змн.	Арк.	№ докум.	Підпис	Дата			
Розроб.		Грецькіх В. О.				Літ.	Арк.
Перевір.		Кульчинська Н. З.					Аркушів
Реценз.		Посвятовська О. Б.				6	78
Н. Контр.		Гавришків Н. В.				ГФК. КВТ. КН - 41	
Зав. відділ.		Чубей О. О.					

ВСТУП

Входячи в світовий інформаційний простір, Україна як ніколи потребує швидкого впровадження сучасних комунікаційних технологій. Можна стверджувати, що зараз народжується комунікаційна культура суспільства. Вона складається з мистецтва спілкування за допомогою електронних засобів. Повсякденна реальність України потребує удосконалення засобів телекомунікацій. Зростає цікавість до оперативної передачі інформації, її накопичення, впровадження найновіших досягнень у повсякденне життя, виходу в світові мережі.

Новітні засоби зв'язку дозволили об'єднати комунікаційні системи у глобальну мережу. Завдяки цьому людина отримала можливість обмінюватися інформацією в межах всієї планети, не залежно від кордонів і відстаней. Саме Інтернет є однією з ознак переходу суспільства від індустріальної стадії розвитку до інформаційної і являє собою надзвичайно привабливий засіб комунікації, що може пояснити його розвиток [].

Таким чином, правильно побудована стратегія інформаційної політики держави може значно допомогти будь-якій країні досягти відповідного результату з найменшими втратами та у найкоротший термін.

Отже, Інтернет – мережа мереж, вмістище колосальної кількості інформації, доступної всім, незалежно від національності, віку, статі, місця проживання тощо. Поява Інтернет сильно впливає на діяльність людини. Зараз, за допомогою Інтернет можна знайти роботу, працювати та навчатися, подорожувати по найвизначніших місцях світу, не виходячи з дому.

Інтернет — обов'язкова та невід'ємна частина сучасного життя людини.

В даний час туристична галузь набуває дедалі більшого значення для розвитку економіки та соціальної сфери. Розвиток туризму є важливим

					ДП.КН 22.459.13.000 ПЗ	Арк.
						7
Змн.	Арк.	№ докум.	Підпис	Дата		

чинником підвищення якості життя в Україні, і є одним з найбільш перспективних напрямів структурної перебудови економіки. У свою чергу важливими факторами розвитку туристичної галузі є потреба в рекламній політиці популяризації економічного та туристичного потенціалу міст України саме в електронних засобах масової інформації(ЗМІ), тобто в мережі Інтернет.

Метою даного дипломного проекту є створення Web-сайту м. Збаража з використанням динамічного генерування сторінок за допомогою мови PHP, використання якого дасть можливість проводити рекламну політику економічного та туристичного потенціалу міста в глобальній мережі Інтернет.

На даному Web-сайті міста Збаража розміщено наступну інформацію: символи міста, історія міста, основні вектори стратегічного розвитку міста, інфраструктура міста, туристична карта з розміщенням об'єктів туризму, події, новини, гостьова книга.

					ДП.КН 22.459.13.000 ПЗ	Арк.
						8
Змн.	Арк.	№ докум.	Підпис	Дата		

1 АНАЛІЗ ІСНУЮЧИХ РІШЕНЬ І ПОСТАНОВКА ЗАВДАНЬ

1.1 Дослідження об'єкта інформації

Інтернет – сукупність мереж із спільним адресним простором, які працюють з протоколами обміну типу TCP/IP, які з'єднанні між собою, причому кожний комп'ютер мережі має унікальну адресу. Інтернет об'єднує комп'ютери, що розташовані в найвіддаленіших куточках планети. Дати оцінку розмірів цієї глобальної мережі неможливо – ніхто не знає напевно кількості користувачів Інтернет. Проте відомо, що їх кількість зростає на 100 % щороку, і зараз не залишилося у світі країн, які не мають доступу до мережі Інтернет.

Кілька років тому користуватися Інтернет можна було лише тим, хто працював в організації, під'єднаний до мережі: навчальному закладі, дослідницькій лабораторії, урядовій установі. Сьогодні на першому плані – індивідуальний користувач. Отримати доступ до Інтернет тепер простіше ніж будь-коли, і незабаром Інтернет матиме абсолютну популярність. З структури, що підтримувалася урядом та обслуговувала наукові дослідження, Інтернет перетворилася на потужне комерційне середовище. Хоча 50 % серверів все ще належать уряду та науковим установам, 9 з кожних 10 нових вузлів – комерційні. Не зважаючи на всі спроби обмежити доступ до Інтернет, ця мережа залишається незалежною. Мабуть, спроби встановити цензуру в Інтернет будуть продовжуватись, але безсумнівно, що введення локальних стандартів в глобальній мережі не має жодного сенсу.

Головні організації, які сприяють розвитку Інтернет, – це ISO (International Standardization Corporation), ISOC (Internet Society), IAB (Internet Architecture Board), IETF (Internet Engineering Task Force), NIC (Network Information Center), RIPE (Reseaux IP Europeans), CERT (Computer Emergency Response Team) [1] .

World Wide Web, скорочено WWW, або також називають просто "веб" є розподіленою системою управління інформацією, яка використовує

					ДП.КН 22.459.13.000 ПЗ	Арк.
						9
Змн.	Арк.	№ докум.	Підпис	Дата		

Інтернет або мережу для передачі даних.

Інформація зберігається на великій кількості WWW-серверів (веб-серверів). Веб-сервер - комп'ютер, на який встановлено серверне програмне забезпечення з обов'язковим високошвидкісним підключенням до мережі Internet та резервуванням живлення. На даний час це оптоволоконні канали, потужні безперебійні блоки живлення (UPS) та дизельгенератори. Користувачі, під'єднанні до мережі, отримують цю інформацію за допомогою програм-клієнтів (браузерів). Принцип роботи простий: браузер надсилає через мережу Internet запит серверу, що зберігає необхідну сторінку (сайт). У відповідь на цей запит веб-сервер надсилає браузеру запитуваний браузером файл або повідомляє про відмову, якщо даний файл відсутній. Як правило веб-сайт складається з багатьох веб-сторінок (документів) пов'язаних між собою, які містять графіку, відео і звук, гіпертекст та ін.

Взаємодія браузера та сервера відбувається за певними стандартизованими правилами - протоколом. Протокол в World Wide Web називається HyperText Transfer Protocol (HTTP).

У сучасному WWW, щоб уникати кіберзагроз рекомендується використовувати HTTPS. HTTPS це HTTP протокол з шифруванням (наприклад: шляху та назви ресурсу (сторінки), параметрів запити та інше).

Гіпертекст — текст, який містить інформацію про посилання на інші сторінки/файли/текст, які можуть зберігатися на будь-якому сервері мережі.

Такі посилання називають гіперпосиланнями або гіперзв'язками. Візуально ці посилання у веб-браузерах можуть виглядати як “підсвічені” будь-яким чином чи способом. За допомогою гіперпосилання, користувач може швидко переміщатися від однієї частини документа до іншої, або ж від одного документа до іншого.

Для реалізації цього, була розроблена спеціальна стандартизована мова – HyperText Markup Language (HTML). Файл, написаний на HTML, являє собою текстовий файл з тегами та відповідним розширенням в імені.

					ДП.КН 22.459.13.000 ПЗ	Арк.
						10
Змн.	Арк.	№ докум.	Підпис	Дата		

Теги – це команди, що визначають розмітку документу. Файл на мові HTML набуде вигляду веб-сторінки тільки у програмі браузері.

Програми для роботи з ресурсами WWW, називаються WEB-браузерами. Основна функція web-браузера є надсилання запиту до сервера, інтерпретації отриманих даних і відображення. WEB-браузери виконують ще багато інших допоміжних задач, наприклад, веде історію переглянутих сторінок, щоб користувач мав змогу повернутися до них, за допомогою додаткових розширень може блокувати рекламу, VPN та ще виконувати багато іншого.

Веб-сайти за метою використання можна поділити на:

- персональний;
- корпоративний;
- сайти-візитівки;
- інтернет-магазин;
- новині та пошукові портали.

До персональних веб-сайтів можна віднести веб-сайти, як правило, приватних осіб. На них розміщено відомості про особу, хобі, захоплення, та інше.

Корпоративний веб-сайт - це сайт підприємства, фірми, бізнес-структур. На таких сайтах розміщують інформацію про підприємство, види діяльності та інше.

Сайти-візитівки – невеликий веб-сайт, який містить коротку інформацію про фірму, підприємство, особу. Може бути як односторінковим так і багатосторінковим. Як правило містить перелік та інформацію про послуги, товари, контактні дані та інше. Головна мета - є надання інформації, але в порівнянні з звичайними візитками містить більше інформації (наприклад, карту розташування).

Інтернет-магазин – це веб-сайт, на якому розміщено у зручний спосіб “вітрини” перелік товарів, послуг з детальним описом, характеристикою,

					ДП.КН 22.459.13.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		11

зазначенням ціни та кількістю товарів в наявності. Функціонал таких сайтів обов'язково має містити пошук, можливість купити чи замовити даний товар, послуги та оплати, вказати інформацію про доставку та інше.

1.2 Обґрунтування доцільності створення системи

Входячи в світовий інформаційний простір, Україна як ніколи потребує швидкого впровадження сучасних комунікаційних технологій. Повсякденна реальність України потребує удосконалення засобів телекомунікацій. Зростає цікавість до оперативної передачі інформації, її накопичення, впровадження найновіших досягнень у повсякденне життя, виходу в світові мережі.

Туристична галузь набуває великого значення для розвитку економіки. Розвиток туризму є важливим чинником покращення якості життя в Україні та одним із напрямів розвитку економіки. Тому для розвитку туристичної галузі є потреба в рекламі туристичного потенціалу міст в мережі Інтернет.

Збараж має величезний туристичний потенціал - м. Збараж віднесено до 39 міст України з найбільшою кількістю історичних пам'яток. Відповідно до постанови Ради Міністрів України від 04.11.1976 року м.Збараж оголошено містом-заповідником засноване у 1211 році, розкопки свідчать про II-IV ст. н.е.

У місті нараховується 56 пам'яток, з них 33 пам'ятки архітектури місцевого значення та 23 пам'ятки що перебувають на державному обліку.

У м. Збаражі розміщений головний офіс Національного Заповідника "Замки Тернопілля" до якого входить 12 замків Тернопільської області. Окрасою якого є далеко відомий за межами України Збараський замок у якому проводяться практично всі офіційні заходи Тернопільської області. Зокрема тут постійно проводяться обласний Міжнародний інвестиційний форум та відкриття обласного туристичного сезону, відбуваються різноманітні обласні культурно-масові заходи, фестивалі.

					ДП.КН 22.459.13.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		12

Використання Web-сайта м. Збаража дасть можливість проводити рекламну політику економічного та туристичного потенціалу міста, висвітлювати події, а отже залучення інвестицій для покращення життя мешканців.

1.3 Огляд існуючих рішень

На сучасному етапі розвитку мережі Інтернет, оглядаючи існуючі рішення в популяризації туризму рідного краю, можна окремо виділити туристичні портали, наприклад, <http://www.ternotour.com.ua/> (рис. 1.1) чи сторінки в соціальних мережах <https://www.facebook.com/ternotour> (рис. 1.2).

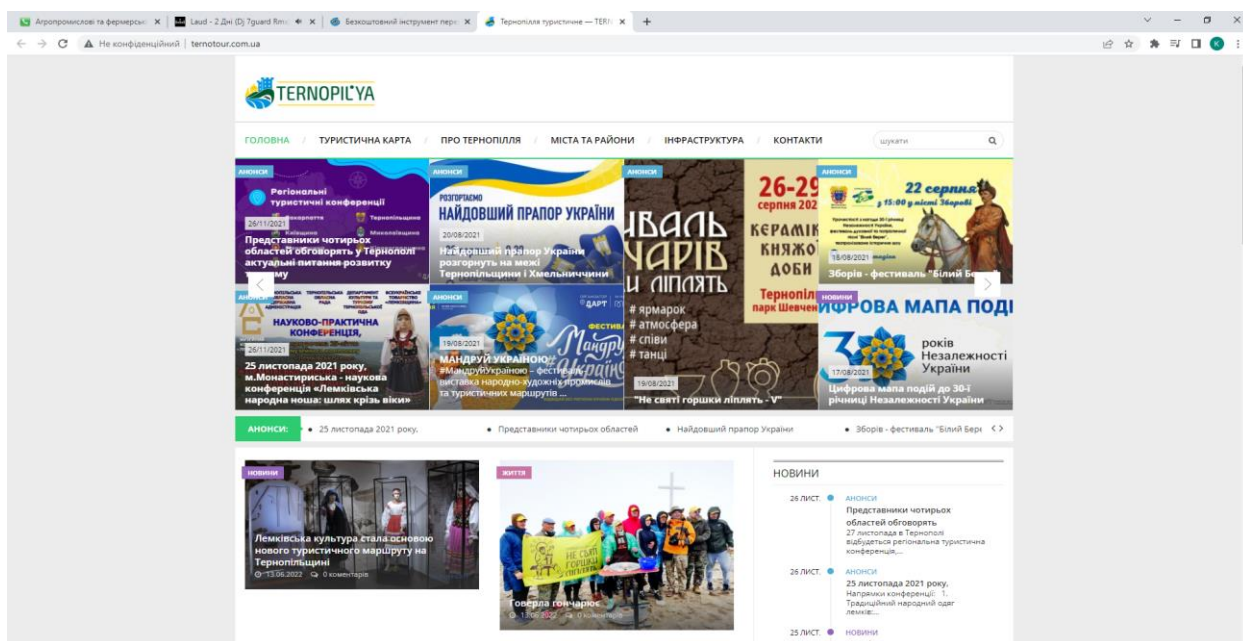


Рисунок 1.1 – Туристичний портал

Змн.	Арк.	№ докум.	Підпис	Дата

ДП.КН 22.459.13.000 ПЗ

Арк.

13

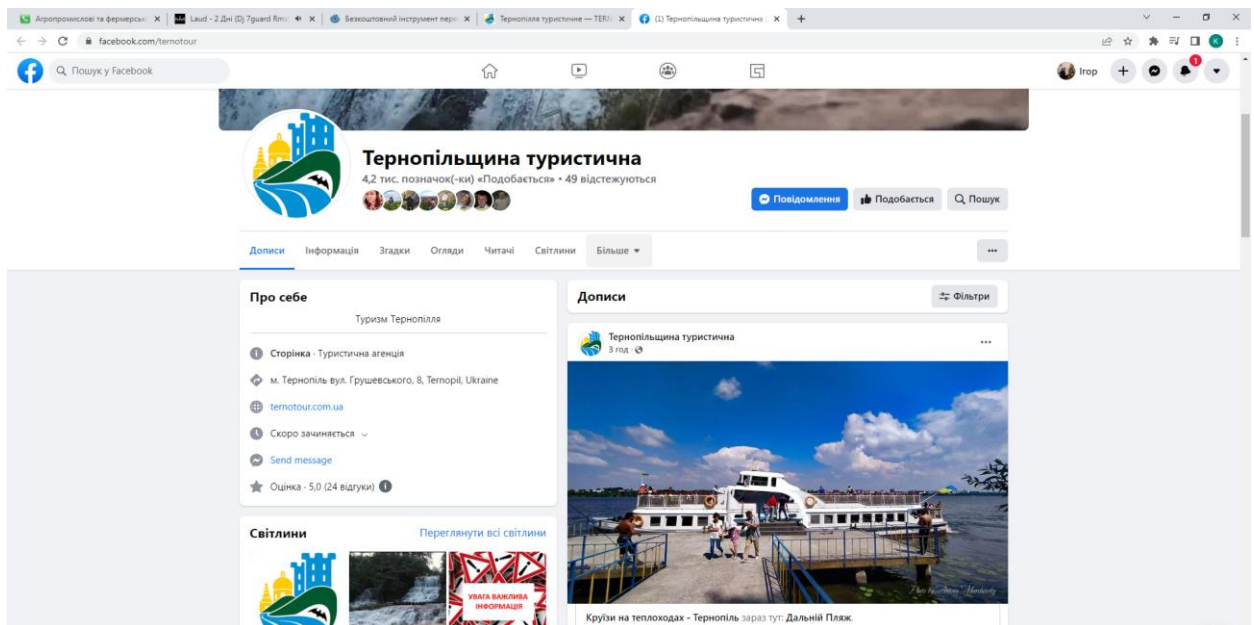


Рисунок 1.2 – Сторінка соціальної мережі facebook

Також до існуючих рішень популяризації туристичного та економічного потенціалу міст можна віднести різного роду каталоги: https://tvoemisto.tv/news/15_unikalnyh_arhitekturnyh_pamyatok_ukrainy_yaki_s_hche_mozhna_vryatuvaty_74928.html (рис. 1.3), <https://109.te.ua> та багато інших.

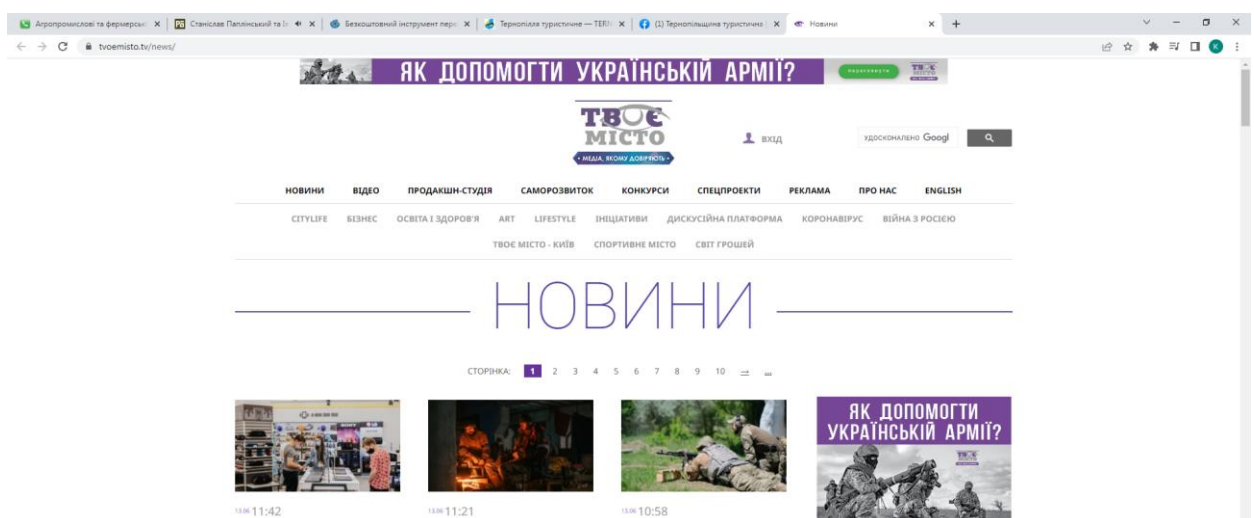


Рисунок 1.3 – Електронний каталог

Змн.	Арк.	№ докум.	Підпис	Дата

ДП.КН 22.459.13.000 ПЗ

Арк.

14

Такі ресурси як правило належать приватним туристичним компаніям чи державним органам влади.

1.4 Аналіз засобів реалізації

Якщо переглянути існуючі рішення, то можна побачити, що багато ресурсів побудовані з використанням безкоштовних систем керування вмісту CMS Wordpress, Joomla!, Drupal та інших. Систе́ма керува́ння вмі́стом (Content Management System, CMS) - програмне забезпечення для організації вебсайтів чи інших інформаційних ресурсів.

WordPress - система керування вмістом з відкритим кодом, яка через свою простоту в встановленні та використанні широко застосовується для створення вебсайтів. Сфера використання - від блогів до складних вебсайтів. Вбудована система тем і плагінів у поєднанні з вдалою архітектурою дозволяє конструювати на основі WordPress практично будь-які вебпроекти (рис. 1.4).

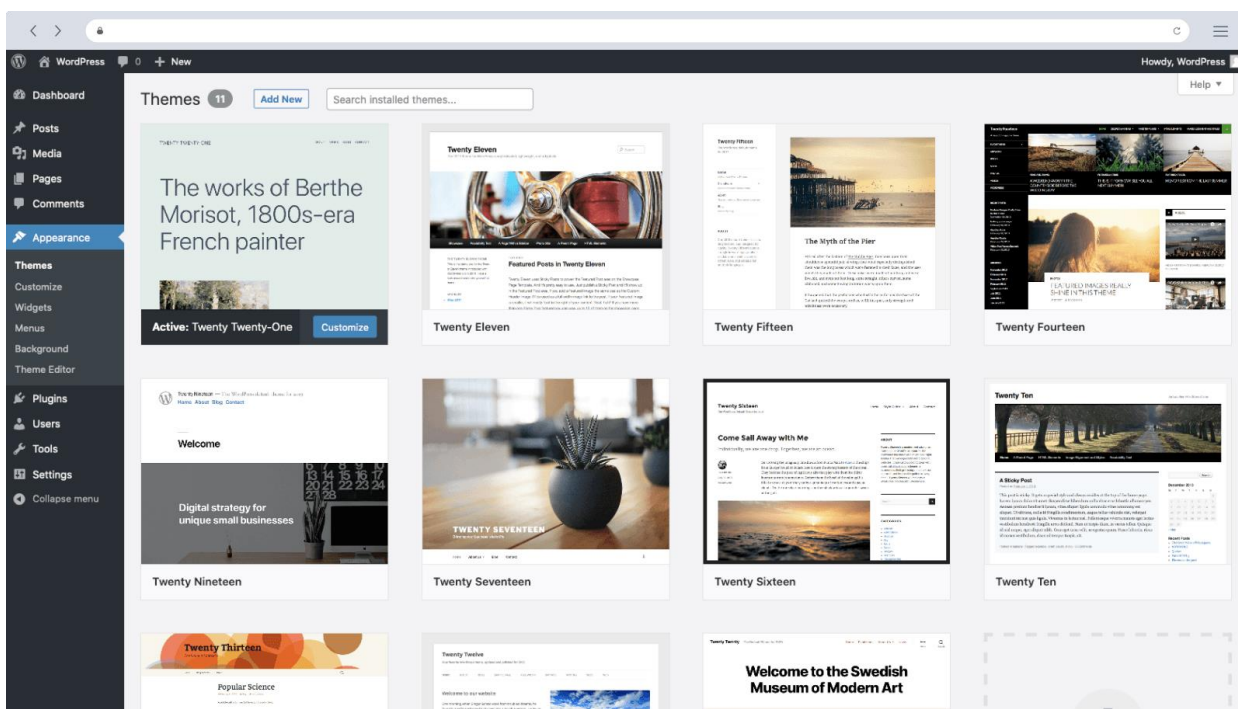


Рисунок 1.4 –WordPress

					ДП.КН 22.459.13.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		15

Joomla! – відкрита універсальна система керування вмістом для публікації. Дуже зручна для створення маленьких чи великих корпоративних сайтів, порталів, онлайн-магазинів. Joomla подібна до Wordpress, але можна відзначити гнучкі інструменти управління обліковими записами, інтерфейс для управління медіа-файлами, підтримка створення багатомовних варіантів сторінок, система управління рекламними кампаніями, адресна книга користувачів, голосування, вбудований пошук, функції категоризації посилань і обліку кліків, WYSIWYG-редактор, система шаблонів, підтримка меню, управління новинними потоками та інше (рис. 1.5)

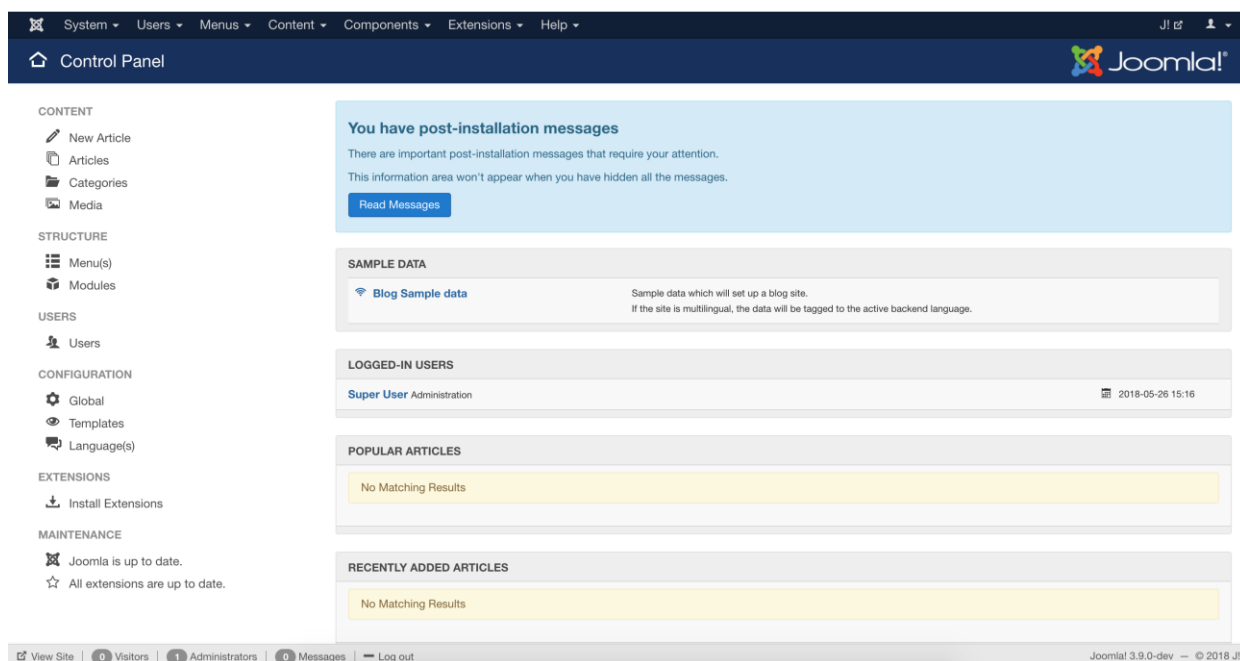


Рисунок 1.5 – Joomla!

1.5 Постановка задачі

Оскільки веб-сайті м. Збараж планується використовувати для реклами економічного та туристичного потенціалу міста та висвітлення подій, необхідно розмістити наступні розділи :

- візитка міста (символи міста, запрошення до міста);
- відгомін віків (історія);

					ДП.КН 22.459.13.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		16

- основні вектори стратегічного розвитку міста;
- інфраструктура міста (міське господарство);
- туризм (туристична карта з розміщенням об'єктів туризму);
- мистецтво, культура і спорт;
- особистості (уродженці міста, видатні люди, які жили чи живуть у місті, почесні громадяни);
- мерія (голова, заступник, секретар міської ради та ін.);
- “вікно в Збараж” (події, новини);
- гостьова книга.

Також потрібно передбачити можливість пошуку інформації по розділах.

Розділи візитка міста, відгомін віків, основні вектори стратегічного розвитку міста, інфраструктура міста, туризм, мистецтво, особистості, мерія повинні зберігати свою актуальність протягом великого часу і не передбачають систематичного оновлення. Тобто дані розділи можуть бути статичними, а розділ „вікно в Збараж” з стрічкою новин, гостьова книга, пошук по розділах реалізуємо з застосуванням PHP та бази даних MySQL.

1.6 Аналіз засобів реалізації

Кожен сайт будується з використанням HTML. HTML (Hyper Text Markup Language, мова гіпертекстової розмітки) - стандартизована мова розмітки документів для перегляду веб-сторінок у браузері. Веб-браузери отримують HTML документ від сервера, інтерпретують та відображають на екрані монітора. Веб-сторінка складається з послідовності команд, які називають тегами(tag). Під інтерпретації тегів на екран виводиться елементи веб-сторінки.

Теги HTML дозволяють, розміщувати текстові блоки, малюнки, таблиці, лінії, посилання, елементи керування та вводу інформації (кнопки, перемикачі, поля вводу тексту), задавати вирівнювання, доповнювати

					ДП.КН 22.459.13.000 ПЗ	Арк.
						17
Змн.	Арк.	№ докум.	Підпис	Дата		

динамічними об'єктами (Java Applet, Java Script, ActiveX, PHP) та багато іншого.

Широке розповсюдження HTML-документи одержали завдяки різноманітним можливостям представлення інформації, легкістю написання та сумісністю з різними операційними системами. Команди мови HTML називають тегами. Теги бувають одинарними і парними. Більшість тегів є парними, наприклад, тег означення HTML-файлу: `<html>:</html>`. Парні теги позначають початок і кінець області дії відповідної команди. Теги записують у кутових дужках, причому той тег, що закриває область дії, має косу риску (слеш/slash). Тег може містити параметри, через пробіл, наприклад, `<body bgcolor="black">`.

Сторінка має починатися тегом `<HTML>` і закінчуватися тегом `</HTML>`.

Всі скрипти HTML мають однакову структуру.

Обов'язково мають бути теги:

```
<html>
<head>
<title> Заголовок сайту </title>
</head>
<body>
...
</body>
</html>
```

Веб-сторінка має мати назву, тому варто вказати заголовок тегами:

```
<HEAD> <TITLE> Заголовок сайту </TITLE> </HEAD>
```

Особливо важливим є тег `<body>`, в якому власне і описується “тіло” web-сторінки. А саме: фон, текст, малюнки, розмір тексту, тобто зміст сторінки та її зовнішній вигляд.

При написанні тегів HTML не має значення, якими буквами -

					ДП.КН 22.459.13.000 ПЗ	Арк.
						18
Змн.	Арк.	№ докум.	Підпис	Дата		

рядковими або прописними Ви пишете команди.

Опишемо декілька тегів, які часто використовуються:

- для переходу на наступний рядок використовується тег `
`;
- для створення пустого рядка використовується тег `<p>`;
- для створення горизонтальної смуги використовується тег `<hr>`.

Якщо смуга створюється не на весь рядок, а на якусь її частину, наприклад, на 50%, то команда виглядає таким чином: `<hr width= 50 %>`, для створення широкої смуги потрібно задати її ширину (в пікселях) `<hr size= 20 >`;

– для вирівнювання смуги ліворуч/праворуч/по центру використовується відповідний параметр `align=left/right/center`.

– заголовки в документі створюються за допомогою команд: `<Hі>` Заголовок `</Hі>`, причому при $i = 1$ заголовок самий великий, а при $i = 6$ - самий дрібний;

– кольори зображення і фону задаються за допомогою команди `<body>`. Ця команда може задаватися тільки на початку HTML файла і не може бути змінена надалі. Її параметри: `bgcolor` = «# код кольору» - колір фону документа, `text` = «# код кольору» - колір тексту документа, `link` = «# код кольору» - колір тексту, що використовується як посилання, `vlink` = «# код кольору» - колір посилання на переглянутий раніше документ, `alink` = «# код кольору» - колір посилання в момент натиснення на неї правої кнопки миші;

– таблиця створюється за допомогою тегів `<table>` і `</table>`. Для формування таблиці, що складається з декількох рядків, використовують команду `<TR>`, що розділяє рядки. Команди `<th>` і `</th>` використовуються для позначення заголовка стовпця кліток. Параметри `rowspan` і `colspan` команд `<td>` і `<th>` використовуються для формування кліток даних, що об'єднують більше за один рядок або стовець;

– для запису гіпертекстового посилання в HTML-файл використовують наступну команду: `Перехід по`

					ДП.КН 22.459.13.000 ПЗ	Арк.
						19
Змн.	Арк.	№ докум.	Підпис	Дата		

гіпертекстовому посиланню [2].

Розділ „вікно в Збараж” (стрічка новини), адміністрування стрічки новин (додавання, редагування чи знищення новин), гостьова книга, модерування гостьової книги (візуальний контроль даних з можливістю внесення змін), пошук по розділах, адміністрування доступу реалізуємо з застосуванням PHP та бази даних MySQL.

PHP (PHP: Hypertext Preprocessor) – це мова, використовувана при написанні серверних програм-сценаріїв, яка вмонтовується в HTML, інтерпретується та виконується на сервері.

PHP з'явилася як розвиток іншого продукту, PHP/FI. PHP/FI був створений Rasmus'ом Lerdorf'ом у 1995 році, спочатку як простий набір Perl-скриптів для відстеження доступу до його власного online-резюме. Він назвав цей набір скриптів 'Personal Contents Page Tools'.

У 1997 році PHP/FI 2.0, друга версія C-реалізації. Оскільки безліч людей надавали блоки коду для цього проекту, вона перестала бути проектом однієї людини.

Незабаром після цього з'явилися перша alpha PHP 3.0.

PHP 3.0 була першою версією, схожої на сьогоденішню PHP. Його створили Andi Gutmans і Zeev Suraski у 1997 році як цілком нову, переписану мову, оскільки PHP/FI 2.0 виявилось недостатньо для розробки їхніх власних е-commerce-додатків. З метою кооперації зусиль, Andi, Rasmus і Zeev створили і випустили PHP 3.0 як офіційний спадкоємець PHP/FI 2.0, і розвиток PHP/FI 2.0 в основному зупинилося.

Однієї із сильних сторін PHP 3.0 була можливість розширення. Крім того, надаючи кінцевим користувачам міцну інфраструктуру для різних БД, можливості розширення PHP 3.0 спонукало десятки розробників розробляти нові модулі розширення. Можливо саме в цьому був секрет приголомшуючого успіху PHP 3.0.

Іншою ключовою особливістю PHP 3.0 стала підтримка об'єктно-

					ДП.КН 22.459.13.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		20

орієнтованого синтаксису. Мова була названа PHP: Hypertext Preprocessor.

Взимку 1998 року, незабаром після офіційної публікації PHP 3.0, Andi Gutmans і Zeev Suraski почали зміну ядра PHP з метою підвищити продуктивність при роботі складних додатків і поліпшити модульність кодової бази PHP. Такі додатки були можливі в PHP 3.0, але PHP 3.0 не розроблявся для ефективного обслуговування складних комплексних додатків.

Нова машина, 'Zend Engine' (по іменах творців – Zeev і Andi), успішно вирішила ці задачі і з'явилася спочатку в середині 1999 року. PHP 4.0, заснована на цій машині й оснащена великою кількістю нових можливостей. Окрім продуктивності в версії PHP 4.0 введено такі нові ключові можливості, як підтримка великої кількості Web-серверів, HTTP-сесій, буферизації, більш безпечні способи роботи. PHP 4 є в даний момент останньою версією PHP.

Розробникам Web-додатків немає необхідності говорити, що web-сторінки - це не тільки текст і картинки. Гідний уваги сайт повинний підтримувати деякий рівень інтерактивності з користувачем: пошук інформації, конференції і т. п. Традиційно все це реалізувалося CGI-скриптами, написаними на Perl. Але CGI-скрипти дуже погано масштабуються. Кожен новий виклик CGI, просить від ядра породження нового процесу, а це забирає процесорний час і витрачає оперативну пам'ять. PHP пропонує інший варіант – він працює як частина Web-сервера, і цим самою схожий на ASP від Microsoft.

Програму на PHP, подібно до скрипта на JavaScript, VBScript або ASP, треба вставляти в HTML-файл. Початок і кінець програми відзначаються спеціальними тегамі `<?php` і `?>`. Текст поза цих тегів PHP не сприймаються - вони передаються Web-броузерові "як є". Самі скрипти знаходяться на сервері, і їхній лістинг відвідувач сайту переглянути теоретично не може. Файли скриптів повинні мати розширення *.php або *.phtml, інакше сервер проігнорує всі php-вставки в html-документі. При

					ДП.КН 22.459.13.000 ПЗ	Арк.
						21
Змн.	Арк.	№ докум.	Підпис	Дата		

активізації скрипта серверна програма виконує всі команди `php`, не торкаючи `html`-код, і повертає результат браузеру відвідувача. Наведемо наглядний приклад:

```
<html>

<head>
<title>Заголовок сайту</title>
</head>

<body>
<?php echo "Привіт, це PHP сценарій!"; ?>
</body>
</html>
```

Оскільки синтаксис PHP дуже схожий на синтаксис C або Perl, то люди, знайомі з програмуванням, дуже швидко зможуть почати писати програми на PHP. У цій мові немає строгої типізації даних і немає необхідності в діях по виділенню/звільненню пам'яті.

Програми, написані на PHP, досить легко читаються. Написаний PHP – код легко прочитати і зрозуміти, на відміну від Perl-програм [].

Наведемо декілька функцій PHP, яку будемо застосовувати в даному проекті:

- `include ()` – функція PHP, яка замінює себе в процесі виконання файлом, який вказаний як аргумент функції;
- `echo (string arg1 [, string argn...])` – виводить один або більше рядків;
- `MYSQLi_CONNECT (string server , string username , string password, string base)` – відкриває з'єднання з MySQL сервером;
- `mysqli_query (resource link_identifier, string query)` – відсилає активні бази даних на сервері MySQL-запит/query;
- `mysqli_close (resource link_identifier)` – закриває з'єднання з базою даних;

					ДП.КН 22.459.13.000 ПЗ	Арк.
						22
Змн.	Арк.	№ докум.	Підпис	Дата		

– - substr (string string, int start , int length) - повертає частина рядка string, починаючи з символу start і кількістю length.

MySQL – сервер баз даних. MySQL характеризується високою продуктивністю, надійністю та легкістю у використанні.

MySQL – це система керування базами даних. В базі даних, дані зберігаються не разом, а в окремих таблицях. Саме це і забезпечує продуктивність та гнучкість. Таблиці пов'язуються між собою зв'язками, чим і забезпечується можливість поєднувати при виконанні запиту інформацію з декількох таблиць.

SQL можна охарактеризувати як мову структурованих запитів, що найбільш розповсюджена стандартна мова, використовувана для доступу до баз даних.

Короткий перелік можливостей MySQL:

- підтримується необмежена кількість користувачів, що одночасно працюють з базою даних;
- кількість рядків у таблицях може досягати 50 млн;
- швидке виконання команд;
- та ін.

MySQL є ідеальним рішенням для малих і середніх додатків. MySQL компілюється на безлічі платформ. Найбільше можливості сервера виявляються на Unix-серверах, де є підтримка багатопотоковості, що дає значний приріст продуктивності.

На даний час актуальна версія 8.0.29.

2 ПРОЄКТУВАННЯ ВЕБ-САЙТУ МІСТА ЗБАРАЖ

Змн.	Арк.	№ докум.	Підпис	Дата

ДП.КН 22.459.13.000 ПЗ

Арк.

23

2.1 Формулювання вимог до системи

Створенню будь-якого програмного комплексу передують аналіз вимог та декомпозиція предметної області.

Більшість спеціалістів в області програмування виділяють в процесі створення програмного забезпечення чотири етапи:

- аналіз - визначення потреб, які ставляться користувачем до майбутньої системи;
- проектування - розробка програмних рішень, що задовольняють потреби користувача;
- реалізація - створення необхідного програмного забезпечення;
- тестування створеного програмного забезпечення.

Деякі розробники вважають за необхідне виконувати вказані дії послідовно, як окремі етапи. Такий підхід носить назву модель водоспаду. В інших випадках передусім виконується найбільш важлива частина проекту, що дозволяє зменшити ризик зриву графіку робіт, після чого подія проходить в вигляді спіралі, поки весь проект не буде виконано.

Перед розробником інформаційної системи в першу чергу постає проблема достатньої зрозумілості вимог користувача до майбутньої системи. Необхідно конкретно визначитись з задачами, покладеними на майбутній проект.

2.2 Проектування структури веб-сайту м. Збараж

На даному Web-сайті потрібно розмістити наступні розділи:

- візитка міста (символи міста, запрошення до міста);
- відгомін віків (історія);
- основні вектори стратегічного розвитку міста;
- інфраструктура міста (міське господарство);
- туризм (туристична карта з розміщенням об'єктів туризму);

Змн.	Арк.	№ докум.	Підпис	Дата

ДП.КН 22.459.13.000 ПЗ

Арк.

24

- мистецтво, культура і спорт;
- особистості (уродженці міста, видатні люди, які жили чи живуть у місті, почесні громадяни);
- мерія (голова, заступник, секретар міської ради та ін.);
- “вікно в Збараж” (події, новини);
- гостьова книга.

Також потрібно передбачити можливість пошуку інформації в розділах.

Розділи візитка міста, відгомін віків, основні вектори стратегічного розвитку міста, інфраструктура міста, туризм, мистецтво, особистості, мерія повинні зберігати свою актуальність протягом великого часу і не передбачають систематичного оновлення. Тобто дані розділи є статичними і вони реалізуються у вигляді HTML-сторінок, що містять текстові та графічні складові.

Розділ „вікно в Збараж” (стрічка новини) є динамічним, тому, що актуальність його зберігається лише протягом обмеженого часу, через що він повинен систематично оновлюватися. Даний розділ реалізуємо з застосуванням бази даних, тобто з використання PHP і MySQL.

Оновлення динамічних матеріалів відбувається адміністратором, і тому потрібно передбачити розділ з обмеженим доступом для адміністрування (додавання, редагування чи знищення матеріалів).

Розділ гостьової книги є потоковим. Оновлення поточкових матеріалів здійснюється регулярно, по мірі надходження, тобто кожного разу коли гість сайту заповнить всю обов’язкову інформацію та натисне кнопку Зберегти на сторінці гостьової книги. Даний розділ реалізуємо з використанням бази даних та PHP.

Враховуючи те, що зміст поточкових матеріалів має контролюватися адміністратором, необхідним компонентом даного процесу є модерування

					ДП.КН 22.459.13.000 ПЗ	Арк.
						25
Змн.	Арк.	№ докум.	Підпис	Дата		

(візуальний контроль даних з можливістю внесення змін). Тобто виникає необхідність в розділі адміністрування чи модерування гостьової книги з обмеженим доступом.

Оскільки є обмежений доступ – потрібно скрипт для зміни імені адміністратора та паролю для адміністрування стрічки новин та гостьової книги.

В Web-дизайні таблиці використовуються в надзвичайно часто – не тільки для традиційних колонок цифр і іншої інформації, яку зручно показувати в вигляді таблиці, але як і елемент дизайну.

Для того щоб розмістити окремі блоки інформації в визначених місцях сторінки, таку сторінку проектують в вигляді сукупності комірок таблиць.

Для даного Web-сайту в цілому застосуємо так званий „табличний дизайн”, тобто „скелетом” буде таблиця без видимих контурів, в якій у відповідних комірках будуть розміщуватися малюнки меню, банери, та інше.

За допомогою графічного редактора Photoshop намалюємо малюнок, який буде як базовий дизайн сторінки і „поріжемо” його на фрагменти (пункти меню, назви розділів, графічні кнопки як окремі малюнки), які збережемо в форматі JPEG Image (.jpg), так щоб вклавши „порізані” фрагменти в відповідні комірки таблиці-„скелету” ми могли зібрати цілісний малюнок.

На рисунку 2.1 зобразимо схематично таблицю-„скелет” базової сторінки сайту.

Всі сторінки (розділи) сайту будуть складатися з одного і того ж „скелету”, тільки в відповідних комірках будуть змінюватися у відповідності з назвою розділу малюнок на якому зображено назву розділу і комірка в якій власне і буде розгортатися основна частина розділу, тобто інформаційна частина розділу.

					ДП.КН 22.459.13.000 ПЗ	Арк.
						26
Змн.	Арк.	№ докум.	Підпис	Дата		







 ЗБАРАЖ		фон		
				
 <p>ПОШУК</p>		Форма з текстовим полем для пошуку	Графічна  кнопка	
<p>Головна</p>		<p>“ВІКНО В ЗБАРАЖ”</p> <p>„Інформаційна” комірка</p>		
<p>Візитка міста</p>				
<p>Відомин віків</p>				
<p>Стратегічний розвиток</p>				
<p>Інфраструктура</p>				
<p>Туризм</p>				
<p>Мистецтво, культура</p>				
<p>Особистості</p>				
<p>Мерія</p>				
<p>Гостюва книга</p>				
		<p>Фон</p>		
				
				

Рисунок 2.1 – Макет сайту

Отже, використовуючи табличний дизайн ми отримаємо, що пункти головного меню будуть окремими малюнками і ми зможемо для кожного пункту меню (малюнку) задати посилання на файл відповідного пункту меню розділу. Візуально дане меню буде виглядати як цілісний малюнок.

					ДП.КН 22.459.13.000 ПЗ	Арк.
						27
Змн.	Арк.	№ докум.	Підпис	Дата		

Також в даній таблиці в відповідних комірках розмістимо форму з полем для вводу і графічною кнопкою, яку будемо використовувати для пошуку по розділах сайту.

2.3 Проектування бази даних

База даних сайту (bd_sait) складається з трьох таблиць: access, table_news, table_g_book.

Таблиця access бази даних bd_sait використовується для збереження імені та паролю для адміністрування розділів „Вікно в Збараж” (стрічки новин) та гостьової книги.

Таблиця 1.1 – Таблиця access бази даних bd_sait

Поле	Тип
<u>Id</u>	int(11)
user	varchar(20)
pass	varchar(20)

- Поле id – унікальний порядковий номер,
- поле user – ім'я користувача,
- поле pass – пароль.

Таблиця table_g_book бази даних bd_sait використовується для збереження повідомлень гостьової книги (інформації про відвідувача та повідомлення).

Таблиця 1.2 – Таблиця table_g_book бази даних bd_sait

Поле	Тип
<u>id</u>	int(11)
name_guest	text
name_city	text
name_country	text
name_e_mail	Text
name_www	text
text_mess	text
text_mess_answer	text
date_mess	datetime

- Поле id – унікальний порядковий номер;
- поле name_guest – ім'я відвідувача;
- поле name_city – назва міста відвідувача;
- - поле name_country – назва країни відвідувача;
- - поле name_e_mail – електронна пошта відвідувача;
- - поле name_www – Web-сайт відвідувача;
- - поле text_mess – текст повідомлення відвідувача;
- поле text_mess_answer – відповідь адміністратора гостьової книги на повідомлення відвідувача;
- поле date_mess – дата повідомлення відвідувача.

Таблиця table_news бази даних bd_sait використовується для збереження новин розділу „Вікно в Збараж”.

Таблиця 1.3 – Таблиця table_news бази даних bd_sait

	Тип
<u>Id</u>	int(11)
date	datetime
name_news	text
text_news	text
link_news	text

- поле id – унікальний порядковий номер,
- поле date – дата повідомлення,
- поле name_news – назва новини,
- поле text_news – текст новини,
- поле link_news – автор новини чи посилання на джерело.

3 РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ ВЕБ-САЙТУ

3.1 Вибір засобів реалізації

Розділи сайту візитка міста, відгомін віків, основні вектори стратегічного розвитку міста, інфраструктура міста, туризм, мистецтво, особистості, мерія є статичним, тобто такими в яких не передбачається систематичного оновлення, а тому їх виконаємо з застосування „чистого” HTML.

На сторінці сайту „Вікно в Збараж” буде розміщено стрічку новин, а у розділі Гостьова книга буде відповідно розміщено гостьову книгу. Дані сторінки будуть динамічними і виникає необхідність в базі даних MySQL та PHP.

Для зручності та наглядності будемо використовувати Visual Studio Code (рис. 3.1). Visual Studio Code - ПЗ від компанії Microsoft для створення, редагування та відлагодження сучасних вебзастосунків і програм для хмарних систем. Visual Studio Code розповсюджується безкоштовно і доступний у версіях для платформ Windows, Linux і OS X.

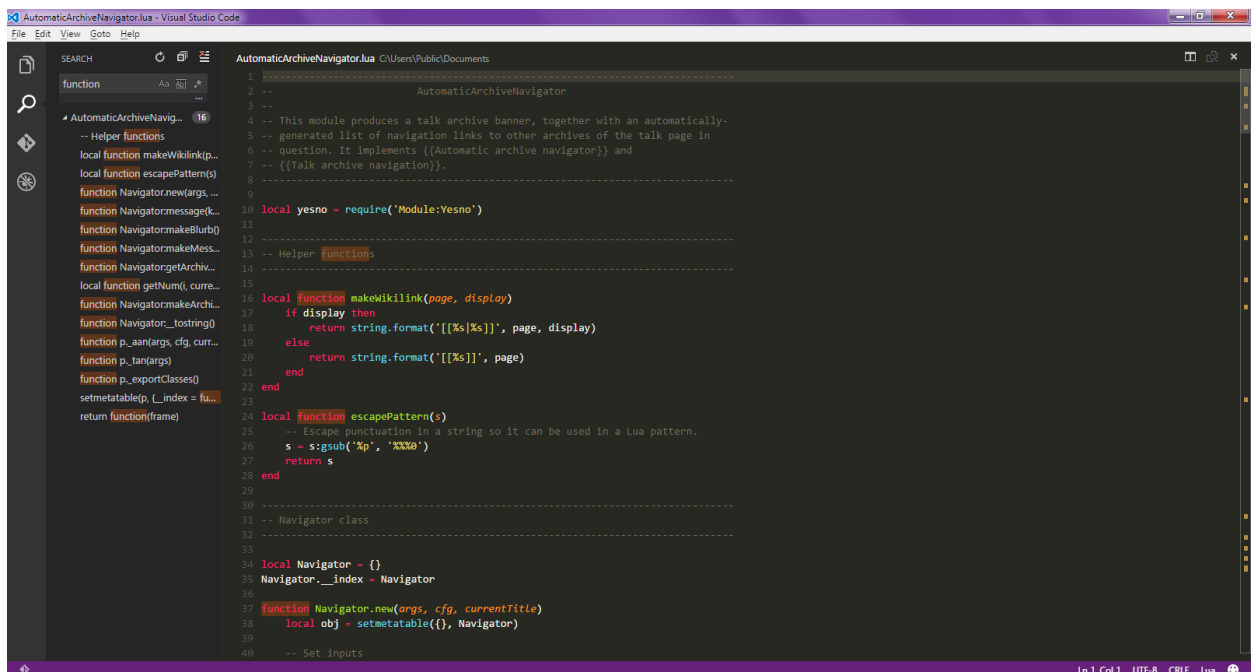


Рисунок 3.1–Visual Studio Code

					ДП.КН 22.459.13.000 ПЗ	Арк.
						31
Змн.	Арк.	№ докум.	Підпис	Дата		

3.2 Встановлення веб сервера

ХАМРР — безкоштовна багатоплатформова збірка веб-сервера з відкритим початковим кодом, що містить HTTP-сервер Apache, базу даних MySQL й інтерпретатори скриптів для мов програмування PHP та Perl, а також додаткові бібліотеки, що дозволяють запустити повноцінний вебсервер. Розробником є Apache Friends.

Повний пакет містить:

- Web-сервер Apache;
- СКБД MySQL
- Утиліту phpMyAdmin.
- PHP
- FTP-сервер FileZilla
- Perl
- POP3/SMTP сервер
- Для windows надається панель для управління встановленими на

сервері засобами ХАМРР Control Panel

ХАМРР працює з ОС Microsoft Linux, Mac OS X і Solaris. Програма вільно розповсюджується згідно з ліцензією GNU General Public License і є безкоштовним, зручним у роботі web-сервером, здатним обслуговувати динамічні сторінки. На сьогоднішній день хамрр є однією з найкращих збірок вебсервера.

Для початку інсталяції ХАМРР необхідно завантажити інсталяційний пакет, для цього переходимо на сторінку завантаження Apache Friends <https://www.apachefriends.org/download.html> і завантажуюємо версію 8.1.6 / PHP 8.1.6.

Далі переходимо у каталог Завантаження і запускаємо до виконання файл з назвою хамрр-windows-x64-8.1.6-0-VS16-installer.

					ДП.КН 22.459.13.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		32

На екрані з'явиться повідомлення щодо контролю облікових записів (рис. 3.2).

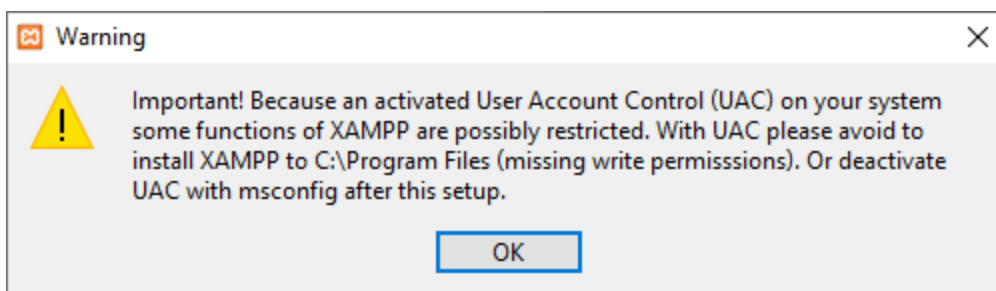


Рисунок 3.2 – повідомлення щодо контролю облікових записів

Натискаємо ОК. Запустився майстер інсталяції і натискаємо кнопку Продовжити (рис. 3.3).

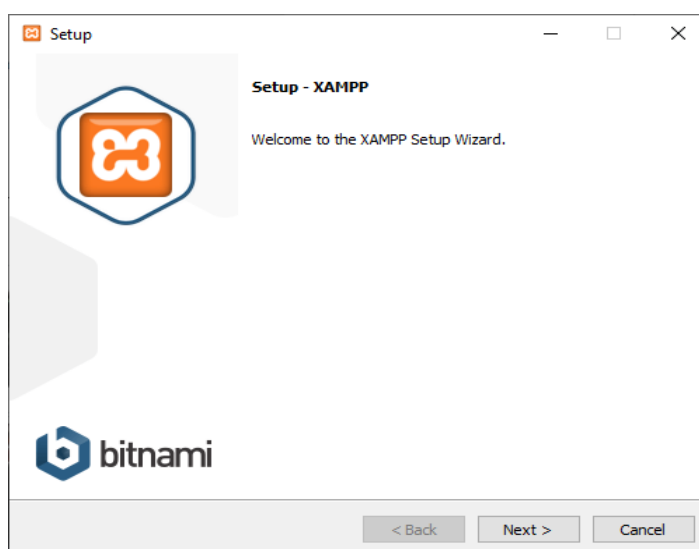


Рисунок 3.3– Вітання майстра інсталяції XAMPP

На наступному кроці необхідно вибрати, які саме компоненти необхідно встановити. Але оскільки ці всі компоненти є важливими і варто залишити перелік вибраних без змін, тому просто натискаємо Далі (рис. 3.4).

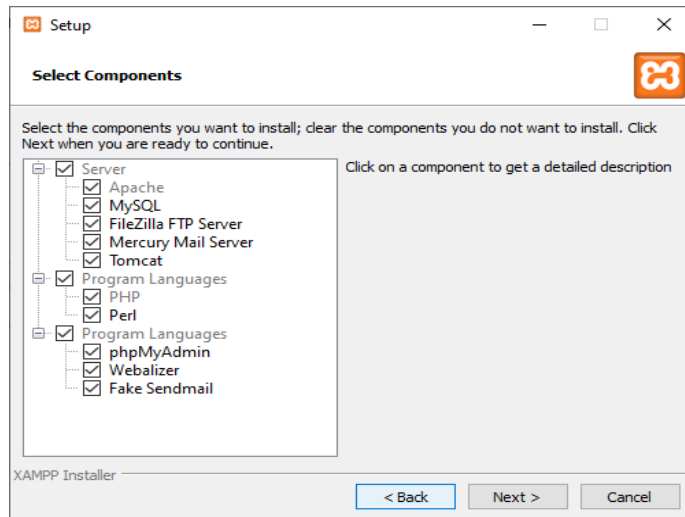


Рисунок 3.4 – Діалог майстра інсталяції ХАМРР

Далі інсталятор пропонує нам вибираємо каталог куди встановлювати веб-сервер. Залишаємо без змін і натискаємо Далі(рис. 3.5).

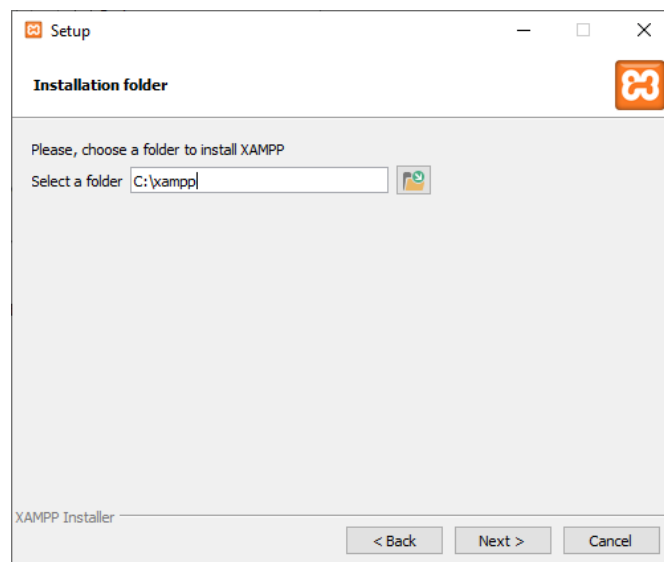


Рисунок 3.5 – Діалог майстра інсталяції ХАМРР

На цьому етапі встановлення майстер інсталяції запитує якою мовою хочете бачити панель керування ХАМРР. Залишаємо Англійську, оскільки Українська не пропонується і натискаємо Далі (рис. 3.6).

На останньому кроці перед початком встановлення заберемо відмітку “Дізнатися більше про Bitnami для ХАМРР ” і натиснемо Далі(рис. 3.7).

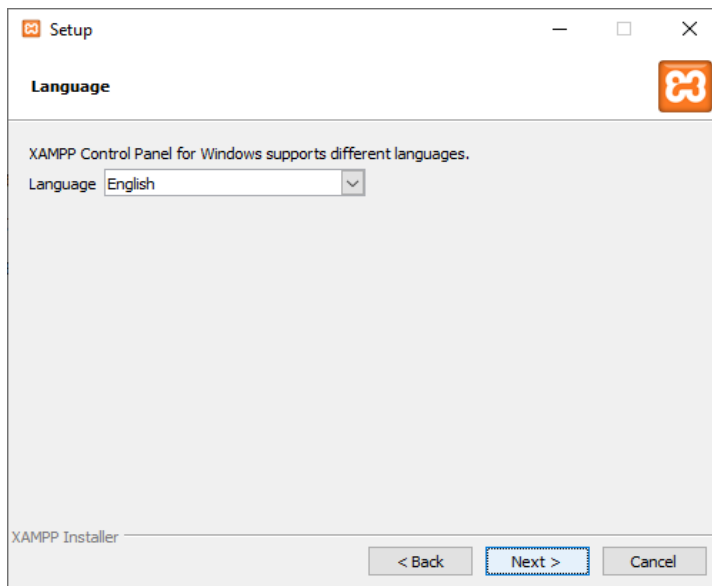


Рисунок 3.6 – Діалог майстра інсталяції ХАМРР

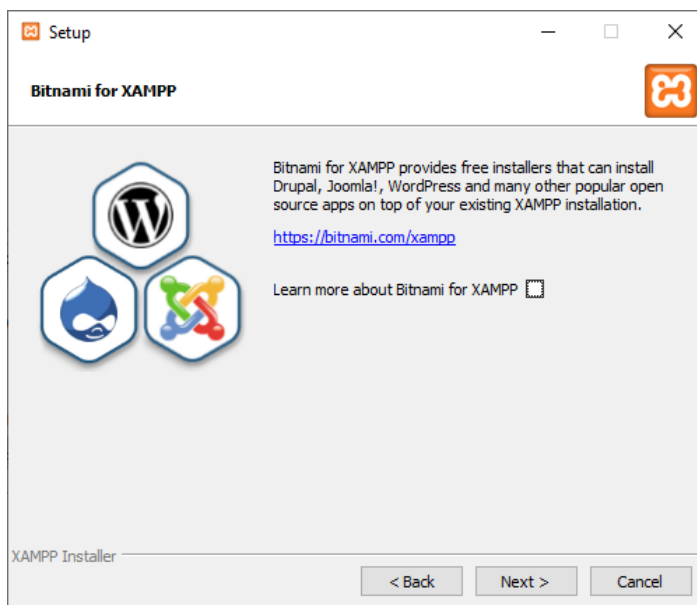


Рисунок 3.7 – Діалог майстра інсталяції ХАМРР

Тепер майстер встановлення готовий встановити ХАМРР, натискаємо Далі (рис. 3.8).

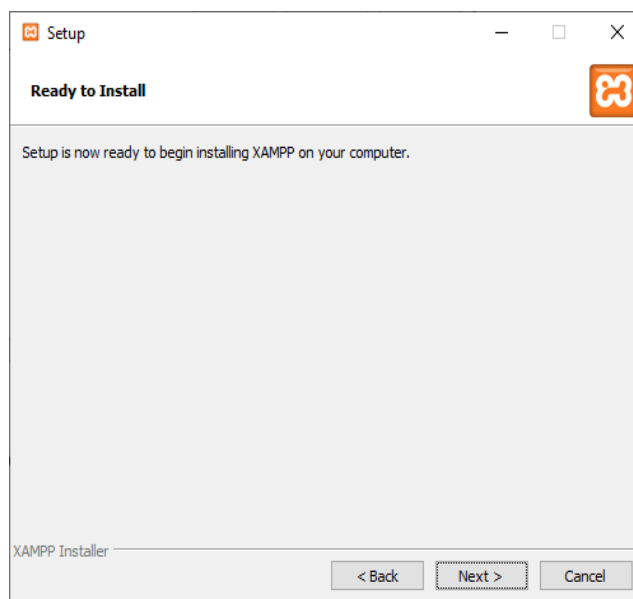


Рисунок 3.8 – Діалог майстра інсталяції ХАМРР

Після встановлення, Майстер інсталяції повідомить про закінчення встановлення і запропонує “Чи хочемо ми запустити панель керування”. Відмічаємо пташкою, що хочемо і натискаємо Завершити.

На цьому етапі встановлення веб-сервера завершено і ми бачимо панель керування (рис. 3.9).

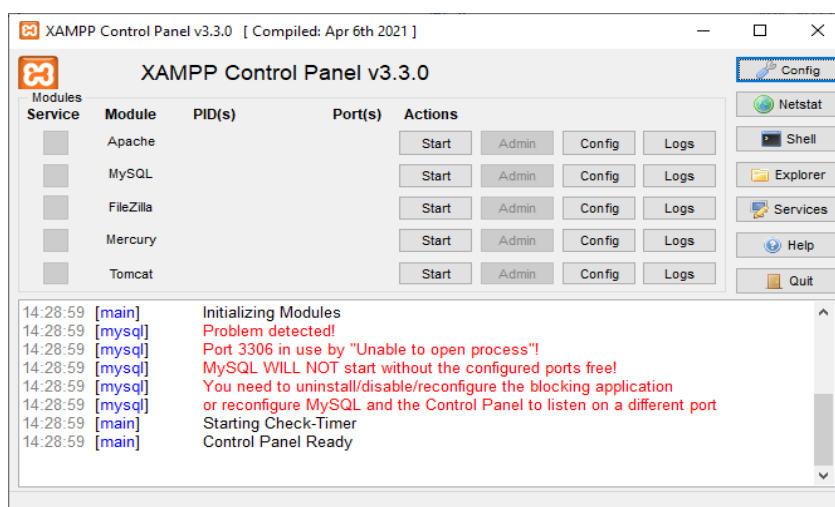


Рисунок 3.9 – Панель керування ХАМРР

					ДП.КН 22.459.13.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		36

Для того, щоб веб-сервер Apache та СКБД MySQL почали працювати спочатку необхідно активувати служби Apache та MySQL, натиснувши відповідні кнопки "Пуск" у стовпці "Дії".

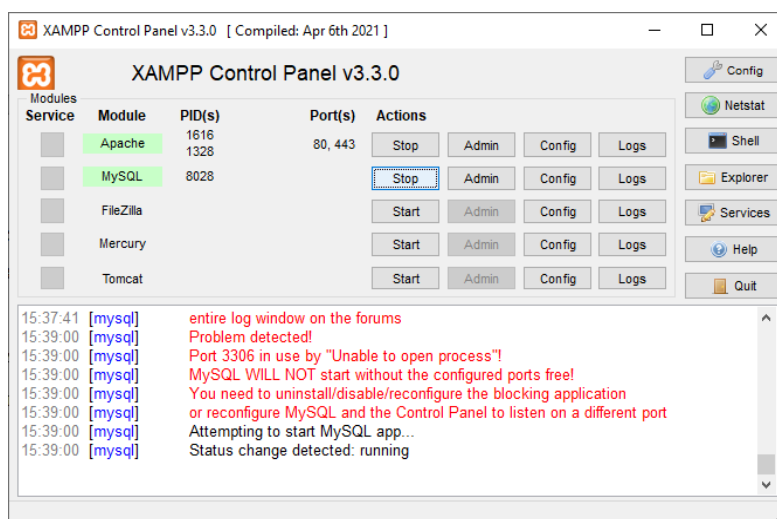


Рисунок 3.10 – Панель керування XAMPP

Коли запускаються веб-сервер Apache та СКБД MySQL, колір модулів Apache та СКБД MySQL стають зеленими, з'явиться номер PID та номер порту (рис. 3.10).

Аналогічно при потребі можна запустити і інші модулі.

На цьому встановлення веб-сервера Apache, СКБД MySQL, FTP-сервер FileZilla, PHP завершено.

3.3 Реалізація інтерфейсу

Основою даних розділів буде таблиця, яка описана в розділ 2.2.

В кожному розділі в „інформаційну” комірку (див. Рисунок 2.2.1) таблиці вкладемо ще одну таблицю, також без видимих контурів, в якій і розмістимо матеріали, що стосують відповідного розділу, тобто:

- в розділі Web-сайту візитка міста зобразимо символіку міста (герб та

прапор міста) та їх походження;

- в розділі Web-сайту відгомін віків опишемо коротко історичне минуле міста Збаража, наведемо декілька малюнок;

- в розділі Web-сайту основні вектори стратегічного розвитку міста опишемо основні вектори стратегії планування розвитку територіальної громади м. Збараж в міському господарстві, екології, економіці, культурі і спорті, туризмі, розвитку місцевого самоврядування, соціального захисту населення;

- в розділі Web-сайту інфраструктура міста – надамо перелік промислових, транспортних, будівельних та інших підприємств м. Збаража;

- в розділі Web-сайту туризм варто зобразити туристичну карту м. Збаража та провести короткий екскурс по історичному туризму (пам'ятках архітектури), археологічному туризму (розкопки древніх поселень);

- в розділ Web-сайту мистецтво розкажемо про творчі колективи та майстрів Збаражчини;

- в розділ Web-сайту особистості опишемо про Київське земляцтво, діаспору, видатних людей та їх досягнення;

- в розділ Web-сайту мерія розмістимо коротку інформації про мерію міста.

На сторінці сайту „Вікно в Збараж” розмістимо стрічку новин. В такому випадку дана сторінка сайту буде динамічною, тому, що актуальність новин зберігається лише протягом обмеженого часу, через що потрібно буде її систематично оновлювати. Розділі Гостьова книга буде містити власне і саму гостьову книгу, який відповідно також буде динамічним.

3.4 Реалізація продукту

Розділи, які є статичні описувати не будемо, оскільки в них використовується чистий HTML. По аналогії з статичними сторінками сайту в „інформаційну” комірку таблиці-„скелету” вкладемо окрему таблицю

					ДП.КН 22.459.13.000 ПЗ	Арк.
						38
Змн.	Арк.	№ докум.	Підпис	Дата		

розміром 1 рядок на 1 стовпець і якій задамо ширину 75% від ширини „інформаційної” комірки для того щоб стрічка новин виводилася не на всю ширину „інформаційної комірки” (для дизайну). В новостворену таблицю вкладемо наступний PHP скрипт: `<?php include ("news_index_script.php") ?>`, де `include ()` – функція PHP, яка замінює себе в процесі виконання файлом, який вказаний як аргумент функції. Тобто при виконанні в даній таблиці буде відображено стрічку новин за участю PHP скрипта з файлу `news_index_script.php`.

Сторінку „Вікно в Збараж” зображено на рисунку 3.11.

Опишемо PHP скрипт стрічки новин (`news_index_script.php`).

Для збереження новин ми будемо використовувати базу даних (БД) MySQL, тому для початку нам будуть потрібні ім’я сервера бази даних, назва бази даних, ім’я користувача та пароль для підключення до бази даних.

```
$host="localhost"; //задаємо хост
$user="duplom"; //задаємо ім'я користувача
$pass="duplom"; //задаємо пароль
$base="bd_sait"; // база даних.
```

Також на початку визначимося скільки повідомлень новин буде виводитися відразу на першій сторінці `$mess=4`.

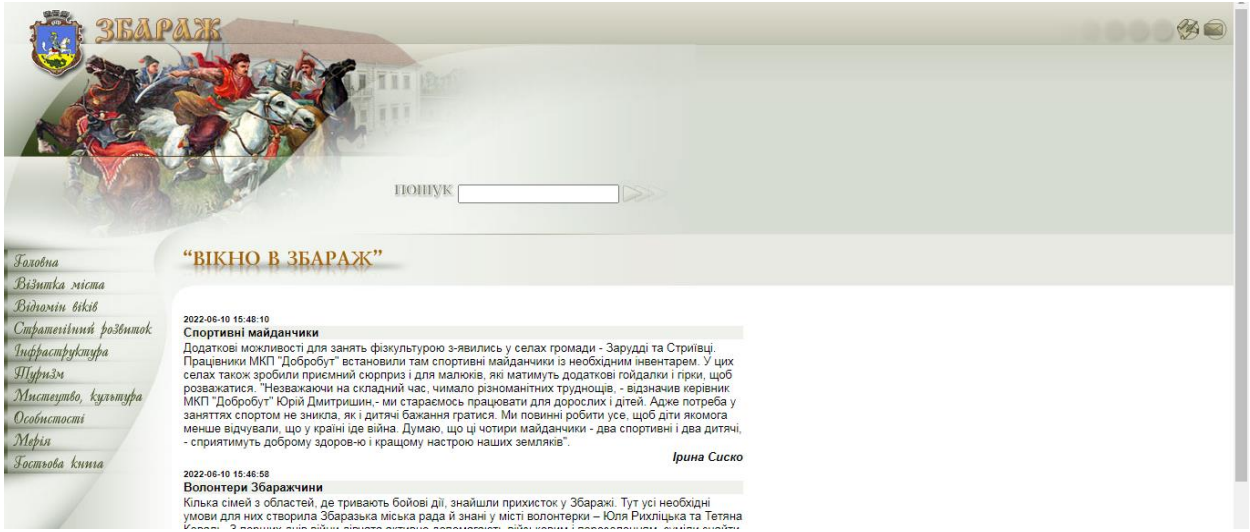


Рисунок 3.11 - Сторінка „Вікно в Збараж”

Відразу накладемо умову, що \$p (змінна, в якій буде зберігатися номер повідомлення) не задана - присвоюємо їй 0. Надалі коли буде здійснюватися перехід по сторінках стрічки новин – змінна \$p буде зберігати номер першого повідомлення відповідної сторінки стрічки новин.

Далі підключаємося до бази даних за допомогою функції PHP MySQLi_connect():

```
$connect=MYSQLi_CONNECT($host, $user, $pass, $base) or die("Unable to connect to database server");
```

Далі нам потрібно буде посортувати в таблиці table_news БД bd_sait повідомлення новин за датою в порядку спадання. Для цього буде достатньо посортувати поле id в порядку спадання, тобто „найсвіжіші„ новини будуть мати більші значення поля id і відповідно перемістяться на початок таблиці). Наступним кроком нам буде потрібно вибрати повідомлення новин з \$p-ї позиції і кількістю \$mess). На практиці реалізуємо це за допомогою функції PHP mysql_query ():

```
$result = mysql_query($connect, "SELECT * FROM table_g_book ORDER BY id DESC limit $p,$mess");
```

Отже, \$result –масив полів таблиці table_news, створений в результаті вищенаведеного запиту.

Вичислюємо загальну кількість повідомлень. Це нам знадобиться для того щоб побудувати рядок-посилання на сторінки новин коли їх буде більше заданої кількості новин на сторінці, тобто більше \$mess:

```
$count = mysql_num_rows(mysql_query($connect,"SELECT * FROM table_g_book"));
```

Закриваємо з'єднання з БД: mysql_close(\$connect).

Отримавши всі необхідні дані від БД можемо виводити стрічку новин, скориставшись функцією PHP mysql_fetch_row(), яка отримує результуючий рядок таблиці БД як масив. Вивід стрічки новин доцільно буде організувати за допомогою циклу: while (\$row = mysql_fetch_row(\$result)). Для зручності

					ДП.КН 22.459.13.000 ПЗ	Арк.
						40
Змн.	Арк.	№ докум.	Підпис	Дата		

(з боку дизайну) в тіло циклу вкладемо таблицю, в відповідних комірках якої будуть знаходитися: дата новини (\$row[0]), заголовок новини (\$row[1]), текст новини (\$row[2]) і автор (\$row[3]).

Для навігації по сторінках стрічки новин нам потрібна буде навігаційна панель. Навігаційну панель побудуємо у вигляді рядка-посилання з номерами сторінок стрічки новин.

Задаємо початковий номер першого повідомлення в таблиці БД \$page=0. Далі задаємо цикл з умовою, що виводимо кожен новий номер сторінки коли номер сторінки (на сторінці \$mess повідомлення) не більший загальної кількості повідомлень While (\$page<\$count).

Далі в тіло циклу вкладемо посилання у вигляді відносних номерів сторінок в форматі: початкове повідомлення - кінцеве повідомлення на сторінці, тобто номер повідомлення \$page+1 (тому, що відносний номер першого повідомлення в таблиці БД = 0) - номер повідомлення \$page+\$mess (+\$mess тому, що на сторінці \$mess повідомлення): \" , \$page+1 , \"-\" , \$page+\$mess , \" | \" , \"\".

Також в тілі циклу ще задамо наступний номер повідомлення: \$page=\$page+\$mess.

Навігаційну панель зобразимо на рисунку 3.12.

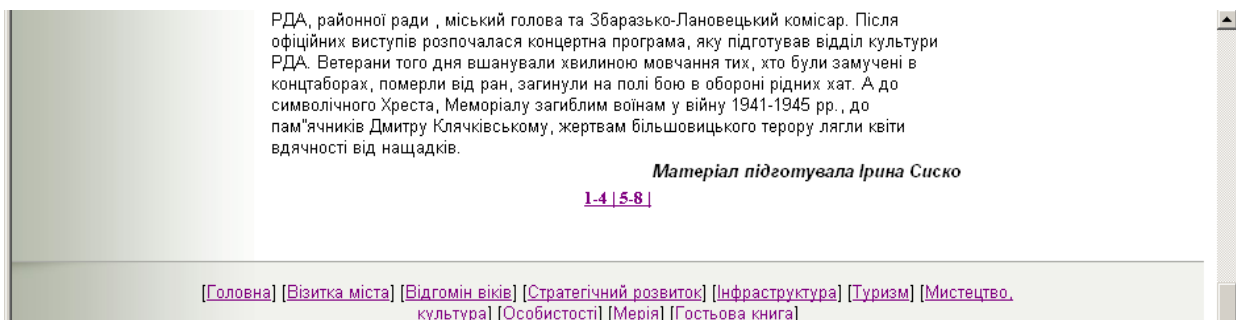


Рисунок 3.12 – Навігаційна панель

На цьому опис PHP скрипта стрічки новин (news_index_script.php) завершимо.

					ДП.КН 22.459.13.000 ПЗ	Арк.
						41
Змн.	Арк.	№ докум.	Підпис	Дата		

Для адміністрування стрічки новин, тобто для додавання, редагування чи видалення повідомлень новин використовується PHP скрипт news_admin.php.

Для початку потрібно вказати ім'я сервера бази даних, назва бази даних, ім'я користувача та пароль для підключення до бази даних. Оскільки даний скрипт працює з тією ж самою БД, що і скрипт стрічки новин, тому дана частина скрипта буде аналогічною.

Для ведення чи редагування повідомлень новин насамперед потрібно форму з текстовими полями і кнопкою „Зберегти”. А оскільки дану форму будемо використовувати двічі - доцільно її оформити у вигляді функції: function form_enter_update(\$name_news, \$text_news, \$link_news), де \$name_news – коротка назва новини, \$text_news – текст новини, \$link_news – посилання на джерело чи автора. Тілом даної функції буде таблиця з текстовими полями.

Оскільки є присутні текстові поля для вводу тому виникає необхідність для контролю за введеною інформації, тобто перевірки на наявність ненормативної лексики, тегів, SQL-запитів. Але так як перевірку введеної інформації потрібно буде двічі (введення нового повідомлення новини та редагування повідомлення новини) тому доцільно буде викласти даний „контроль” у вигляді функції: function str (\$text, \$k).

За допомогою функції substr(), яка повертає частину рядка \$text починаючи з 0 по \$k символ, тобто обмежимо кількістю символів не більше \$k:

```
$text=substr($text,0,$k).
```

За допомогою функції ltrim() вирізуємо пробіли з початку кожного рядка:

```
$text=ltrim($text).
```

					ДП.КН 22.459.13.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		42

Складемо масив `replace` не допустимих символів, тобто таких, які можуть бути використанні для написання не літературних термінів, тегів чи `sql` запитів:

```
$replace = array ("'", "<", ">", "/", "sql_query", "sql_connect", "password",  
"Password", "PASSWORD" , "passwd", "PASSWORD", "Passwd", "User", "user",  
"USER", "LOGIN", "Login", "login", "SET", "set", "Set", "(", ")", "=", "$").
```

За допомогою функції `str_replace` з введеного рядка вирізуємо не допустимі символи задані в масиві `replace`:

```
$text=str_replace($replace, "-", $text).  
return $text; }.
```

Так як адмініструванням стрічки новин має виконувати обмежене коло людей, то нам потрібно передбачити механізм обмеження доступу до даної сторінки.

Для авторизації користувача застосуємо авторизацію засобами HTML – через код 401. Будь-який броузер зустрічаючи даний код висвітить вікно авторизації і попросить ввести ім'я та пароль.

Після введення імені користувача і паролю в змінних `$PHP_AUTH_USER` і `$PHP_AUTH_PW` будуть ім'я та пароль відповідно.

Вікно авторизації зображено на рисунку 3.13.

Рисунок 3.13 – Вікно авторизації

У випадку коли буде натиснута кнопка Отмена – „Ви не пройшли авторизації, тому доступ до адміністрування ЗАБОРОНЕНО!!!”.

Отже, накладемо умову, що якщо \$PHP_AUTH_USER не визначена виводимо вікно авторизації:

```
if(isset($_POST['send_mess_g_book']))  
{ Header("WWW-Authenticate: Basic realm=\"Новини\"");  
Header("HTTP/1.1 401 Unauthorized");
```

На випадок коли користувач натисне кнопку “Відміна” виведемо повідомлення:

```
echo "Ви не пройшли авторизації, тому доступ до адміністрування  
ЗАБОРОНЕНО!!!\n";
```

```
і на цьому завершимо виконання скрипта: exit;}
```

Правильне ім'я та пароль зберігають в таблиці access бази даних bd_sait.

Пароль в базі даних зберігається, як md5-хеш даного паролю, тобто у вигляді 32-символьного 16-річного числа.

Отримавши введені користувачем ім'я та пароль, та прочитавши ім'я та пароль з бази даних, потрібно їх порівняти:

```
if ($aut[1]==$_SERVER['PHP_AUTH_USER'] and $aut[2] == md5  
($_SERVER['PHP_AUTH_PW'])).
```

У випадку не співпадання – виведемо повідомлення "Ви не пройшли авторизації (не вірне ім'я користувача чи пароль), тому доступ до адміністрування ЗАБОРОНЕНО!!!” і виконання скрипта на цьому припиняється.

Надалі визначимося, що \$p – ідентифікатор номера сторінки стрічки новин, \$id – номер повідомлення новин в базі даних, \$job – змінна, в якій буде вказано, що робити з повідомленням новин (delete – видалити повідомлення за номер \$id з БД, update – оновити повідомлення новин за номер \$id в базі даних).

					ДП.КН 22.459.13.000 ПЗ	Арк.
						44
Змн.	Арк.	№ докум.	Підпис	Дата		

Для уникнення різного роду помилок приведемо змінні \$p та \$id до цілих чисел за допомогою функції PHP intval():

```
if (isset($_GET['p'])) $p = intval($_GET['p']);  
if (isset($_GET['id'])) $id = intval($_GET['id']);
```

Наступним кроком нам буде потрібно накласти умову, що коли \$job <> „update”, тобто коли потрібно вивести форму для введення нового повідомлення та вивести повідомлення стрічки новин для перегляду, які розкладемо по \$mess повідомлень на сторінці, де \$mess кількість повідомлень на сторінці. Зупинятися детально на даному фрагменті скрипта не будемо, тому що вивід стрічки новин описаний в попередньому розділі. Доповненням до даного фрагменту буде два посилання „Відредагувати повідомлення” і „Знищити повідомлення”:

```
echo "<font size=\"2\" face=\"Arial, Helvetica, sans-serif\"><a  
href=\"news_admin.php?id=$row[0]&job=update\">Відредагувати  
повідомлення</a></font>";
```

```
echo "<font size=\"2\" face=\"Arial, Helvetica, sans-serif\"><a  
href=\"news_admin.php?p=$p&id=$row[0]&job=delete\">Знищити  
повідомлення</a></font>".
```

Задамо умову, що коли натиснемо на формі введення нового повідомлення новини кнопку Зберегти виконаємо SQL-запит: `MYSQLi_QUERY($connect, "INSERT INTO table_news (`id` , `date`, `name_news` , `text_news` , `link_news`) VALUES ('', Now(), '$name_news', '$text_news', '$link')")`, тобто запишемо введені повідомлення в таблицю table_news бази даних.

Для навігації по сторінках стрічки новин побудуємо навігаційну панель аналогічно як і в попередньому скрипті розділу „Вікно в Збараж”.

Виконання описаної вище частини скрипта наведемо на рисунку 3.14.

Отже, якщо адміністратор стрічки новин натисне на посилання „Знищити повідомлення”, то змінна \$job буде рівна „delete”. Накладемо

					ДП.КН 22.459.13.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		45

умову перед виведенням стрічки новин для перегляду, що коли \$job=„delete” то підключаємося до БД і виконуємо за допомогою функції mysql_query SQL-запит „DELETE FROM `table_news` WHERE id = \$id”.

Якщо ж адміністратор стрічки новин натисне посилання „Змінити повідомлення”, \$job буде рівна "update".

Рисунок 3.14 – Сторінка адміністрування стрічки новин

Знову ж таки накладемо умову if (\$job=="update"), то підключаємося до бази даних, виконуємо запит вибірки повідомлення новин за id номером:

```
$result = mysqli_query($connect, "SELECT * FROM table_news WHERE id = $id LIMIT 1");
```

Викликаємо функцію з формою редагування повідомлення новин form_enter_update (\$row[2], \$row[3], \$row[4]). Дану функцію ми вже використовували для введення нових повідомлень новин, тому зупинятися на детальному описі не будемо.

Форма для редагування повідомлення новин зображено на рис. 3.15.

Рисунок 3.15 – Форма для редагування повідомлення новин

					ДП.КН 22.459.13.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		46

Накладаємо знову умову, якщо адміністратор натисне на кнопку „Зберегти” if (isset(\$_POST['Update_button'])), то застосовуємо функцію фільтрації введеною інформації:

```
$name_news =str ($name_news, 100);
```

```
$text_news =str ($text_news, 500);
```

```
$link =str ($link, 100), підключаємося до бази даних і пишемо запит на оновлення: mysqli_query($connect, "UPDATE table_news SET name_news = '$name_news' , text_news ='$text_news' , link_news ='$link' WHERE id = '$id' LIMIT 1;");
```

Сторінка де буде розміщена гостьова книга по аналогії з попередніми складається з таблиці-„скелету” в якій у „інформаційній” комірці PHP скрипт: `<?php include ("gb_index_script.php")?>`.

Сторінка гостьової книги зображена на рисунку 3.16.

Рисунок 3.16 – Сторінка гостьової книги

Розглянемо скрипт гостьової книги gb_index_script.php. По аналогії з попередніми розділами першим кроком нам потрібно вказати необхідну

інформацію для з'єднання з базою даних та побудувати форму з текстовими полями для вводу та кнопкою Зберегти.

Накладемо умови, що коли гість сайту натисне на кнопку Зберегти:

if \$_POST['send_mess_g_book'], то ведену інформацію занесемо в таблицю базу даних table_g_book. Отже, коли виконується дана умова, тобто коли відвідувач сайту натисну на кнопку Зберегти ми отримаємо в наступних змінних: \$name_guest – ім'я відвідувача сайту, \$name_city – місто, \$name_county – країна, \$text_mess – текст повідомлення, \$name_e_mail – адреса електронної пошти, \$name_www – адреса WWW. Далі потрібно профільтрувати ведену інформацію за допомогою функції фільтрації str(), яка детально описана в попередньому розділі:

```
$name_guest =str ($_POST['name_guest'], 30);
```

```
$name_city =str ($_POST['name_city'], 30);
```

```
$name_country =str ($_POST['name_country'], 30);
```

```
$text_mess =str ($_POST['text_mess'], 255);
```

```
$name_e_mail =str ($_POST['name_e_mail'], 30);
```

```
$name_www =str ($_POST['name_www'], 30);
```

Профільтрувавши введену інформацію, перевіримо чи заповненні всі текстові поля форми і якщо так – запишемо значення текстових полів форми в таблицю бази даних, тобто виконаємо наступний SQL-запит за допомогою функції: MySQLi_QUERY():

```
MySQLi_QUERY($connect, "INSERT INTO table_g_book ( `id` ,  
`name_guest` , `name_city` , `name_country` , `name_e_mail` , `name_www` ,  
`text_mess` , `date_mess`) VALUES (", '$name_guest', '$name_city',  
'$name_country', '$name_e_mail', '$name_www', '$text_mess', Now())").
```

В протилежному випадку виведемо повідомлення „!!!Ваше повідомлення не збережене, перевірте чи заповненні обов'язкові поля!!!”.

Окрім форми для вводу нових повідомлень нам потрібно вивести власне самі записи гостьової книги. Для цього нам знову ж таки потрібно

					ДП.КН 22.459.13.000 ПЗ	Арк.
						48
Змн.	Арк.	№ докум.	Підпис	Дата		

встановити з'єднання з базою даних, вибрати таблицю і у вигляді таблиці вивести повідомлення гостьової книги, розклавши як і в попередньому скрипті стрічки новин по \$mess – повідомленню на сторінці. Виключенням в даному випадку буде умова, що якщо в таблиці бази даних буде відповідь адміністратора на конкретне повідомлення то потрібно вивести ще один рядок таблиці з даним коментарем адміністратора.

Останнім кроком в даному скрипті буде побудова навігаційної панелі, яку ми запозичимо з скрипта стрічки новин.

Враховуючи те, що зміст потокових матеріалів має контролюватися адміністратором, необхідним компонентом даного процесу є модерування (візуальний контроль даних з можливістю внесення змін). Тобто виникає необхідність в розділі модерування гостьової книги.

Для модерування гостьової книги використовується скрипт `gb_admin.php`.

Скрипт модерування гостьової книги буде подібними до скрипта адміністрування стрічки новин.

Оскільки повідомлення зберігаються в базі даних, відповідно потрібно логін, пароль, ім'я сервера бази даних для з'єднання з базою даних аналогічно як і в попередньому скрипті гостьової книги.

Результат виконання даного скрипта наведемо на рисунку 3.17.

Так як, модерування гостьової книги відбувається тільки модератором (адміністратором), тому потрібно обмежити доступ до даної сторінки. А тому знову застосуємо авторизацію засобами HTML – через код 401. Даний метод авторизації описаний раніше.

Наступним кроком нам потрібно показати повідомлення гостьової книги. Вивід повідомлень з посиланнями „Знищити повідомлення” та „Редагувати повідомлення” гостьової книги аналогічний виводу повідомлень стрічки новин.

					ДП.КН 22.459.13.000 ПЗ	Арк.
						49
Змн.	Арк.	№ докум.	Підпис	Дата		

Згадаємо, що \$id – номер повідомлення новин в базі даних, \$job – змінна, в якій буде вказано, що робити з повідомленням новин (delete – видалити повідомлення за номер \$id з БД, update – оновити повідомлення новин за номер \$id в базі даних), \$mess - кількість повідомлень на сторінці.

Отже, зайшовши перший раз на сторінку модерування, потрібно показати повідомлення по \$mess повідомлень на сторінці.

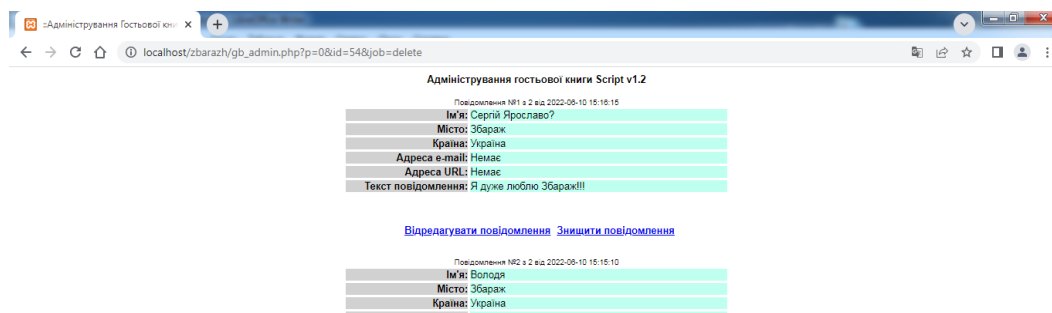


Рисунок 3.17 – Виконання скрипта gb_admin.php

Натиснувши на посилання „Відредагувати повідомлення” \$job==”update” потрібно буде вивести форму редагування повідомлення.

У випадку обрання посилання „Знищити повідомлення” \$job==”delete” видаляємо повідомлення з бази даних, тобто виконуємо SQL-запит за допомогою функції `mysqli_query($connect, " DELETE FROM `table_g_book` WHERE id = $id");`.

В таблиці - „скелеті” кожного розділу є комірки де розміщено форму з текстовим полем і графічною кнопкою, яку зображено на рисунку 3.18.



Рисунок 3.18 – Форма з текстовим полем і кнопкою для пошуку по сайту

Натискаючи на графічну кнопку пошуку передаємо управління скрипту вказану в атрибуті action форми, який буде опрацьовувати отриману інформацію з текстового поля форми. Тобто передамо управління скрипту search.php.

За основу сторінки пошук взято знову ж таки таблиця – „скелет”, в якій в інформаційній комірці через функцію include () вставляємо скрипт пошуку search_script.php. Сторінку з результатами пошуку зображено на рисунку 3.19.

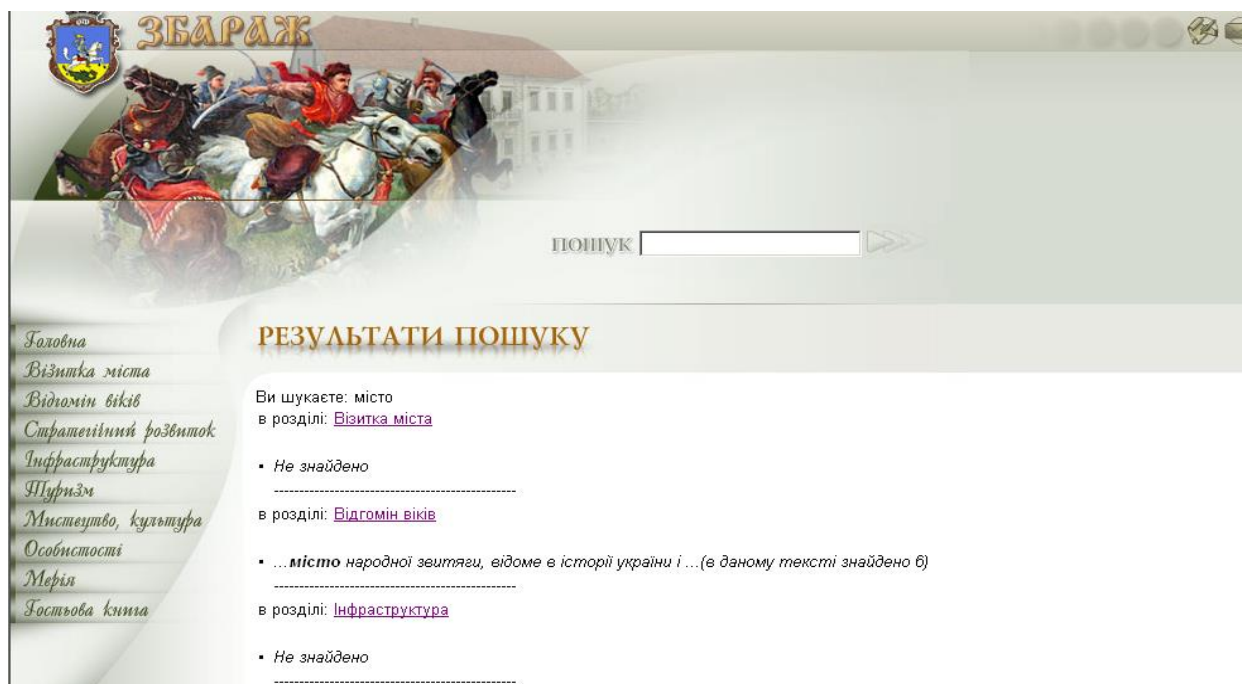


Рисунок 3.19 – Сторінка з результатами пошуку

Для початку визначимося, що пошук будемо проводити в розділах „візитка міста”, „відгомін віків”, „основні вектори стратегічного розвитку міста”, „інфраструктура міста”, „туризм”, „мистецтво”, „особистості”, „мерія”. В розділ „Вікно в Збараж” пошук буде проводитися в таблиці table_news бази даних.

Для підключення до бази даних потрібно насамперед логін, пароль, ім’я сервера бази даних та ім’я таблиці бази даних в якій будемо проводити

пошук:

```
$host="localhost"; //задаємо хост  
$user="duplom"; //задаємо ім'я користувача  
$pass="duplom"; //задаємо пароль  
$base="bd_sait"; // база даних  
$table_base="table_news".
```

Оскільки пошук буде проводитися в декількох статичних розділах, тобто в файлах, то доцільно алгоритм пошуку організувати у вигляді функції `search_text ($link_rozdil, $rozdil, $myquery)`, де `$link_rozdil` – назва html-файлу, `$rozdil` – назва розділу, `$myquery` - стрічка запиту.

Опишемо дану функцію :

```
function search_text ($link_rozdil, $rozdil, $myquery)
```

За допомогою функції PHP `file ()` прочитаємо файл в масив `$filearray`:

```
$filearray = file("$link_rozdil");
```

За допомогою функції PHP `strip_tags()` виріжемо з масиву `$filearray` всі теги html і PHP та скориставшись функцією PHP `strtolower()` переведемо всі символи масиву `$filearray` в нижній регістр:

```
$mystring=strtolower(strip_tags(implode (" ", $filearray)));
```

Перетворюємо запит `$myquery` в нижній регістр:

```
$myquery =strtolower($myquery);
```

Обчислимо довжину рядка запиту `$myquery`, це нам знадобиться для виводу знайденої інформації:

```
$myquery_len=strlen($myquery).
```

За допомогою функції PHP `strpos ()`, власне і на якій будується весь механізм пошуку знайдемо перше входження рядка запиту `$myquery` `$mystring` в масив `$mystring`, який прочитаний з файлу.

```
$pos = strpos($mystring, $myquery);
```

Далі накладемо умову, якщо `$pos` не визначена, то виведемо повідомлення „Не знайдено”, в іншому випадку виведемо рядок в форматі

					ДП.КН 22.459.13.000 ПЗ	Арк.
						52
Змн.	Арк.	№ докум.	Підпис	Дата		

...запит + частину рядку \$mystring починаючи з \$pos+myquery_len і довжиною 50 символів ... (в даному тексті знайдено (порахуємо кількість появ підрядка \$myquery в рядку \$mystring) substr_count(\$mystring, \$myquery)):

```
if ($pos === false) $rezultat="Не знайдено" ;
else $rezultat="...". "<b>".$myquery. "</b>".substr ($mystring,
$pos+$myquery_len, 50). "...". "(в даному тексті знайдено ".substr_count
($mystring, $myquery).
```

Коли користувач натисне на графічну кнопку пошуку, накладемо умову, за допомогою якої перевіримо чи введений запит в текстове поле для пошуку:

```
if (isset($myquery))
```

Виведемо повідомлення, де вкажемо, що шукаємо:

```
echo "<font face=\"Arial, Helvetica, sans-serif\" size=\"2\">Ви шукаєте:
$myquery</font>";
```

Застосуємо функцію пошуку в файлах вказавши в який файлах шукати і назви розділів:

```
search_text ("Vizutka_mista.php", "Візитка міста", $myquery);
search_text ("Vidgomin_vikiv.php", "Відгомін віків", $myquery);
search_text ("Infrastruktura.php", "Інфраструктура", $myquery);
search_text ("Turuzm.php", "Туризм", $myquery);
search_text ("Mustectvo.php", "Мистецтво", $myquery);
search_text ("Osobustosti.php", "Особистості", $myquery);
search_text ("Meriya.php", "Мерія", $myquery);
```

Наступним кроком буде пошук в таблиці table_news бази даних bd_sait.

Аналогічно до попередніх розділів потрібно підключитися до бази даних, вибрати таблицю table_news, прочитати з неї повідомлення новин, посортувавши повідомлення новин по полю id в порядку спадання, тобто

					ДП.КН 22.459.13.000 ПЗ	Арк.
						53
Змн.	Арк.	№ докум.	Підпис	Дата		

найновіше повідомлення – найбільше значення поля id буде першим:

```
$result = mysql_query("SELECT * FROM $table_base ORDER BY id  
DESC"), а також вчислимо загальну кількість повідомлень:
```

```
$n = mysqli_num_rows(mysqli_query($connect,"SELECT * FROM  
$table_base")).
```

Визначимо довжину рядка-запиту для пошуку:
\$myquery_len=strlen(\$myquery);

Визначимося надалі, що \$count_not_found, буде лічильником знайдених повідомлень, задамо початковий параметр: \$count_not_found=0.

Далі можна вивести коментар, що ми шукаємо і де:

```
echo "<font face=\"Arial, Helvetica, sans-serif\" size=\"2\">в розділі:  
</font>";
```

```
echo "<font face=\"Arial, Helvetica, sans-serif\" size=\"2\"><a  
href=\"index.php\"> Стрічка новин </a></font>";
```

Механізм пошуку в таблиці бази даних буде подібний до пошуку в файлах.

Отримавши загальну кількість повідомлень в таблиці ми можемо задати цикл:

```
for ($i=1;$i<=$n;$i++)  
{ $row = mysqli_fetch_row($result);
```

Переведемо рядок \$row[2] (поле таблиці бази даних з заголовком новини) в нижній регістр:

```
$mystring=strtolower($row[2]);
```

Знайдемо позицію входження рядка-запиту в рядок \$row[2]:

Аналогічно переведемо в нижній регістр і знайдемо позиції знайденого рядка-запиту для поля текст новини і актор новини:

```
$pos_name_news=strpos($mystring, strtolower($myquery));
```

```
$mystring=strtolower($row[3]);
```

```
$pos_text_news=strpos($mystring, strtolower($myquery));
```

					ДП.КН 22.459.13.000 ПЗ	Арк.
						54
Змн.	Арк.	№ докум.	Підпис	Дата		

```
$mystring=strtolower($row[4]);
```

```
$pos_aut_news=strpos($mystring, strtolower($myquery));
```

Накладемо умову, що коли рядок-запит не знайдено ні в полі заголовок новин, ні в полі текст новини та в полі автор новин, то застосуємо збільшимо лічильник \$count_not_found++ і використаємо функції PHP continue, тобто припинимо виконання далі скрипта і повернемося на початок циклу:

```
if ($pos_name_news === false and $pos_text_news === false and  
$pos_aut_news === false) {$count_not_found++; continue; };
```

Якщо ж попередня умова не виконується, тобто коли знайдено позицію входження рядка-запиту хоча б в одному рядку (чи в назві новини, чи в тексті, чи в посиланні на джерело), то виведемо фрагмент тексту рядка де знайдено входження рядка-запиту. Отже, знову накладаємо умову, що коли \$pos_name_news визначена, тобто є визначена позиція входження рядка-запиту, то змінні \$result_find_name_news присвоїмо фрагмент рядка з назви новини, форматом аналогічним як і в функції search_text для пошуку в файлах. А коли \$pos_name_news не визначена, тобто коли не знайдено входження рядка-запиту, то виведемо рядок починаючи з першого символу до \$long, де \$long – кількість символів в фрагменті:

```
if ($pos_name_news === false) $result_find_name_news =  
substr($row[2],0,$long) . "..."; else $result_find_name_news="..." . "<b>" .  
$myquery. "</b>" . substr($row[2], $pos_name_news+$myquery_len, $long). "...".  
"(в даному тексті знайдено ".substr_count(strtolower($row[2]),  
strtolower($myquery)). ")"
```

Аналогічно виведемо результати пошуку в тексті новини (\$pos_text_news, \$result_find_text_news) та в посиланні на автора (\$pos_aut_news, \$result_find_aut_news).

Коли лічильник \$count_not_found рівний = загальні кількості повідомлень \$n виведемо повідомлення про те, що задану інформацію не

					ДП.КН 22.459.13.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		55

знайдено: if (\$count_not_found==\$n) { echo "<p> </p>"; echo "<i>#149 Нічого не знайдено</i>".

Для зміни імені та паролю для доступу до адміністрування стрічки новин та модерування гостьової книги використовуємо скрипт acces_admin.php.

Розглянемо даний скрипт.

Оскільки ім'я та пароль зберігаються в базі даних, то нам необхідно логін, пароль, ім'я сервера бази даних, ім'я таблиці для підключення до бази даних. Даний фрагмент скрипта буде аналогічний до попередніх.

Для авторизації користувача застосуємо авторизацію засобами HTML – через код 401, механізм якої описаний раніше.

Побудуємо форму з таблицею, в комірках якої будуть текстові поля для ведення поточного імені користувача, нового імені і підтвердження його, поточного пароль і нового пароль та його підтвердження. Також на даній формі розмістимо кнопку Зберегти. (рисунок 3.20).

*Для зміни імені користувача і паролю Вам необхідно заповнити всі поля (не більше 30 символів) і натиснути Зберегти

*Всі поля обов'язкові до заповнення (ім'я та пароль повинні бути тільки текстові)

Поточне ім'я користувача	<input type="text"/>	Поточний пароль	<input type="text"/>
Нове ім'я користувача	<input type="text"/>	Новий пароль	<input type="text"/>
		Підтвердження паролю	<input type="text"/>

Рисунок 3.20 – Сторінка для зміни імені та паролю для доступу до адміністрування стрічки новин та модерування гостьової книги

Якщо адміністратор натисне на кнопку Зберегти: if (isset (\$_POST['Submit'])), за допомогою функції str(), яка описана також раніше перевіримо введену інформацію і якщо виконується умова: if (\$_POST ['new_log'] <>"" and \$_POST ['cur_log']==\$aut[1] and md5(\$_POST ['cur_pass'])==\$aut[2] and \$_POST ['new_pass']==\$_POST ['re_new_pass'] and

`$_POST['new_pass']<>""`) , тобто коли заповненні всі необхідні поля, то підключимося до бази даних, виконаємо за допомогою функції PHP SQL-запит: `"UPDATE access SET user = " . $_POST['new_log'] . ", pass = ".MD5($_POST['new_pass'])." WHERE user = " . $_aut[1] . " LIMIT 1;"`, тобто оновимо в таблиці `access` значення полів `user` та `pass`.

Для поінформування про оновлення інформації виведемо повідомлення: `echo "Ім'я та/чи пароль успішно збережені".`

3.5 Тестування роботи веб-сайту м. Збараж.

Мета тестування – перевірка функціонування сайту на відповідність до поставленої задачі. Щоб перевірити веб-сайт, необхідно створити свідомо ситуації, які можуть виникати при роботі з сайтом. Провести аналіз поведінки сайту. При виявленні “багів”, помилок, потрібно знайти ці недоліки, усунути їх та знову провести тестування. Тестування буде проводитися до тих пір, поки сайт буде працювати практично за будь-яких умов.

Для початку протестуємо візуально наш сайт на коректну роботу всіх розділів. Розпочнемо з розділу “Вікно в Збараж” на якому розміщено стрічку новин (рис. 3.21).

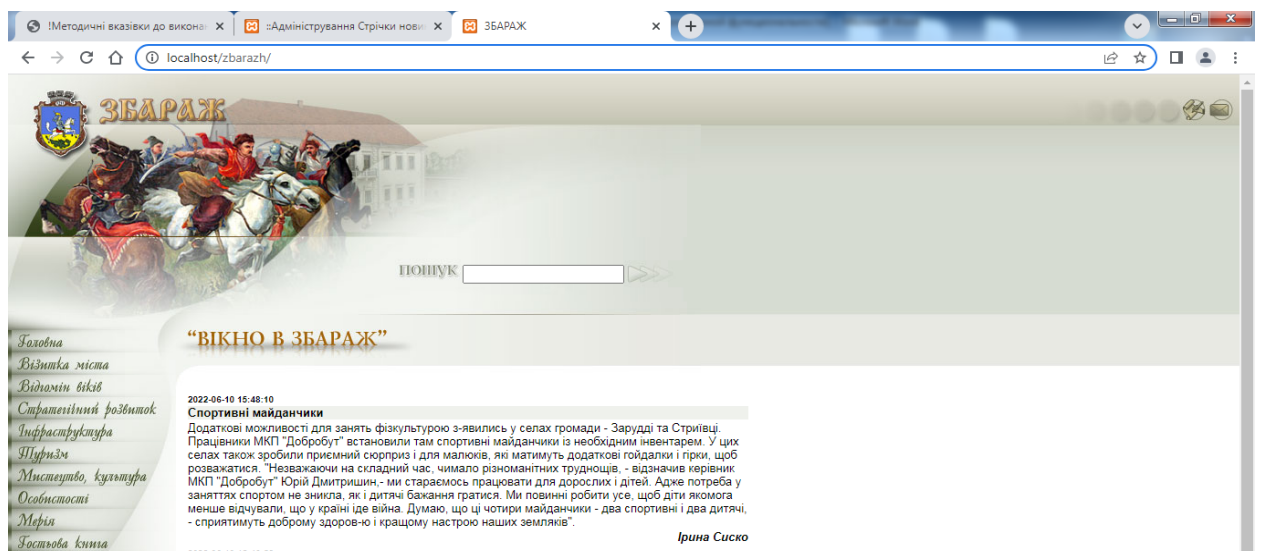


Рисунок 3.21 – Розділ «Вікно в Збараж»

					ДП.КН 22.459.13.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		57

Далі переходимо до розділу “Візитка міста”, де ми бачимо опис символіки міста та інше.

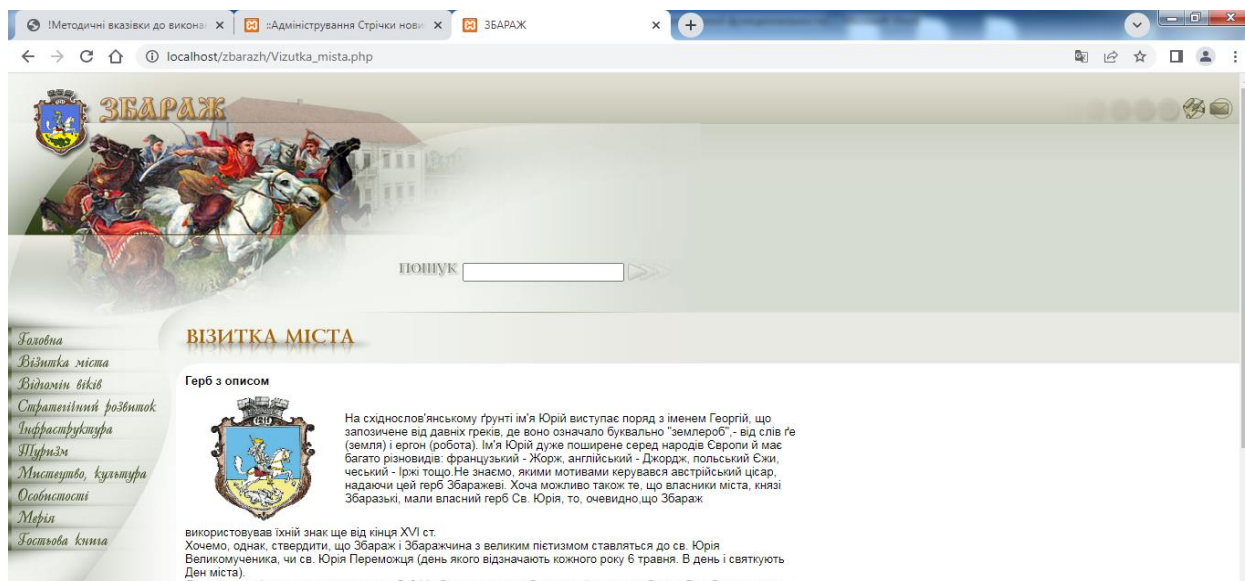


Рисунок 3.22 – Розділ «Візитка міста»

Переглянемо розділ “Відгомін віків”, де розміщено історичні факти про місто. Також звертаємо увагу, що всі елементи сайту відображаються коректно.

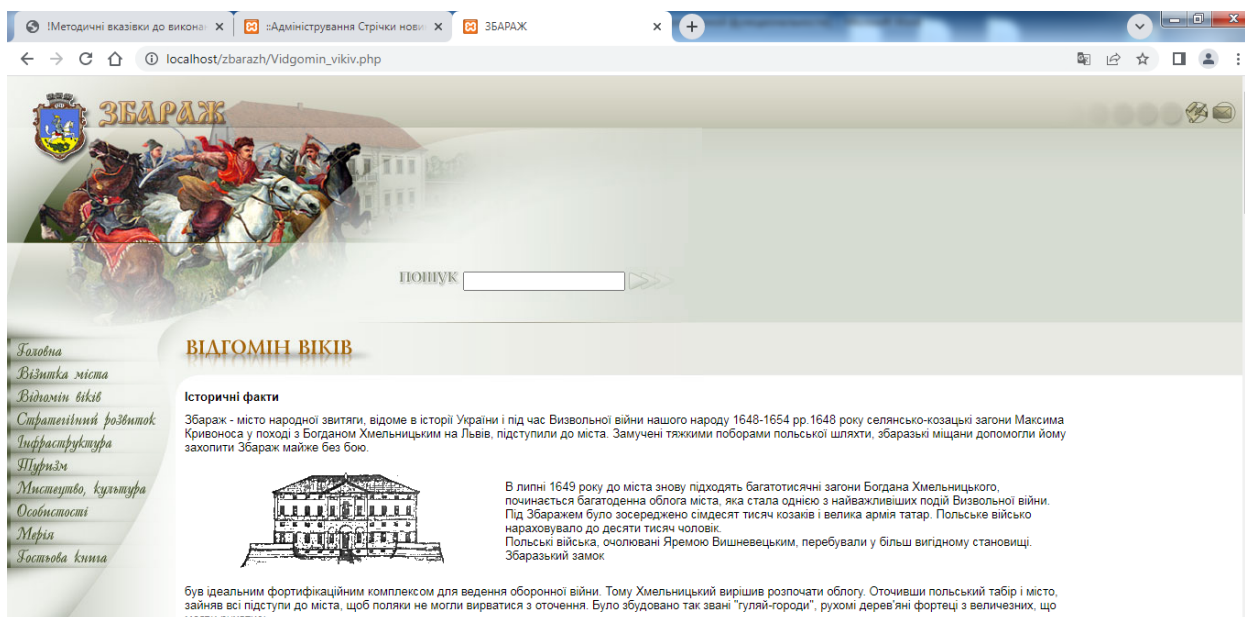


Рисунок 3.23 – Розділ «Відгомін віків»

					Арк.
					58
Змн.	Арк.	№ докум.	Підпис	Дата	

ДП.КН 22.459.13.000 ПЗ

Щодо розділу “Стратегічний розвиток”, то ми бачимо, що все відображається коректно.

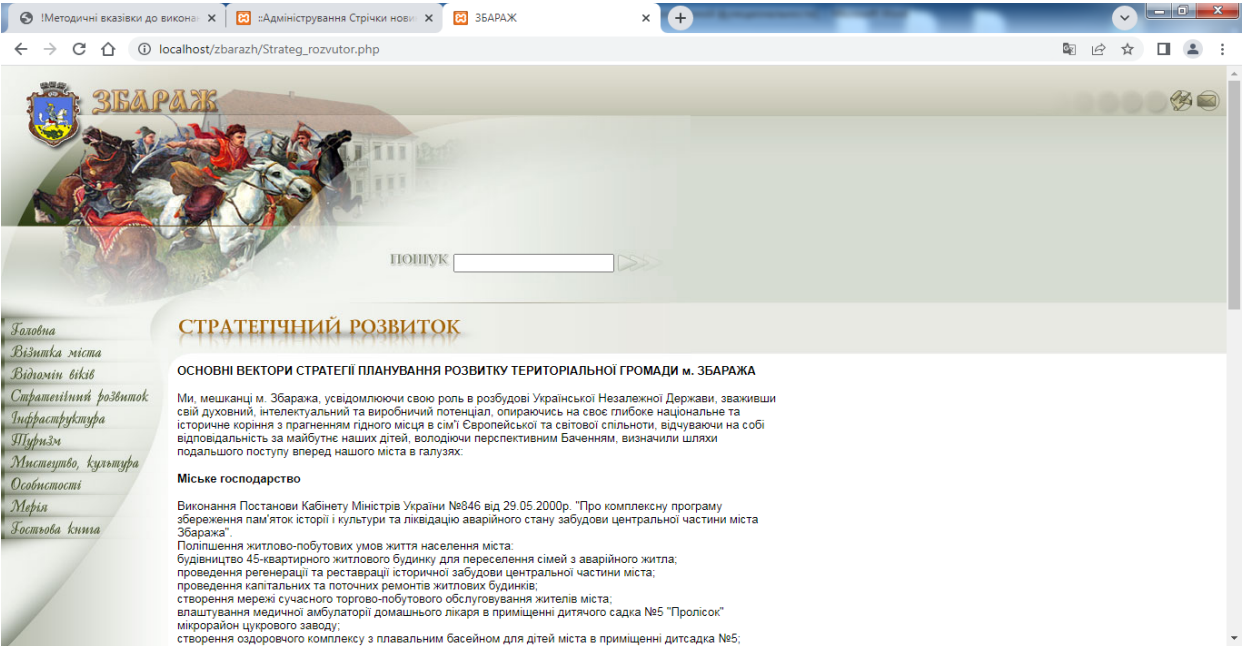


Рисунок 3.24 – Розділ «Стратегічний розвиток»

Розділ “Інфраструктура” є подібним до попереднього розділу, відображається згідно поставленого завдання.

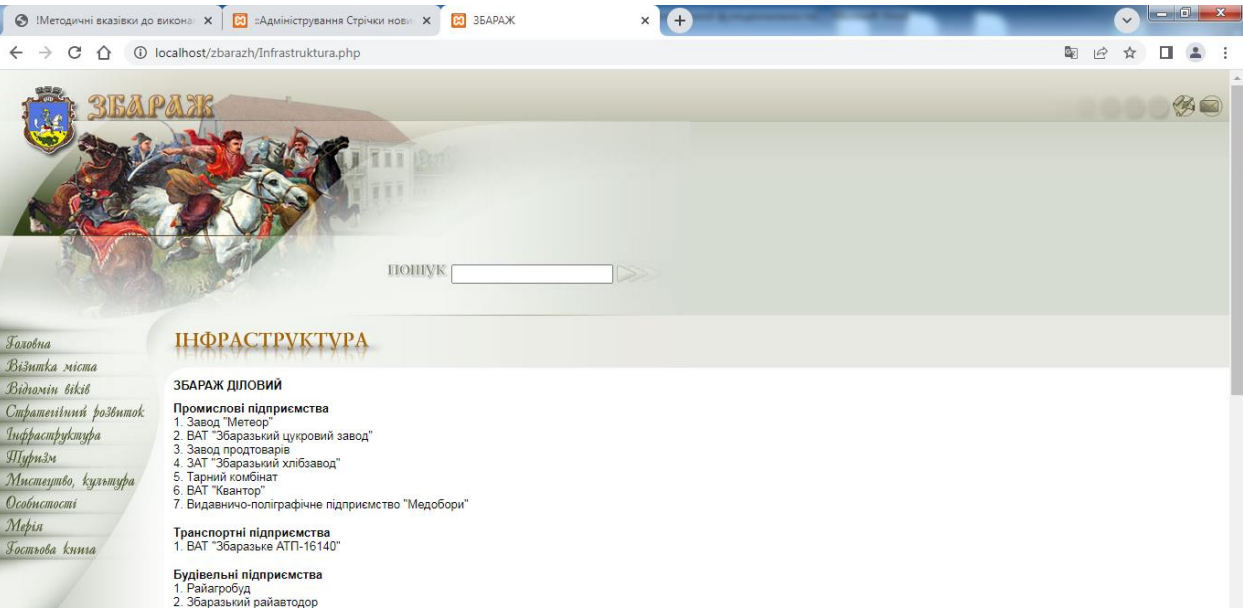


Рисунок 3.25 – Розділ «Інфраструктура»

У розділі “Туризм ” розміщена карта, описаний туристичний потенціал міста. Розділ відображається коректно.

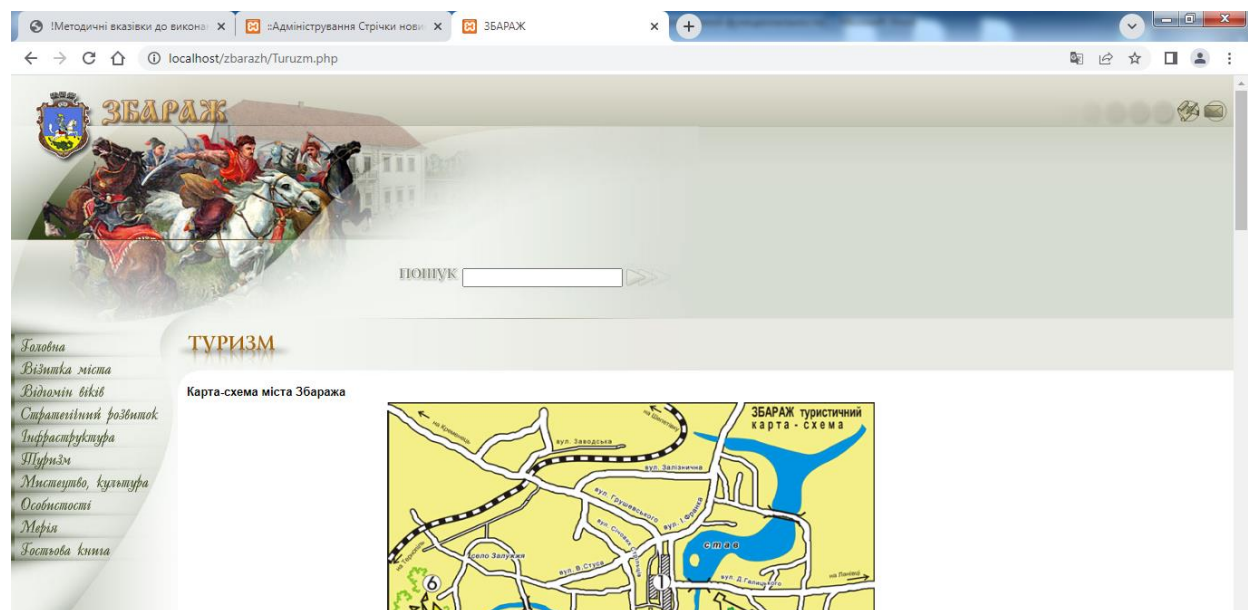


Рисунок 3.26 – Розділ «Туризм»

Переглянувши розділ “Мистецтво, культура”, ми бачимо що все розміщено добре, структура дизайну сайту не порушені.

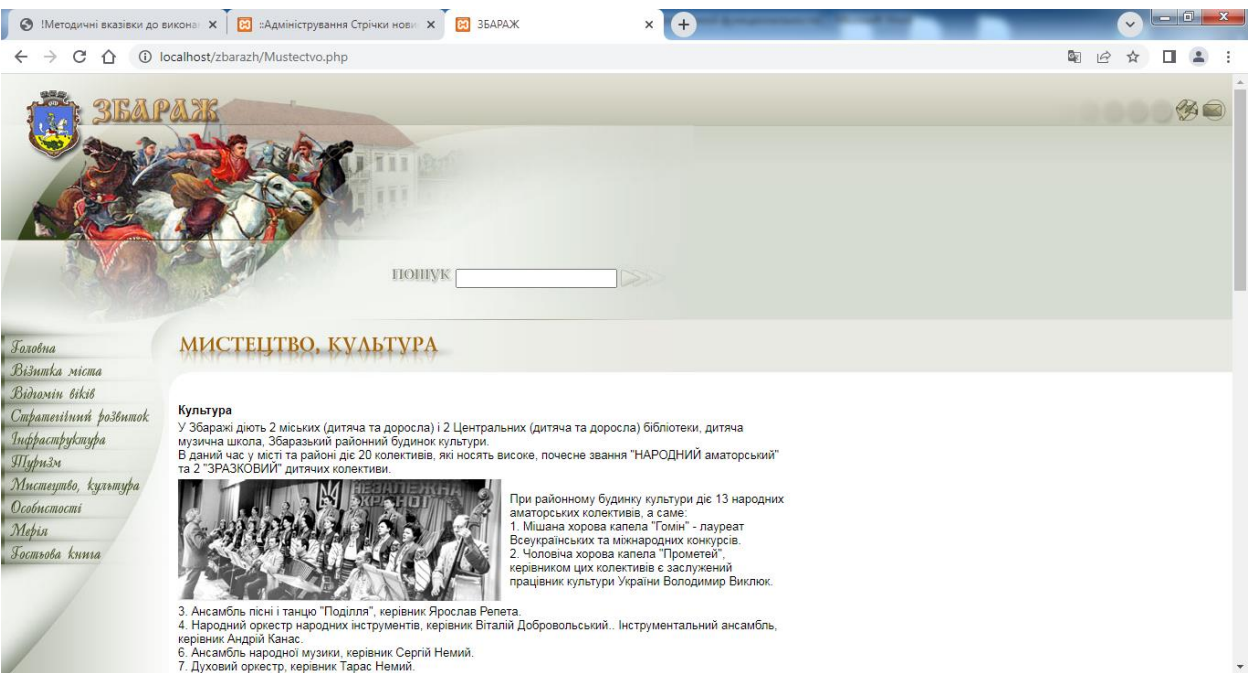


Рисунок 3.27 – Розділ «Мистецтво, культура»

У розділі “Особистості” розказано про життєвий шлях видатних людей Збаражчини. Малюнки і текст відображаються належним чином.

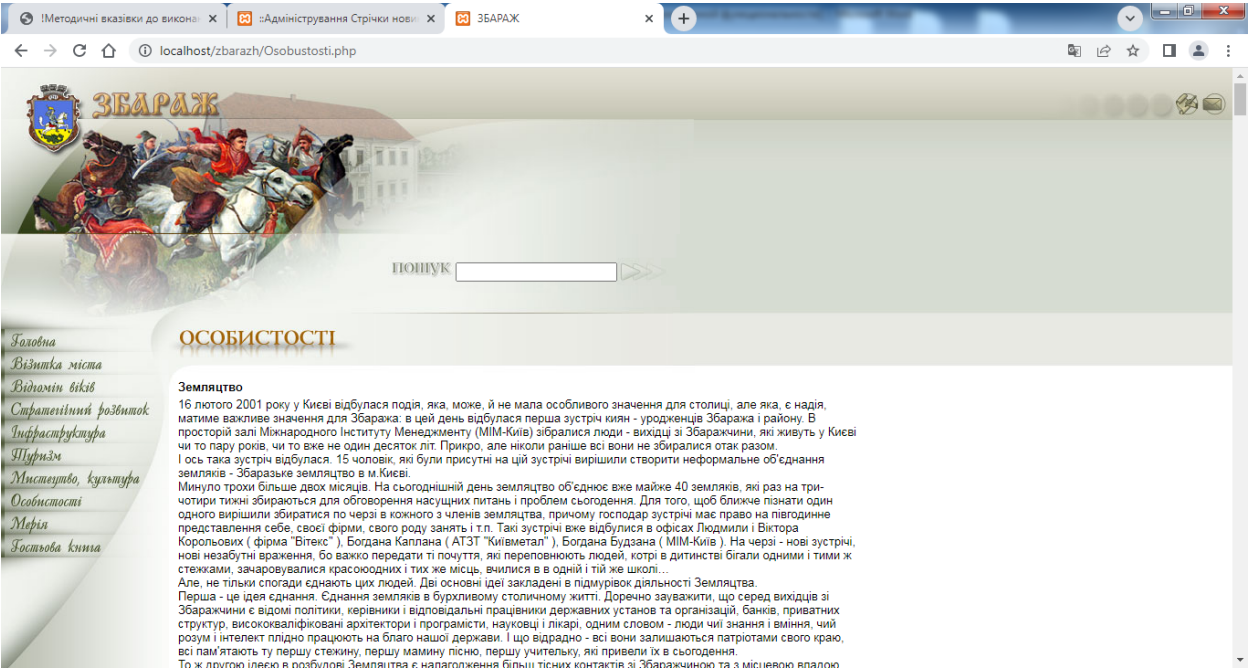


Рисунок 3.28 – Розділ «Особистості»

У розділі “Мерія” вказано керівний склад міської ради та їхні робочі телефони. Розділ відображається коректно. Текст знаходиться у встановлених межах.

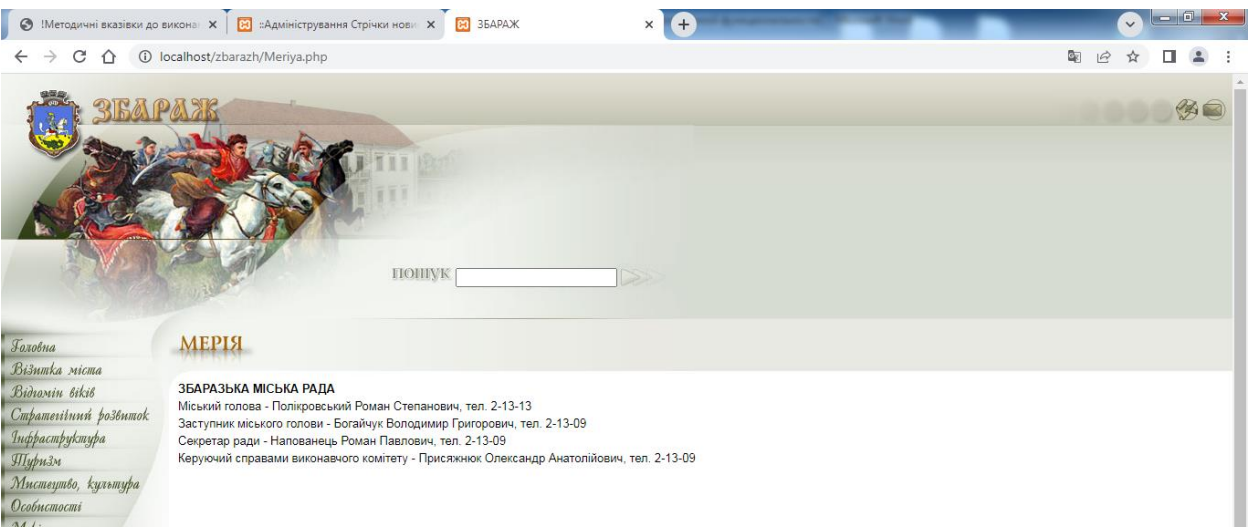


Рисунок 3.29 – Розділ «Мерія»

У розділі Гостьова книга, ми бачимо, що форма введення відгуку відображається правильно. Стрічка відгуків відвідувачів сайту також працює належним чином.

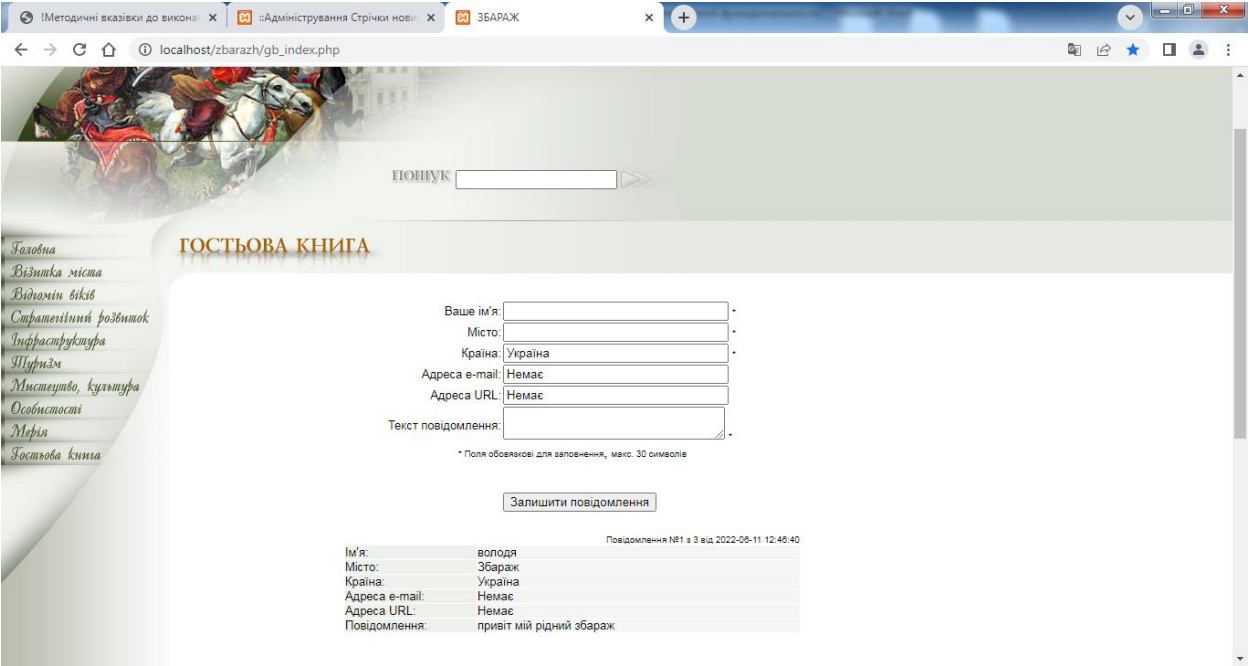


Рисунок 3.30 – Розділ «Гостьова книга»

Далі протестуємо функцію пошуку по сайту, для чого введемо інформацію у рядок пошуку і натиснемо відповідно кнопку (рис. 3.31):

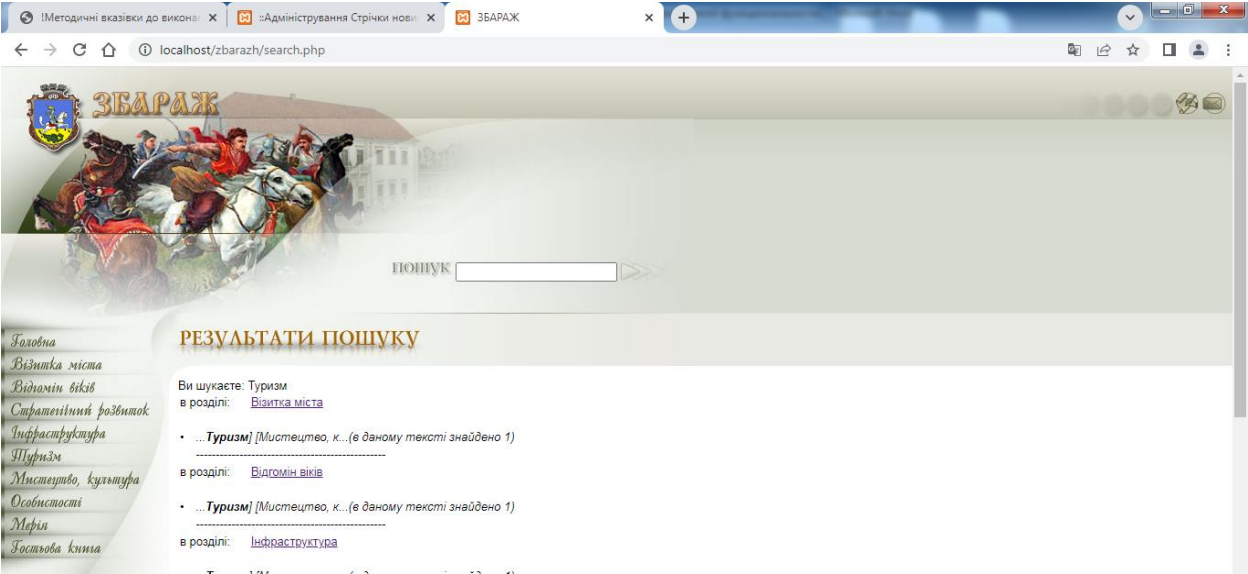


Рисунок 3.31 – Результат пошуку

Варто спробувати змодельовати ситуацію коли в поле вводу запиту не буде введено сам запит, тобто рядок пошуку буде пустий (рис. 3.32):

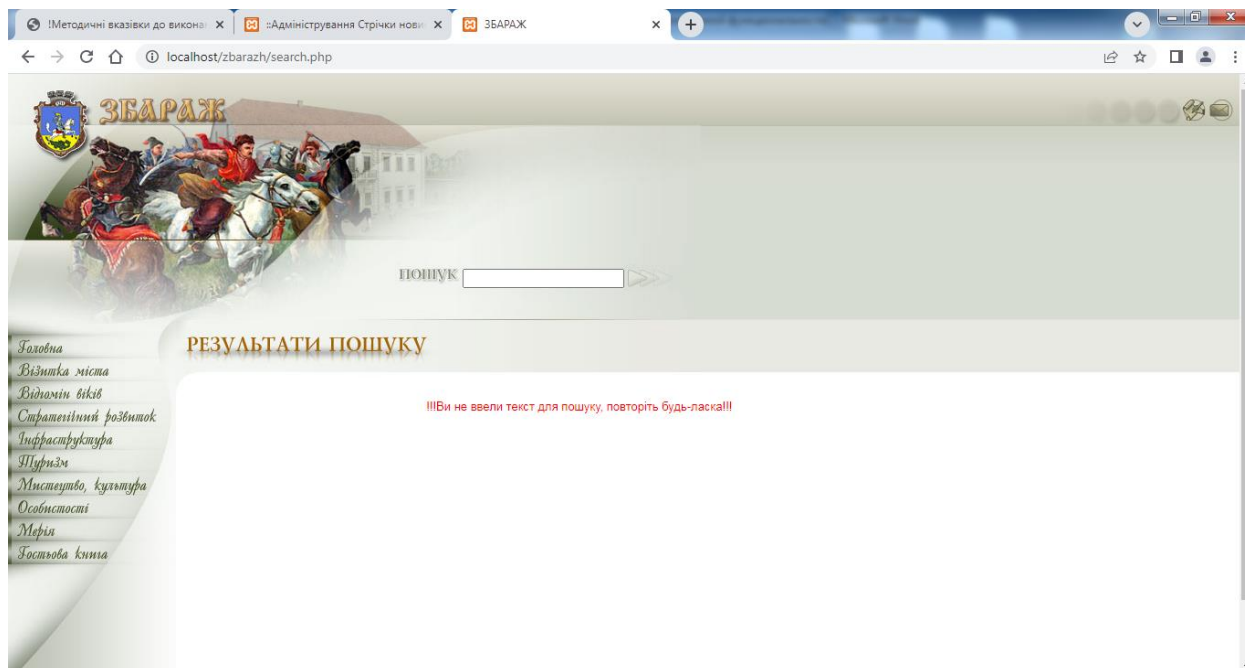


Рисунок 3.32 – Результат пошуку

Наступник кроком тестування — залишимо повідомлення у гостьовій книзі, для чого перейдемо у розділ Гостьова книга, заповнимо поля форми і натиснемо кнопку Залишити повідомлення (рис. 3.33).

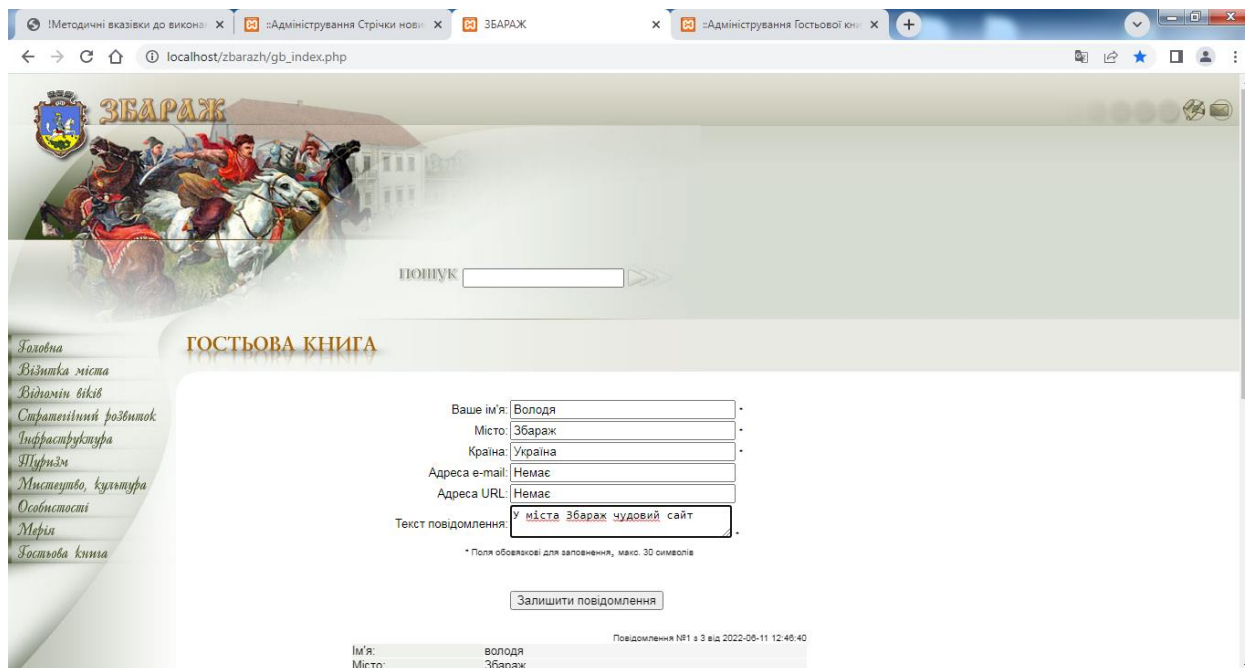


Рисунок 3.33 – Опублікування відгуку в гостьовій книзі

Також змодельюємо ситуацію коли не всі поля введені (рис. 3.34).

Рисунок 3.34 – Опублікування відгуку в гостьовій книзі

Аналогічним способом протестуємо скрипт зміни логіну та паролю до адмінпанелі новин та гостьової книги `acces_admin.php` (рис. 3.35)

Рисунок 3.35 – Зміна логіна та паролю

Змодельюємо коли неправильно введено логін чи пароль (рис. 3.36)

Рисунок 3.36 – Зміна логіна та паролю

Перевіримо панель адміністрування стрічки новин, спробуємо опублікувати нову новину , відредагувати її та знищити опубліковану (рис. 3.37).

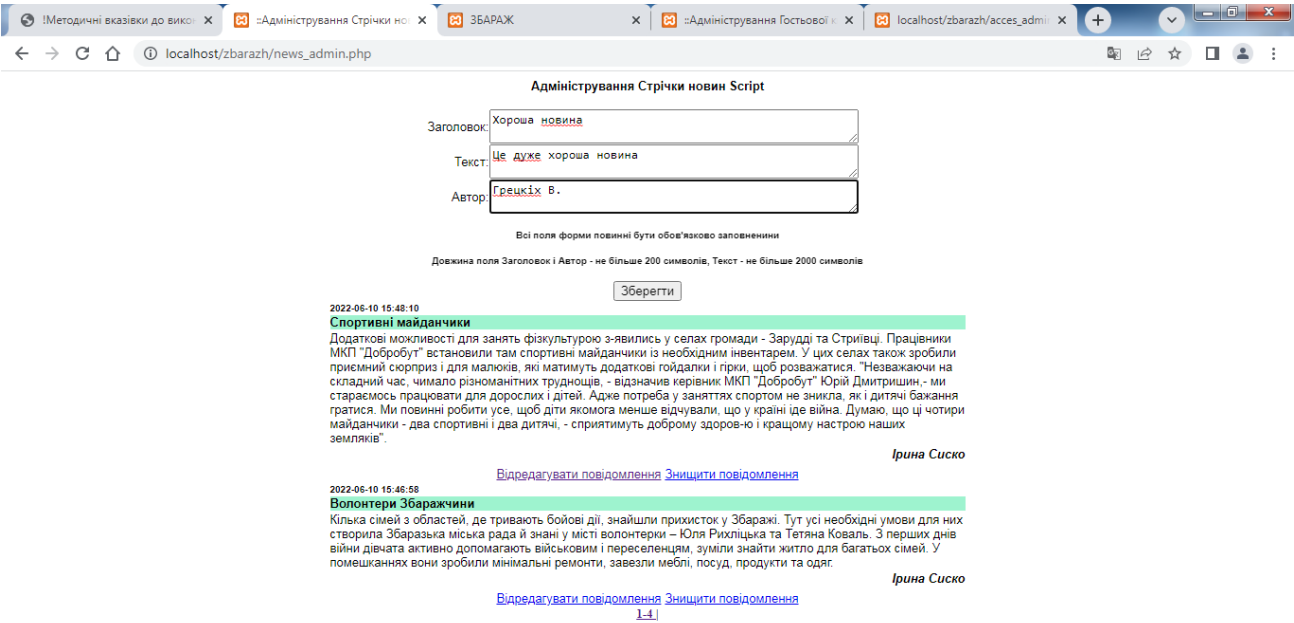


Рисунок 3.37 – Публікація новин

Аналогічно перевіримо панель адміністрування гостьової книги, а саме редагування та знищення вже висвітлений відгуків (рис. 3.38).

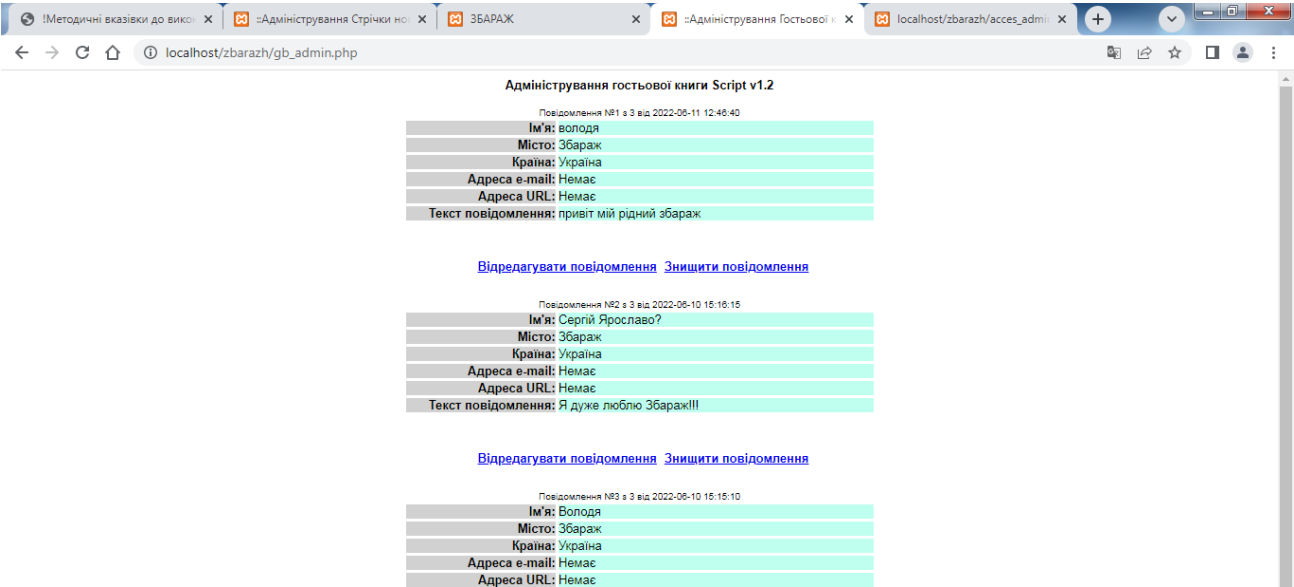


Рисунок 3.38 – Адміністрування гостьової книги

По завершенню даного тестування, можемо зробити висновок, що даний сайт в повній мірі відповідає поставленому завданню, працює коректно та без помилок.

					ДП.КН 22.459.13.000 ПЗ	Арк.
						66
Змн.	Арк.	№ докум.	Підпис	Дата		

4 ТЕХНІКО-ЕКОНОМІЧНЕ ОБҐРУНТУВАННЯ

Основною метою виконання розділу організаційно-економічного обґрунтування дипломного проекту є встановлення економічної доцільності проведення даної розробки [3].

Завданням роботи є розробка інформаційного Web-сайту м. Збаража, на якому буде розміщено історію, стратегічний розвиток, інфраструктуру міста та інше.

Цей проект дозволить проводити інтенсивну рекламну політику міста в електронних ЗМІ в плані висвітлення економічного та туристичного потенціалу міста.

У даному розділі буде проведено розрахунок економічної доцільності розробки та створення вищевказаного Web-сайту міста Збаража.

4.1 Розрахунок норм часу на виконання розробки Web-сайту

Ефективне планування і використання часу має велике значення, тому що коефіцієнт корисної дії залежить від оптимального використання часу.

Для полегшення і структуризації виконання розробки, її поділяють на декілька етапів.

Основні етапи при виконанні розробки Web-сайту наступні:

постановка проблеми;

реалізація алгоритмів;

написання програмного забезпечення;

оформлення документації.

Для оцінки тривалості виконання окремих робіт використовуємо нормативи часу. До таких нормативів відносять тривалість написання операції (команди), яка на може становити:

1-ї операції — 0,5-1,6 год;

5-ти операцій — 8 год (тривалість зміни).

					ДП.КН 22.459.13.000 ПЗ	Арк.
						67
Змн.	Арк.	№ докум.	Підпис	Дата		

Для визначення загальної тривалості проведення розробки програмного продукту доцільно дані витрат часу на виконання окремих стадій (етапів) звести у таблицю 4.1, при цьому витрати часу керівника на виконання окремих стадій (етапів) прийматимемо в межах 5 % сумарних витрат часу інженерів на виконання цих стадій (етапів).

Таблиця 4.1 – Середній час виконання розробки

Номер і назва етапу	Середній час виконання етапу, год.	
	інженер	керівник
Постановка проблеми	24	24
Реалізація алгоритмів роботи	64	10
Побудова програми	80	10
Оформлення документації	16	2
Разом	~184	~46

4.2 Розрахунок витрат на проведення розробки Web-сайту

Розрахунок поточних витрат на проведення розробки програмного продукту проводимо в розрізі таких калькуляційних статей:

- 1) основна заробітна плата (з/п);
- 2) додаткова з/п;
- 3) нарахування на з/п;
- 4) консультаційні витрати;
- 5) матеріали для виконання робіт по НДР (розробки програмного продукту);
- 6) експериментально-виробничі витрати;
- 7) загальновиробничі витрати;
- 8) адміністративні витрати;
- 9) позавиробничі витрати.

Системна обробка інформації щодо обліку праці і заробітної плати є однією з найрізноманітніших, складних і трудомістких ділянок роботи. Як показують дослідження, за трудомісткістю ця ділянка становить майже одну третину від всього обліку по підприємству.

Реалізація на практиці процесу справедливого розподілу матеріальних благ, а також пов'язаний з цим контроль цілком і повністю залежить від вірної організації (на науковій основі) праці і достовірного нарахування заробітної плати. Без цього навряд чи можна установити необхідний контроль за мірою праці і мірою винагородження за неї, за виконання виробничого плану, плану по праці і заробітній платі, а також за рівнем і зростом їх продуктивності.

При системному розв'язанні питання про облік праці і заробітної плати велике значення має умовно-постійна (нормативна, довідкова та інша) інформація, яка в даному разі характеризує переважно постійних виконавців (людей і механізми) та постійні процеси (технологічні операції). Тому у першу чергу зміст по обліку праці і заробітної плати неодмінно повинна входити інформація про виконавців (облік складу працівників).

Для правильної оплати потрібні достовірний і своєчасний облік виробітку та визначення його якості. Облік виробітку і його оплати — найбільш трудомістка ділянка роботи, вона складається з ряду проміжних операцій.

Застосування ПК у вигляді АРМ, а також широке використання умовно-постійної інформації значно поліпшує облік показників про виробіток, що безпосередньо відноситься до обліку результатів і одночасно ці ж данні використовуються для нарахування відрядної заробітної плати. Так здійснюється зв'язок між цими ділянками обліку, а також взаємоконтроль цих показників.

Основна з/п складається із прямої з/п і доплати, яка при узагальнених розрахунках становить 25 %-35 % від прямої з/п. При розрахунку з/п

					ДП.КН 22.459.13.000 ПЗ	Арк.
						69
Змн.	Арк.	№ докум.	Підпис	Дата		

кількість робочих днів в місяці приймаємо рівною 25,4 дні / міс, що відповідає 203,2 год. / міс. Розмір місячних окладів керівника приймаємо 7200 грн. та інженерів — 5000 грн.

Пряма з/п визначається наступним чином:

$$ЗП = (O_i - T_i) / 203,2$$

де O_i — розмір місячних окладів 1-х категорій працівників;

T_i — трудомісткість робіт виконаних працівниками /*-х категорій.

Для інженера:

$$ЗП = (5000 \cdot 184) / 203,2 = 4\,527,5 \text{ (грн.)};$$

Для керівника:

$$ЗП = (7200 \cdot 46) / 203,2 = 1\,629,92 \text{ (грн.)}.$$

Величина доплат визначається наступним чином:

$$ЗП_1 = ЗП \cdot K_1$$

де K_1 — коефіцієнт доплат (0,25-0,35).

Приймаємо коефіцієнт доплат рівним 0,3:

Для інженера:

$$ЗП_1 = 4\,527,5 \cdot 0,3 = 1\,358,25 \text{ (грн.)};$$

для керівника:

$$ЗП_1 = 1\,629,92 \cdot 0,3 = 488,98 \text{ (грн.)};$$

Основна з/п визначається наступним чином:

$$ЗП_o = ЗП + ЗП_1$$

Для інженера:

$$ЗП_o = 4\,527,5 + 1\,358,25 = 5\,885,75 \text{ (грн.)};$$

для керівника:

$$ЗП_o = 1\,629,92 + 488,98 = 2\,118,9 \text{ (грн.)}.$$

Величина додаткової з/п визначається наступним чином:

$$ЗП_\delta = ЗП_o \cdot K_\delta$$

Де K_δ — коефіцієнт додаткової з/п (0,05-0,1).

Приймаємо коефіцієнт додаткової з/п рівним 0,1, тоді для інженера:

					ДП.КН 22.459.13.000 ПЗ	Арк.
						70
Змн.	Арк.	№ докум.	Підпис	Дата		

$$ЗП_{\partial} = 5\,885,75 \cdot 0,1 = 588,57 \text{ (грн.)};$$

для керівника:

$$ЗП_{\partial} = 2\,118,9 \cdot 0,1 = 211,89 \text{ (грн.)}.$$

Витрати, на проведення розробки програмного продукту крім річного фонду заробітної плати включають ще й єдиний соціальний внесок у розмірі 22%: $ЄСВ = (ЗП_{\partial} + ЗП_{\partial}) \cdot 22\%$,

для інженера -

$$ЄСВ = (5\,885,75 + 588,57) \cdot 0,22 = 1\,424,35 \text{ (грн.)};$$

для керівника -

$$ЄСВ = (2\,118,9 + 211,89) \cdot 0,22 = 512,77 \text{ (грн.)}.$$

Таким чином, результати розрахунку заробітної плати та нарахувань на неї зведемо в таблицю 4.2.

Таблиця 4.2 – Зведена відомість витрат на заробітну плату, грн.

	Категорія працівників	Основна заробітна плата, грн.			Додаткова заробітна плата	ЄСВ	Всього витрат на заробітну
		Пряма заробітна плата	Доплати	Всього			
1	Інженер	4527,5	1358,25	5885,75	588,57	1424,35	2128,50
2	Керівник	1629,92	488,98	2118,9	211,89	512,77	798,29
Всього		6157,42	1847,23	8004,65	800,46	1937,12	2926,79

При необхідності, проводячи організаційно-економічне обґрунтування, слід враховувати витрати на консультації. Такі витрати можна врахувати окремою калькуляційною статтею виходячи з реальних цін на певний вид консультаційних послуг. Як правило, при отриманні консультацій витрати рахуються на оплату праці консультантів за певний консультаційний час.

Для розрахунку витрат на консультації, врахуємо, що консультації були надані в обсязі 2 год, вартість їх становить 100 грн.

Витрати на матеріали розраховуються на основі норм їх витрат і відповідних оптових цін.

$$M_3 = \sum_{i=1}^n H_{mi} \cdot C_{oi}$$

де M_3 — затрати на матеріали;

H_{mi} — норма затрат i -х матеріалів;

C_{oi} — оптова ціна за одиницю витрат i -х матеріалів;

P — кількість найменувань матеріалів.

До одержаного результату додамо транспортно-заготівельні затрати на рівні 6-10 % їх преїскурантної вартості.

Результати розрахунку затрат на матеріали зведемо в таблицю 4.3.

Таблиця 4.3 – Визначення величини затрат на матеріали

Найменування матеріальних ресурсів	Одиниця виміру	Норма витрат	Ціна за одиницю, грн.	Затрати матеріалів, грн.	Транспортно заготівельні витрати, грн.	Загальна сума витрат на матеріали, грн.
1. Основні матеріали						
Папір	Пачка	1	210	210	10	220
Тонер	Банка	1	50	50	10	60
CD-RW	Штук	1	4,5	4,5	0	4,5
Ручка	Штук	1	15	15	0	15
Всього						299,5

Експериментально-виробничі витрати визначаються як витрати на машинний час, який є потрібним для виконання необхідного об'єму робіт

виходячи з його вартості за одиницю часу. Вартість роботи на ПК і користування мережею Інтернет встановлюємо виходячи з реальних даних. Приймаємо середній тариф роботи на ПК 1 грн. / год.

Загальновиробничі витрати при укрупнених розрахунках приймаємо на рівні 50 % від суми основної і додаткової з/п інженера, яка була нарахована за роботу по проведенні розробки програмного продукту. Аналогічно визначаються адміністративні витрати, які доцільно прийняти на рівні 50 % від суми основної і додаткової з/п інженера. Позавиробничі витрати приймаємо на рівні 5 % від виробничої собівартості.

Розрахунок поточних витрат на проведення розробки програмного продукту зведемо в таблицю 4.4.

Таблиця 4.4 – Калькуляція собівартості проведення НДР (розробки програмного продукту)

Статті витрат	Витрати, грн.	В % до загальної суми, %
1. Основна заробітна плата	8004,65	42,39
2. Додаткова заробітна плата	800,46	4,24
3. ЄСВ	1937,12	10,26
4. Консультації	200	1,06
5. Матеріали	299,50	1,59
6. Експериментально-виробничі витрати	184,0	0,97
Загальновиробничі витрати	3237,16	17,14
Разом виробнича собівартість	14662,89	77,65
7. Адміністративні витрати	3237,16	17,14
8. Позавиробничі витрати	323,72	1,71
Повна собівартість	~ 18883,77	100%

4.3 Розрахунок ціни розробки Web-сайту

Ціну розробки програмного продукту можна визначити

$$Ц = (C_{np} / N_z + C_{kop}) + П$$

де C_{np} — собівартість розробки програмного продукту), грн.;

N_z — кількість замовлень, од.;

C_{kop} — собівартість копіювання (ксерокопії, компакт-диски, відрядження спеціалістів для запуску та наладки програмного забезпечення тощо), грн.;

$П$ — нормативна величина прибутку (15 % - 30 % від собівартості C_{kop}).

Нехай собівартість копіювання програмного продукту становитиме 100 грн.

Приймемо нормативну величину прибутку рівною 20 % від собівартості. Таким чином ціна розробки програмного продукту складатиме;

$$Ц = (18883,77 / 1 + 1000) + 3776,75 = 23660,52$$

Отже ціна розробки програмного продукту дорівнює 23660,52 грн.

4.4 Оцінка економічної ефективності розробки програмного продукту

При визначенні економічної ефективності необхідно порівняти використовуваний (базовий) програмний продукт і пропонуваний. З допомогою відповідних розрахунків (в разі значної складності використання експертних оцінок) визначається скорочення трудовитрат і (або) машинного часу, і як наслідок — економія коштів при використанні нового програмного продукту.

Для визначення ефективності продукту потрібно розрахувати чисту приведену цінність NPV і термін окупності $T_{ок}$:

$$NPV = \sum ((D_t - B_t) / (1 + i)^i),$$

де D_t — повний дохід за рік (при використанні нового програмного продукту;

B_t — повні витрати за рік і при використанні нового програмного

					ДП.КН 22.459.13.000 ПЗ	Арк.
						74
Змн.	Арк.	№ докум.	Підпис	Дата		

продукту;

t — відповідний рік проекту;

i — дисконтна ставка (0,3).

Економічний ефект від впровадження ПП є очевидний і беззаперечний, оскільки використання його дасть можливість проводити рекламу економічного та туристичного потенціалу міста, залучення інвесторів, а отже покращення життя в місті Збаражі.

					ДП.КН 22.459.13.000 ПЗ	Арк.
						75
Змн.	Арк.	№ докум.	Підпис	Дата		

ВИСНОВОК

В даному дипломному проєкті ми розробили Web-сайт міста Збаража з використанням PHP та бази даних MySQL. Використання цього сайту дасть можливість проводити рекламу економічного та туристичного потенціалу міста в глобальній мережі Інтернет.

В цьому Web-сайті ми розмістили наступні розділи:

- візитка міста (символи міста, запрошення до міста);
- відгомін віків (історія);
- основні вектори стратегічного розвитку міста;
- інфраструктура міста (міське господарство);
- туризм (туристична карта з розміщенням об'єктів туризму);
- мистецтво, культура і спорт;
- особистості (уродженці міста, видатні люди, які жили чи живуть у місті, почесні громадяни);
- мерія (голова, заступник, секретар міської ради та ін.);
- “вікно в Збараж” (події, новини);
- гостьова книга.

Розділи візитка міста, відгомін віків, стратегічний розвиток, інфраструктура міста, туризм, мистецтво, особистості, мерія є статичними сторінками, тобто такими які не будуть часто оновлюватися, а тому виконанні з використанням HTML.

В розділах „Вікно в Збараж” (стрічка новин), гостьова книга, пошук по сайту застосовано динамічне генерування сторінок за допомогою мови PHP та для збереження інформації застосували базу даних MySQL.

Також за допомогою PHP та MySQL реалізовані скрипти адміністрування стрічки новин, модерування гостьової книги, зміни логіну та

					ДП.КН 22.459.13.000 ПЗ	Арк.
						76
Змн.	Арк.	№ докум.	Підпис	Дата		

пароллю для даних скриптів.

Мова PHP та базу даних MySQL обрано тому, що вони розповсюджуються безкоштовно та є мультиплатформовими (Windows, Linux). Перевагою PHP є підтримка популярних при написанні серверних скриптів можливостей інших мов, таких як Java, C, Perl. В PHP є багато вбудованих функцій для написання серверних задач та для роботи з базами даних.

					ДП.КН 22.459.13.000 ПЗ	Арк.
						77
Змн.	Арк.	№ докум.	Підпис	Дата		

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Матов А. Проблеми і можливості входження України до глобального інформаційного суспільства. Комп'ютери у Європі. Минуле, сучасне та майбутнє. Міжнародний симпозіум з історії створення перших ЕОМ та внеску європейців в розвиток комп'ютерних технологій. Київ. 5-9 жовтня 1998 року. с. 316-324.
2. Васкевич Д. Стратегія клієнт-сервера. К.: Диалектика. 1996.
3. Глинський Я.М. Практикум з інформатики. Львів: Підприємство Деол, 1998.- 168 с.
4. ХАМРР: веб-сайт. URL: <https://www.apachefriends.org> (дата звернення: 18.04.2022).
5. Документація MySQL: веб-сайт. URL: <http://www.mysql.com> (дата звернення: 18.04.2022).
6. Методичні вказівки по виконанню організаційно-економічної частини дипломних проектів науково-дослідницького характеру для студентів спеціальності.- Тернопіль, ТДТУ, 2003.
7. Документація PHP: веб-сайт. URL: <http://www.php.net> (дата звернення: 18.04.2022).

					ДП.КН 22.459.13.000 ПЗ	Арк.
						78
Змн.	Арк.	№ докум.	Підпис	Дата		