

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА»
ФАКУЛЬТЕТ ІНФОРМАЦІЙНО-КОМП'ЮТЕРНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

ПОЯСНЮВАЛЬНА ЗАПИСКА

до кваліфікаційної роботи освітнього ступеня «бакалавр»
за спеціальністю 121 «Інженерія програмного забезпечення»
(освітня програма «Інженерія програмного забезпечення»)

на тему:

«Меморіал пам'яті»

Виконала студентка групи ЗІПЗк-22-1
ПЕРЕХВАТОВА Алевтина Олександрівна

Керівник роботи:
СЕМЕРІКОВ Сергій Олексійович

Консультант:
САВІЦЬКИЙ Роман Святославович

Рецензент:
ЧИЖМОТРИЯ Олена Геннадіївна

Житомир – 2024

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА»
ФАКУЛЬТЕТ ІНФОРМАЦІЙНО-КОМП'ЮТЕРНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

«ЗАТВЕРДЖУЮ»

Зав. кафедри інженерії
програмного забезпечення
Тетяна ВАКАЛЮК
«26» лютого 2024 р.

ЗАВДАННЯ
на кваліфікаційну роботу

Здобувач вищої освіти: **ПЕРЕХВАТОВА Алевтина Олександрівна**

Керівник роботи: **СЕМЕРІКОВ Сергій Олексійович**

Тема роботи: **«Меморіал пам'яті»,**

затверджена Наказом закладу вищої освіти від «23» лютого 2024 р., №75/с

Вихідні дані для роботи: **систематизація даних про жертв російських ракетних ударів по Україні.**

Консультанти з бакалаврської кваліфікаційної роботи із зазначенням розділів, що їх стосуються:

Розділ	Консультант	Завдання видав	Завдання прийняв
1	Савіцький Р. С. Семеріков С.О.	27.02.2024	27.02.2024
2	Савіцький Р. С. Семеріков С.О.	27.02.2024	27.02.2024
3	Савіцький Р. С. Семеріков С.О.	27.02.2024	27.02.2024

РЕФЕРАТ

Випускна робота бакалавра складається з інформаційної веборієнтованої системи, яка демонструє наслідки російської агресії в Україні та пояснювальної записки.

Пояснювальна записка до випускної роботи містить текстову частину, викладену на 66 сторінках друкованого тексту, має 10 сторінок додатків. Список використаних джерел містить 38 найменувань і займає 4 сторінки. В роботі наведено 43 ілюстрацій та 3 таблиці. Загальний обсяг роботи – 77 сторінок.

Метою роботи є систематизація та відображення всієї інформації про жертв російських ракетних ударів по Україні, з можливістю додавання інформації про нові удари та нові жертви за допомогою технічних засобів PHP, HTML, CSS, JavaScript та з використанням бази даних MySQL.

У роботі було визначено та детально обґрунтовано основні завдання на розробку проекту та проаналізовано аналоги програмного продукту. Обґрунтовано вибір інструментальних та апаратних засобів та описано взаємодію окремих елементів продукту. Описано структуру інтерфейсу веборієнтованої системи та проведено тестування, результати якого показали справність та надійність продукту.

КЛЮЧОВІ СЛОВА: МЕМОРІАЛ, ВІЙНА, ВЕБСТОРІНКА, ОБСТРІЛИ.

					ІПЗ.КР.Б-121-23-ПЗ			
Зм.	Арк.	№ докум.	Підпис	Дата	Меморіал пам'яті	Літ.	Арк.	Аркуші
Розроб.		Перехватова А.О.						
Керівник		Семеріков С.О.					2	
н. контр.		Тичина О.П.				Житомирська політехніка, група ЗІПЗк-22-1		
Зав. каф.		Вакалюк Т.А.						

ABSTRACT

The bachelor's thesis consists of a web-based information system that demonstrates the consequences of Russian aggression in Ukraine and an explanatory note.

The explanatory note to the thesis contains a textual part set out on 65 pages of printed text and has 34 pages of appendices. The list of references includes 38 titles and covers 4 pages. The work contains 43 illustrations and 3 tables. The total volume of the work is 99 pages.

The purpose of the work is to systematize and display all information about the victims of Russian missile strikes on Ukraine, with the possibility of adding information about new strikes and new victims using PHP, HTML, CSS, JavaScript and the MySQL database.

The paper identifies and substantiates in detail the main tasks for the development of the project and analyzes analogues of the software product. The choice of tools and hardware is substantiated and the interaction of individual product elements is described. The structure of the interface of the web-oriented system is described and testing is carried out, the results of which showed the serviceability and reliability of the product.

KEYWORDS: MEMORIAL, WAR, WEB PAGE, SHELLING.

					III3.KP.Б-121-24-ПЗ	
						3

ЗМІСТ

ВСТУП.....	7
РОЗДІЛ 1. АНАЛІЗ НАПРЯМКІВ ВИКОРИСТАННЯ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ДЛЯ ОБЛІКУ ЖЕРТВ РОСІЙСЬКОГО ВТОРГНЕННЯ....	3
1.1. Постановка задачі	3
1.2. Існуючі аналоги програмного продукту	10
1.3. Інструментальні засоби реалізації програмного продукту та вимоги до апаратних засобів та обґрунтування їх вибору	17
Висновки до розділу 1.....	20
РОЗДІЛ 2. ПРОЄКТУВАННЯ ВЕБОРІЄНТОВАНОЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ МЕМОРІАЛУ ПАМ'ЯТІ	22
2.1. Варіанти використання системи.....	22
2.2. Структура БД та алгоритми роботи системи	24
2.3. Опис реалізації функціоналу системи.....	26
2.4. Опис UML діаграм.....	28
Висновки до 2 розділу.....	38
РОЗДІЛ 3. ІНТЕРФЕЙС ТА ПОРЯДОК РОБОТИ З ВЕБОРІЄНТОВАНОЮ СИСТЕМОЮ МЕМОРІАЛУ ПАМ'ЯТІ.....	40
3.1. Порядок встановлення та налаштування параметрів	40
3.2. Структура інтерфейсу веборієнтованої системи меморіалу пам'яті	43
3.3. Тестування роботи програмного продукту	47
Висновки до 3 розділу.....	52
ВИСНОВКИ	54
ЛІТЕРАТУРА.....	55
ДОДАТКИ.....	59

					ПЗ.КР.Б-121-24-ПЗ	
						5

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

БД — база даних

СУБД — Система управління базами даних

UI — User Interface

HTML — HyperText Markup Language

CSS — Cascading Style Sheets

PHP — Hypertext Preprocessor

SQL — Structured Query Language

UML — Unified Modeling Language

					ІПЗ.КР.Б-121-24-ПЗ	
						6

ВСТУП

Актуальність теми полягає в тому, що Україна переживає різноманітні конфлікти, зокрема військові дії та ракетні обстріли. Створення пам'ятників дозволяє вшанувати пам'ять жертв цих подій.

Зберігання та систематизація інформації про постраждалих від ракетного обстрілу здійснюється для подальшого розслідування події, встановлення відповідальності та надання допомоги сім'ям загиблих.

Меморіал слугуватиме нагадуванням про руйнівні наслідки війни, а роль мирної дипломатії у вирішенні конфліктів сприятиме підвищенню обізнаності громадськості про ці важливі події.

Аналіз даних про ракетні атаки може допомогти виявити та запобігти тому, що призвело до таких ситуацій, і розробити стратегії для запобігання повторенням у майбутньому.

Метою випускної роботи є розробка веборієнтованої інформаційної системи, що дозволяє систематизувати та відображати всю інформацію про жертв російських ракетних ударів по Україні, з можливістю додавання інформації про нові удари та нові жертви.

Встановлена мета обумовлює наступні завдання:

- дослідження потреб та очікування користувачів щодо інформаційної системи, визначення їх вимог та пріоритетів;
- розробка структури бази даних для зберігання інформації про ракетні удари та їхні жертви, забезпечення можливості додавання нових записів та редагування існуючих;
- створення зручного та інтуїтивно зрозумілого інтерфейсу для взаємодії з інформаційною системою, з урахуванням потреб користувачів;
- розробка механізму додавання нових записів про ракетні удари та жертви з можливістю заповнення необхідних полів;
- забезпечення можливості редагування та оновлення існуючих даних про удари та жертви;

- проведення тестування інформаційної системи з метою виявлення помилок та валідація роботи всіх її функцій;

- забезпечення ефективного впровадження розробленої системи та підтримка її функціонування, враховуючи можливість майбутнього розширення та оновлення.

Об'єктом дослідження є систематизація даних про жертв російських ракетних ударів по Україні.

Предметом дослідження є розробка веборієнтованої інформаційної системи меморіалу пам'яті жертв ракетних ударів по Україні.

В роботі було використано методи системного аналізу методи об'єктно орієнтованого проектування, вебпрограмування та дизайну.

Практичне значення отриманих результатів полягає в тому що:

- розроблена інформаційна система буде служити важливим джерелом інформації для дослідників, істориків, журналістів та громадськості загалом, допомагаючи краще зрозуміти та вивчати події, пов'язані з ракетними ударами по Україні;

- меморіал створить можливість вшанування пам'яті жертв та відображення їхньої важливої ролі в історії країни, сприяючи підтримці та співчуттю до сімей постраждалих;

- аналіз даних про ракетні удари може допомогти виявити закономірності та причини таких подій, що в майбутньому сприятиме розробці стратегій для запобігання подібним ситуаціям;

- публічний доступ до інформації про ракетні удари може сприяти залученню уваги міжнародного співтовариства до конфлікту та сприяти мирному вирішенню подібних конфліктів;

- вивчення та аналіз даних про ракетні удари сприятиме підвищенню обізнаності суспільства про наслідки війни та руйнівний вплив воєнних конфліктів на людей та інфраструктуру;

- створення меморіалу та інформаційної системи може привернути увагу міжнародних громадських та політичних організацій до подій, що відбулися, і спонукати до міжнародної реакції та допомоги;

- інформаційна система меморіалу може слугувати платформою для підтримки жертв і їх родин, надаючи їм можливість ділитися своїм досвідом, знаходити спільноту та отримувати підтримку;

- інформаційна система може використовуватися для освіти та виховання молодого покоління, сприяючи розвитку толерантності, визнанню та повазі до різних культур та ідентичностей;

- на період воєнного стану буде можливість додавання нових даних і оновлення існуючої інформації, система буде продовжувати служити як джерело актуальної та достовірної інформації про ракетні удари та їхні наслідки.

Тези на тему «Розробка меморіалу пам'яті: ідея, концепція та реалізація» приведені в Перехватова А.О., Савіцький Р.С. Розробка меморіалу пам'яті: ідея, концепція та реалізація. Тези XI Всеукраїнської науково–практичної конференції молодих учених, 16 травня 2024 року. Київ : «Київський столичний університет ім. Бориса Грінченка», 2024. С.85. [1].

РОЗДІЛ 1. АНАЛІЗ НАПРЯМКІВ ВИКОРИСТАННЯ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ДЛЯ ОБЛІКУ ЖЕРТВ РОСІЙСЬКОГО ВТОРГНЕННЯ

1.1. Постановка задачі

Постановкою задачі є необхідність розробити веборієнтовану інформаційну систему меморіалу пам'яті жертв російських ракетних ударів по Україні.

Проблематика розробки даної інформаційної системи полягає у наступному:

- одна з основних проблем – збір достовірної інформації про ракетні удари та їхні жертви. Умови конфлікту можуть ускладнювати доступ до об'єктивних даних, тому потрібно розробити механізми перевірки та підтвердження інформації;

- конфіденційність даних. Інформація про жертв ракетних ударів може містити конференційні дані, важливо забезпечити високий рівень захисту цих даних від несанкціонованого доступу та зловживання;

- розробка системи повинна враховувати етичні аспекти збору, зберігання та використання інформації про жертв ракетних ударів, такі як повага до приватності та використання даних у відповідності з міжнародними нормами та стандартами;

- забезпечення точності та об'єктивності зібраних даних, щоб уникнути спотворення фактів та підвищити довіру до інформаційної системи. Оскільки агресор веде активну інформаційну війну проти України [2];

- конфлікт має міжнародний характер, інформаційна система повинна бути здатна взаємодіяти з різними мовами та культурними контекстами, а також сприяти співпраці з міжнародними громадськими та політичними організаціями;

Метою системи є створення централізованого та доступного джерела інформації про жертв ракетних ударів українського конфлікту з метою вшанування пам'яті загиблих, надання підтримки їхнім родинам та сприяння дослідженню та аналізу подій, що стосуються цього конфлікту.

Завданнями системи є:

- автоматизація процесу збору та обробки інформації про ракетні удари та жертви, що дозволить збільшити швидкість та ефективність реагування на події;
- розробка аналітичних засобів для вивчення та аналізу динаміки ракетних ударів, виявлення тенденцій та патернів, що сприятиме у виявленні ризикових зон та попередженні майбутніх подій;
- забезпечення можливості інтеграції з іншими інформаційними джерелами, такими як засоби масової інформації, соціальні мережі та організації з прав людини, для отримання додаткових даних та підтримки досліджень;
- розробка механізмів сповіщення та системи оповіщення про нові удари та жертви для оперативного реагування та надання допомоги потерпілим;
- забезпечення можливості створення різноманітних звітів та статистичних аналізів для інформування громадськості, дослідників, правозахисників та органів влади про ситуацію з ракетними ударами та їхніми наслідками;
- впровадження механізмів забезпечення якості даних та перевірки їх достовірності для забезпечення надійності та об'єктивності інформації, що надається користувачам;
- підтримка інтерактивного взаємодії з користувачами, включаючи можливість залишати коментарі, відгуки та подання власних даних про ракетні удари та їхні наслідки;
- забезпечення відкритого доступу до даних для наукових досліджень, аналізу та моніторингу ситуації, а також для створення інструментів публічного освітлення проблеми ракетних ударів та їхніх наслідків.

Предметна область, яку обрано для випускної роботи – це розробка веборієнтованої інформаційної системи меморіалу пам'яті жертв ракетних обстрілів України.

Для того, щоб розробити веборієнтовану інформаційну систему було прийнято рішення розглянути хронологію російсько-українського збройного конфлікту, що розпочався з 20-го лютого 2014 року, обґрунтування було наведено шляхом аналізу всієї хронології конфлікту, а саме розглянуто збройне

вторгнення у Крим, війну на сході України, а також повномасштабне вторгнення в Україну.

Було розглянуто наслідки окупації Криму та Донбасу та наслідки повномасштабного вторгнення в Україну, яке триває й наразі. Описано економічні, екологічні, прямі та непрямі наслідки всього конфлікту, що відбувається впродовж 10-ти років на території України, висвітлено реакцію міжнародного суспільства на події.

Російське збройне вторгнення у Крим розпочалося 20-го лютого 2014 року, шляхом того, що російські війська без знаків розрізнення вторглися на територію Криму, що призвело до окупації півострова. Це стало першою частиною збройного конфлікту між агресором та Україною. Потім розпочалася війна на сході, відразу після окупації Криму [3].

Отже, агресивна окупація Криму Російською Федерацією становить порушення міжнародного права та загрозу суверенітету України, що порушує основи сучасного міжнародного порядку. Цей вчинок свідчить про відкрите порушення норм міжнародного права та створює небезпечний прецедент.

У відповідь на незаконну окупацію Криму, Російська Федерація була виключена з «великої вісімки», і провідні країни світу запровадили політичні та економічні санкції проти неї [4].

Економічні наслідки мали наступні характеристики:

1. Автономна залежність: Незважаючи на статус автономії, Крим є тісно пов'язаним економічно з Україною. Маючи лише 1.5-1.6% в українській міжнародній торгівлі. Крим створював 3.6-3.7% ВВП України при 4.3% населення. Основні галузі: обробна промисловість, торгівля та транспорт. Проте регіон був дотаційним - 40% бюджету. Дотації від Росії оцінюються в 3 млрд. доларів на рік. Крим отримує 80% води та електроенергії з України. Туристи з України складала 70-75%, а з Росії - 25% [5].

Скорочення бажаючих відпочити в Криму - як українців, так і росіян, частка яких в турпотоці оцінюється приблизно в 25%, не викликає сумнівів [6].

					ІПЗ.КР.Б-121-24-ПЗ	
						5

Основний висновок наступний: Крим економічно та технічно сильно залежить від України, проте зворотної залежності не спостерігається.

2. Прямі втрати: Україна втратила доступ до чорноморських родовищ газу[7]. Блокада пропуску торговельних кораблів через Керченську протоку негативно вплине на експорт аграрної і металургійної продукції, які активно використовують Маріупольський порт [8].

3. Опосередковані втрати: Кримська криза призвела до втрат у торгівлі з Росією, яка становила 24% експорту України у 2013 році.

З огляду на ці фактори, Україна має планувати можливі наслідки, такі як зміни у виробничих процесах та ринкових стратегіях. Наприклад, зміцнення торговельних відносин з іншими країнами та пошук нових ринків збуту можуть допомогти зменшити залежність від Росії [9].

Війна на Донбасі, розпочата у 2014 році, мала серйозні наслідки [10]. Донбас - промислово-енергетичний центр з великими ресурсами, але бойові дії спричинили екологічні та промислові втрати. Багато підприємств, включаючи хімічні заводи та електростанції, постраждали від обстрілів. Внаслідок пожеж на природних територіях знищено багато гектарів лісів та степів. На окупованих територіях закриваються шахти і заводи, що призводить до екологічних катастроф.

Серед багатьох промислових підприємств, що постраждали внаслідок бойових дій, особливо виділилися ті, які представляють найбільше екологічне небезпеку.

Деякі природно-заповідні об'єкти отримали пошкодження в результаті обстрілів. Серед них національні парки, такі як Святі гори, Меотида, відділення Українського степового заповідника Кальміуське, Крейдяна флора, а також регіональні ландшафтні парки Донецький кряж і Слов'янський курорт. Деякі об'єкти природно-заповідного фонду стали мішенню для фортифікаційних споруд, навіть без прямої участі у бойових діях, наприклад, національний парк Дворічанський і філія Українського степового заповідника — Крейдяна флора.

Під час бойових дій внаслідок обстрілів виникали пожежі на хімічних підприємствах, таких як Донецький казенний завод технічних виробів, Донецький завод «Точмаш», Лисичанський нафтопереробний завод, Ясинуватський коксохімічний завод та Авдіївський коксохімічний завод. Вплив цих пожеж на навколишнє середовище був катастрофічним.

Ситуація з екологічним моніторингом на територіях, що перебувають під окупацією, настільки критична, що постів контролю якості поверхневих вод та атмосферного повітря в ряді міст, таких як Донецьк, Макіївка, Горлівка, Єнакієве, Луганськ і Алчевськ, не функціонують. Діяльність Державних екологічних інспекцій паралізована, а радіаційний фон не контролюється.

Наприклад, з 39 постів Сіверськодонецького басейнового управління водних ресурсів в Донецькій області і 24 в Луганській, зараз працюють лише 20 і 6 відповідно. Така ж ситуація із постами контролю якості атмосферного повітря обласних гідрометеоцентрів: у Донецькій області працює 11 з 25, а в Луганській - 4 з 11 постів [11].

Через пожежі та безконтрольні вирубування захисних лісосмуг швидко відбувається деградація сільськогосподарських земель. Захоплені території можуть просто існувати протягом певного періоду, руйнуючи наявні ресурси, і не забезпечуючи їх відновлення.

Російське вторгнення в Україну почалося 24 лютого 2022 року. Конфлікт триває, включаючи розірвання Мінських угод, вторгнення російських військ та їх витіснення з деяких областей України. Росія порушила міжнародне гуманітарне право, що підтверджено розслідуванням Міжнародного кримінального суду в Гаазі, включаючи злочини проти людяності та геноцид українців.

Російські ракетні обстріли руйнують цивільні об'єкти, використовуючи заборонені види зброї. Це порушує міжнародні угоди та обмежує судноплавство в Азовському морі. Війна спонукала Фінляндію та Швецію розглянути вступ до НАТО, а Україна зміцнює євроатлантичну інтеграцію, створюючи нові формати співпраці, такі як «Малий альянс» [12].

					ІПЗ.КР.Б-121-24-ПЗ	
						7

Вторгнення призвело до політизації світової економіки, загострення геополітичних відносин, і вразливості глобалізації. Політичні чинники тепер домінують у прийнятті економічних рішень. Китай шукає альтернативні маршрути для вантажів, обходячи Росію. Санкції проти Росії мають обмежений ефект [13].

Цивільний захист стає критично важливим для України. За даними ООН, Україна втратила 27 149 цивільних осіб, включаючи 9 614 загиблих і 17 535 поранених [14].

Російське вторгнення створило широкий спектр загроз для мирного населення, включаючи напади з повітря, землі і води. Російські ракети, бомби, артилерія та безпілотники щодня атакують українців, включаючи цілі далеко за лінією фронту. У такому контексті дослідження особливостей публічного управління системою цивільного захисту стає вельми актуальним, оскільки це важлива складова у протидії військовій агресії путінського режиму проти України [15].

Система оповіщення про повітряні напади в Україні успішно працювала на всій території країни, за винятком районів, що прилягають до фронту. Подальше вдосконалення цієї системи дозволило створити механізми оповіщення громадян за допомогою мобільних телефонів, ЗМІ та сирен.

Високий рівень цифровізації державних послуг дозволив уряду, місцевим органам влади та іншим постачальникам критичних послуг ефективно функціонувати під час конфлікту. Стабільність мобільного зв'язку та інтернет-мережі відігравала ключову роль у рятувальних операціях та координації цивільного захисту. У той же час, приватні постачальники швидко забезпечили альтернативні джерела живлення, щоб забезпечити безперервний доступ до зв'язку [16].

Проте вторгнення показало, що протиповітряна оборона є надзвичайно важливою для захисту цивільного населення та критичної інфраструктури, а Україна виявилася недостатньо підготовленою у цьому плані. Єдиним значним густонаселеним районом, який достатньо захищений від російських атак, є

столиця Київ, де система протиракетної оборони може забезпечити деякий захист. Однак навіть у Києві це не гарантує повного захисту, оскільки падаючі уламки все ж загрожують місту та його інфраструктурі [17].

Крім того, Україні не вдалося зберегти критичну енергетичну інфраструктуру, яка стала головною мішенню для російських ракетно-дронових атак. Протягом усього конфлікту російські військові наносили удари по цивільній інфраструктурі електропостачання, опалення та водопостачання, що призводило до масових тимчасових відключень електроенергії.

Особливою проблемою стала фактична неготовність системи укриттів від бомбардувань, які є важливим елементом цивільного захисту. Планування цивільного захисту в Україні було спрямовано переважно на евакуацію та порятунок у тимчасових надзвичайних ситуаціях, а не на створення укриття для захисту та виживання під час війни. Це призвело до того, що більшість захисних споруд перебували в аварійному стані, затоплені або позбавлені необхідного обладнання, що робило їх непридатними для використання [18].

Опис задач, що автоматизуються

До задач, що автоматизуються відносяться в рамках системи збору та аналізу даних про жертв ракетних ударів, можна віднести наступні:

- збір та зберігання інформації даних про жертв ракетних ударів з різних джерел, такі як офіційні звіти, засвідчені свідчення очевидців, новинні статті тощо. Ця інформація може включати імена загиблих, дати та місця інцидентів, додаткові деталі про їхнє життя та події;

- система може автоматично класифікувати дані за різними критеріями, такими як місце події, кількість жертв, тип ракет тощо. Це допоможе користувачам швидко знаходити інформацію, яка їх цікавить;

- система може надавати візуальні засоби для відображення даних у зручному для сприйняття форматі, такі як графіки, діаграми чи карти. Це допоможе користувачам краще розуміти структуру та характеристики даних;

- користувачі можуть мати можливість додавати нові події ракетних ударів до системи, заповнюючи відповідну інформацію про ці події;

- система може зберігати відомості про кожного загиблого, такі як ім'я, фотографії, вік, додаткові деталі про їхнє життя та інші релевантні дані.

Автоматизація цих задач дозволить значно підвищити ефективність збору та аналізу даних, зменшити ймовірність помилок та забезпечити оперативний доступ до релевантної інформації для всіх зацікавлених сторін.

1.2. Існуючі аналоги програмного продукту

До існуючих аналогів варто віднести два онлайн-меморіали:

- 1) Платформа пам'яті Меморіал [19].
- 2) Мартиролог «Закриті очі» [20].

Огляд платформи пам'яті Меморіал

Платформа пам'яті Меморіал заснована у 2022 році через повномасштабне російське вторгнення в Україну, її головна сторінка наведена на рис. 1.1.

Дана платформа збирає всю інформацію про загиблих через російську агресію.

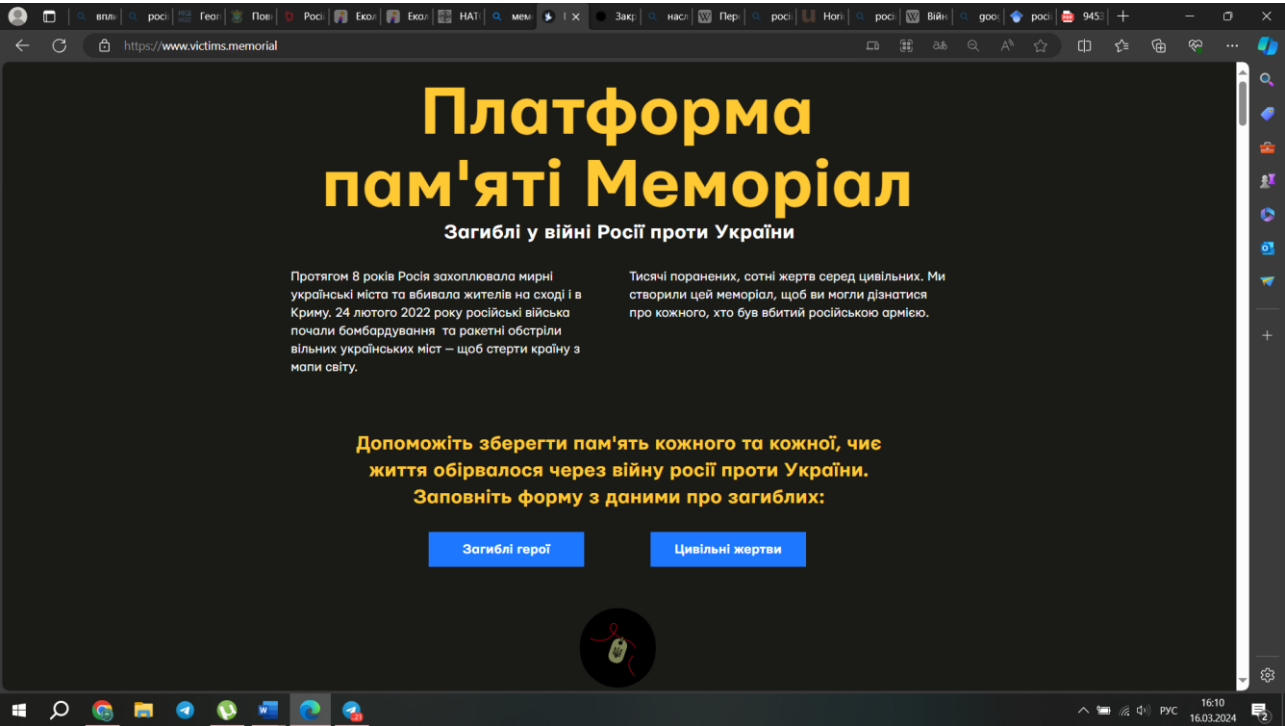


Рисунок 1.1. – Головна сторінка платформи пам'яті Меморіал

Її функціонал полягає в тому, що можна додати свою інформацію, як про загиблого героя, так і про цивільну жертву. Також по категоріям поділено те, що до загиблих відносяться, як герої, так і цивільні, і ті, хто наближав перемогу та загиблі діти, та люди, які загинули під час війни (рис.1.2.-1.6). Розділено на населенні пункти (рис.1.7), де з тих, хто є у певній категорії загинув, а також, можна вказати прізвище та дату загибелі певної людини або героя та здійснити пошук.

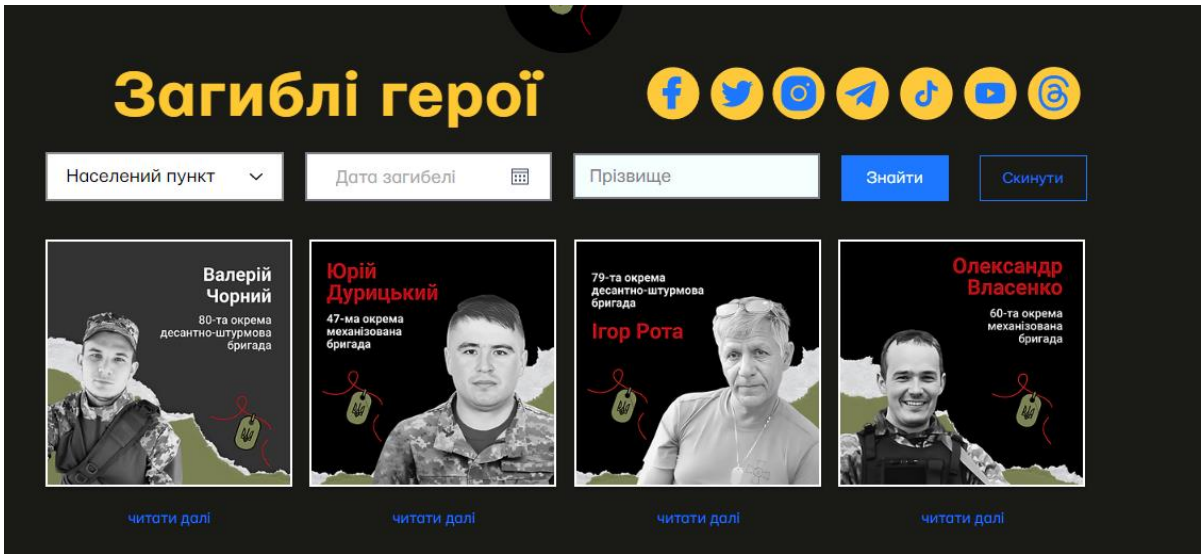


Рисунок 1.2. – Категорія «Загиблі герої»

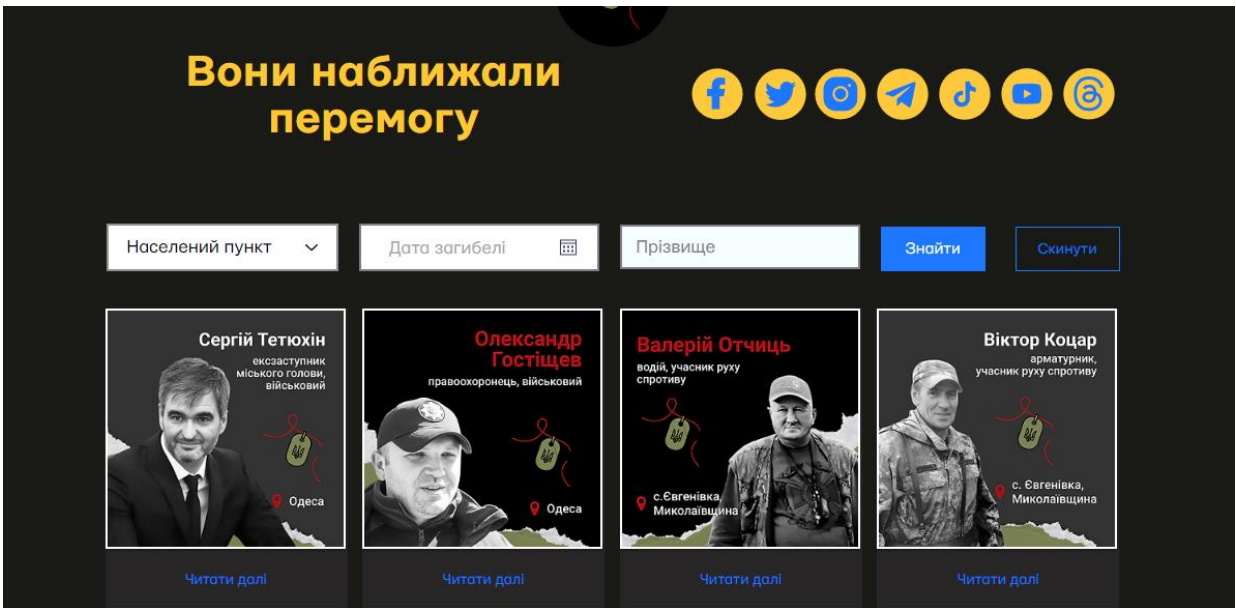


Рисунок 1.3. – Категорія «Вони наближали перемогу»

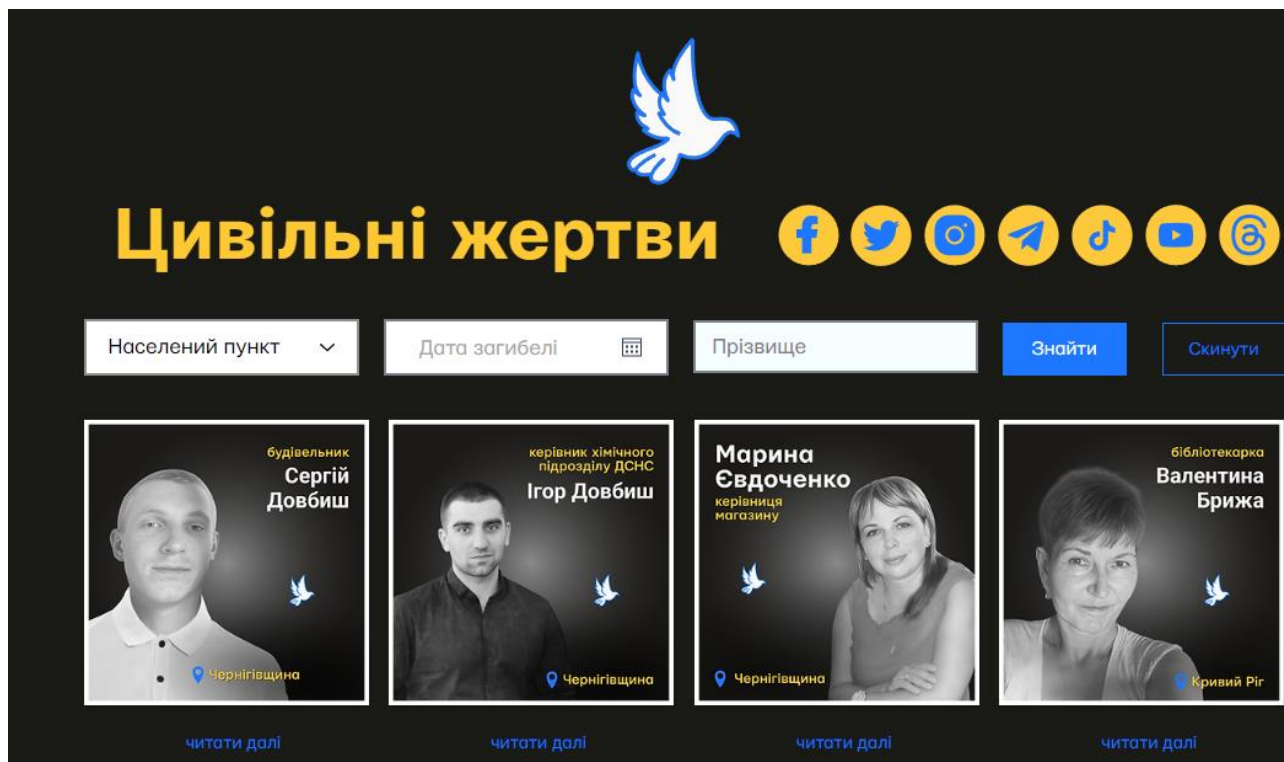


Рисунок 1.4. – Категорія «Цивільні жертви»

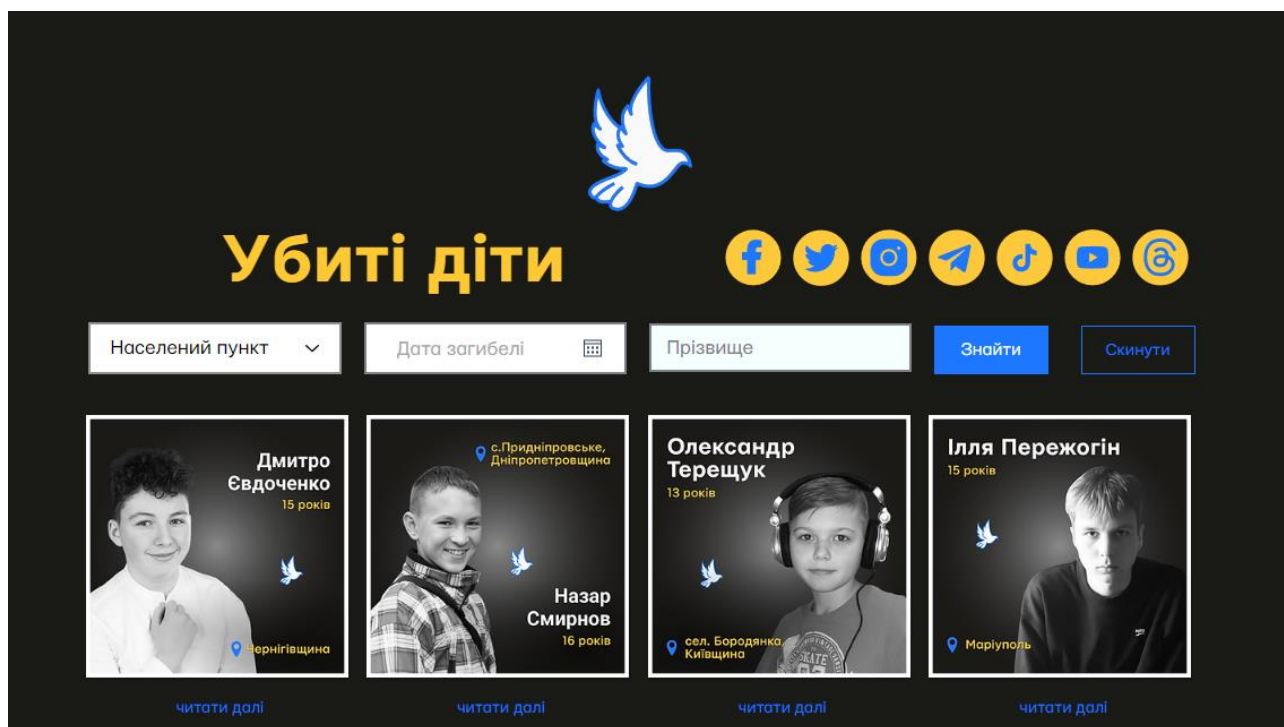


Рисунок 1.5. – Категорія «Убиті діти»

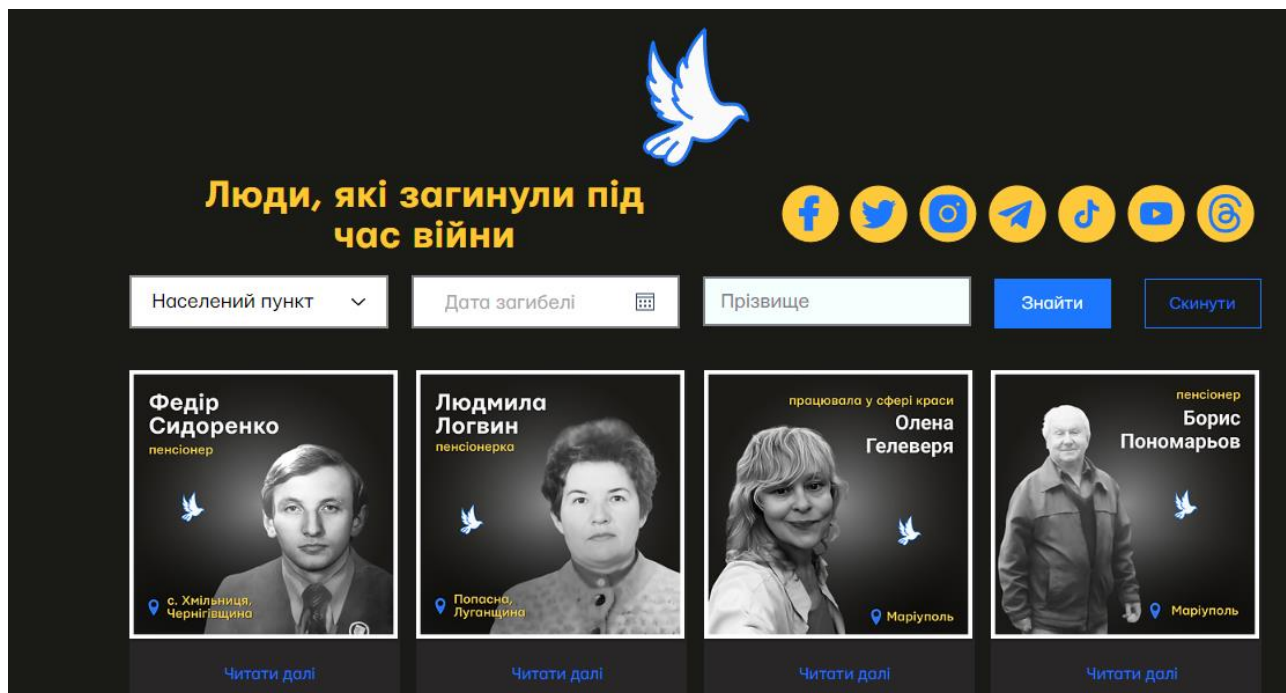


Рисунок 1.6. – Люди, які загинули під час війни

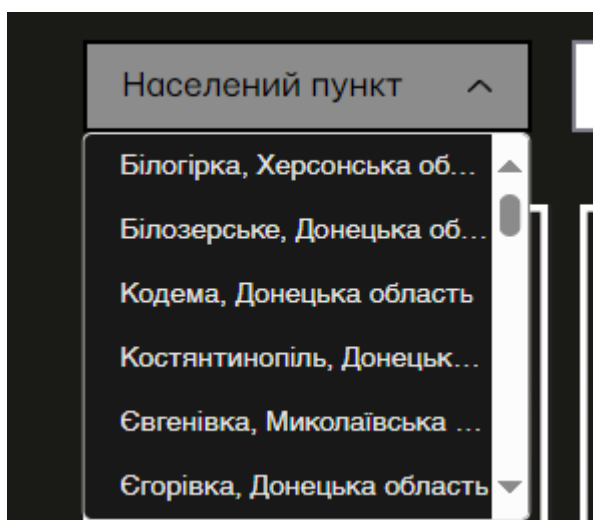


Рисунок 1.7. – Вибір населених пунктів, де шукати загиблого

Також на платформі є тексти пам'яті загиблих (рис. 1.8.), фотопроєкт, що присвячений коханим загиблих героїв (рис. 1.9.), досвід пам'яті (рис. 1.10) та медіаархів пам'яті (рис.1.11).

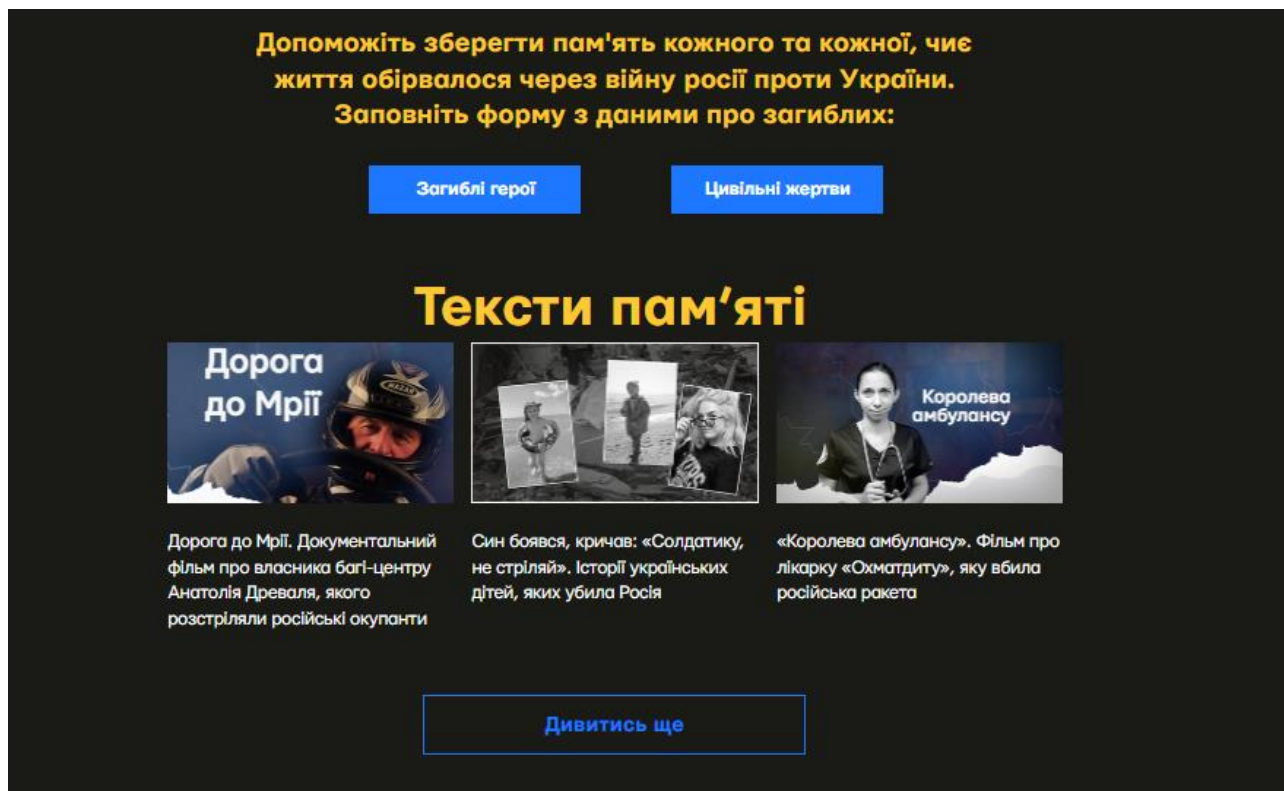


Рисунок 1.8. – Тексти пам'яті загиблих

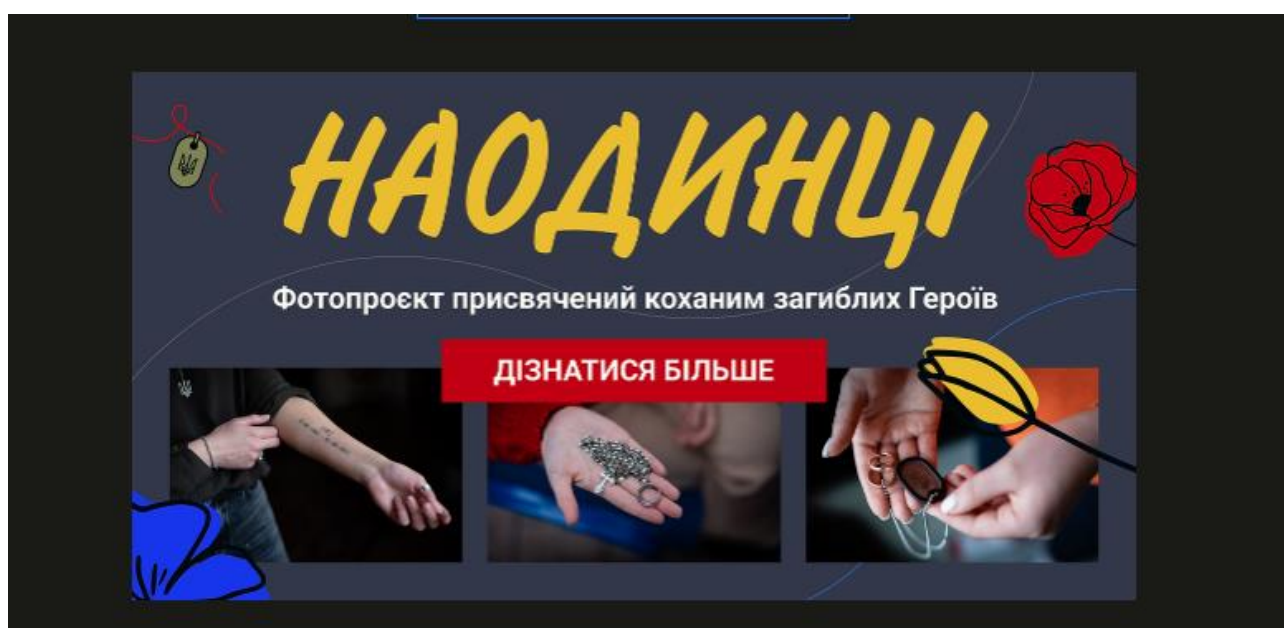


Рисунок 1.9. – Фотопроект, що присвячено коханим загиблих героїв

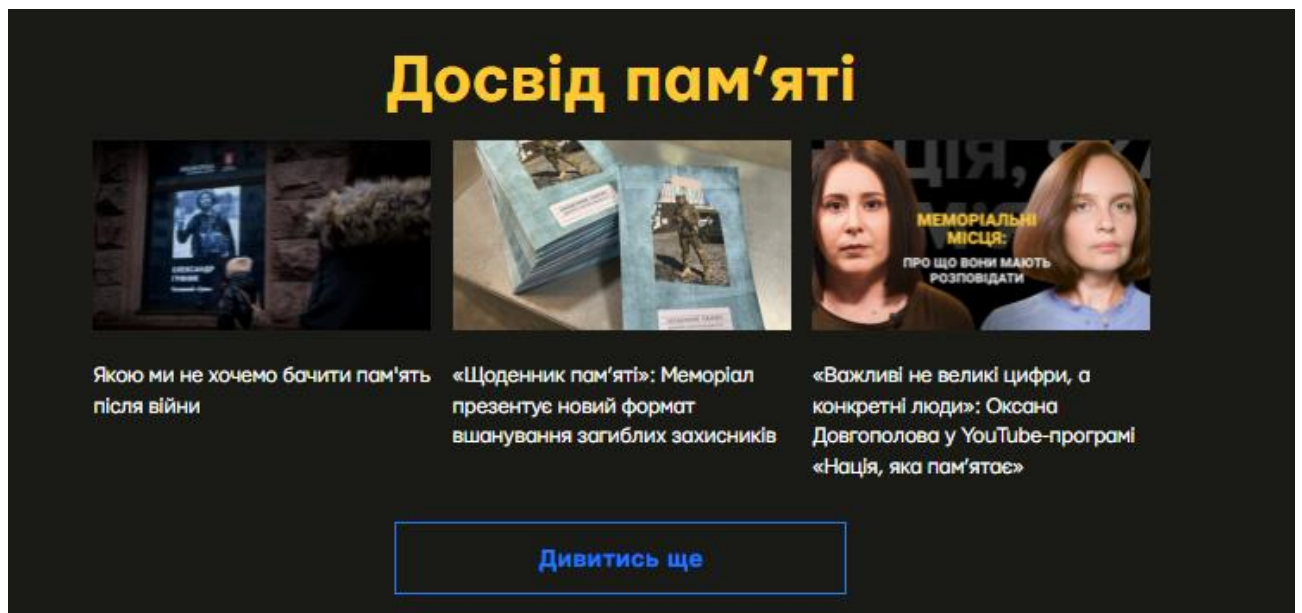


Рисунок 1.10. – Досвід пам'яті

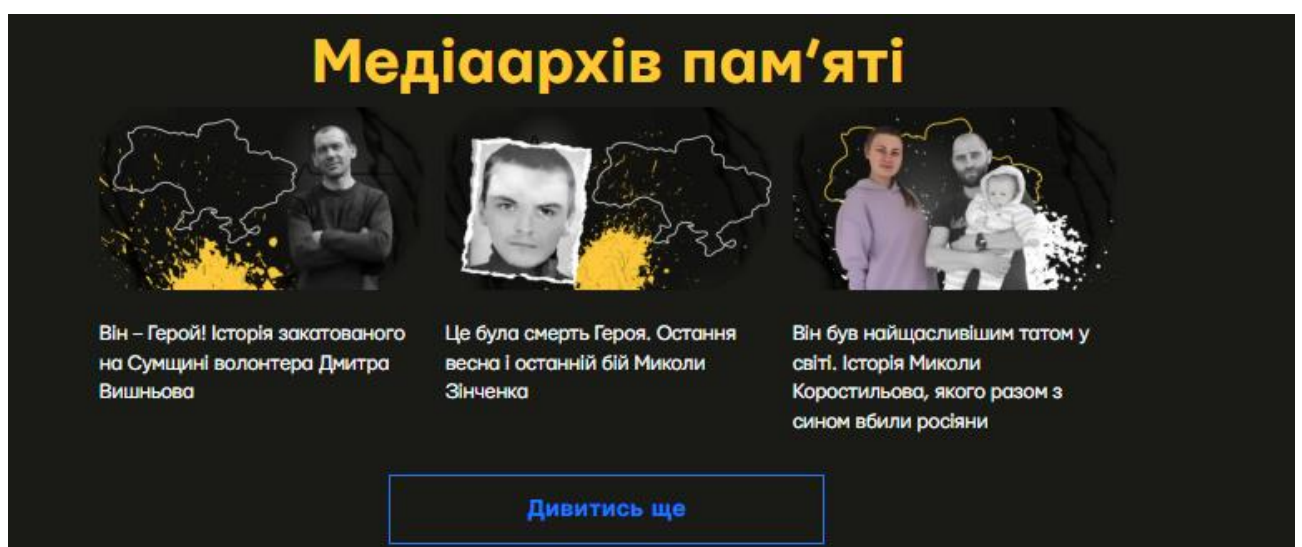


Рисунок 1.11. – Медіаархів пам'яті

Мартиролог «Закриті очі» – проєкт, який збирає інформацію на основі відкритих джерел про всіх тих, хто загинув під час російсько-української війни серед цивільного населення, щоб зберегти пам'ять про загиблих і розповісти світу про немислимий у 21 столітті геноцид, якому були піддані українці. Проєкт збирає інформацію про всіх українців, які загинули з початку 2014-го року. Головна сторінка проєкту наведена на рис. 1.12.

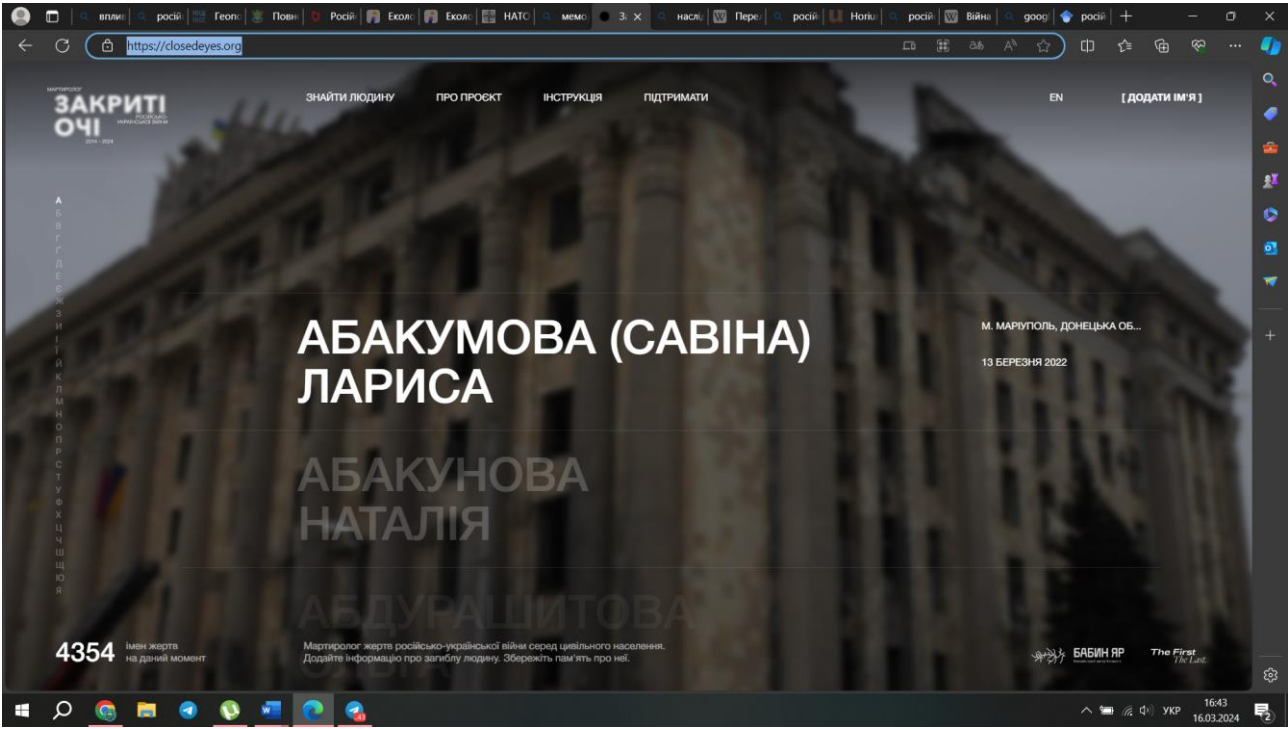


Рисунок 1.12. – Головна сторінка проєкт «Закриті очі»

До цього проєкту також можна додати свою інформацію про жертву, та її коротку біографію та фото. Сайт двомовний (українська та англійська).
Вся доступна інформація про жертв відображається у алфавітному списку.
Інформація, що знаходиться у базі даних цього проєкту береться з відкритих джерел.

Таблиця 1.1

Порівняльна таблиця

	«Закриті очі»	Платформа пам'яті Меморіал	Власний продукт
Додання інформації на сайт	+	+	+
Збирання інформації на основі відкритих джерел	+	+	+

Таймлайн російського вторгнення в Україну	-	-	+
Інфографіка про вбивства в містах	-	-	+
Інформація про російські удари	-	-	+

Згідно з прикладами наведеними в Таблиці 1.1 можемо зробити висновок, вебсторінка що розробляється має всі потрібні функції, є конкурентоспроможною та в деяких випадках кращою за її аналоги.

1.3. Інструментальні засоби реалізації програмного продукту та вимоги до апаратних засобів та обґрунтування їх вибору

Вибір мов програмування та інструментів розробки залежить від багатьох факторів, включаючи обсяг роботи проекту, вимоги до функціональності, продуктивності, методи реалізації і багато іншого. У випадку розробки меморіалу пам'яті було обрано HTML [21], CSS [22], PHP [23], бази даних MySQL [24] та JavaScript [25]. Платформою для розробки та локального тестування вебсайтів було обрано Open Server.

Open Server [26] – безкоштовна, портативна, проста у встановленні та налаштуванні платформа, має інтерфейс користувача, підтримує різні версії PHP та MySQL, має вбудований вебінтерфейс phpMyAdmin, що дозволяє керувати базами даних. Містить у собі набір всіх необхідних програм для веброзробника, завдяки своїй портативності звільняє розробника від роботи за конкретним комп'ютером.

Цю платформу було вибрано завдяки її портативності, легкості у встановленні, налаштуванні та набору програм, яка вона у собі містить. Також вона не вимагає особливості характеристик системи та її можна запускати навіть не на найсучаснішому комп'ютері.

HTML – це стандарт для створення вебсторінок і забезпечує структурну розмітку. За допомогою HTML можна створювати вебсторінки різного типу, з текстом, зображеннями, відео та іншими мультимедійними елементами. HTML – основа для будь-якої вебсторінки та забезпечує доступ до контенту сторінки чи сторінок з будь-якого пристрою, що має підключення до Інтернету.

CSS – мова для створення стилів та забезпечує зовнішній вигляд вебсторінки. За допомогою CSS можна задавати безліч параметрів, наприклад, такі як: кольори, розмір шрифту, шрифт та інші параметри відображення вебсторінки. CSS забезпечує можливість створення красивих та привабливих вебсторінок з гармонійним дизайном [27].

База даних (БД) – це набір інформації або даних, яка зберігається впорядкованим способом, інакше кажучи, сховище даних.

Система управління базами даних (СУБД), використовує мову запитів SQL, створення, зберігання, організації та управління даними. Схема роботи з базами даних продемонстрована на рисунку 1.12 [28].

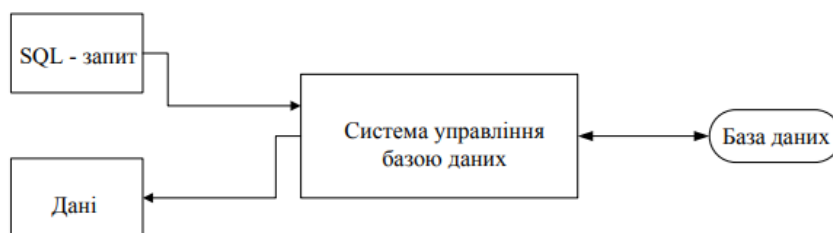


Рисунок 1.12 – Схема роботи БД

Наведемо коротке пояснення рисунку 1.12, у СУБД зберігаються дані і завдяки SQL-запитам вони відображаються або проводяться інші маніпуляції з ними.

СУБД поділяють на однокористувацькі та багатокористувацькі (мультикористувацькі), як можна зрозуміти з назви, то у однокористувацьких доступ має тільки один користувач, а у багатокористувацьких декілька (багато) користувачів може працювати з базою даних одночасно.

PHP і MySQL є провідними технологіями для розробки веборієнтованих програм. Ці технології є найрозповсюдженими через свою продуктивність, гнучкість та зручність у використанні.

Мова PHP – це скриптова мова програмування, яку використовують для створення динамічних Вебсторінок і Вебдодатків. PHP дає змогу взаємодіяти з базою даних, обробляти дані форми та взаємодіяти з іншими серверами. Вона також підтримує безліч бібліотек і фреймворків, які спрощують створення складних Вебдодатків.

MySQL – це реляційна система управління базами даних, яка використовується для зберігання та управління даними. MySQL є однією з найшвидших і найнадійніших СУБД, та її легко інтегрувати з PHP.

MySQL надає ряд функцій для керування даними, включаючи сортування, фільтрацію та пошук.

Давайте підкреслимо переваги MySQL:

Швидкість і надійність роблять MySQL популярною альтернативою пропрієтарним системам баз даних. Система працює на кількох платформах і має багато привабливих функцій. Вона проста у використанні, установці та обслуговуванні та добре задокументована, має чудову підтримку через групу користувачів, а також пропонує комерційну підтримку.

MySQL може керувати зв'язками між 1,5 мільярдами рядків даних у кількох віртуальних таблицях. Це означає, що вона може обслуговувати сотні клієнтів, які підключаються до сервера, і може використовувати кілька баз даних одночасно.

Вона також використовується для оновлення серверів у браузері та розповсюдження даних на сервери за межами браузера.

MySQL розроблено для обробки великих баз даних набагато швидше, ніж існуючі рішення, і успішно використовується у виробничих середовищах з високими вимогами протягом кількох років.

Таким чином, MySQL Server ідеально підходить для доступу до баз даних в Інтернеті з точки зору підключення, швидкості та функцій безпеки.[29]

Для розробки кваліфікаційної роботи використовувався вебінтерфейс phpMyAdmin, який не є базою даних, а надає інтерфейс для роботи з базою даних, оскільки працює на сервері. За допомогою цього вебінтерфейсу розробник може створити багатокористувацьку базу даних та проводити різні маніпуляції з керуванням та організацією даних.

Представлення та таблиці у phpMyAdmin можна створити як і за допомогою конструктора, так і за допомогою мови запитів SQL. Також phpMyAdmin має вбудований «Дизайнер», який може продемонструвати фізичну модель бази даних, так само, як і MSSQL Server Management. phpMyAdmin надає користувачам зручний інтерфейс для роботи з базами даних MySQL, який може бути встановлений на вебсервері та доступний на персональному комп'ютері, що має доступ до Інтернету. phpMyAdmin підтримує багато мов програмувань, наприклад, PHP, JavaScript, C#[30] та інші, вони можуть використовуватися для взаємодії з базою даних MySQL, яка була розроблена та керується за допомогою phpMyAdmin [31].

Слід відмітити, що phpMyAdmin дуже пов'язаний з мовою програмування PHP, завдяки чому спрощується організація виведення та керування даними на вебсторінках, за допомогою цього вебінтерфейсу також можна забезпечити цілісність та безпеку даних.

Висновки до розділу 1

У розділі 1 розглянуто напрями використання інформаційних технологій для обліку жертв російського вторгнення в Україну. Було визначено необхідність реалізації вебсторінки меморіалу пам'яті жертв ракетних ударів та виявлено основні проблеми пов'язані з її розробкою.

					ІПЗ.КР.Б-121-24-ПЗ	
						20

Проаналізовано хронологію вторгнення Росії в Україну з 2014 року та проведено аналіз аналогів подібних платформ, таких як Меморіал та Мартиролог «Закриті очі», що були створені під час повномасштабного вторгнення для збору інформації про загиблих під час воєнних дій. Проаналізовано їхні можливості, а також висвітлено переваги та недоліки. Аналіз довів, що розроблена веборієнтована інформаційна система має низку переваг перед існуючими аналогами, такі як відображення таймлайну російського вторгнення, інфографіки та інформації про російські удари.

Описано обрані технології для розробки веборієнтованої системи, включаючи HTML, CSS, PHP, базу даних MySQL та JavaScript. Також обрано платформу Open Server. Описані обґрунтовані принципи вибору технологій, переваги та недоліки.

РОЗДІЛ 2. ПРОЄКТУВАННЯ ВЕБОРІЄНТОВАНОЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ МЕМОРІАЛУ ПАМ'ЯТІ

2.1. Варіанти використання системи

Варіантами використання розроблюваної системи може бути:

1. Розробка цифрової карти, яка б відображала місця ракетних ударів по Україні, деталі про жертви, пошкодження і фотографії. Це може допомогти в усвідомленні розмірів та наслідків конфлікту;
2. Зібрання та аналіз даних про ракетні удари для наукових досліджень, що досліджують ефективність гуманітарної допомоги, реакцію міжнародної спільноти та інші аспекти конфлікту.
3. Створення вебсайту або мобільного додатку, де користувачі можуть отримувати інформацію про ракетні атаки та їх наслідки, а також можуть долучатися до обговорень і акцій;
4. Створення віртуального меморіалу, де можна зберігати історії жертв, фотографії, а також спогади сімей і друзів. Це допоможе у запобіганні забуттю про трагедії та вшануванні пам'яті загиблих;
5. Використання системи для привертання уваги міжнародної спільноти до ситуації в Україні та поширення інформації про порушення прав людини внаслідок конфлікту;
6. Забезпечення доступу до даних для гуманітарних організацій, які надають допомогу постраждалим від ракетних атак, щоб вони могли ефективніше планувати та розподіляти ресурси;
7. Моніторинг реакції уряду на ракетні атаки та їх наслідки, щоб з'ясувати ефективність заходів управління кризою і можливі шляхи поліпшення;
8. Аналіз даних про ракетні атаки може допомогти в розробці рекомендацій для майбутнього щодо підвищення безпеки та захисту цивільних осіб від подібних конфліктів;
9. Надання даних про ракетні атаки правозахисним організаціям та міжнародним трибуналам для розслідування можливих воєнних злочинів;

					ІПЗ.КР.Б-121-24-ПЗ	
						22

10. Використання даних про ракетні атаки для створення інформаційних буклетів, презентацій та відеоматеріалів, які можуть використовуватися для освіти, просвітництва громадськості;

11. Надання інформації про ракетні атаки та їх наслідки медичним та психологічним організаціям для підтримки реабілітації постраждалих осіб;

12. Розробка централізованої бази даних, яка дозволить різним організаціям та державам обмінюватися інформацією про ракетні атаки та їх наслідки для підтримки спільних дій;

13. Включення даних про ракетні атаки до навчальних програм для формування свідомих громадян і підвищення усвідомленості щодо глобальних конфліктів та їх наслідків;

14. Надання даних для координації гуманітарних місій та доставки допомоги постраждалим;

15. Використання даних про ракетні атаки для розробки та вдосконалення стратегій захисту цивільного населення в умовах війни або конфлікту;

16. Використання інформації про місця пам'яті та меморіали ракетних атак для розвитку туристичної індустрії та пам'ятному туризму;

17. Використання даних про ракетні атаки для створення художніх творів, літератури, фільмів та інших культурних проєктів, які висвітлюють тему війни та трагедії;

18. Надання підтримки, допомоги та послуг для родин та близьких загиблих у ракетних атаках;

19. Включення даних про ракетні атаки до навчальних програм для студентів, які вивчають міжнародні відносини, політичну науку та інші суміжні дисципліни;

20. Застосування системи для надання психологічної підтримки та консультацій для людей, які пережили ракетні атаки або втратили близьких;

21. Розвиток системи для формування та підтримки мережі психологічної та соціальної підтримки для військовослужбовців, які пережили ракетні атаки;

22. Використання даних про ракетні атаки для розробки та впровадження стратегій захисту критичної інфраструктури від подібних атак;

23. Використання системи для розробки програм пам'яті, комеморації та реконструкції районів, пошкоджених ракетними ударами;

24. Розробка підручників, уроків та навчальних матеріалів для вчителів, які допоможуть пояснити дітям причини та наслідки ракетних атак;

25. Організація програми волонтерської допомоги для залучення добровольців до надання підтримки та допомоги постраждалим від ракетних атак;

2.2. Структура БД та алгоритми роботи системи

У базі даних є дві таблиці:

1. Таблиця «Victims» (Жертви):

- id – порядковий номер у таблиці;
- first_name – ім'я жертви;
- last_name – прізвище жертви;
- date_of_death – дата смерті;
- place_of_death – місце смерті;
- description – біографія жертви;
- path – шлях до картинки;
- ShootingID – код ракетного удару.

2. Таблиця «Shooting» (Ракетні удари):

- ShootingID – порядковий номер у таблиці;
- Location – місце (удару);
- Date – дата удару;
- Description – тип удару;
- ImagePath – фото удару.

Представлення для статистики по підрахункам, де був удар:

- Місто;
- Кількість загинлих.

Логічна структура таблиць у базі даних наведена у табл 2.1.-2.2.

Таблиця 2.1 – Логічна структура таблиці Victims» (Жертви)

Програмна назва	Тип даних	PK	FK	Опис
id	int	+	-	Порядковий номер у таблиці
first_name	varchar(255)	-	-	Ім'я
last_name	varchar(255)	-	-	Прізвище
date_of_death	date	-	-	Дата смерті
place_of_death	varchar(255)	-	-	Місце смерті
description	text	-	-	Біографія жертви
path	varchar(255)	-	-	Шлях до фото
shootingID	int	-	+	Код ракетного удару

Таблиця 2.2 – Логічна структура таблиці Shootings» (Ракетні удари)

Програмна назва	Тип даних	PK	FK	Опис
ShootingID	int	+	-	Порядковий номер у таблиці
Location	varchar(255)	-	-	Місце удару
Date	date	-	-	Дата
Description	text	-	-	Тип обстрілу
ImagePath	varchar(255)	-	-	Фото удару

2.3. Опис реалізації функціоналу системи

Підхід до реалізації функціоналу системи може бути різноманітним залежно від конкретних потреб та вимог проєкту. Під час розробки даного проєкту було доречно використовувати такий алгоритм.

1. Аналіз вимог включає в себе:
 - оцінка потреб користувачів та бізнес-вимоги;
 - визначення основних функцій та функціональності, яка потрібна для задоволення вимог.
2. Проектування бази даних включає в себе:
 - визначення сутностей та відносин між ними;
 - створення схеми бази даних;
 - розробка моделей даних.
3. Розробка користувацького інтерфейсу включає в себе:
 - створення макетів та дизайну користувацького інтерфейсу;
 - реалізація вебсторінок, форм, кнопок, меню та інших елементів інтерфейсу.
4. Розробка бізнес-логіки включає в себе:
 - створення класів та функцій, які виконують основні операції системи;
 - обробка запитів від клієнтів та генерація відповідей;
 - реалізація алгоритмів, які визначають логіку системи.
5. Розробка серверної частини включає в себе:
 - встановлення та налаштування веб-сервера;
 - реалізація серверного коду для обробки запитів та взаємодії з базою даних;
 - налаштування безпеки та аутентифікації.
6. Розробка клієнтської частини включає в себе:
 - розробка клієнтського коду (JavaScript, HTML, CSS);
 - взаємодія з сервером через AJAX або WebSocket;
 - тестування та оптимізація користувацького інтерфейсу.

7. Тестування та налагодження включає в себе:

- проведення функціонального, інтеграційного та модульного тестування;
- виявлення та виправлення помилок;
- оптимізація продуктивності та швидкодії системи.

8. Впровадження та супровід включає в себе:

- розгортання системи на виробничному сервері;
- навчання користувачів та адміністраторів;
- підтримка та вдосконалення системи після впровадження.

Функціонал системи дозволяє вносити інформацію стосовно загиблої людини та обирати у якому місці, це трапилося.

Функціонал розроблюваної системи дозволяє переглянути статистику, де було найбільше жертвам внаслідок ракетних ударів по містах.

Функціонал системи дозволяє переглянути інформацію стосовно загиблих жертв та переглянути за допомогою фільтру по містах.

Компонентна будова веборієнтованої системи описується шляхом розбиття системи на окремі компоненти, які взаємодіють між собою для досягнення певної функціональності. Вона складається з таких компонентів, як:

1.Клієнтська частина (Frontend):

- користувацький інтерфейс (UI): включає в себе всі елементи, які взаємодіють з користувачем, такі як сторінки, форми, кнопки, меню тощо;
- клієнтський код (JavaScript, HTML, CSS): клієнтська логіка, яка виконується на браузері користувача і забезпечує інтерактивність і візуалізацію сторінок.

2. Серверна частина (Backend):

- бізнес-логіка: це логіка застосунку, яка визначає, як система повинна обробляти запити та виконувати різні завдання;
- доступ до даних (Data Access Layer): компоненти, що відповідають за взаємодію з базою даних або іншими джерелами даних;

- серверний код (наприклад, PHP, Python, Node.js): код, який виконується на сервері і відповідає за обробку запитів від клієнтів, обробку даних та відправку відповідей.

3.База даних:

- зберігання даних: місце, де зберігаються дані системи, такі як користувачі, товари, замовлення тощо;

- система управління базами даних (СУБД): програмне забезпечення, що дозволяє управляти та маніпулювати даними у базі даних.

- запити до бази даних: SQL-запити або інші механізми для доступу, додавання, видалення та оновлення даних

4. Сервіси:

- вебсервіси: компоненти, які надають функціональність системи через API, який може бути викликаний від клієнтської або серверної частини;

- інтеграційні сервіси: сервіси, які забезпечують взаємодію з іншими системами або сервісами через різноманітні протоколи і API.

5. Інфраструктура:

- хостинг та сервери: фізичне або віртуальне середовище, де розгорнута система;

- мережеві компоненти: компоненти, які забезпечують мережеву комунікацію між різними частинами системи.

2.4. Опис UML діаграм

Діаграма варіантів використання (Use Case Diagram) — це графічне зображення, яке показує, які дії можуть виконувати актори в системі. Актори можуть бути користувачами системи, іншими системами або зовнішніми сутностями, які взаємодіють з системою. Основна мета діаграми варіантів використання - показати функціональність системи з точки зору її користувачів [32].

Основними її елементами є:

1. Актори (Actors): Це будь-які сутності або зовнішні системи, які взаємодіють з системою. Актори зазвичай представляють користувачів, але можуть бути також інші системи або процеси.

2. Варіанти використання (Use Cases): Варіанти використання представляють конкретні функціональні можливості системи, які вона надає своїм користувачам. Кожен варіант використання описує одну конкретну дію або послідовність дій, яку може виконати користувач.

3. Відношення між акторами та варіантами використання: Відношення показують, як актори взаємодіють з варіантами використання. Наприклад, актор може бути причетним до виконання конкретного варіанту використання або мати доступ до певної функціональності системи.

Діаграма варіантів використання допомагає команді розробників уточнити вимоги до системи, зрозуміти потреби користувачів і визначити ключові функціональні можливості, які потрібно реалізувати. Вона може бути використана як основа для подальшої розробки детальних специфікацій і проектування системи [33].

Перевагами діаграми варіантів використання є:

1. Діаграми варіантів використання легко зрозуміти навіть для людей без технічного фахового фону. Вони відображають основні взаємодії користувачів з системою у формі сценаріїв, що робить їх доступними для всіх учасників проекту.

2. Діаграми варіантів використання дозволяють ідентифікувати функціональність системи з точки зору того, як користувачі взаємодіють з системою для досягнення своїх цілей. Це допомагає уникнути зайвих технічних деталей на ранніх етапах проекту.

3. Аналіз варіантів використання допомагає ідентифікувати потреби та очікування користувачів системи. Вони дозволяють зосередитися на тому, що саме користувачі хочуть від системи, і враховувати це при розробці.

4. За допомогою діаграм варіантів використання можна чітко визначити, які функції повинна виконувати система, і які варіанти використання вона підтримує. Це допомагає встановити межі системи і уникнути розпливчастості.

5. Варіанти використання можуть служити основою для управління розвитком системи. Вони дозволяють ідентифікувати пріоритетність функціональності та послідовність реалізації, що сприяє керованості проекту.

6. Використання діаграм варіантів використання допомагає забезпечити повноту та консистентність функціональних вимог до системи, що сприяє покращенню якості розробки.

Діаграма варіантів використання наведена на рис. 2.1. Розглянемо детальніше які функції має Користувач та Оператор.

Користувач має доступ до нижче представлених функцій:

- можливість перегляду інформації наявної в веборієнтованій системі;
- самостійне внесення інформації до бази даних, для подальшого її відображення на вебсторінці;
- можливість виконувати вибірку по містам для швидшого та якіснішого пошуку потрібної інформації;
- взаємодія із системою для отримання вибірки з БД на основі статистичних даних про кількість жертв.

Оператор має доступ до таких функцій, як:

- можливість відповідати за налаштування та оптимізацію роботи вебсайту для забезпечення його ефективності;
- здійснювати управління вебсайтом, включаючи технічне обслуговування та моніторинг;
- можливість налаштування та оптимізації бази даних для підвищення її продуктивності та швидкості доступу до даних;
- можливість вводити нові дані в БД та редагувати, оновлювати існуючу інформацію.

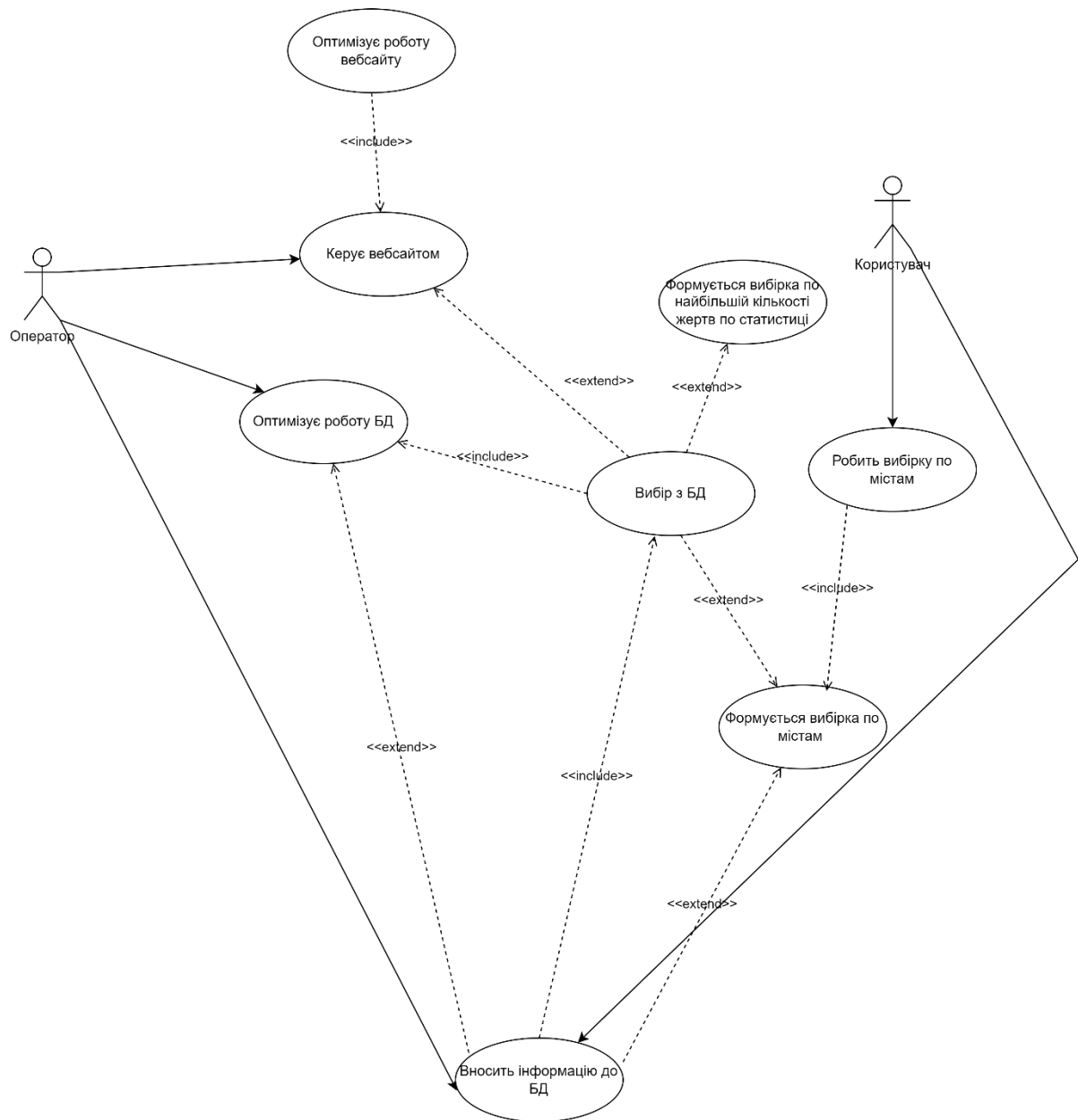


Рисунок 2.1 – Діаграма варіантів використання

Діаграма кооперації (Collaboration Diagram) - це структурна діаграма, яка показує, як об'єкти взаємодіють один з одним у межах системи або підсистеми. Вона ілюструє взаємодію між об'єктами у вигляді взаємодій або повідомлень, які передаються між об'єктами [34].

Основні компоненти діаграми кооперації:

1. Об'єкти (Objects): Це конкретні екземпляри класів, які існують у системі. Об'єкти відображаються у вигляді прямокутників із іменами.

2. Взаємодії (Interactions): Взаємодії показують, як об'єкти спілкуються між собою. Вони представлені стрілками або лініями, які вказують напрям передачі повідомлень або викликів методів між об'єктами.

3. Повідомлення (Messages): Повідомлення вказують на комунікацію між об'єктами. Вони можуть бути синхронними (коли відправник чекає на відповідь) або асинхронними (коли відправник продовжує виконання без очікування відповіді).

4. Роль (Role): Роль описує, яку функцію виконує об'єкт у взаємодії. На діаграмі вона може вказуватися поруч з іменем об'єкта.

Діаграма кооперації часто використовується для моделювання взаємодії між об'єктами у рамках певної операції або сценарію використання. Вона допомагає розробникам отримати краще розуміння того, як об'єкти взаємодіють у реальному часі під час виконання програми [35-36].

Перевагами використання таких діаграм є:

1. Діаграма кооперації дозволяє вам детально розглянути способи взаємодії між об'єктами у системі. Вона допомагає зрозуміти, як об'єкти обмінюються інформацією та виконують взаємодії під час виконання певних операцій;

2. Використання діаграм кооперації може допомогти виявити помилки або недоліки в проекті ще до початку розробки. Ви можете виявити проблеми у взаємодії об'єктів та вирішити їх на етапі аналізу;

3. Діаграми кооперації є ефективним інструментом для комунікації між учасниками проекту. Вони дозволяють розробникам, аналітикам та іншим зацікавленим сторонам легко обговорювати взаємодії об'єктів і розуміти способи їхньої реалізації [37];

4. Діаграми кооперації можуть служити як частина документації проекту. Вони допомагають документувати взаємодії між об'єктами та зберігати цю інформацію для майбутніх використань;

5. Використання діаграм кооперації сприяє кращому проектуванню системи, оскільки вони дозволяють вам детально розглянути взаємодії між об'єктами і зробити відповідні висновки для подальшого розроблення [38].

Діаграма кооперації наведена на рис. 2.2. На основі сторінки «Надати нову інформацію». На її основі ми можемо побачити детальніше структуру нашої інформаційної веборієнтованої системи. На рисунку представлена структура «шапки» сайту, яка включає в себе такі елементи як: «Головна», «Стрічка», «Статистика», «Загиблі» та «Надати нову інформацію». На рис. 2.2 детальніше розглянута структура сторінки «Надати нову інформацію», на ній ми можемо побачити форму для додавання нової інформації користувачем про загиблих, для відправки цих даних в БД.

Форма складається з наступних елементів:

- рядок для введення імені;
- рядок для введення прізвища;
- елемент для введення дати смерті загиблого;
- елемент для вибору міста;
- рядок для введення біографії людини;
- рядок для вибору типу обстрілу;
- елемент для додавання фото загиблого;
- кнопка для відправки інформації до бази даних.

Також нам представлена схема кооперації наведена на рис. 2.3, яка більш детально розглядає структуру сторінки «Статистика». На ній ми можемо побачити інформаційний графік, який показує найбільшу кількість вбивств в певних містах, а також за датами та типом обстрілів, спричинених російською агресією.

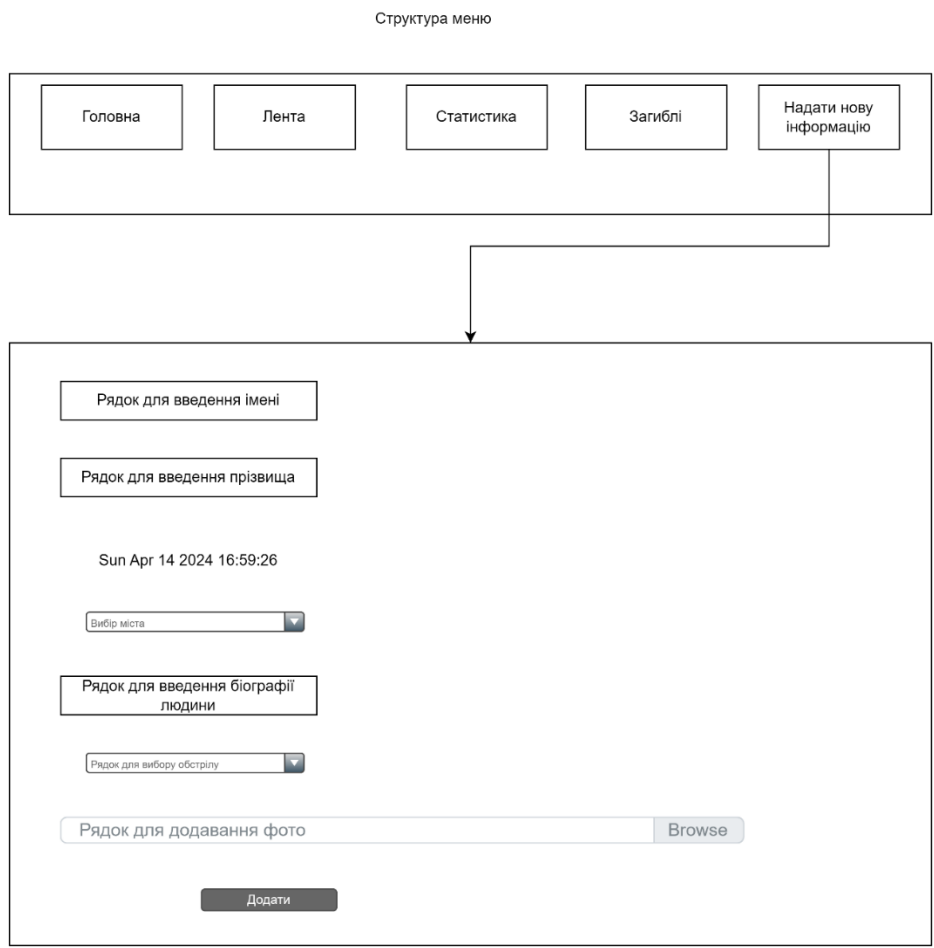


Рисунок 2.2 – Діаграма кооперації сторінки «Надати нову інформацію»



Рисунок 2.3 – Схема кооперації сторінки «Статистика»

Розглянемо схематичне подання даних до розроблюваної бази даних, яке наведено на рис. 2.4. На ньому представлено три таблиці Shootings, Victims, і citymurders, дві з яких поєднані між собою через зовнішній ключ ShootingID. Це означає, що кожна жертва пов'язана з конкретним типом обстрілу, що відбувалися. Розберемо кожну таблицю окремо.

Таблиця Victims:

- id (int): Унікальний ідентифікатор для кожної жертви;
- first_name (varchar(255)): зберігає в собі ім'я жертви;
- last_name (varchar(255)): прізвище жертви;
- date_of_death (date): дата смерті жертви;
- place_of_death (varchar(255)): місто смерті жертви;
- description (text): опис біографії жертви або обставин її смерті;
- path (varchar(255)): шлях до зображення, для демонстрації фото жертви;
- ShootingID (int): ідентифікатор обстрілу (зовнішній ключ, пов'язаний з таблицею Shootings).

Таблиця Shootings:

- ShootingID (int): унікальний ідентифікатор для кожного обстрілу;
- Location (varchar(255)): місто, де відбувся обстріл;
- Date (date): дата, коли відбувся обстріл;
- Description (text): опис події;
- ImagePath (varchar(255)): шлях до зображення, пов'язаного зі стріляниною.

Таблиця citymurders:

- City (varchar(255)): назва міста;
- TotalMurders (bigint): загальна кількість вбивств у цьому місті.

За допомогою цієї бази даних ми можемо зберігати дані які надають користувачі або оператор для подальшої демонстрації інформації на вебсторінці. Вона забезпечує організоване зберігання даних та полегшує доступ до них.

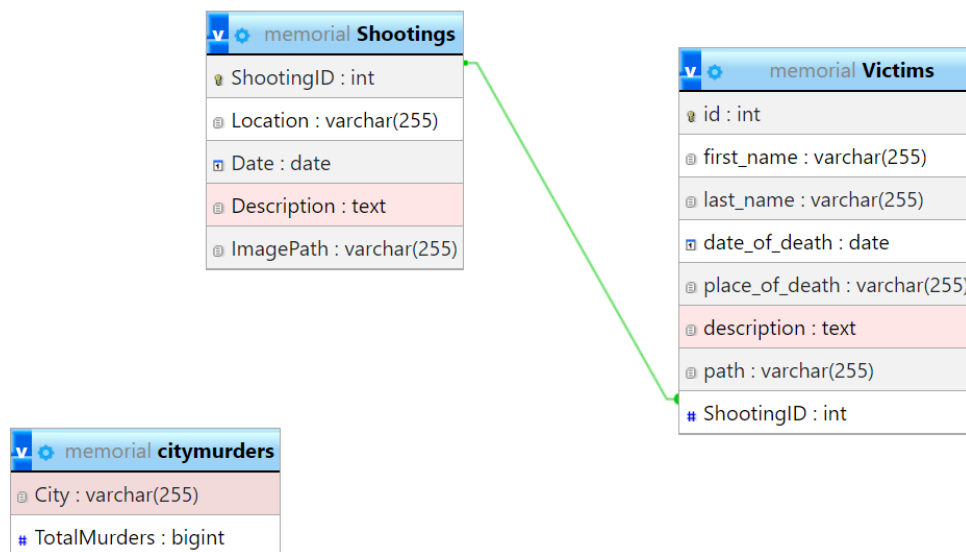


Рисунок 2.4 – Фізична схема БД

2.5. Схема взаємодії програмних модулів та алгоритми роботи

Алгоритм роботи вебсайту полягає в тому, що спочатку завантажується головна сторінка, де відображається хронологія обстрілу у вигляді таймлайну (анімація).

Зверху є меню, що дозволяє переходити по відповідним вебсторінкам.

Схематична структура меню наведена на рис. 2.5. Структура меню у всіх вебсторінок однакова.

Структура меню



Рисунок 2.5 – Схематичне подання структури меню

Структура сторінки «Надати нову інформацію» вже була наведена на діаграмі кооперації, на рис. 2.2.

Структура сторінки «Стрічка» наведена на рис. 2.6. На цьому рисунку ми можемо побачити такі елементи даної сторінки, як:

- рядок для виведення міста;
- рядок для виведення дати та часу обстрілу;
- рядок для виведення типу обстрілу;
- виведення фотографії з місця обстрілу.

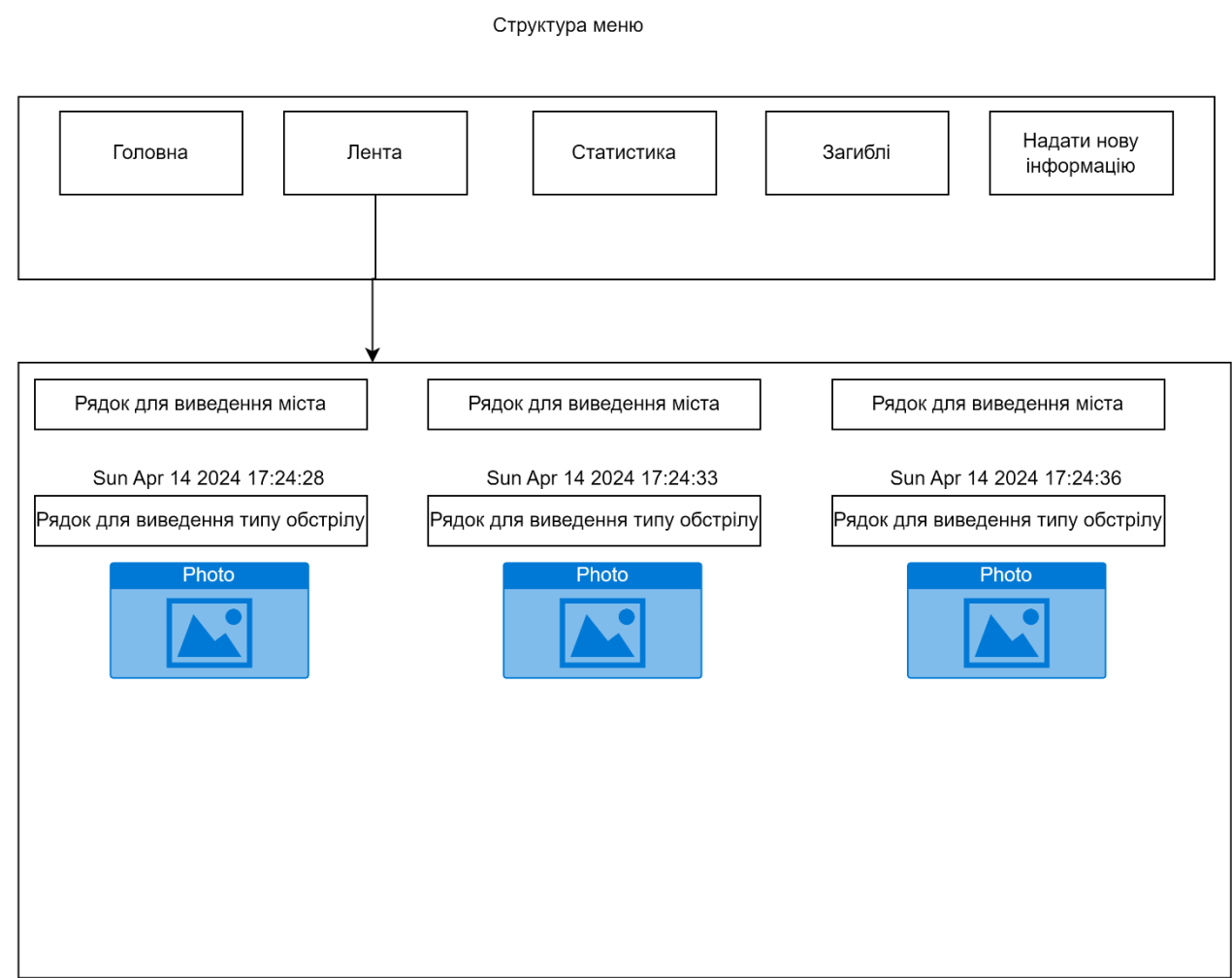


Рисунок 2.6 – Схематична структура сторінки «Стрічка»

Структура сторінки «Статистика» вже була наведена на діаграмі кооперації, на рис. 2.3.

Структура сторінки «Загиблі» наведена на рис. 2.7.

Структура меню

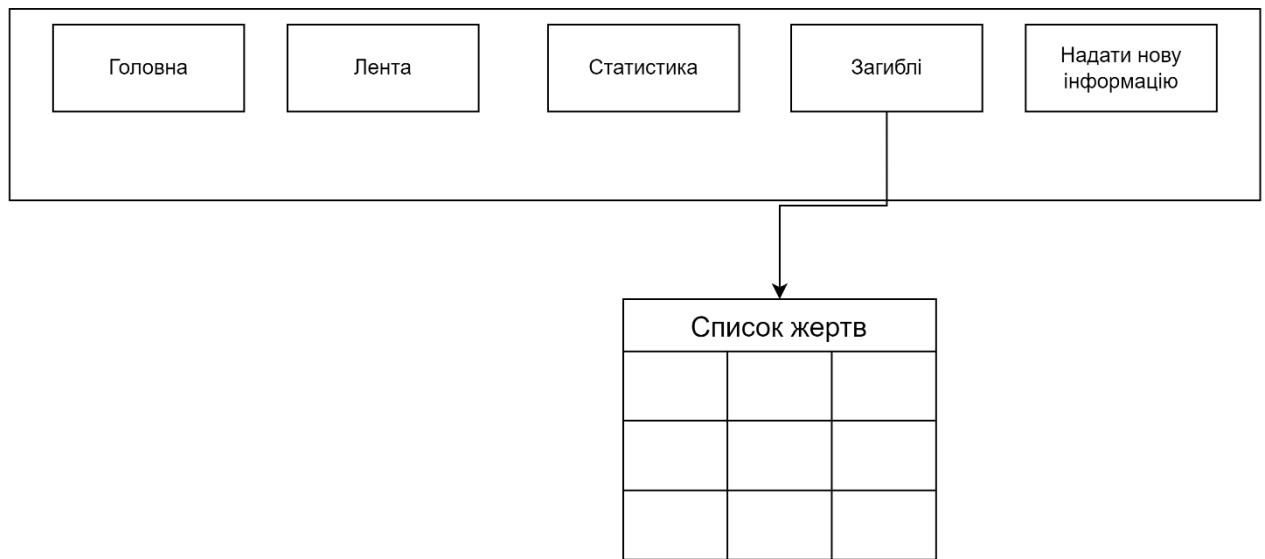


Рисунок 2.7 – Структура сторінки «Загиблі»

Висновки до 2 розділу

У даному розділі детально висвітлюються варіанти використання та удосконалення розроблюваної системи, від створення цифрової карти до моніторингу реакції уряду на ракетні атаки. Це дозволить системі бути корисною для більшої аудиторії.

Основна увага була приділена опису структури бази даних, яка складається з таблиць "Victims" (Жертви) та "Shooting" (Ракетні удари), з детальним визначенням полів кожної таблиці. Схеми даних та програмних модулів, а також алгоритм роботи вебсайту, опису UML-діаграм, варіантів використання (Use Case Diagrams), які допомагають визначити основні функціональні можливості системи ілюструючи функціональність з боку користувачів, та кооперації (Collaboration Diagrams), що ілюструють взаємодію між об'єктами системи для кращого розуміння структури та взаємодії між компонентами. Представлені діаграми забезпечують зрозумілість та послідовність у розробці. Також була описана логічна структура таблиць бази даних.

У цьому розділі наочно подано схематичне подання структури меню і всього вебсайту. Він підкреслює важливість ретельного проектування для створення надійної та ефективної веборієнтованої інформаційної системи.

РОЗДІЛ 3. ІНТЕРФЕЙС ТА ПОРЯДОК РОБОТИ З ВЕБОРІЄНТОВАНОЮ СИСТЕМОЮ МЕМОРІАЛУ ПАМ'ЯТІ

3.1. Порядок встановлення та налаштування параметрів

Процес встановлення Open Server до відображення вебсайту дуже простий. Для початку необхідно скачати Open Server з офіційного вебсайту та встановити його собі на ПК. При цьому на обраному жорсткому диску має бути принаймні 10 ГБ вільного місця. Після успішного встановлення необхідно запустити Open Server, перейти у налаштування та налаштувати його згідно з модулями, що наведені на рис. 3.1.

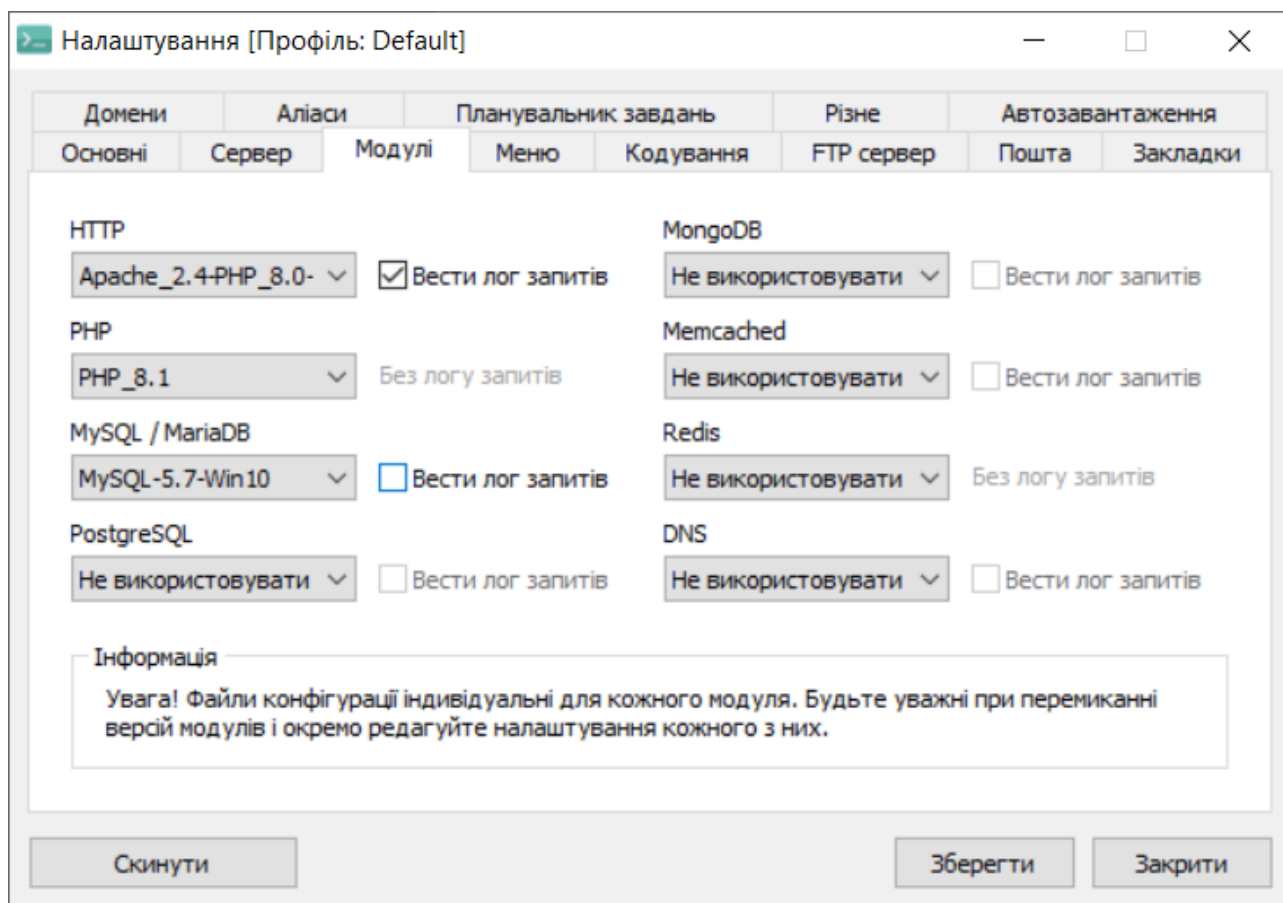


Рисунок 3.1 – Налаштування Open Server

Потім необхідно натиснути на зберегти. Спробувати запустити Open Server обравши функцію «Запустити». Після успішного запуску треба зайти у меню «Додатково» та обрати phpMyAdmin.

Завантажитися сторінка авторизації у phpMyAdmin (рис.3.2) там, де логін необхідно увести стандартний логін root, поле паролю залишити порожнім.

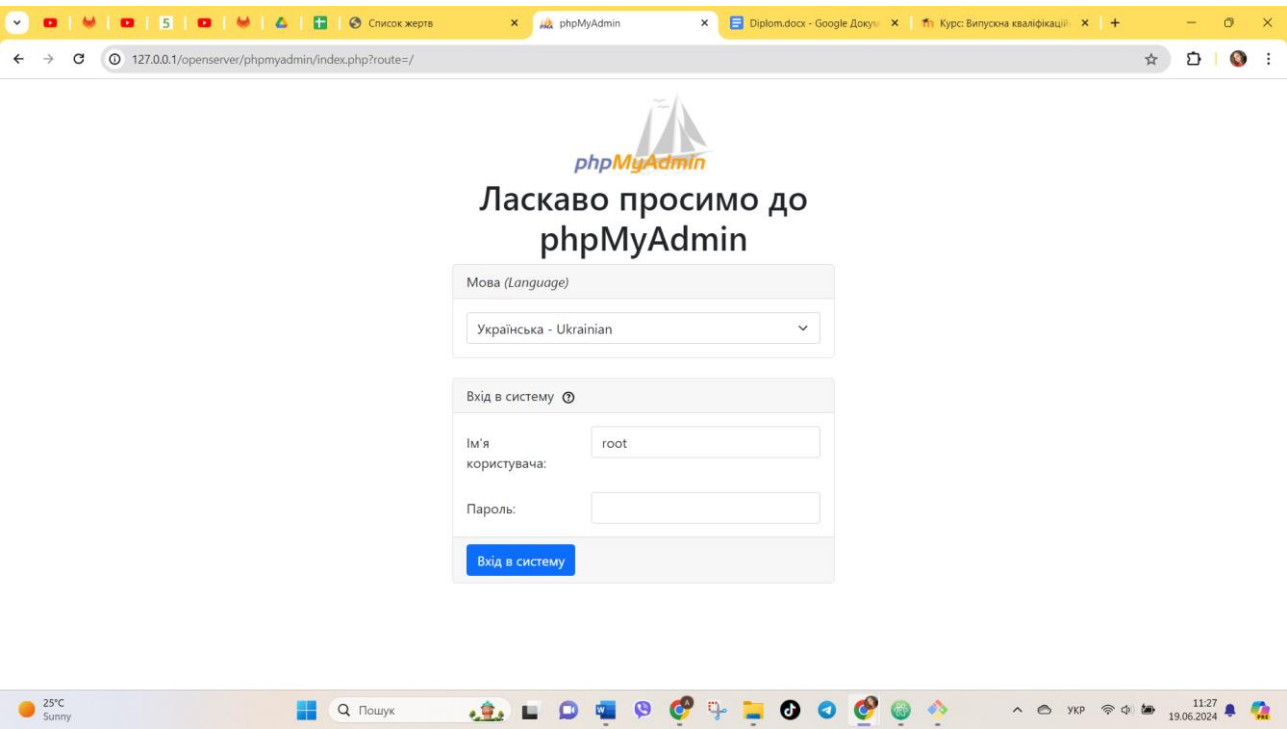


Рисунок 3.2 – Сторінка авторизації у phpMyAdmin

Далі необхідно створити нову базу даних обравши «Нова» з домашньої сторінки phpMyAdmin (рис.3.3-3.4).

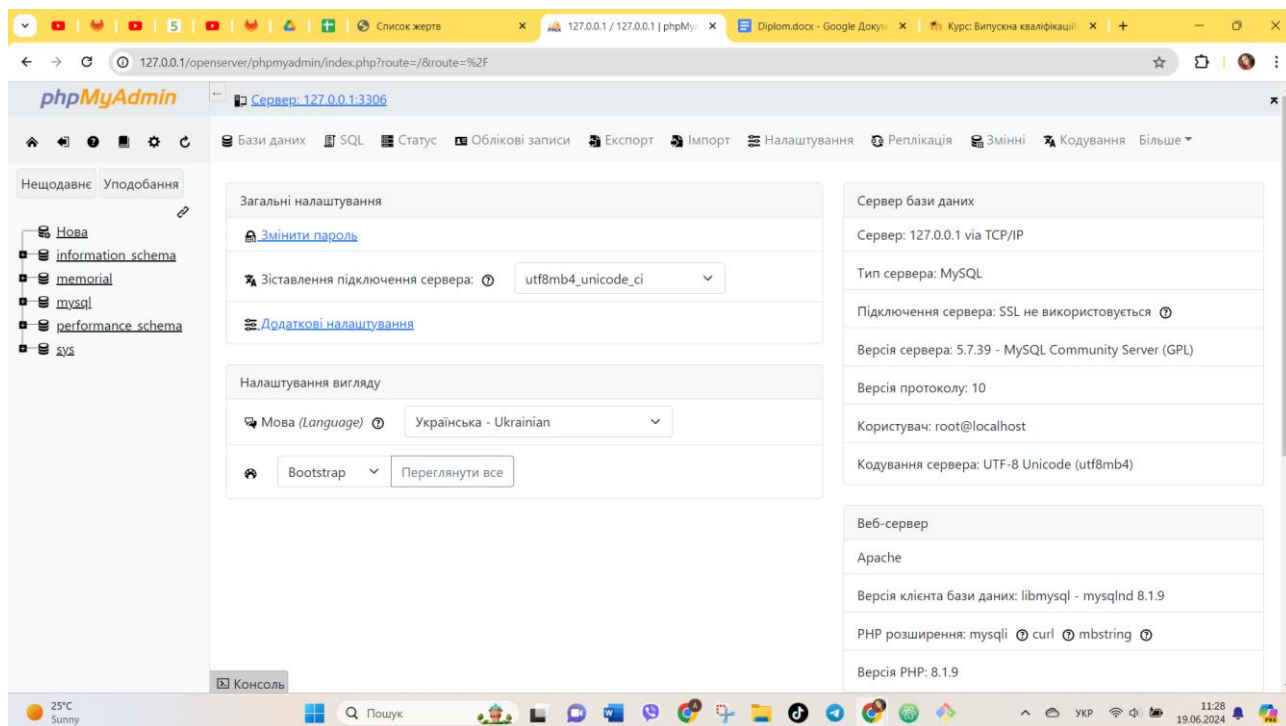


Рисунок 3.3 – Домашня сторінка phpMyAdmin

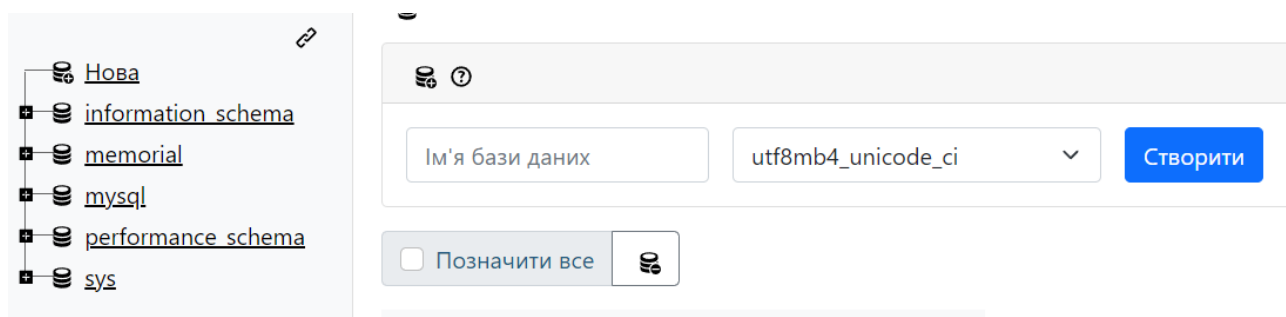


Рисунок 3.4 – Сторінка створення нової бази даних

Необхідно увести ім'я бази даних латинським алфавітом, виставити відповідне кодування та натиснути «Створити».

Якщо у вас вже є таблиці, які ви хочете додати до бази даних, то необхідно натиснути «Імпорт» або ви хочете скачати таблиці з вашої поточної бази даних, то необхідно натиснути «Експорт» (рис.3.5).

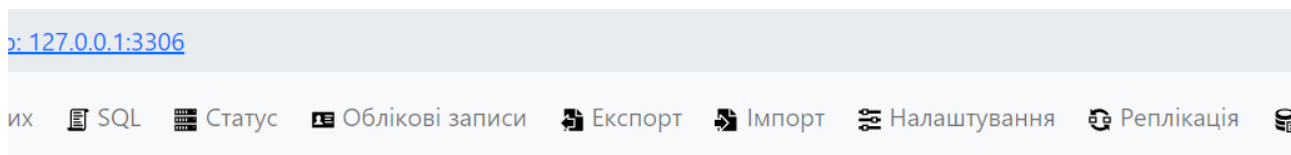


Рисунок 3.5 – Імпорт та Експорт

Якщо ви імпортуєте дані, то в разі успіху вас буде про це проінформовано.

Якщо ви бажаєте вручну створити вашу базу даних, то необхідно обрати або SQL запити, або за допомогою конструктора, який знаходиться на вкладці «Структура» (рис.3.6).

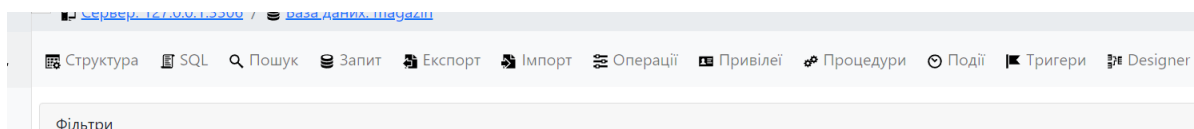


Рисунок 3.6 – Вкладка «Структура» та SQL запити

Далі ваш сайт необхідно розмістити у паці з проєктами, щоб туди потрапити треба натиснути на Open Server (прапорець) та обрати «Папка з проєктами». Відкриється папка domains і туди треба помістити ваш вебсайт, що може містити у собі не тільки php, html файли, а ще й CSS та JavaScript-файли, тобто, помістити туди всю папку з вебсайтом.

Потім необхідно перезавантажити Open Server та відкрити ваш сайт за допомогою підпункту «Мої проєкти» в Open Server (прапорець).

3.2. Структура інтерфейсу веборієнтованої системи меморіалу пам'яті

Інформаційна система складається з 5 сторінок. Розглянемо сторінку «Головна» (рис.3.7). На цій сторінці відображається таймлайн російського вторгнення в Україну.

Таймлайн було вирішено взяти у вигляді GIF-файлу та додати його у структуру вебсайту.

					ІПЗ.КР.Б-121-24-ПЗ	43

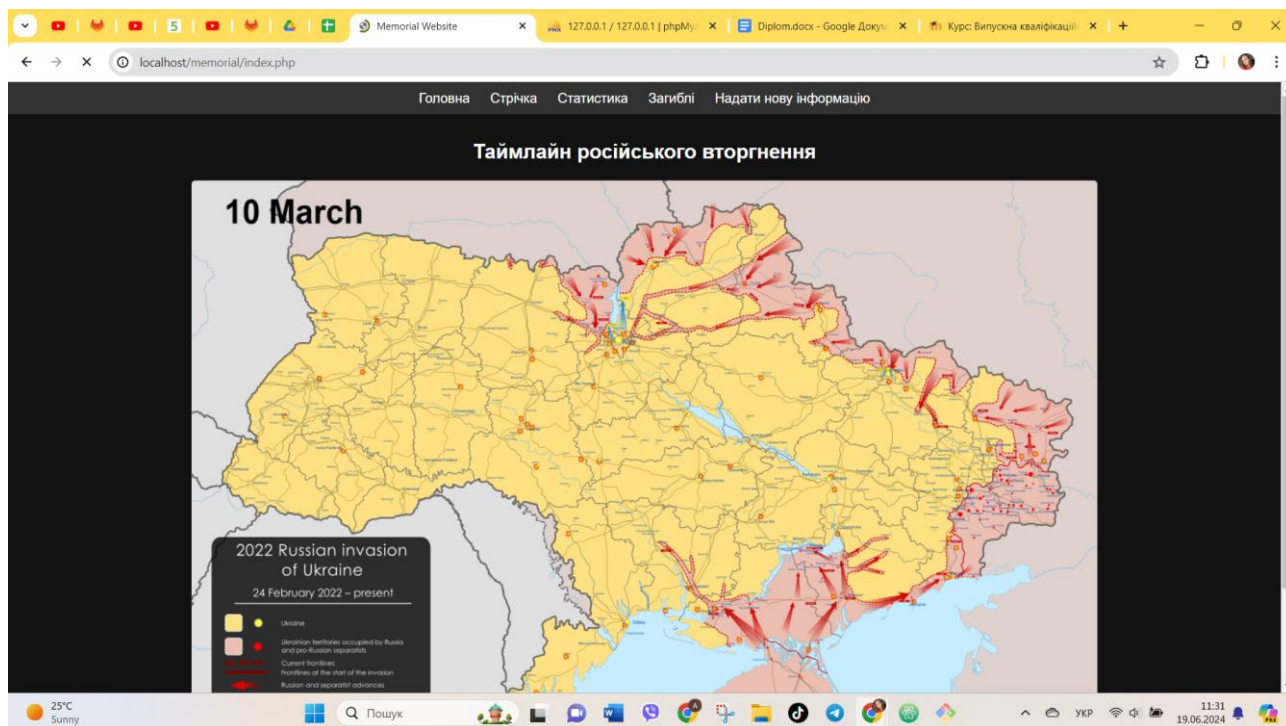


Рисунок 3.7 – Головна сторінка

Розглянемо сторінку «Стрічка» (рис.3.8). На цій сторінці відображається інформація про російські удари у форматі: місто, дата, тип озброєння з якого було вдарено та фотографія наслідку обстрілу.

Варто зазначити, що фотографії та інша інформація динамічно завантажуються без уповільнення вебсайту. Інформація на цій сторінці зберігається у відповідній таблиці.

Інформацію на цю сторінку може завантажувати тільки адміністратор вебсайту для збереження інформаційної тиші та попередження розповсюдження недостовірної інформації.

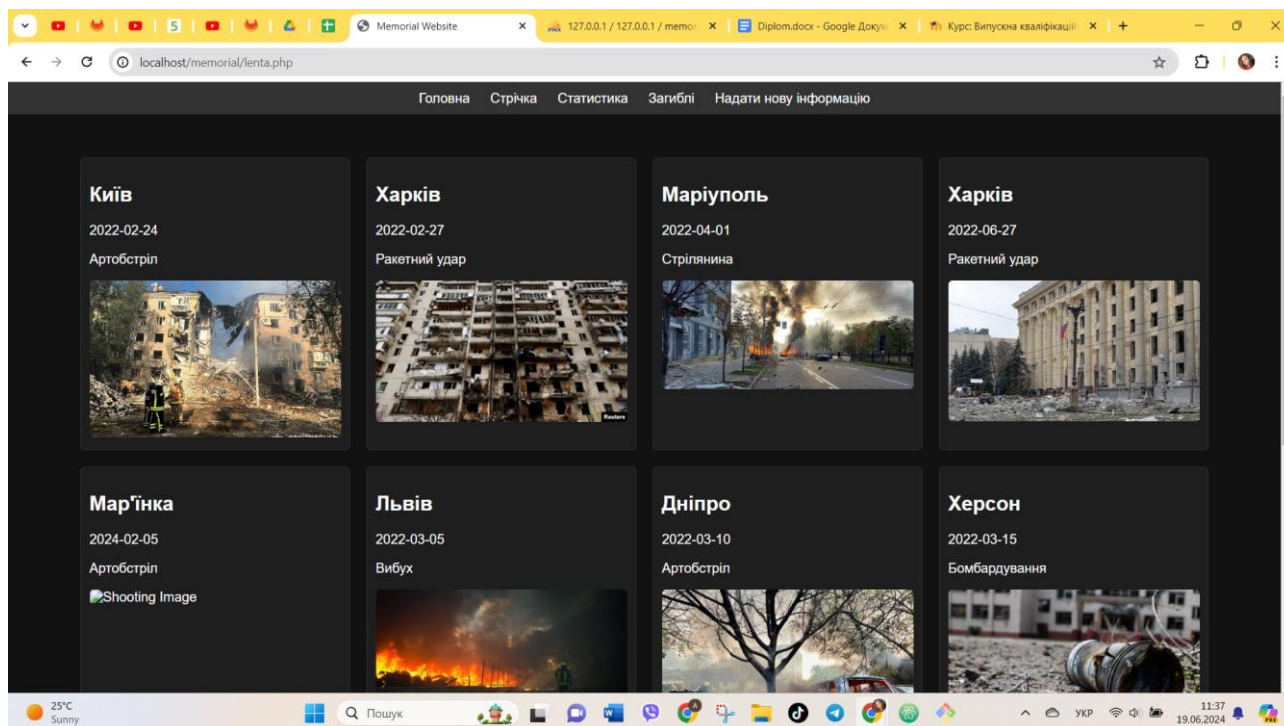


Рисунок 3.8 – Сторінка «Стрїчка»

Переглянемо вебсторінку «Статистика» (рис. 3.9 – рис. 3.11). На цій сторінці відображається статистика з відповідного представлення, що підраховує кількість загиблих в певних містах.

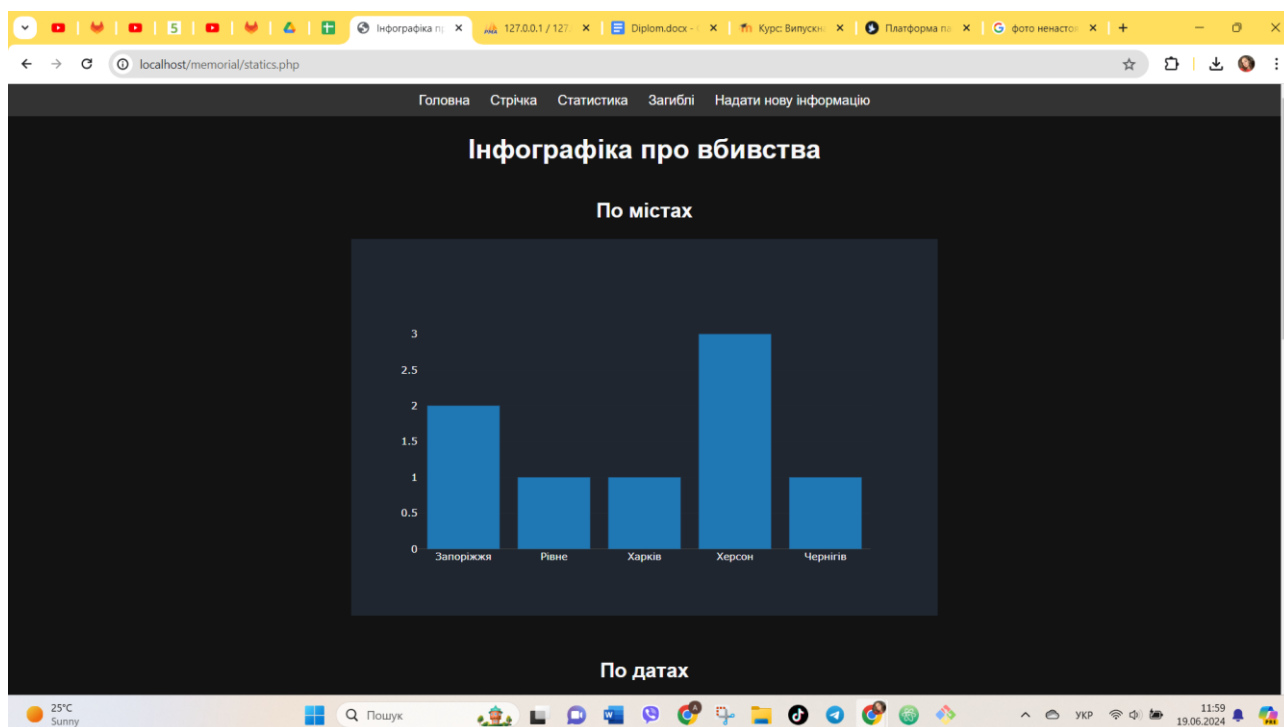


Рисунок 3.9 – Сторінка «Статистика»

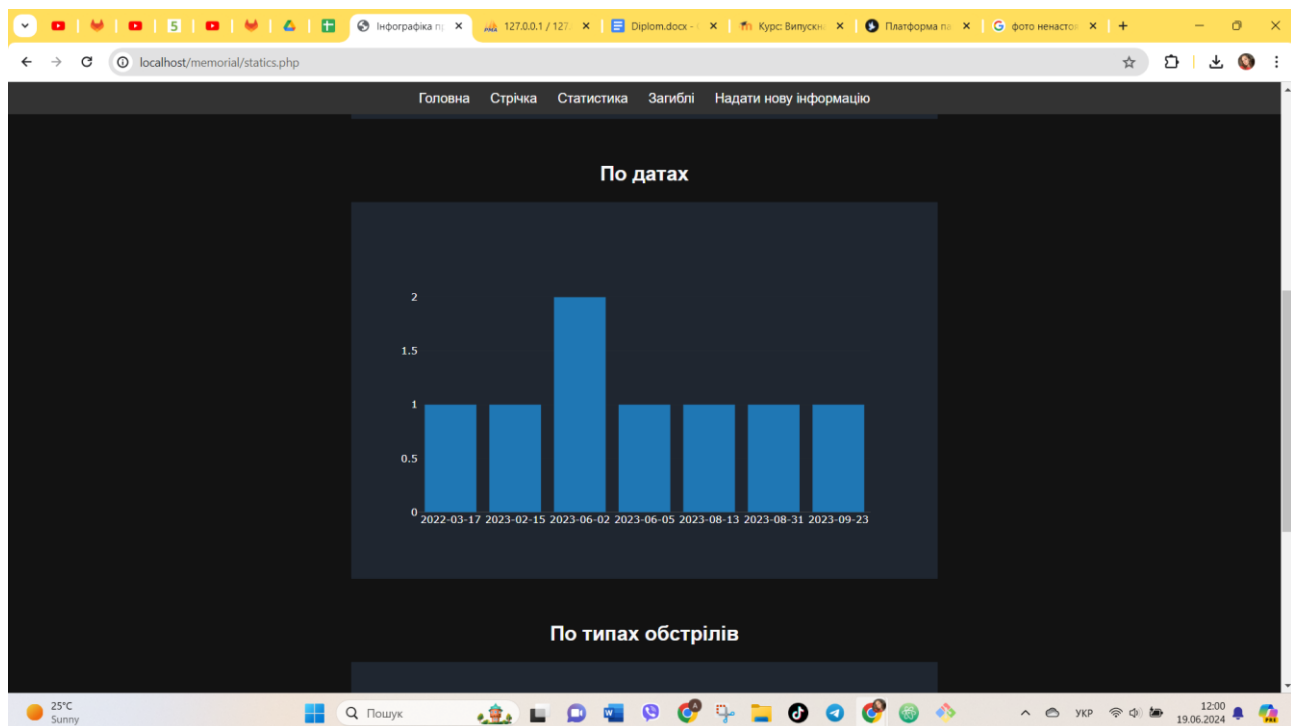


Рисунок 3.10 – Сторінка «Статистика»

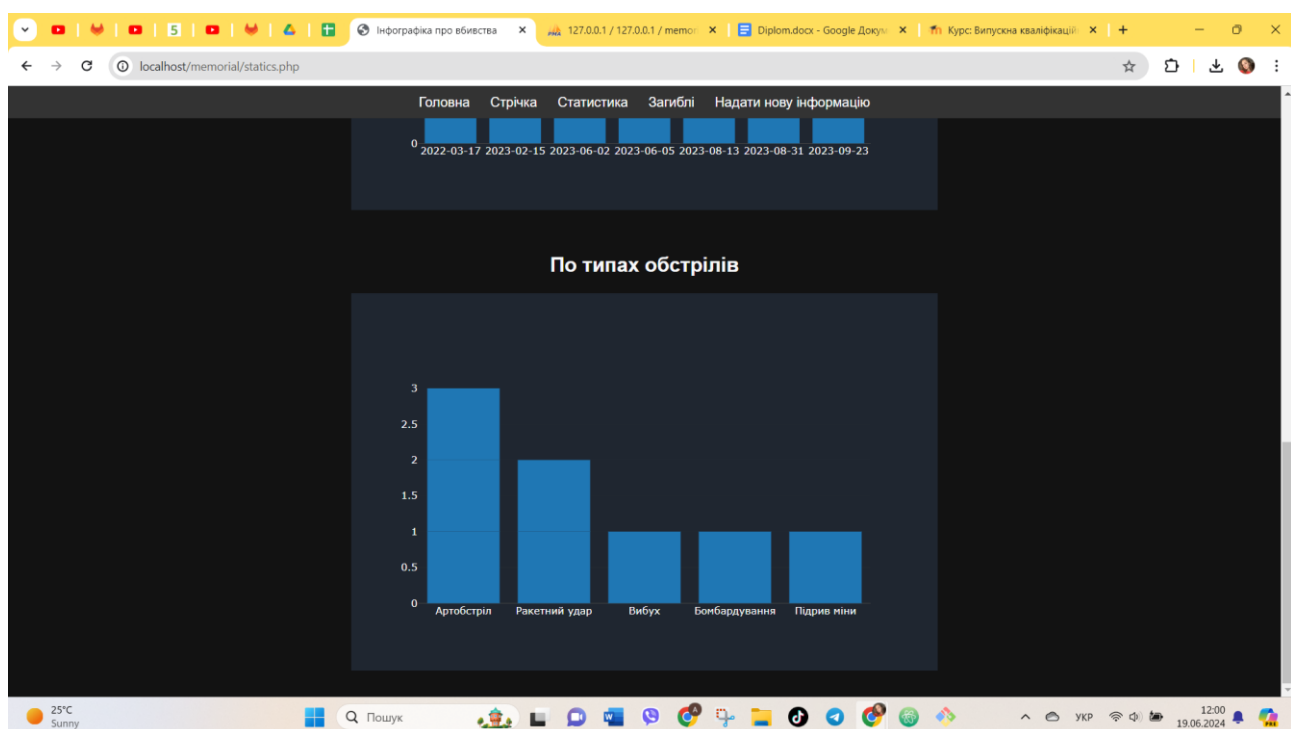


Рисунок 3.11 – Сторінка «Статистика»

Переглянемо сторінку «Загиблі» (рис. 3.12). На цій сторінці відображається вся інформація про загиблих людей. До параметрів, що відображаються відносяться: ім'я, прізвище, дата смерті, місце смерті, смерть внаслідок і фото загиблого. На цій же сторінці врахована можливість відсортувати загиблих громадян по містам, де вони загинули. По замовчуванню відображається вся інформація.

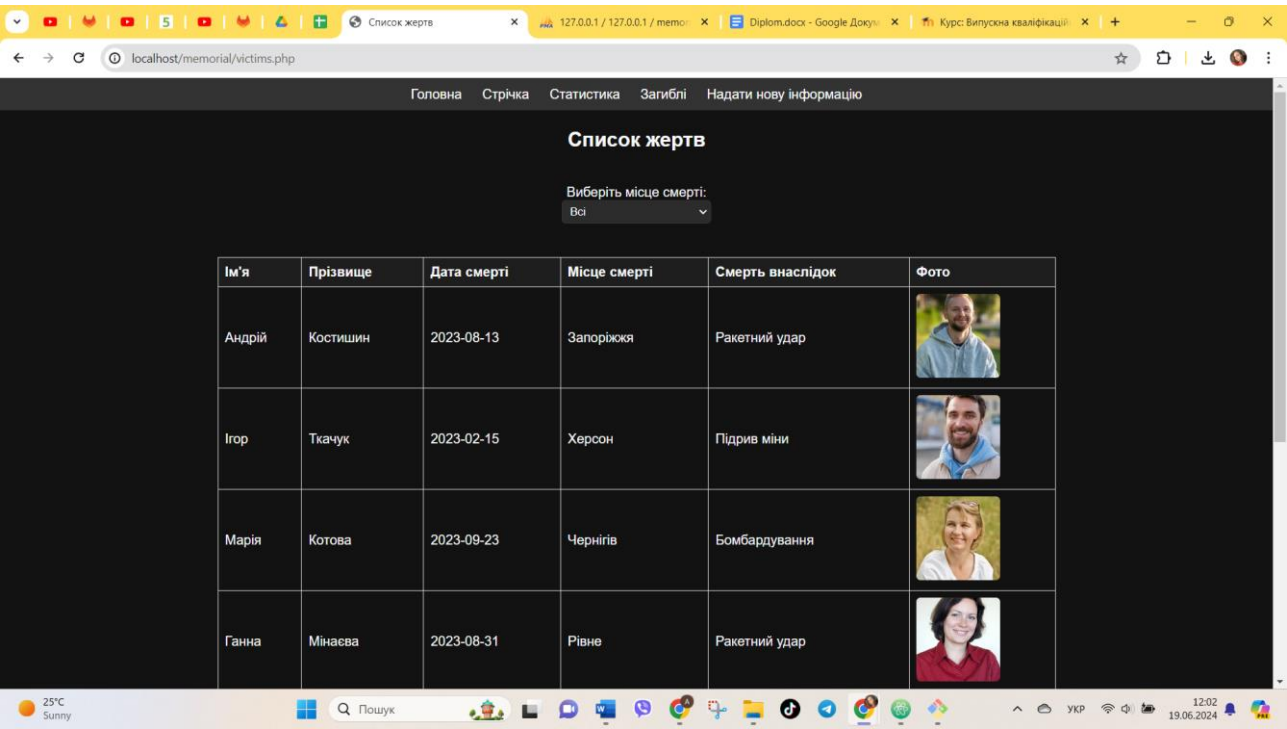


Рисунок 3.12 – Сторінка «Загиблі»

3.3. Тестування роботи програмного продукту

Перейдемо до сторінки «Надати нову інформацію» (рис.3.13). Дана сторінка відповідає за додавання нової інформації про загиблого. Треба увести ім'я, прізвище, дату смерті, місце смерті, написати біографію загиблого, обрати тип обстрілу внаслідок якого він загинув та додати його фото.

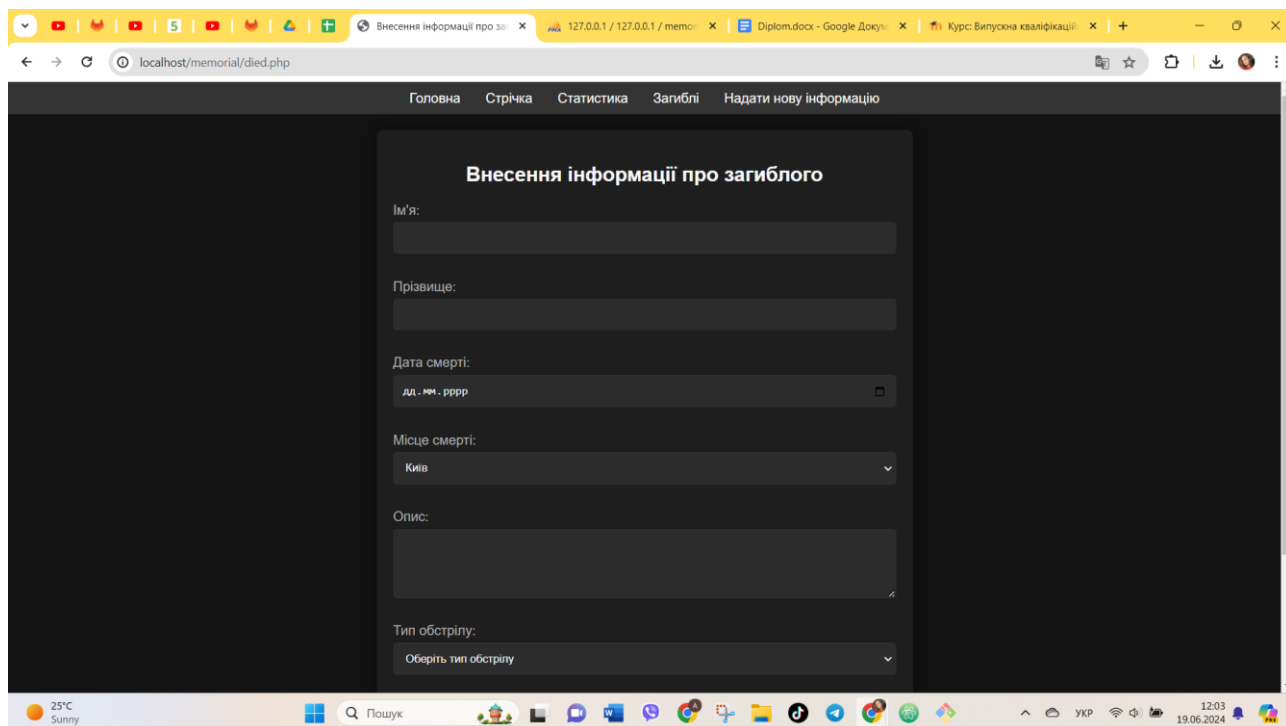


Рисунок 3.13 – Сторінка «Надати нову інформацію»

Заповнимо форму новою інформацією про загиблого та натиснемо на кнопку «Відправити». На рис. 3.14 наведена підготовлена інформація для додавання.

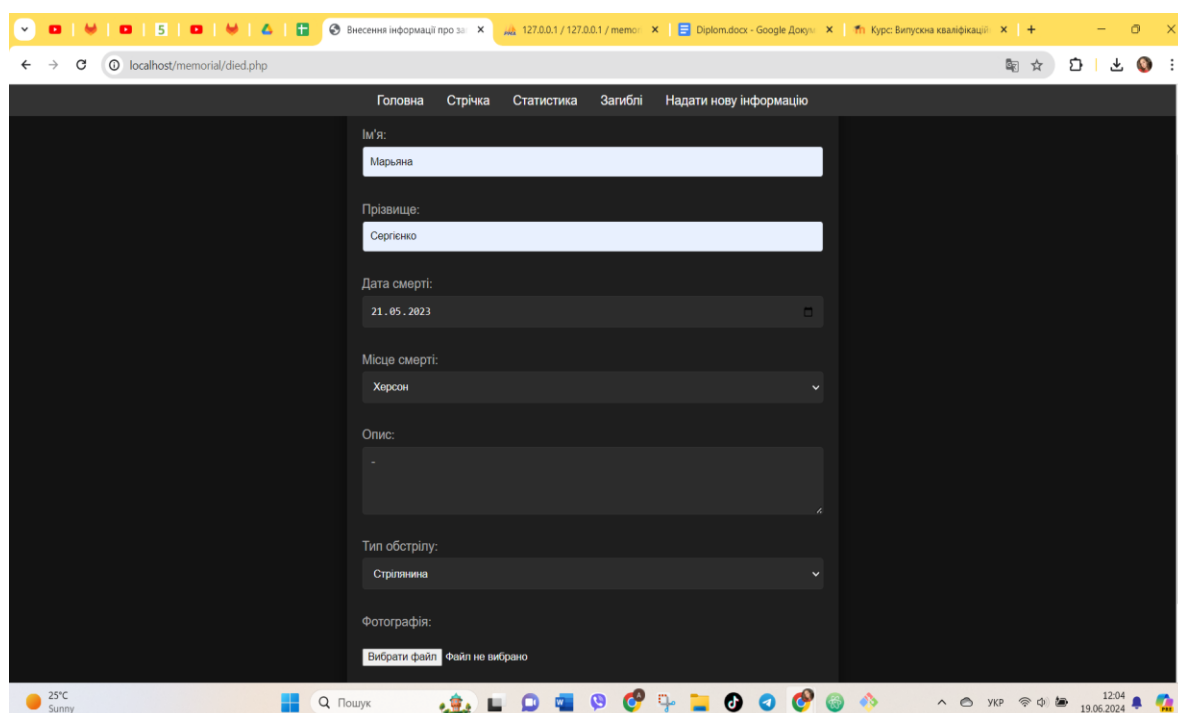


Рисунок 3.14 – Підготовлена інформація для додавання

Для перевірки інформації зайдемо у таблицю в базі даних (рис.3.15) та побачимо, що додалась нова інформація, а також відкриємо вебсторінку «Загиблі» та переглянемо і там нову інформацію (рис.3.16).

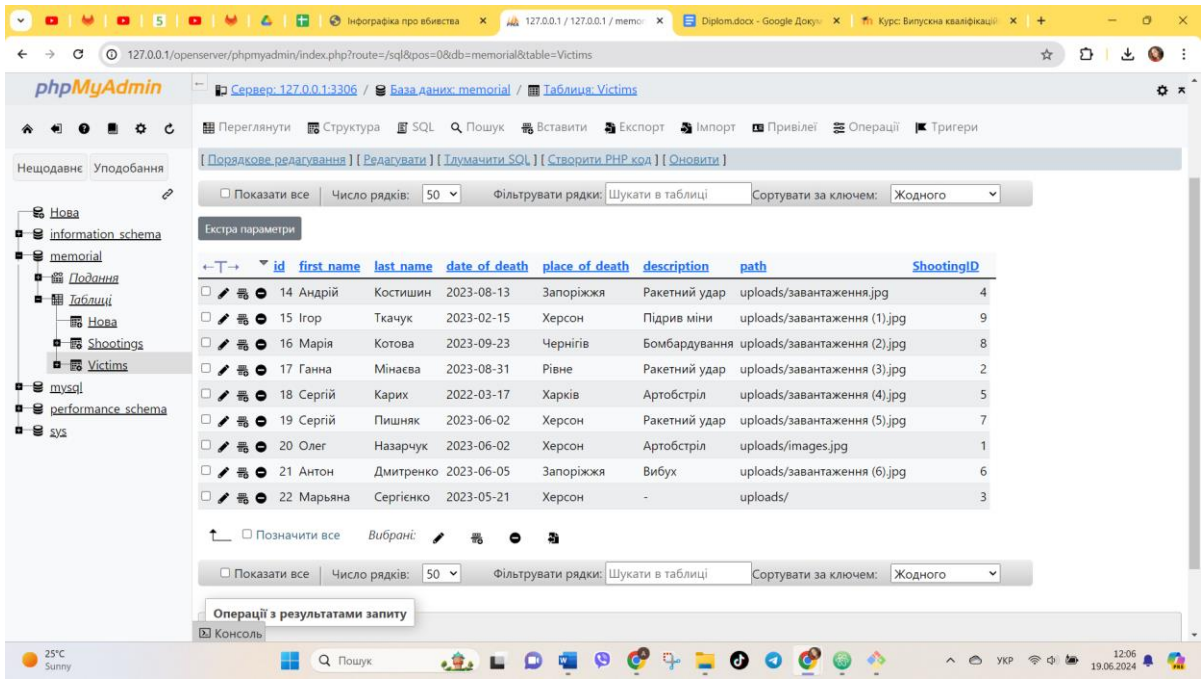


Рисунок 3.15 – Нова інформація у базі даних

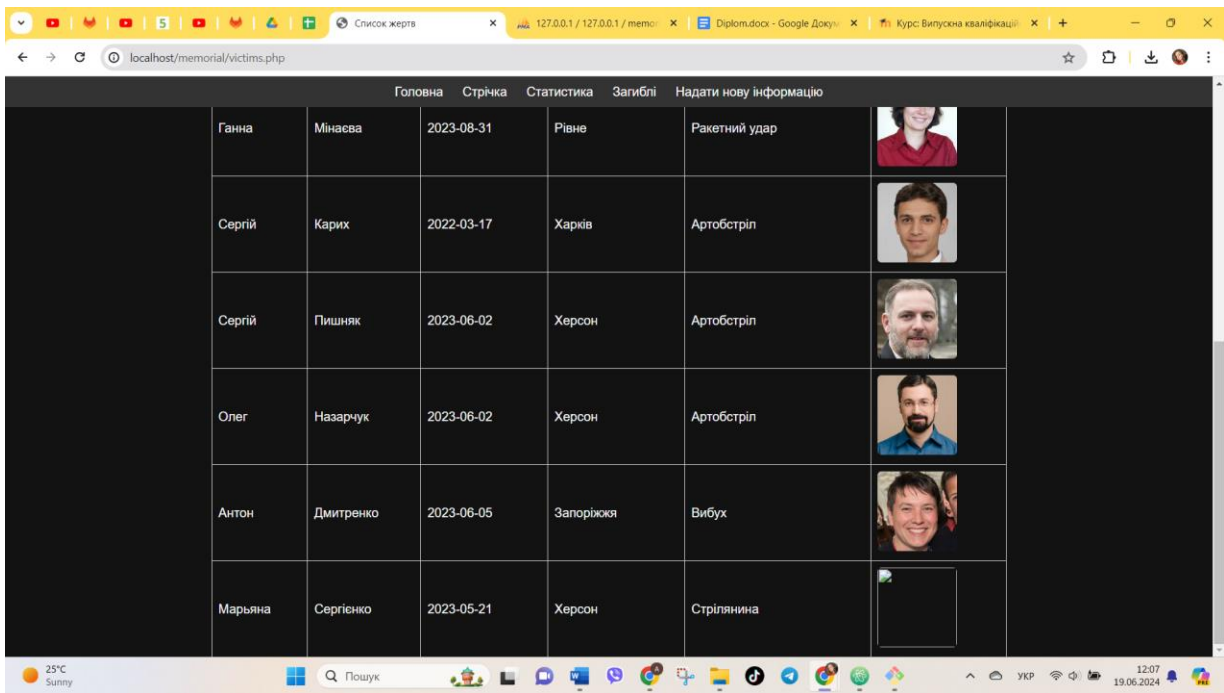


Рисунок 3.16 – Нова інформація на сторінці загиблі

Переглянемо сторінку зі статистикою (рис. 3.17 – рис. 3.19). Як видно з рисунку, то можна побачити, що статистика зростає з кожною новою жертвою у певному місті, даті смерті та по типах обстрілів.

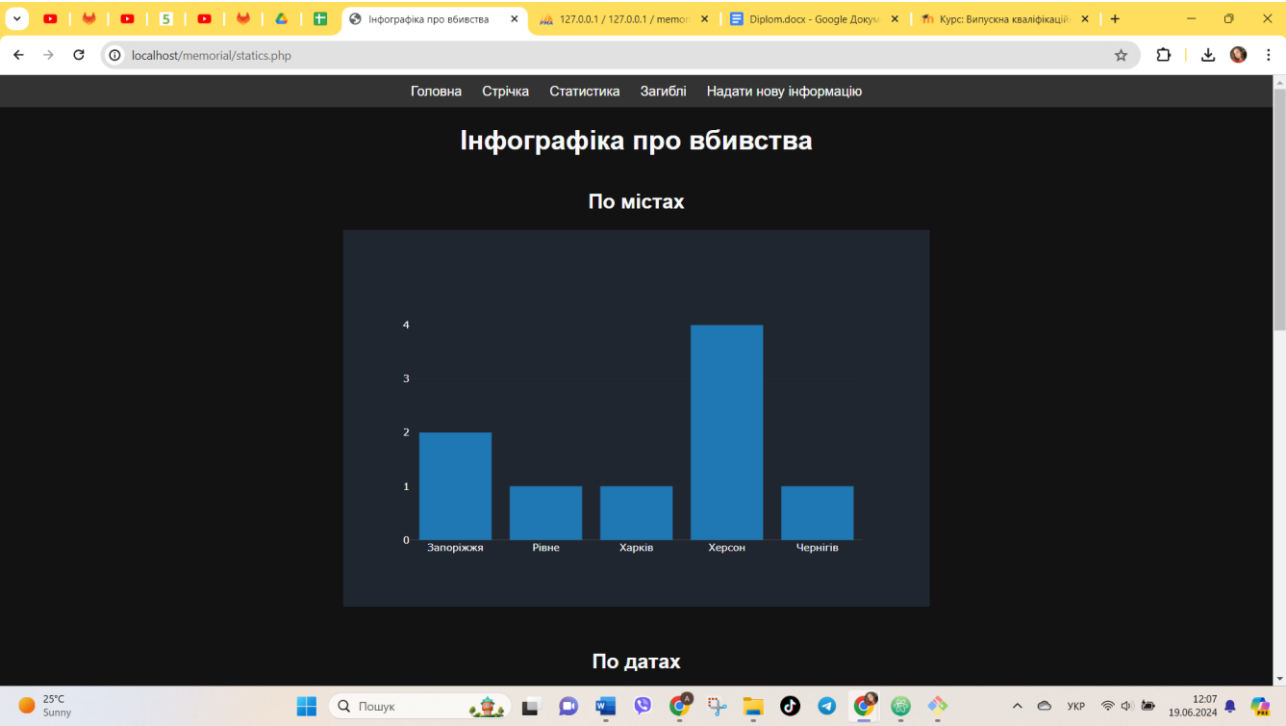


Рисунок 3.17 – Статистика по жертвам

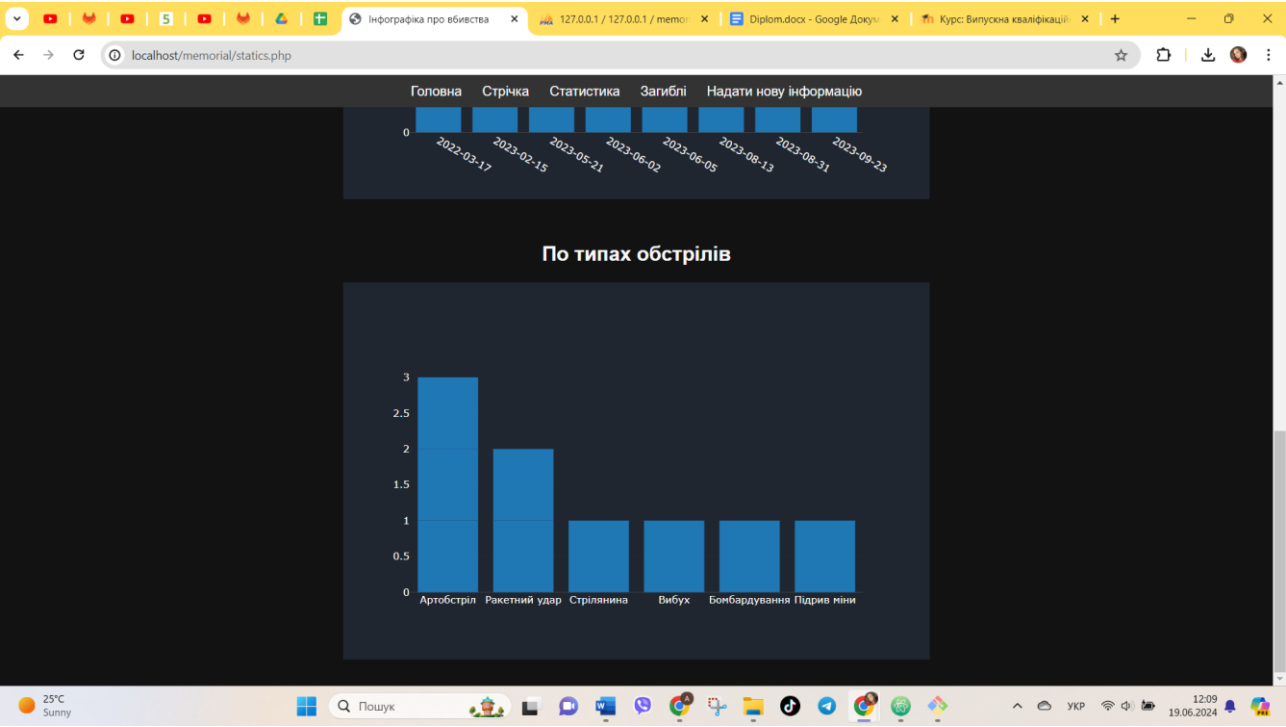


Рисунок 3.18 – Статистика по жертвам

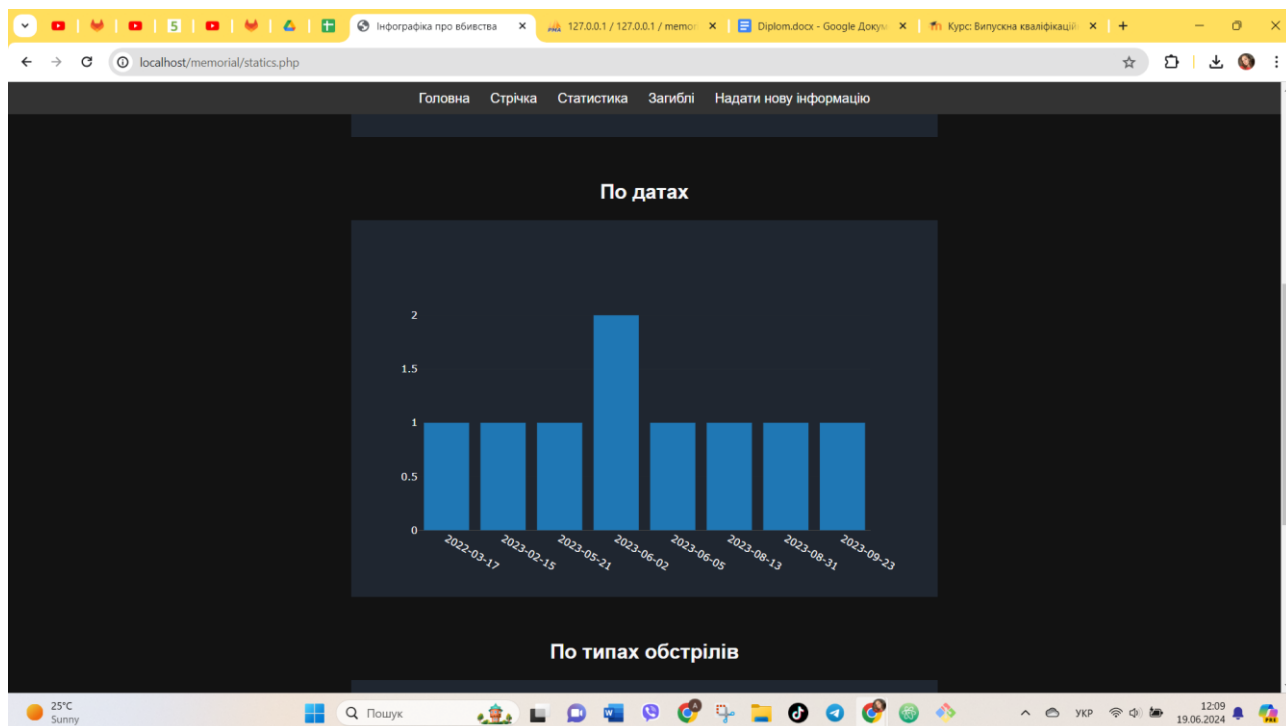


Рисунок 3.19 – Статистика по жертвам

Переглянемо сторінку зі списком загиблих, а саме фільтрацію даних по містам(рис. 3.20 – рис. 3.21), як ми бачимо згідно рисунку функція працює справно та полегшує пошуки користувачу.

Ім'я	Прізвище	Дата смерті	Місце смерті	Смерть внаслідок	Фото
Сергій	Карих	2022-03-17	Харків	Артобстріл	

Рисунок 3.20 – Сторінка «Загиблі»

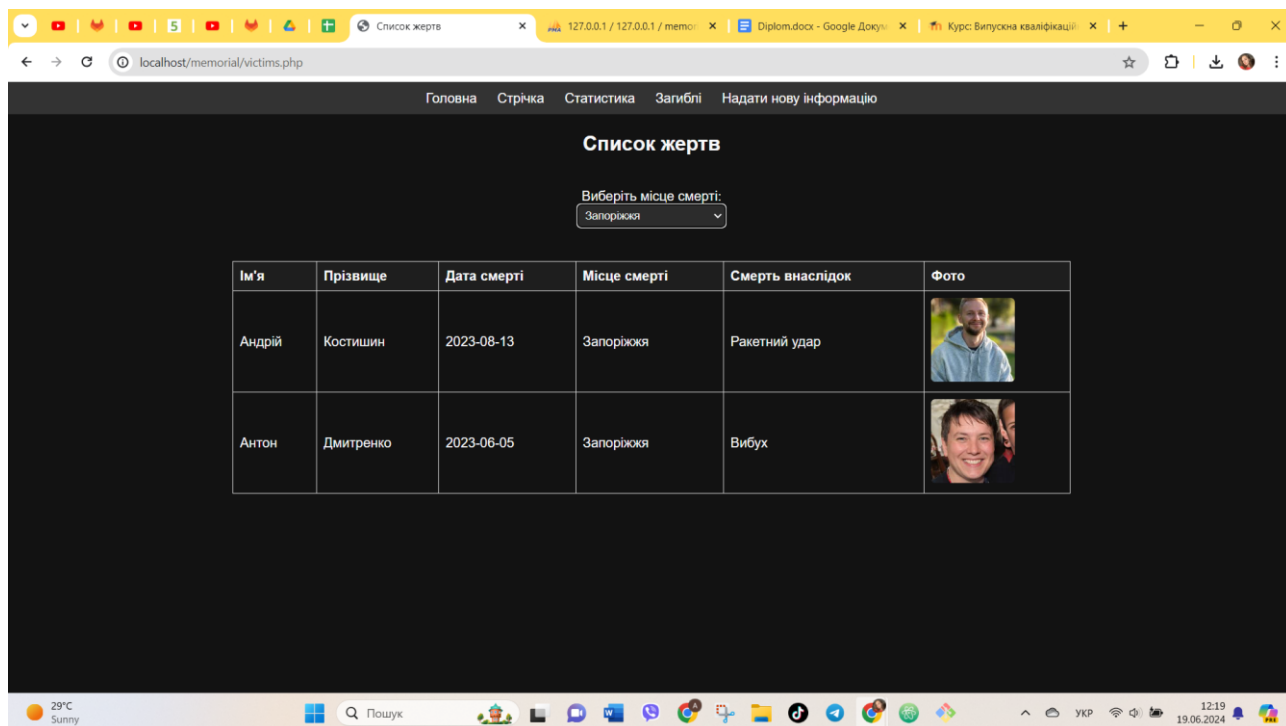


Рисунок 3.21 – Сторінка «Загиблі»

Висновки до третього розділу

У даному розділі розглянуто процес встановлення Open Server дуже простий і займає мало часу. Після завершення встановлення необхідно налаштувати його параметри відповідно до вимог проєкту проєкту.

Налаштування Open Server включає в себе вибір необхідних модулів для роботи вашого вебсайту.

Після успішного налаштування Open Server необхідно перевірити, чи запускається сервер правильно, і переконатися, що phpMyAdmin доступний для роботи з базою даних.

Для роботи з базою даних використовується phpMyAdmin, де можна створити нову базу даних або імпортувати / експортувати таблиці.

Після створення або імпортування бази даних необхідно розмістити файли вашого вебсайту в папці проєктів Open Server, перезавантажити сервер і перевірити роботу вебсайту через браузер.

Інформаційна система складається з 5 основних сторінок: «Головна», «Стрічка», «Статистика», «Загиблі» і «Надати нову інформацію».

Кожна сторінка має свою функціональність і призначення: відображення таймлайну подій, інформації про удари, статистики по жертвах, списку загиблих і можливості додавання нової інформації.

Система дозволяє динамічно завантажувати інформацію без тормозів сторінки, що покращує користувацький досвід.

Під час тестування сторінки «Надати нову інформацію» було успішно додано нову інформацію про загиблого користувача.

Інформація про загиблого користувача відображається як у базі даних, так і на вебсторінці «Загиблі».

Сторінка «Статистика» відображає зростання кількості жертв в певних містах, що дозволяє відстежувати тенденції та аналізувати дані.

ВИСНОВКИ

Під час виконання кваліфікаційної роботи було розроблено інформаційну систему, розробка якої показала, що у результаті аналізу постановки завдання та актуальності розроблюваної системи стало очевидним, що необхідність у такій платформі виникла внаслідок потреби в системі, яка б забезпечувала інформацією про російське вторгнення в Україну та його наслідки.

Огляд існуючих аналогів показав, що багато з них вже були створені, проте наша система має свої особливості та переваги, такі як динамічне завантаження інформації та можливість додавання нової інформації користувачами.

Обрані технології для розробки системи обґрунтовані та відповідають вимогам проекту, проте вимагають належного налаштування та інтеграції з іншими компонентами, такими як база даних.

Інсталяція та налаштування серверного середовища були успішно завершені, проте вимагали певного рівня технічних знань та уваги до деталей.

Створена система має чітку структуру та інтерфейс, що спрощує навігацію користувачів та забезпечує зручний доступ до інформації.

Під час тестування системи було виявлено та виправлено деякі недоліки, проте загалом робота системи була стабільною та ефективною.

Отже, отримані результати дозволяють зробити висновок, що розроблена система успішно відповідає своїм початковим цілям та вимогам і може ефективно використовуватися для подальшого вивчення та аналізу російського вторгнення в Україну та його наслідків.

ЛІТЕРАТУРА

1. Перехватова А. О. РОЗРОБКА МЕМОРІАЛУ ПАМ'ЯТІ: ІДЕЯ, КОНЦЕПЦІЯ ТА РЕАЛІЗАЦІЯ / А. О. Перехватова, Р. С. Савіцький. // КИЇВСЬКИЙ СТОЛИЧНИЙ УНІВЕРСИТЕТ ІМЕНІ БОРИСА ГРІНЧЕНКА. – 2024. – №11. – С. 85–86.
2. Бучин М. Інформаційна війна Росії проти України: особливості та механізми протидії [Електронний ресурс] / М. Бучин, Ю. Курус. – 2018. – Режим доступу до ресурсу: <https://science.lpnu.ua/sites/default/files/journal-paper/2018/jul/13541/buchynarticle120182.pdf>.
3. Юфіцинський Ю. СУЧАСНА РОСІЙСЬКО-УКРАЇНСЬКА ВІЙНА / Юрій Юфіцинський. – Ужгород: ТОВ "РІК-У", 2018. – 312 с.
4. Окупація Криму [Електронний ресурс] – Режим доступу до ресурсу: https://vue.gov.ua/%D0%9E%D0%BA%D1%83%D0%BF%D0%B0%D1%86%D1%96%D1%8F_%D0%9A%D1%80%D0%B8%D0%BC%D1%83#.D0.9B.D1.96.D1.82.D0.B5.D1.80.D0.B0.D1.82.D1.83.D1.80.D0.B0.
5. Анексія Криму. Головні наслідки для України [Електронний ресурс] – Режим доступу до ресурсу: <https://glavcom.ua/publications/124364-aneksija-krimu.-golovni-naslidki-dlja-ukrajini-doslidzhennja.html>.
6. Задорожний О. Анексія Криму — міжнародний злочин. Монографія. Київ : К.І.С., 2015. 576 с.
7. Марусяк О. Анексія Криму Російської Федерацією як злочин агресії проти України. Міжнародно-правові аспекти. Монографія. Чернівці : Місто, 2016. 220 с.
8. Кульчицький С., Якубова Л. Кримський вузол. Київ : КЛІО, 2018. 496 с.
9. Головка В., Якубова Л. Збройна агресія Росії проти України // 25 років незалежності. Нариси створення нації та держави. Київ : Ніка-Центр, 2016. С. 599–768.

10. Донбас-2014: як розгоралася велика війна [Електронний ресурс] // BBC News Україна. – 2024. – Режим доступу до ресурсу: <https://www.bbc.com/ukrainian/articles/cqend3g259lo>.
11. Зіневич А. О. ЕКОЛОГІЧНІ НАСЛІДКИ ВІЙНИ НА СХОДІ УКРАЇНИ / А. О. Зіневич, Г. А. Сербенюк. – 2023. – С. 24–25.
12. Чалюк Ю. ГЛОБАЛЬНІ СОЦІАЛЬНО-ЕКОНОМІЧНІ НАСЛІДКИ РОСІЙСЬКО-УКРАЇНСЬКОЇ ВІЙНИ [Електронний ресурс] / Юлія Чалюк. – 2022. – Режим доступу до ресурсу: <https://doi.org/10.32782/2524-0072/2022-37-54>.
13. Піляєв І. ТРАНСФОРМАЦІЯ ГЛОБАЛІЗАЦІЙНИХ ПРОЦЕСІВ У КОНТЕКСТІ НАСЛІДКІВ ПОВНОМАСШТАБНОГО РОСІЙСЬКОГО ВТОРГНЕННЯ В УКРАЇНУ [Електронний ресурс] / І.С. Піляєв. – 2022. – Режим доступу до ресурсу: http://economyukr.org.ua/?page_id=774&aid=547.
14. Бойко. О. Державне управління у сфері цивільного захисту: подальше вдосконалення законодавства в умовах дії воєнного стану. Науковий вісник: Державне управління. 2022. № 2(12). С. 195 – 217.
15. Веселов М., Забєділіна А. Цивільний захист населення в умовах воєнного стану. Зовнішня торгівля: економіка, фінанси, право. 2023. № 4. С. 98-114.
16. Демчук В. В. Реформування сфери цивільного захисту як складової безпекового простору в контексті європейської інтеграції України. Науковий вісник: державне управління. 2023. № 1. С. 102-122.
17. Клименко Н. Особливості забезпечення цивільного захисту в умовах воєнного стану. Науковий вісник: Державне управління. 2022. № 2(12). С. 218-233.
18. Кулєшов М. М., Ященко О. А. Актуальні питання реалізації завдань цивільного захисту в умовах сучасних викликів та загроз. Вісник Національного університету цивільного захисту України. Серія : Державне управління. 2022. Вип. 2. С. 294-303.

19. Платформа пам'яті меморіал [Електронний ресурс] – Режим доступу до ресурсу: <https://www.victims.memorial/>.
20. Мартиролог "Закриті очі" [Електронний ресурс] – Режим доступу до ресурсу: <https://closedeyes.org/>.
21. HTML5 [Електронний ресурс] – Режим доступу до ресурсу: <https://webplatform.github.io/docs/html/html5/>
22. CSS [Електронний ресурс] – Режим доступу до ресурсу: <https://developer.mozilla.org/en-US/docs/Web/CSS>
23. PHP [Електронний ресурс] – Режим доступу до ресурсу: <https://www.php.net/manual/en/index.php>
24. MySQL [Електронний ресурс] – Режим доступу до ресурсу: <https://dev.mysql.com/doc/>
25. JavaScript [Електронний ресурс] – Режим доступу до ресурсу: <https://devdocs.io/javascript/>
26. OSPanel [Електронний ресурс] – Режим доступу до ресурсу: <https://ospanel.io/docs/>
27. Манако В. Основи будування сайтів: підручник. Київ : видавничий дім «Шкільний світ», 2006. 133 с.
28. Злепко С. М., Тимчик С. В., Федосова І. В. та ін. Сучасні інформаційні технології в науці та освіті: навч. посіб. Вінниця : ВНТУ, 2018. с. 161.
29. Андреев А. С. Перспективи використання PHP та MYSQL в проектах / А. С. Андреев. – Харків, 2023. – 336 с. – (ADED).
30. C# language documentation [Електронний ресурс] – Режим доступу до ресурсу: <https://learn.microsoft.com/en-us/dotnet/csharp/>
31. Франчук В.М. Адміністрування веборієнтованих навчальних комп'ютерних систем: навч.-метод. посібник для студентів фізико-математичних та інформатичних спеціальностей вищих педагогічних навчальних закладів. Київ : Вид-во К: НПУ імені М.П. Драгоманова, 2016. 112 с.

32. Використання UML діаграм для аналізу і проектування інформаційних систем [Електронний ресурс] – Режим доступу до ресурсу: https://pns.hneu.edu.ua/pluginfile.php/321048/mod_resource/content/3/%D0%92%D0%B8%D0%BA%D0%BE%D1%80%D0%B8%D1%81%D1%82%D0%B0%D0%BD%D0%BD%D1%8F%20UML%20%D0%B4%D1%96%D0%B0%D0%B3%D1%80%D0%B0%D0%BC.pdf.
33. Інструкція, як будувати UML-діаграми [Електронний ресурс] – Режим доступу до ресурсу: <https://dou.ua/forums/topic/40575/>.
34. Розділ 2. Основи UML [Електронний ресурс] – Режим доступу до ресурсу: <https://docs.kde.org/trunk5/uk/umbrello/umbrello/uml-basics.html>.
35. ВИКОРИСТАННЯ UML ДІАГРАМ ДЛЯ ТЕХНОЛОГІЧНОГО ПРОЦЕСУ НА ВИРОБНИЦТВІ [Електронний ресурс] / Л.Самчук, Ю. Повстяна, Н. Ліщина, А. Кліменко // Сторінки: 268-275. Номер: №2, 2023 (319). – 2023. – Режим доступу до ресурсу: <http://journals.khnu.km.ua/vestnik/wp-content/uploads/2023/05/vknu-ts-2023-n2319-268-275.pdf>.
36. Огляд інструментів для побудови UML-діаграм класів [Електронний ресурс] – Режим доступу до ресурсу: https://essuir.sumdu.edu.ua/bitstream-download/123456789/31610/1/Kodola_Volinets.pdf;jsessionid=5D722F7B84A586A60F270ABC7DF5CFFB.
37. Хацько Н.Є., Гавриленко С.Ю. Методичні вказівки до виконання лабораторної роботи “Розробка діаграми варіантів використання у середовищі Umbrello UML Modeller” : для студентів спеціальності 121 – Інформаційні технології. Англійською та українською мовами. – Харків : НТУ «ХПІ», 2019. – 40 с.
38. Лекція 8. Елементи нотації діаграм UML для моделювання програмного забезпечення [Електронний ресурс] – Режим доступу до ресурсу: <https://moodle.znu.edu.ua/mod/folder/view.php?id=36352>.

ДОДАТКИ

ДОДАТОК А – ЛІСТИНГ БАЗИ ДАНИХ

```
-- База даних: `memorial`  
CREATE TABLE `citymurders` (  
  `City` varchar(255)  
  , `TotalMurders` bigint(21)  
);  
-- Структура таблиці `Shootings`  
--  
CREATE TABLE `Shootings` (  
  `ShootingID` int(11) NOT NULL,  
  `Location` varchar(255) COLLATE utf8mb4_unicode_ci DEFAULT NULL,  
  `Date` date DEFAULT NULL,  
  `Description` text COLLATE utf8mb4_unicode_ci,  
  `ImagePath` varchar(255) COLLATE utf8mb4_unicode_ci DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;  
  
INSERT INTO `Shootings` (`ShootingID`, `Location`, `Date`, `Description`, `ImagePath`) VALUES  
(1, 'Київ', '2022-02-24', 'Артобстріл', 'uploads/image1.jpeg'),  
(2, 'Харків', '2022-02-27', 'Ракетний удар', '/img/shooting/image2.jpg'),  
(3, 'Одеса', '2022-03-01', 'Стрілянина', '/img/shooting/image3.jpg'),  
(4, 'Харків', '2022-02-27', 'Ракетний удар', '/img/shooting/image2.jpg'),  
(5, 'Одеса', '2022-03-01', 'Стрілянина', '/img/shooting/image3.jpg'),  
(6, 'Львів', '2022-03-05', 'Бомбардування', '/img/shooting/image4.jpg'),  
(7, 'Дніпро', '2022-03-10', 'Артобстріл', '/img/shooting/image5.jpg'),  
(8, 'Херсон', '2022-03-15', 'Вибух', '/img/shooting/image6.jpg'),  
(9, 'Запоріжжя', '2022-03-20', 'Підрих міни', '/img/shooting/image7.jpg');  
-- Структура таблиці `Victims`  
CREATE TABLE `Victims` (  
  `id` int(11) NOT NULL,  
  `first_name` varchar(255) COLLATE utf8mb4_unicode_ci DEFAULT NULL,  
  `last_name` varchar(255) COLLATE utf8mb4_unicode_ci DEFAULT NULL,  
  `date_of_death` date DEFAULT NULL,  
  `place_of_death` varchar(255) COLLATE utf8mb4_unicode_ci DEFAULT NULL,  
  `description` text COLLATE utf8mb4_unicode_ci,  
  `path` varchar(255) COLLATE utf8mb4_unicode_ci DEFAULT NULL,  
  `ShootingID` int(11) DEFAULT NULL
```

```

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;

--
INSERT INTO `Victims` (`id`, `first_name`, `last_name`, `date_of_death`, `place_of_death`,
`description`, `path`, `ShootingID`) VALUES
(1, 'Марія', 'Антонюк', '2024-02-27', 'Київ', 'Артобстріл', 'uploads/image1.jpeg', 1),
(2, 'Олена', 'Шивак', '2024-04-11', 'Київ', 'Артобстріл', 'uploads/image1.jpeg', 1),
(3, 'Степан', 'Стефанчук', '2022-05-14', 'Маріуполь', 'Артобстріл', 'uploads/image1.jpeg', 1),
(4, 'Христина', 'Опришко', '2023-06-04', 'Одеса', 'Ракерний удар', NULL, 1),
(5, 'Соломія', 'Козачок', '2023-07-15', 'Одеса', 'Бомбардування', NULL, 1),
(6, 'Олександр', 'Винченко', '2024-03-04', 'Миколаїв', 'Ракерний удар', NULL, 1),
(7, 'Петро', 'Поліщук', '2023-05-06', 'Харків', 'Ракетний удар', NULL, 1);
-- Структура для представлення `citymurders`
CREATE ALGORITHM=UNDEFINED DEFINER=`root`@`%` SQL SECURITY DEFINER VIEW
`citymurders` AS
SELECT
    `v`.`place_of_death` AS `City`,
    COUNT(0) AS `TotalMurders`
FROM
    `victims` `v`
JOIN
    `shootings` `s`
ON
    (`v`.`ShootingID` = `s`.`ShootingID`)
GROUP BY
    `v`.`place_of_death`;
ALTER TABLE `Shootings`
    ADD PRIMARY KEY (`ShootingID`);
ALTER TABLE `Victims`
    ADD PRIMARY KEY (`id`);
ALTER TABLE `Victims`
    MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=14;
COMMIT;
/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;

```

ДОДАТОК Б – РОЗРОБКА ВЕБ-ОРІЄНТОВАНОЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ

database.php

```
<?php
$mysqli = false;
// Функція для підключення до бази даних
function connectDB() {
    global $conn;
    // Підключення до MySQL сервера
    $conn = new mysqli("localhost", "root", "", "memorial");
    // Встановлення кодування для з'єднання
    $conn->query("SET NAMES 'utf8'");
    // Перевірка підключення
    if ($conn->connect_error) {
        die("Connection failed: " . $conn->connect_error);
    }
}
?>
```

died.php

```
<title>Внесення інформації про загиблого</title>
<style>
body {
    font-family: 'Arial', sans-serif;
    background-color: #121212;
    ....
button:hover {
    background-color: #0056b3;
}
</style>
<h2>Внесення інформації про загиблого</h2>
<form action="insert_victim.php" method="post" enctype="multipart/form-data">
....
    <option selected value="0">Оберіть тип обстрілу</option>
<?php
// Підключення до бази даних
```

```

require_once "database.php";
global $conn;
connectDB();
// Запит до бази даних для отримання унікальних значень Description
$sql = "SELECT DISTINCT Description, ShootingID FROM Shootings";
$result = $conn->query($sql);
// Перевірка наявності результатів запиту
if ($result) {
    // Виведення опцій для вибору типу обстрілу
    while($row = $result->fetch_assoc()) {
        $description = $row['Description'];
        $shooting_id = $row['ShootingID'];
        // Перевірка, чи такий опис обстрілу ще не був виведений
        $sql_check = "SELECT COUNT(*) AS count FROM Victims WHERE
ShootingID = ?";
        $stmt_check = $conn->prepare($sql_check);
        $stmt_check->bind_param("i", $shooting_id);
        $stmt_check->execute();
        $result_check = $stmt_check->get_result();
        $row_check = $result_check->fetch_assoc();
        if ($row_check['count'] == 0) {
            echo "<option value=\"" . $shooting_id . "\">" . $description . "</option>";
        } } }
    // Закриття з'єднання з базою даних
    $conn->close();    ?>
</select>
// Приховане поле для передачі shootingID в POST
<input type="hidden" id="shooting_id" name="shooting_id">
</div>
<script>
function getShootingID() {
    let selectBox = document.getElementById("shooting_type");
    let selectedValue = selectBox.options[selectBox.selectedIndex].value;
    // Записуємо значення shootingID в приховане поле
    document.getElementById("shooting_id").value = selectedValue;

```

```
}  
</script>
```

get.shootings.php

```
<?php  
// Підключення до бази даних  
require_once "database.php";  
global $conn;  
connectDB();  
// Виконання запиту до бази даних для отримання інформації з таблиці "Shootings"  
$query = "SELECT * FROM Shootings";  
$result = mysqli_query($conn, $query);  
// Перевірка наявності результатів  
if ($result) {  
    // Цикл для перегляду кожного рядка результатів  
    while ($row = mysqli_fetch_assoc($result)) {  
        // Виведення інформації про кожен обстріл у форматі HTML  
        echo "<div class='card'>";  
        echo "<h2>{$row['Location']}</h2>";  
        echo "<p>{$row['Date']}</p>";  
        echo "<p>{$row['Description']}</p>";  
        // Перевірка наявності шляху до зображення  
        if (!empty($row['ImagePath'])) {  
            echo "<img src='{$row['ImagePath']}' alt='Shooting Image'>";  
        } else {  
            echo "<p>Зображення відсутнє</p>";  
        }  
        echo "</div>";  
    }  
    // Звільнення результатів  
    mysqli_free_result($result);  
} else {  
    // Якщо виникла помилка запиту  
    echo "Помилка: " . mysqli_error($conn);  
}
```

					ІПЗ.КР.Б-121-24-ПЗ	
						63

?>

insert_victim.php

```
<?php
```

```
// Підключення до бази даних
```

```
require_once "database.php";
```

```
global $conn;
```

```
connectDB();
```

```
// Перевірка наявності значення в $_POST['shooting_type']
```

```
if(isset($_POST['shooting_id'])) {
```

```
    // Отримання ідентифікатора обстрілу (ShootingID) на основі вибраного типу обстрілу
```

```
    $shooting_id = $_POST['shooting_id'];
```

```
    // Обробка і збереження введених даних
```

```
    $first_name = $_POST['first_name'];
```

```
    $last_name = $_POST['last_name'];
```

```
    $date_of_death = $_POST['date_of_death'];
```

```
    $place_of_death = $_POST['place_of_death'];
```

```
    $description = $_POST['description'];
```

```
    // Збереження фотографії
```

```
    $target_dir = "uploads/";
```

```
    $target_file = $target_dir . basename($_FILES["photo"]["name"]);
```

```
    move_uploaded_file($_FILES["photo"]["tmp_name"], $target_file);
```

```
    // SQL-запит для вставки даних з ShootingID
```

```
    $sql = "INSERT INTO Victims (first_name, last_name, date_of_death, place_of_death, description, path, ShootingID)
```

```
    VALUES (?, ?, ?, ?, ?, ?, ?)";
```

```
    $stmt = $conn->prepare($sql);
```

```
    $stmt->bind_param("sssssi", $first_name, $last_name, $date_of_death, $place_of_death, $description, $target_file, $shooting_id);
```

```
    // Виконання запиту
```

```
    if ($stmt->execute()) {
```

```
        echo "Інформація про загиблого успішно додана.";
```

```
        header("Location: index.php"); // Перенаправлення на index.php
```

```
        exit; // Важливо додати exit, щоб забезпечити коректну роботу перенаправлення
```

```
    } else {
```

					ІІЗ.КР.Б-121-24-ІІЗ	
						64

```

        echo "Помилка: " . $sql . "<br>" . $conn->error;
    }
} else {
    echo "Помилка: Не отримано значення ідентифікатора обстрілу.";
}
// Закриття з'єднання
$conn->close();
?>

```

lenta.php

```

<!-- JavaScript для взаємодії з сервером та відображення даних -->
<script>
    // Коли сторінка завантажується, відправте запит на сервер для отримання даних
    про обстріли
    document.addEventListener("DOMContentLoaded", function() {
        // Використовуйте XMLHttpRequest або fetch для відправки запиту на сервер
        var xhr = new XMLHttpRequest();
        xhr.open("GET", "get_shootings.php", true);
        xhr.onload = function() {
            if (xhr.status == 200) {
                // Отримані дані про обстріли у форматі HTML
                document.getElementById("shootingsContainer").innerHTML =
xhr.responseText;
            }
        };
        xhr.send();
    });
</script>

```

statics.php

```

<?php
// Підключення до бази даних
require_once "database.php";
global $conn;
connectDB();
// SQL-запит для отримання статистики по містах
$sqlTable1 = "SELECT

```

```

        `v`.`place_of_death` AS `City`,
        COUNT(0) AS `TotalMurders`
FROM
    `victims` `v`
JOIN
    `shootings` `s`
ON
    (`v`.`ShootingID` = `s`.`ShootingID`)
GROUP BY
    `v`.`place_of_death`;
// SQL-запит для отримання статистики по датах
$sqlTable2 = "SELECT
    `v`.`date_of_death` AS `Date`,
    COUNT(0) AS `TotalMurders`
FROM
    `victims` `v`
JOIN
    `shootings` `s`
ON
    (`v`.`ShootingID` = `s`.`ShootingID`)
GROUP BY
    `v`.`date_of_death`;
// SQL-запит для отримання статистики по типах обстрілів
$sqlTable3 = "
SELECT
    `s`.`description` AS `FireType`,
    COUNT(0) AS `TotalMurders`
FROM
    `victims` `v`
JOIN
    `shootings` `s`
ON
    (`v`.`ShootingID` = `s`.`ShootingID`)
GROUP BY
    `s`.`ShootingID`

```

```

";
$resultTable1 = $conn->query($sqlTable1);
$resultTable2 = $conn->query($sqlTable2);
$resultTable3 = $conn->query($sqlTable3);
// Створення масивів для даних графіка
$cities = array();
$murders1 = array();
$dates = array();
$murders2 = array();
$fireTypes = array();
$murders3 = array();
// Обробка результатів запиту
if ($resultTable1->num_rows > 0) {
    while ($row = $resultTable1->fetch_assoc()) {
        array_push($cities, $row["City"]);
        array_push($murders1, $row["TotalMurders"]);
    }
}
{
// Обробка результатів запиту
if ($resultTable2->num_rows > 0) {
    while ($row = $resultTable2->fetch_assoc()) {
        array_push($dates, $row["Date"]);
        array_push($murders2, $row["TotalMurders"]);
    }
}
{
// Обробка результатів запиту
if ($resultTable3->num_rows > 0) {
    while ($row = $resultTable3->fetch_assoc()) {
        array_push($fireTypes, $row["FireType"]);
        array_push($murders3, $row["TotalMurders"]);
    }
}

// Закриття з'єднання з базою даних
$conn->close();
?>

```

victims.php

```
<table id="victimTable">
  <tr>
    <th class="hidden">ID</th>
    <th>Ім'я</th>
    <th>Прізвище</th>
    <th>Дата смерті</th>
    <th>Місце смерті</th>
    <th>Смерть внаслідок</th>
    <th>Фото</th>
  </tr>
<?php
// Підключення до бази даних
require_once "database.php";
global $conn;
connectDB();

// SQL-запит для вибору даних
$sql = "SELECT v.id, v.first_name, v.last_name, v.date_of_death, v.place_of_death,
s.description, v.path, v.shootingID
FROM victims v
INNER JOIN shootings s ON v.shootingID = s.ShootingID"; // Змінено запит для
вибору даних та додано JOIN з таблицею shootings
$result = $conn->query($sql);
// Виведення даних
if ($result->num_rows > 0) {
  while($row = $result->fetch_assoc()) {
    echo "<tr>
      <td class='hidden'>".$row["id"]."</td>
      <td>".$row["first_name"]."</td>
      <td>".$row["last_name"]."</td>
      <td>".$row["date_of_death"]."</td>
      <td>".$row["place_of_death"]."</td>
      <td>".$row["description"]."</td> <!-- Змінено виведення description -->
      <td><img src='".$row["path"]."' style='width:100px;height:100px;'></td>
    </tr>";
```

```

    }
  } else {
    echo "0 results";
  }
  $conn->close();
?>
</table>
<script>
function filterVictims() { //фільтр таблиці за місцем смерті
  var selectBox = document.getElementById("place_of_death");
  var selectedValue = selectBox.options[selectBox.selectedIndex].value;
  var tableRows = document.querySelectorAll("#victimTable tr");
  // Перевірка чи обрано "Все"
  if (selectedValue === "all") {
    // Показати всі рядки
    for (var i = 1; i < tableRows.length; i++) {
      tableRows[i].style.display = "";
    }
  } else {
    // Фільтрувати за обраним місцем смерті
    for (var i = 1; i < tableRows.length; i++) {
      var row = tableRows[i];
      var placeOfDeathCell = row.cells[4]; // 5-й стовпчик (індекс 4) - "Місце смерті"
      if (placeOfDeathCell.innerHTML !== selectedValue) {
        row.style.display = "none";
      } else {
        row.style.display = ""; // Відобразити рядок
      }
    }
  }
}
</script>

```