

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО”

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

До захисту допущено:

Завідувач кафедри

Сергій СТИРЕНКО

(підпис)

“__” _____ 2024 р.

Дипломний проєкт

на здобуття ступеня бакалавра

за освітньо-професійною програмою “Комп’ютерні мережі та системи”
спеціальності 123 “Комп’ютерна інженерія”

на тему: Веб-ресурс для підтримки та інформування громадян під час
воєнних дій

Виконав : студентка 4 курсу, групи ІО-04
(шифр групи)

Бондаренко Юлія Андріївна

(прізвище, ім’я, по батькові)

(підпис)

Керівник асистент, Щур В.Ю.

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

(підпис)

Консультант(нормоконтроль) асистент, Іваніщев Б.В.

(назва розділу)

(посада, вченезвання, науковий ступінь, прізвище та ініціали)

(підпис)

Рецензент доц. каф. ІСТ, к.т.н., доц., Крилов Є.В.

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

(підпис)

Засвідчую, що у цьому дипломному
проєкті немає запозичень з праць інших
авторів без відповідних посилань.

Студентка _____

(підпис)

Київ – 2024 р.

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО”**

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

Рівень вищої освіти – перший (бакалавр)

Освітньо-професійна програма

“Комп’ютерні мережі та системи”

спеціальності 123 “Комп’ютерна інженерія”

ЗАТВЕРДЖУЮ

Завідувач кафедри

Сергій СТИРЕНКО

(підпис)

“ ___ ” _____ 2024 р.

ЗАВДАННЯ

на бакалаврський дипломний проєкт студента

Бондаренко Юлії Андріївни

1. Тема проєкту Веб-ресурс для підтримки та інформування громадян під час воєнних дій
керівник проєкту Щур Вадим Юрійович, асистент,
(прізвище, ім’я, по батькові, науковий ступінь, вчене звання)
затверджені наказом по університету 27 травня 2024 року № 2112-с
2. Термін здачі студентом закінченого проєкту 3 червня 2024 р.
3. Вихідні дані до проєкту технічна документація, теоретичні дані
4. Зміст розрахунково-пояснювальної записки (перелік питань, які розробляються)
Розділ 1. Огляд та аналіз існуючих інформаційних систем
Розділ 2. Огляд і вибір технологій для розробки.

Розділ 3. Проектування та розробка інформаційного веб-ресурсу.

Розділ 4. Аналіз та тестування розробленого продукту.

5. Перелік графічного матеріалу (з точним позначенням обов'язкових креслень) структурна схема системи, діаграма даних (функціональна схема), алгоритм дії форми зворотнього зв'язку (принципова схема)

6. Консультанта проєкту, з вказівкою розділів проєкту, які до них вносяться

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв
Нормоконтроль	Іваніщев Б.В.		

7. Дата видачі завдання «23» жовтня 2023 р.

Календарний план

№ П/П	Найменування етапів дипломного проєкту	Терміни виконання етапів проєкту	Примітки
1.	<i>Затвердження теми проєкту</i>	<i>08.04.2024-13.04.2024</i>	
2.	<i>Вивчення та аналіз завдання</i>	<i>15.04.2024-20.04.2024</i>	
3.	<i>Розробка архітектури та загальної структури системи</i>	<i>22.04.2024-03.05.2024</i>	
4.	<i>Розробка структур окремих підсистем</i>	<i>02.05.2024-11.05.2024</i>	
5.	<i>Програмна реалізація системи</i>	<i>13.05.2024-28.05.2024</i>	
6.	<i>Оформлення пояснювальної записки</i>	<i>27.05.2024-31.05.2024</i>	
7.	<i>Захист програмного продукту</i>	<i>03.06.2024</i>	
8.	<i>Передзахист</i>	<i>06.06.2024</i>	
9.	<i>Захист</i>	<i>17.06.2024</i>	

Студент-дипломник _____ Юлія БОНДАРЕНКО
(підпис)

Керівник проєкту _____ Вадим ЩУР
(підпис)

АНОТАЦІЯ

Даний дипломний проєкт присвячений розробці інформаційного веб-ресурсу з додатковою підтримкою для цивільного населення під час бойових дій. Був проведений аналіз вже існуючих інформаційних порталів та враховано всі переваги та недоліки.

Сервіс надає користувачеві змогу отримати актуальну та необхідну інформацію, щодо дій та поведінки в кризових ситуаціях, надає можливість оформити запис на семінар та включає в себе інструменти для зворотнього зв'язку. Розроблено архітектуру серверної та користувацької частини веб-сервісу, графічні елементи та інтерфейс веб-сторінок. Особлива увага була приділена для забезпечення зручності та доступності використання ресурсу широкій аудиторії. З метою підвищення продуктивності та безпеки були використанні сучасні технології розробки.

Ключові слова: інформаційна система, веб-ресурс, допомога громадян в умовах воєнних дій, MySQL, SPA.

ANNOTATION

This diploma project is devoted to the development of an information web resource with additional support for the civilian population during hostilities. An analysis of already existing information portals was carried out and all advantages and disadvantages were taken into account.

The service enables the user to obtain relevant and necessary information on actions and behavior in crisis situations, provides an opportunity to register for a seminar and includes tools for feedback. The architecture of the server and user parts of the web service, graphic elements and the interface of web pages have been developed. Special attention was paid to ensuring accessibility and ease of use of the resource for a wide audience. In order to increase productivity and safety, modern development technologies were used.

Keywords: information system, web resource, assistance to citizens in conditions of military operations, MySQL, SPA.

Довідки	Формат	Значення	Найменування	Кіл. листів	№ ек-земпляр	Додаток
			Документація загальна			
			Знову розроблена			
	<i>A4</i>	<i>ІАЛЦ.467200.002 ТЗ</i>	Веб-ресурс для підтримки та інформування громадян під час воєнних дій	4		
			Технічне завдання			
	<i>A4</i>	<i>ІАЛЦ.467200.003 ПЗ</i>	Веб-ресурс для підтримки та інформування громадян під час воєнних дій	66		
			Пояснювальна записка			
	<i>A4</i>	<i>ІАЛЦ.467200.004 Д1</i>	Веб-ресурс для підтримки та інформування громадян під час воєнних дій	1		
			Структурна схема системи			
	<i>A4</i>	<i>ІАЛЦ.4672008.005 Д2</i>	Веб-ресурс для підтримки та інформування громадян під час воєнних дій	1		
			Діаграма даних (функціональна схема)			
	<i>A4</i>	<i>ІАЛЦ.467200.006 Д3</i>	Веб-ресурс для підтримки та інформування громадян під час воєнних дій	1		
			Алгоритм дії форми зворотнього зв'язку (принципова схема)			
	<i>A4</i>	<i>ІАЛЦ.467200.007 Д4</i>	Веб-ресурс для підтримки та інформування громадян під час воєнних дій	8		
			Текст програмного коду			

					<i>ІАЛЦ.467200.001 ОА</i>		
<i>Зм</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Підп</i>	<i>Дата</i>			
<i>Розроб</i>		Бондаренко Ю.А.			<i>Літ.</i>	<i>Аркуш</i>	<i>Аркушів</i>
<i>Перев</i>		Щур В.Ю.				1	1
					<i>Веб-ресурс для підтримки та інформування громадян під час воєнних дій</i> Опис альбому		
					<i>КПІ ім. Ігоря Сікорського, ФІОТ, ІО-04</i>		

ТЕХНІЧНЕ ЗАВДАННЯ
ДО ДИПЛОМНОГО ПРОЄКТУ

на тему: «Веб-ресурс для підтримки та інформування громадян
під час воєнних дій»

Київ – 2024

ЗМІСТ

1. НАЙМЕНУВАННЯ ТА ОБЛАСТЬ ЗАСТОСУВАННЯ	2
2. ПІДСТАВИ ДЛЯ РОЗРОБКИ	2
3. МЕТА ТА ПРИЗНАЧЕННЯ РОЗРОБКИ.....	2
4. ДЖЕРЕЛА РОЗРОБКИ.....	2
5. ТЕХНІЧНІ ВИМОГИ.....	3
5.1. Вимоги до розробленого продукту	3
5.2. Вимоги до програмного забезпечення.....	3
5.3. Вимоги до апаратної частини	3
6. ЕТАПИ РОЗРОБКИ	4

					ІАЛЦ.467200.002 ТЗ			
		№ докум.	Підпис	Дата				
Розробив	Бондаренко Ю.А.				Веб-ресурс для підтримки та інформування громадян під час воєнних дій Технічне завдання	Літ.	Аркуш	Аркушів
Перевірив	Щур В.Ю.						1	4
Н. Контр.	Іваніщев Б.В.					КПІ ім. Ігоря Сікорського, ФІОТ, ІО-04		
Затвердив								

1 НАЙМЕНУВАННЯ ТА ОБЛАСТЬ ЗАСТОСУВАННЯ

Дане технічне завдання включає в себе розробку інформаційного веб-застосунку з додатковими засобами підтримки громадян.

Область застосування цієї системи є забезпечення необхідною інформацією населення про поточну ситуацію, шляхи вирішення критичних проблем та можливість отримання підтримки та порад з питань психологічної та медичної допомоги під час воєнних дій.

2 ПІДСТАВИ ДЛЯ РОЗРОБКИ

Підставою для розробки даної системи є завдання для виконання роботи кваліфікаційно-освітнього рівня «бакалавр комп'ютерної користувач інженерії», який був затверджений факультетом “Інформатики та обчислювальної техніки” кафедрою обчислювальної техніки Національного технічного Університету України «Київський Політехнічний інститут ім. Ігоря Сікорського».

3 МЕТА ТА ПРИЗНАЧЕННЯ РОЗРОБКИ

Метою та призначенням даної роботи є розробка веб-ресурсу, який буде слугувати надійним ресурсом для оперативного отримання актуальної та достовірної інформації, щодо захисту власного життя під час воєнних дій. Основна мета цього проєкту – допомогти людям зорієнтуватися в ситуації, знайти необхідну допомогу та звернутися на консультації з питань медичного обслуговування та психологічного стану, які можуть стати необхідним інструментом для подолання труднощів війни.

4 ДЖЕРЕЛА РОЗРОБКИ

Джерелом розробки даного дипломного проєкту є офіційні документації, технологій, які використовувались для розробки, науково-технічна література.

					ІАЛЦ.467200.002 ТЗ	Арк.
						2
Зм.	Арк.	№ докум.	Підпис	Дата		

5 ТЕХНІЧНІ ВИМОГИ

5.1. Вимоги до розробленого продукту

Розроблена система має виконувати такі вимоги:

- Інтуїтивний та зрозумілий інтерфейс для користувачів.
- Надати швидкий доступ до необхідної інформації у випадках кризових ситуацій.
- Надати можливість реєстрації користувача на семінарі по медичній допомозі та психологічній підтримці.
- Мати вичерпну та зрозумілу документацію.

5.2. Вимоги до програмного забезпечення

- Операційна система ОС Mac, Linux чи Windows.
- Веб-браузер Google Chrome, Mozilla Firefox, Opera.
- Доступ до інтернету.

5.3. Вимоги до апаратної частини

- ЦП не менше ніж Intel® Core (TM) i3-2100T.
- ROM не менше ніж 64 ГБ.
- RAM не менше ніж 4 ГБ.

					ІАЛЦ.467200.002 ТЗ	Арк.
						3
Зм.	Арк.	№ докум.	Підпис	Дата		

6 ЕТАПИ РОЗРОБКИ

Назва етапів виконання	Термін виконання
Затвердження теми роботи	08.04.2024-13.04.2024
Вивчення та аналіз завдання	15.04.2024-20.04.2024
Розробка архітектури та загальної структури системи	22.04.2024-03.05.2024
Розробка структур окремих частин системи	02.05.2024-11.05.2024
Програмна реалізація системи	13.05.2024-28.05.2024
Виправлення помилок	28.05.2024
Оформлення пояснювальної записки	27.05.2024-31.05.2024

					ІАЛЦ.467200.002 ТЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		4

**ПОЯСНЮВАЛЬНА ЗАПИСКА
ДО ДИПЛОМНОГО ПРОЄКТУ**

на тему: «Веб-ресурс для підтримки та інформування громадян
під час воєнних дій»

ЗМІСТ

ВСТУП	4
РОЗДІЛ 1 ОГЛЯД ТА АНАЛІЗ ІСНУЮЧИХ ІНФОРМАЦІЙНИХ СИСТЕМ	5
1.1 Загальні положення.....	5
1.2 Опис та аналіз інформаційних систем.....	5
1.2.1 Опис компонентів інформаційних систем.....	6
1.2.2 Класифікація інформаційних систем	8
1.3 Огляд існуючих рішень	12
1.3.1 Портал Dovidka.info	12
1.3.2 Сайт Державної служби України з надзвичайних ситуацій (ДСНС).....	14
1.3.3 Порівняння існуючих рішень.....	16
1.4 Постановка задачі.....	17
ВИСНОВОК ДО РОЗДІЛУ 1	19
РОЗДІЛ 2 ОГЛЯД І ВИБІР ТЕХНОЛОГІЙ ДЛЯ РОЗРОБКИ	20
2.1 Архітектура веб-ресурсу.....	20
2.2 Вибір технологій розробки серверної частини.....	22
2.2.1 Огляд мов програмування	22
2.2.2 Мова програмування Java.....	23
2.2.3 Мова програмування Python	24
2.2.4 Мова програмування C#.....	26
2.2.5 Мова програмування PHP	28
2.3 Вибір технологій розробки клієнтської частини.....	29
2.3.1 HTML, CSS, JavaScript.....	30
2.3.2 CSS-фреймворки	31
2.3.3 JS-фреймворки.....	33
2.4 Вибір системи керування базами даних.....	37
2.5 Хостинг сервіс	37
ВИСНОВОК ДО РОЗДІЛУ 2	39

					ІАЛЦ.467200.003 ПЗ			
Зм.	Арк.	№ докум.	Підпис	Дата				
Розробив	Бондаренко Ю.А.				Веб-ресурс для підтримки та інформування громадян під час воєнних дій Пояснювальна записка	Літ.	Аркуш	Аркушів
Перевірив	Щур В.Ю.						1	66
Реценз.						КПІ ім. Ігоря		
Н. Контр.	Іваніщев Б.В.					Сікорського, ФІОТ, ІО-04		
Затвердив								

РОЗДІЛ 3 ПРОЄКТУВАННЯ ТА РОЗРОБКА ІНФОРМАЦІЙНОГО ВЕБ-РЕСУРСУ	40
3.1 Створення та підключення бази даних.....	40
3.1.1 Функції взаємодії з базою даних.....	42
3.2 Опис основних компонентів.....	46
3.2.1 Форма реєстрації	46
3.2.2 URL-адреса.....	48
3.2.3 Конструкція “require”	49
3.2.4 Форма зворотнього зв’язку.....	50
3.2.5 Доступ до сторінки адміністратора	52
3.3 Клієнтська частина	52
ВИСНОВОК ДО РОЗДІЛУ 3	53
РОЗДІЛ 4 АНАЛІЗ ТА ТЕСТУВАННЯ РОЗРОБЛЕНОГО ПРОДУКТУ.....	54
4.1 Основні сторінки веб-застосунку.....	54
4.2 Форма реєстрації.....	59
4.3 Сторінка адміністратора	61
4.4 Форма зворотнього зв’язку.....	62
ВИСНОВОК ДО РОЗДІЛУ 4	63
ВИСНОВОКИ.....	64
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	65

ПЕРЕЛІК СКОРОЧЕНЬ

HTTP	(Hypertext Transfer Protocol) Протокол передачі гіпертексту
ОС	Операційна Система
TCP	(Transmission Control Protocol) Протокол контролю передачі
IP	(Internet Protocol) Протокол адресації пакетів
FTP	(File Transfer Protocol) Протокол передачі файлів
SMTP	(Simple Mail Transfer Protocol) Протокол пересилання електронної пошти
POP3	(Post Office Protocol version 3) Протокол отримання листів з віддаленого сервера
HTML	(Hyper Text Markup Language)
CSS	(Cascading Style Sheets)
JS	(JavaScript)
Front-End	(Front-End development) Розробка інтерфейсу користувача
Back-End	(Back-end development) Розробка на серверному рівні
SQL	(Structured query language) Мова структурованих запитів
AI	Artificial intelligence
JVM	Java Virtual Machine
JIT	Оптимізація Just-In-Time на платформі .NET
DOM	Document Object Model
SPA	Single-page application
SSL	Secure Sockets Layer
DdoS attack	Distributed denial-of-service attack
PDO	PHP Data Objects
СУБД	Database Management System

ВСТУП

24 лютого 2022 року реальність мільйонів українців розділилась на «до» та «після». Початок повномасштабного вторгнення став трагічною подією, яка залишила невід’ємний відбиток в житті кожного з нас і навчила думати та діяти по новому. На жаль, війна становить не лише фізичну небезпеку та загрозу для громадян, а й викликає значне психологічне навантаження, під час якого більшість відчувають страх перед невідомістю та невизначеністю. У зв'язку з цим правдивість і точність інформації є принциповим питанням, і вона потрібна кожній людині в постраждалих місцях для прийняття життєво важливих рішень і зменшення негативних наслідків.

Інформаційний простір в Україні зазнав значних трансформацій, що були викликані появою великої кількості ненадійних джерел та ресурсів, які поширювали дезінформацію та пропаганду. Зрештою, це призвело до розповсюдження паніки серед населення та появи помилкових уявлень про сьогоднішнє. Також психологічна допомога особам, які постраждали від війни, підвищила своє значення. Військові дії, втрата членів родини та житла лише посилює тривогу та стрес, що вимагає кваліфікаційної допомоги не лише дорослим, а й молодшому поколінню. В результаті, виникла необхідність створити спільне джерело інформації, яке б дозволило громадянам отримувати необхідні перевірені дані та звертатися за допомогою до професіоналів.

Метою роботи є розробка веб-застосунку, який вирішить ці проблеми та стане вичерпним посібником для громадян України під час повномасштабного вторгнення. Робота включає в себе аналіз вже існуючих інформаційних порталів, визначення основних вимог та потреб цільової групи користувачів та опрацювання концепції зі створення моделі веб-ресурсу.

					ІАЛЦ.467200.003 ПЗ	Арк.
						4
Зм.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 1 ОГЛЯД ТА АНАЛІЗ ІСНУЮЧИХ ІНФОРМАЦІЙНИХ СИСТЕМ

1.1 Загальні положення

Війна в Україні спровокувала непомірний попит громадян до ментальної та емоційної допомоги. Інформаційні системи сьогодні відіграють вирішальну роль у покращенні здатності долати катастрофи, надаючи необхідні дані, пов'язані з новинами, ресурсами та можливостями спілкування.

Через стрімке зростання недостовірних джерел інформації та фейкових новин, гостро постало питання в пошуку ресурсу, якому можна довіряти та не погіршувати психологічний стан населення. На сьогоднішній день державні установи, включаючи Міністерство оборони, Управління національної безпеки та відповідні цивільні та військові відомства, мають надійні веб-сайти, а також мобільні додатки, де публікується актуальна інформація про передову діяльність, поради щодо безпеки, а також пропозиції заходів захисту.

Отже, враховуючи всі вищезгадані аспекти, метою мого проєкту є створення надійного веб-застосунку, який буде містити лише офіційну та перевірену інформацію та заходи безпеки під час незвичайних ситуацій. Більш того для охоплення більшої кількості постраждалих людей, ресурс матиме можливість надавати доступ до своєчасних психологічних тренінгів та семінарів по першій медичній допомозі.

1.2 Опис та аналіз інформаційних систем

Веб-ресурс для підтримки громадян під час воєнних дій можна розглянути як одну з форм інформаційної системи. У свою чергу інформаційна система (ІС) являє собою організовану структуру, утворену взаємозв'язком компонентів, які дозволяють збирати і зберігати, обробляти і поширювати дані та цифрову інформацію.

					ІАЛЦ.467200.003 ПЗ	Арк.
						5
Зм.	Арк.	№ докум.	Підпис	Дата		

В результаті, це сукупність елементів, які взаємозалежні між собою для перетворення необроблених даних у корисні знання [1].

1.2.1 Опис компонентів інформаційних систем

Більш детально розглянемо структуру та призначення всіх складових інформаційних систем.

Кожна ІС складається з таких компонентів: [2]

- Апаратне забезпечення
- Програмне забезпечення
- Бази даних
- Мережі
- Людський фактор

На рисунку 1.1 показано основні компоненти ІС та взаємодію між ними.

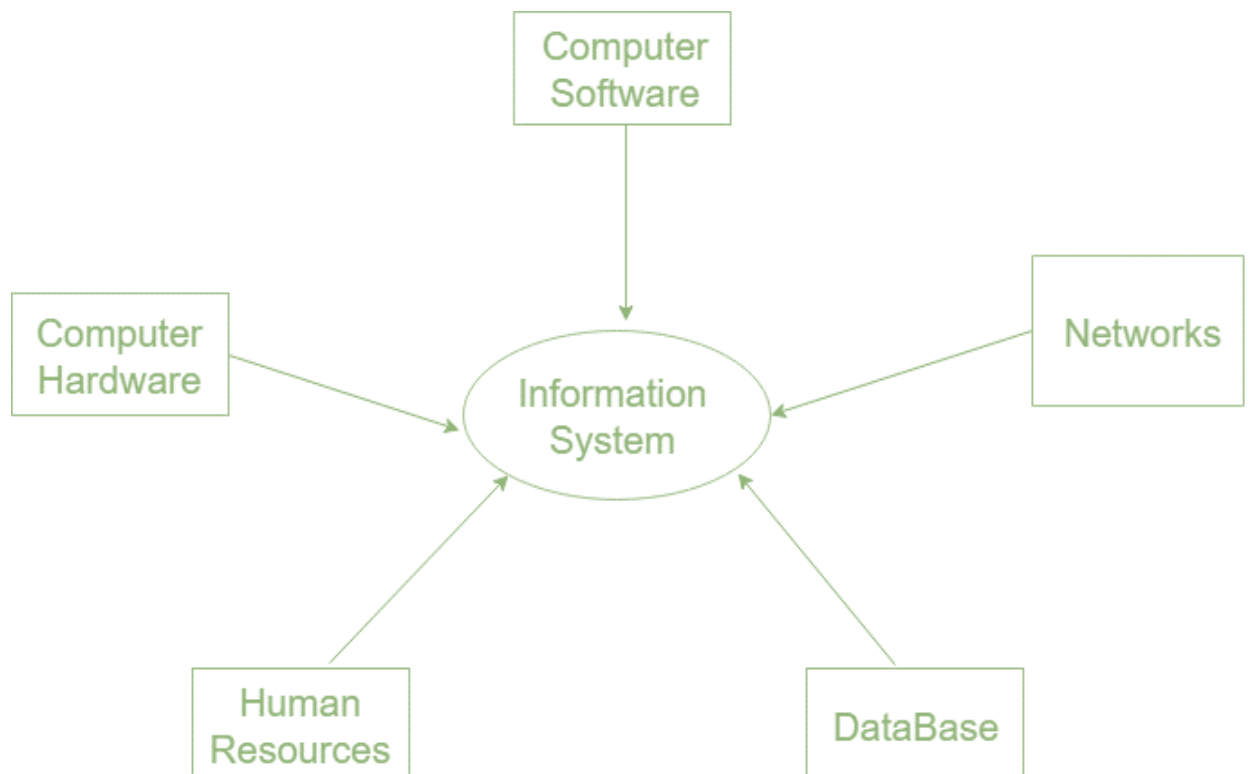


Рисунок 1.1 – Основні компоненти ІС

Апаратне забезпечення включає в себе обладнання, що використовується для передачі даних всередині або поза системою.

Зм.	Арк.	№ докум.	Підпис	Дата

Саме комп'ютери та його комплектуючі мають велике значення, адже вони надають доступ до введення, виведення та обробки інформації. Не менш вагомою частиною є носії даних та сервери, які слугують простором для зберігання інформації, програмного забезпечення та додатків, які складають систему.

Програмне забезпечення складається з набору інструментів, що дозволяють обробляти отриману інформацію.

Першою складовою є операційна система, наприклад Windows – слугує фундаментом для керування апаратними ресурсами комп'ютера та середовищем для виконання команд.

Другим ресурсом є прикладне програмне забезпечення, що надає можливість користувачам функціонувати з продуктом. Яскравим прикладом можуть бути системи управління базами даних – MySQL, управління контентом – WordPress та хмарні середовища – Google Drive.

Та останньою складовою вважають процедури, які полегшують роботу з системою, представляючи собою інструкцію по експлуатації. Це допомагає покращити роботу та уникати помилок, адже зазначають точні кроки для отримання бажаного результату.

Бази даних зберігають в собі великий обсяг необхідної інформації та дозволяють ефективно використовувати та обробляти всі дані. Це важливий компонент системи, тому що БД забезпечують збереження цілісності інформації, легкий та контрольований доступ до неї та можливість масштабуватись для отримання більшої кількості ресурсу.

Мережеві процеси слугують зв'язком між компонентами ІС та зовнішнім середовищем, яке складається з мережі Інтернет та користувачів. Мережі дозволяють легко передавати дані між компонентами ІС, наприклад сервером та клієнтом; комунікувати використовуючи електронну пошту або чати; отримувати доступ до даних через мережу Інтернет для віддаленого користування та контролювати стан всіх компонентів інформаційної системи.

					ІАЛЦ.467200.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		7

Прикладами протоколів в мережевих процесах можна назвати: TCP/IP для використання в мережі Інтернет, HTTP для передачі веб-контенту, FTP для передачі файлів та SMTP/POP3 для надсилання та отримання електронної пошти [3].

Людський ресурс насамперед це робоча сила, яка необхідна для виконання та управління інформаційною системою. Системні аналітики, оператори комп'ютерів та програмісти відповідають за розробку та експлуатацію ІС, включаючи налаштування баз даних, розробку Front-End та Back-End частин продукту та тестуванням готової системи. Люди, які є кінцевим користувачем інформаційної системи, споживають вже готовий продукт та ознайомлюються з інтерфейсами та функціоналом. Між цими двома складовими завжди стоять системні адміністратори, які налагоджують роботу ІС та слідкують за її працездатністю. В їх обов'язки також входить виправляти помилки системи та надавати нову актуальну інформацію для клієнтів.

1.2.2 Класифікація інформаційних систем

Вивчення критеріїв класифікації інформаційних систем допомагає аналізувати можливості та напрямки їх застосування. На прикладі мого проекту розглянемо характеристики та властивості ІС, які важливо визначити перед розробкою фінального сервісу.

Існує кілька основних категорій для визначення та класифікації ІС: [4]

За сферою застосування:

- Системи обробки транзакцій
- Системи прийняття рішень
- Інформаційно-довідкові системи
- Офісні інформаційні системи

Системи обробки транзакцій займаються автоматизацією процесу виконання транзакцій.

					ІАЛЦ.467200.003 ПЗ	Арк.
						8
Зм.	Арк.	№ докум.	Підпис	Дата		

Вони збирають інформацію щодо покупця та суми товару і перевіряє чи задовільний баланс або наявність повноважень для завершення транзакції. Такі системи є гарантією швидкого та безпечного обміну коштами.

Системи прийняття рішень підтримує користувачів, які здатні приймати об'єктивні та зважені рішення. Вони використовуються в багатьох сферах і потребують аналізу даних, оцінки варіантів та створенню найкращого висновку.

Офісні інформаційні системи – це програмні комплекси, розроблені для автоматизації внутрішніх завдань і основних процесів у офісі. Найчастіше використовуються для створення електронних документів, управління робочою поштою, бухгалтерського обліку та аналізу даних для звітностей, скорочуючи час на виконання рутинних завдань.

Веб-ресурс для допомоги та підтримки громадян під час війни можна віднести до інформаційно-довідкової системи, бо саме вони надають своєчасні та актуальні дані та покликані надавати відповіді та докази на поширенні питання. Системи збирають дані з багатьох ресурсів, впорядковують та зберігають їх в базах даних та надають доступ до готового продукту з удосконаленим інтерфейсом.

За способом організації:

- Архітектури файл-сервер
- Багаторівневої архітектури
- Інтернет/інтранет-технології
- Архітектури клієнт-сервер

Порівнюючи класифікацію за способом організації, можна зрозуміти, що файл-серверна архітектура належить до порядку організації комп'ютерної мережі, коли одна машина, працюючи як сервер, дозволяє об'ємний спільний доступ до файлів і ресурсів з іншими пристроями, які називаються клієнтами. Тобто користувач надсилає запит серверові через мережу, вказуючи на потрібний файл та у відповідь другий його надає клієнтові.

					ІАЛЦ.467200.003 ПЗ	Арк.
						9
Зм.	Арк.	№ докум.	Підпис	Дата		

Багаторівнева архітектура представляє собою структуру, в якій всі функції розділені між різними рівнями. Основні шари можуть бути представлені, як: [5]

- Presentation Layer – це рівень інтерфейсу, який працює з користувачем,
- Business Layer – рівень, який відповідає за обробку даних та встановлює бізнес-правила,
- Persistence Layer – слугує для збереження інформації між об’ємними сеансами роботи програми,
- Database Layer – рівень, який спілкується зі сховищами даних і надає до них доступ.

Інтернет/інтранет-технології включають в себе широкий спектр технологій, які використовуються для створення, підтримки та використання мереж. Інтернет-технології працюють над універсальним зв’язком та обміном даних по всьому світу. А інтранет-технології у свою чергу зосереджені на розвитку внутрішніх мереж для функціонування організації.

Отже, варіант, який найкраще відповідає веб-ресурсу інформаційної допомоги – є архітектура клієнт-сервер. Це модель розподіленої обробки даних і ресурсів, в якій робота з інформацією розподіляється між двома типами машин: клієнтські та серверні. Перша з них надсилає запити і очікує на відповідь, а друга обробляє ці запити та надає доступ до даних. На рисунку 1.2 зображено структуру дворівневої архітектури клієнт-сервер.



Рисунок 1.2 – Структура архітектури клієнт-сервер

Зм.	Арк.	№ докум.	Підпис	Дата

За масштабом:

- Одиночні
- Групові
- Корпоративні

Одиночні інформаційні системи розраховані на одного користувача або невелику кількість людей і пропонують обмежений функціонал та послуги. Головними перевагами є низька вартість придбання та обслуговування, простота у використанні та висока швидкість і продуктивність системи. Найчастіше використовують локальні системи управління персональними базами даних, наприклад Paradox, dBase або Clipper.

Ознайомившись із класифікацією, одразу можна зробити висновок, що тема мого проєкту за масштабом відповідає груповій. Адже, веб-ресурс буде розрахований на ширшу аудиторію людей і надавати доступ до великого обсягу офіційної інформації та своєчасної підтримки. Важливо зазначити, що такі системи надають можливість підвищувати ефективність роботи в команді, використовують інтегровані засоби для обміну повідомленнями та забезпечують контроль над даними з додатковою безпекою. Також для групових ІС використовуються сервери баз даних, а саме SQL-сервери.

Корпоративні ІС створенні для використання великими компаніями та надають доступ до широкого спектра послуг високої якості. Вони розроблені з метою узгодження даних між підрозділами корпорації та налагодити потоки управлінської інформації, зробивши цей процес більш зручним та надійним. Для прикладу корпоративних ІС можна згадати Oracle ERP Cloud – що є системою управління підприємством та Workday – рішення для управління персоналом.

У висновку можна зазначити, що було розглянуто критерії класифікації інформаційних систем, що в майбутньому полегшить вибір параметрів для розробки веб-ресурсу для підтримки громадян під час воєнних дій та допоможе забезпечити його зручну та ефективну роботу.

					ІАЛЦ.467200.003 ПЗ	Арк.
						11
Зм.	Арк.	№ докум.	Підпис	Дата		

1.3 Огляд існуючих рішень

1.3.1 Портал Dovidka.info

На основі вищенаведеного аналізу функціональності та класифікації інформаційних систем було виділено декілька існуючих порталів підтримки громадян для порівняння.

Перший сервіс для розгляду – Dovidka.info – це веб-сайт експертної допомоги в текстовому форматі, розроблений Центром стратегічних комунікацій та інформаційних досліджень. На рисунку 1.3 зображено зовнішній вигляд інтерфейсу веб-порталу Dovidka.info [6].

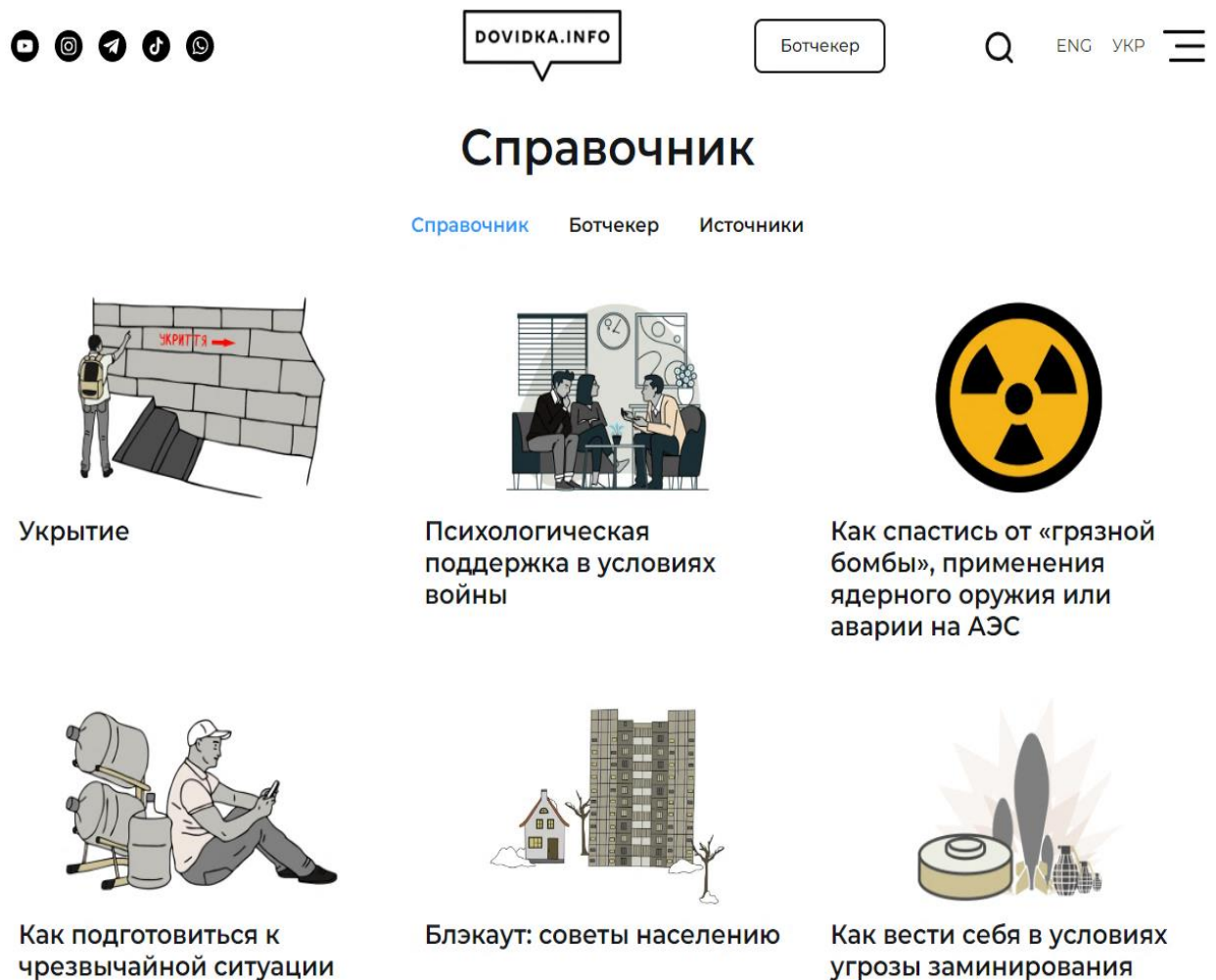


Рисунок 1.3 – Головна сторінка Dovidka.info.

					ІАЛЦ.467200.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		12

Проаналізувавши ресурс одразу можна виділити мінімалістичний та зручний у використанні користувацький інтерфейс, що дозволяє легко знаходити необхідну інформацію. Сайт надає доступ до інформаційних порад про безпеку життєдіяльності та надає чіткий план дій при кризових ситуаціях, таких як: загроза ядерного удару, мінна небезпека, поведінка на територіях, що були окуповані та відключення світла або зв'язку.

Загалом, Dovidka.info є цінним довідником корисної інформації, проте він все ще має потенціал для вдосконалення, щоб забезпечувати більше потреб громадян і надавати ширший доступ до даних. Я виділила декілька недоліків, які необхідно виправити для власного проєкту.

Перший аспект, який необхідно взяти до уваги – це неоднорідність джерел інформації. Не дивлячись на те, що сайт надає користувачам доступ до актуальних та перевірених даних, джерела можуть бути суперечливими і впроваджувати фейки та неправдиві факти. Для того, щоб не підривати довіру населення і не викликати плутанину необхідно використовувати лише офіційні джерела інформації, які надають гарантію в достовірності та актуальності інформації.

Також, важливо виділити відсутність взаємодії з аудиторією, тому що в наш час велика кількість людей потребує допомоги не лише інформаційної, а і психологічної. Для вирішення цього питання слід інтегрувати можливість реєстрації на семінари по наданню першої медичної допомоги або консультації з психологом та додати елемент у вигляді форми зворотнього зв'язку для комунікації між користувачами та адміністраторами сервісу.

У висновку, можна зазначити, що веб-сайт Dovidka.info покриває велику кількість запитань, надаючи змістовні та актуальні відповіді та забезпечуючи необхідною інформацією більшу кількість користувачів. Проте функціонал веб-сервісу може бути розширений за допомогою використання сучасних технологій програмування. Я вважаю, що цей портал є гарним прикладом для фундаменту власного проєкту.

					ІАЛЦ.467200.003 ПЗ	Арк.
						13
Зм.	Арк.	№ докум.	Підпис	Дата		

1.3.2 Сайт Державної служби України з надзвичайних ситуацій (ДСНС)

Наступним джерелом, яке можна вважати гарним прикладом для порівняння та визначення потреб у розробці – це сайт Державної служби України з надзвичайних ситуацій (ДСНС). Слід зауважити, що ДСНС – це важливий відомчий орган, який відповідає за збереження безпеки та захисту населення в таких ситуаціях, як природні катастрофи, пожежі, терористичні загрози та інше. Тому сайт є важливим ресурсом, який покриває всі теми та надає необхідну інформацію.

Враховуючи відсутність в нашому інформаційному просторі сервісів, які налаштовані на надання порад поведінки під час війни, я обрала розділ на сайті ДСНС пов'язаний з цією темою. На рисунку 1.4 можна побачити інтерфейс розділу допомоги громадянам під час воєнних дій на сайті ДСНС [7].

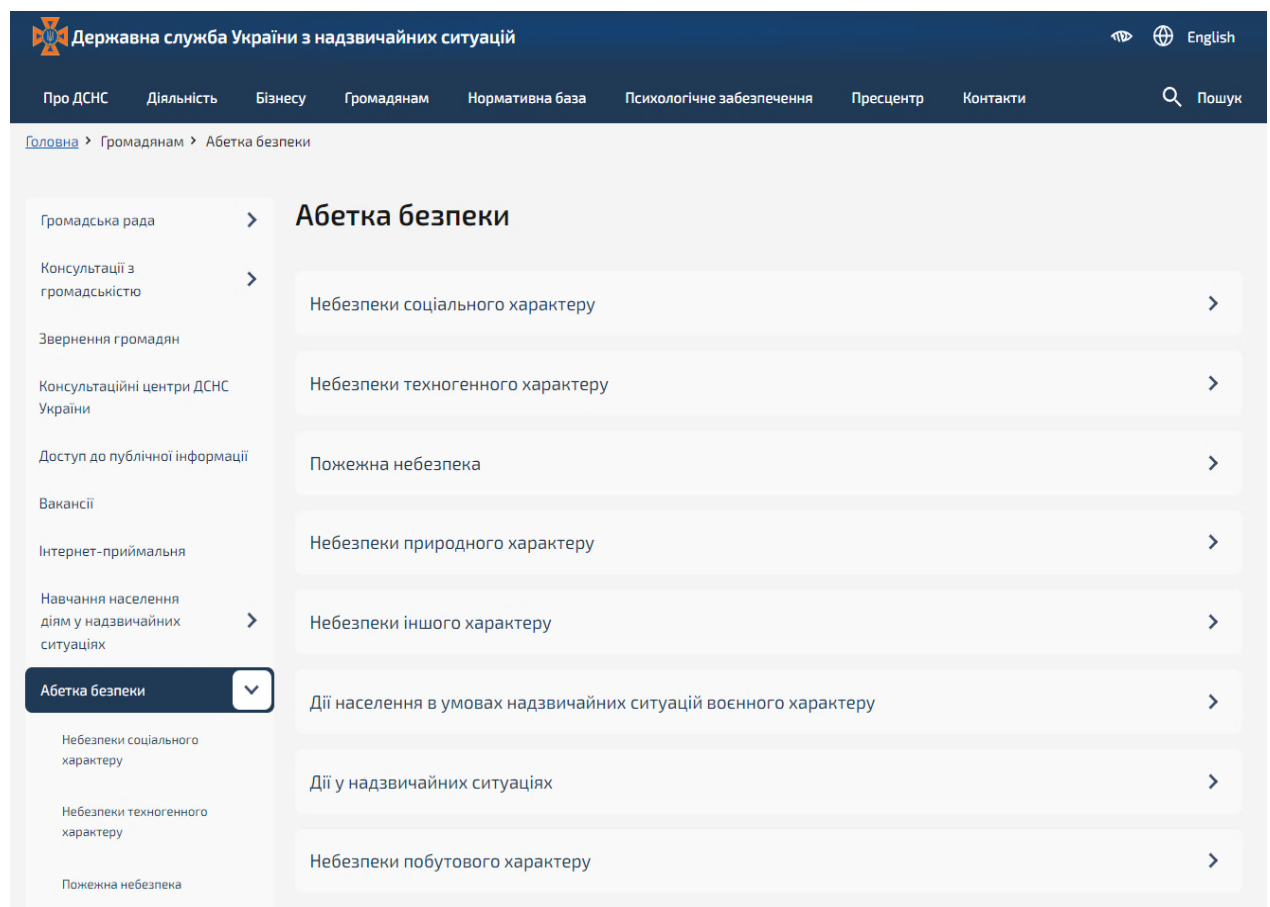


Рисунок 1.4 – Інтерфейс розділу безпеки на сайті ДСНС

Сайт також вирізняється офіційним та зручним дизайном, що дозволяє користувачу легко координуватись по розділах. У доступі є інформація про плани евакуації, план дій у разі хімічної або радіаційної атаки, заходи життєдіяльності та безпеки при відсутності електропостачання.

Щодо недоліків, то сайт Державної служби України з надзвичайних ситуацій точно можна вважати офіційним джерелом і не хвилюватись стосовно достовірності інформації. Відсутність фейків та актуальні дані дають впевненість у безпеці та зменшують емоційну напругу при пошуку необхідних порад. Проте, деякі розділи сайту не наповнені інформацією та мають статус «в розробці». На рисунку 1.5 можна побачити сторінку, яка ще в процесі реалізації.

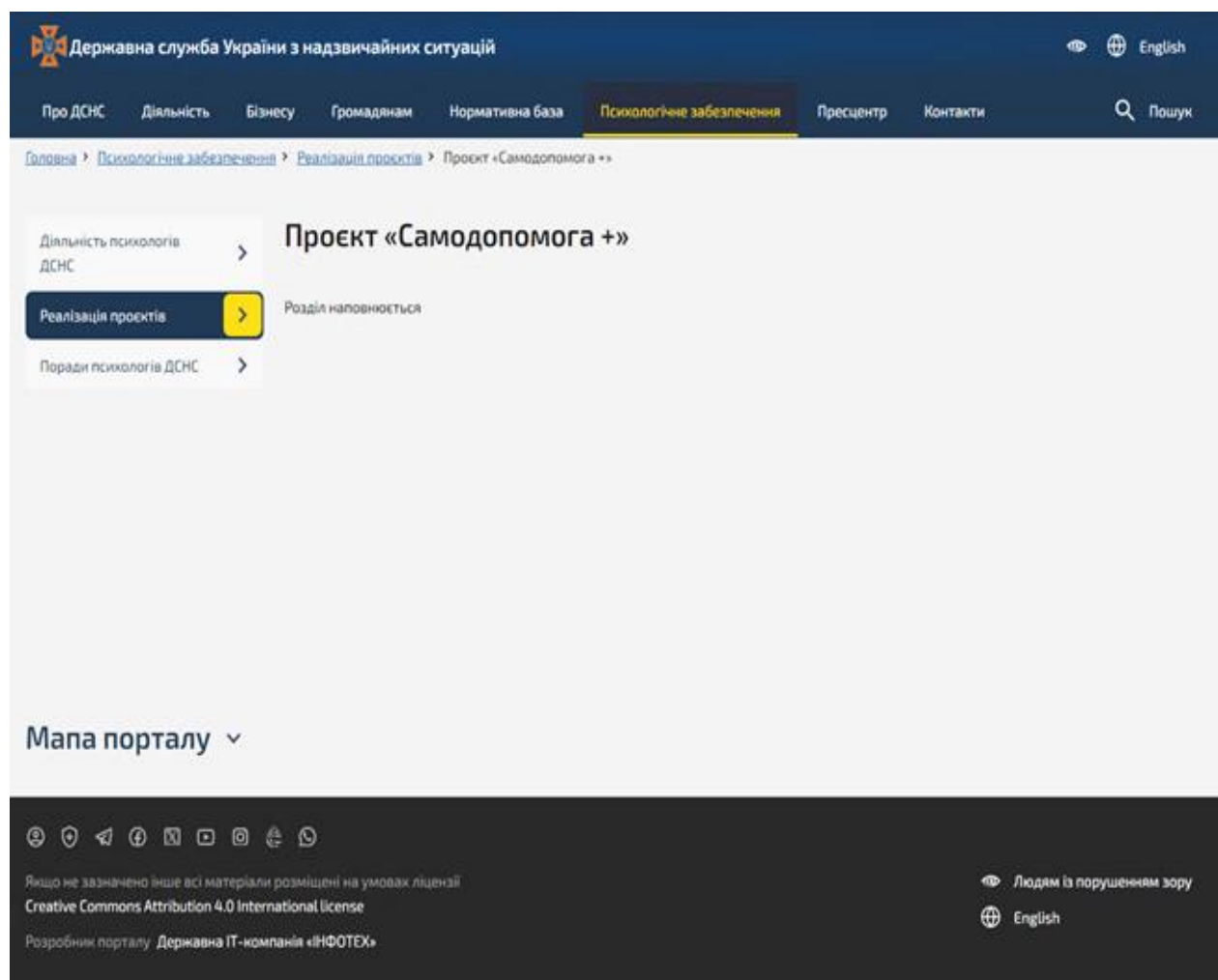


Рисунок 1.5 – Зображення сторінки в статусі «розробка»

					ІАЛЦ.467200.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		15

Проте, враховуючи необхідність наявності фізичної допомоги та підтримки зі сторони кваліфікованих працівників, сайт Державної служби України з надзвичайних ситуацій не має можливості відвідання таких заходів, а лише поради та контакти компаній, що надають необхідні послуги.

1.3.3 Порівняння існуючих рішень

В процесі пошуку існуючих веб-застосунків, які надають широкий спектр інформації для забезпечення контролю над важкою ситуацією, виявилось, що в українському просторі дуже мало таких рішень і більшість з них розрахована на інші тематики. Проте для порівняння були обрані найкращі варіанти.

Для кращого сприйняття функціональності створена таблиця 1.1, яка наведена нижче.

Таблиця 1.1 – Порівняльний аналіз веб-сервісів

Характеристика	Dovidka.info	ДСНС
Тип сайту	Інформаційний портал	Урядовий веб-сайт
Цільова аудиторія	Люди, які шукають інформаційну допомогу	Люди, які шукають інформаційну допомогу
Доступність	Доступний онлайн	Доступний онлайн
Дизайн	Мінімалістичний, простий	Офіційний, інформативний
Актуальність	Інформація, яка потребує більшого оновлення	Актуальна інформація, що постійно оновлюється
Взаємодія з користувачем	Відсутність можливості комунікувати	Відсутність можливості комунікувати

Порівнюючи ці сервіси, можна зробити висновок, що кожен з них має свої переваги та недоліки. І це буде корисним для розробки та проєктування власного веб-застосунку, що допоможе врахувати всі пункти для задовільнення більшої кількості користувачів та їх потреб.

1.4 Постановка задачі

Метою розробки є веб-ресурс для підтримки та інформування громадян під час воєнних дій, який буде слугувати централізованим джерелом актуальних та достовірних даних та платформою для спілкування з кваліфікованими професіоналами викладачами.

Тому, вимогу слід виділити в наступних пунктах:

- Проста та доступна архітектура для будь-якого користувача
- Легкий та мінімалістичний інтерфейс
- Доступ до актуальної інформації, яка зберігається в базі даних
- Можливість реєстрації на семінари
- Наявність форми зворотнього зв'язку
- Контроль працездатності системи
- Можливість розширення та оновлення функціональності веб-порталу

Після вивчення інформації, щодо створення та реалізації веб-ресурсу, необхідно виділити основні пункти для розробки продукту. Це допоможе врахувати всі пункти та організувати роботу за пріоритетністю завдань. Отже, маємо декілька пунктів для реалізації продукту:

- Вибір моделі розробки застосунку
- Зображення діаграми даних
- Вибір мови програмування та необхідних інструментів
- Вибір додаткових ресурсів для розробки
- Реалізація системи
- Мануальне тестування системи

					ІАЛЦ.467200.003 ПЗ	Арк.
						17
Зм.	Арк.	№ докум.	Підпис	Дата		

Дотримуючись цих пунктів та всіх рекомендацій, плануємо в результаті отримати надійний та корисний інформаційний портал. В залежності від ідей та нових впроваджень, які будуть виникати впродовж реалізації проєкту, функціонал та наповнення може буде змінене або доповнене.

					ІАЛІЦ.467200.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		18

ВИСНОВОК ДО РОЗДІЛУ 1

В цьому розділі було проведено детальний аналіз поняття інформаційної системи, розглянуто її складові та класифікацію. Це допомогло краще зрозуміти природу створення ІС та їх функціональність. При виявленні та порівнянні існуючих рішень інформаційних веб-порталів, були виявлені як переваги, так і недоліки, що слугуватимуть фундаментом створення надійного та сучасного ресурсу.

З позитивних рішень, що були виділені при аналізі веб-ресурсів можна назвати зручність використання та сучасний дизайн. Інтуїтивно зрозумілий функціонал та легка навігація надають змогу користувачам швидко знаходити потрібну інформацію. А високий рівень захисту даних та стабільна робота веб-застосунку забезпечують довіру клієнтів при використанні.

Проте, занадто велика кількість інформації на сторінках порталу ускладнюють сприйняття контенту та навантажують увагу користувача. Відсутність інтерактивних елементів, таких як форма зворотнього зв'язку, знижують рівень взаємодії та залученості клієнтів. Саме такі аспекти можна назвати недоліками та надати їм достатньо уваги для вдосконалення у власному проєкті.

Було проведено постановку основних кроків для реалізації продукту та виявлення вимог для ефективного впровадження інформаційного ресурсу.

					ІАЛЦ.467200.003 ПЗ	Арк.
						19
Зм.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 2. ОГЛЯД І ВИБІР ТЕХНОЛОГІЙ ДЛЯ РОЗРОБКИ

У попередньому розділі ми розглянули існуючі аналоги проєкту, їхні технології розробки, переваги та недоліки. Ці аспекти допоможуть детальніше ознайомитись та порівняти інструменти для розробки веб-сервісу. У цьому розділі складемо список вимог та розглянемо засоби для програмної реалізації веб-ресурсу для підтримки та інформування громадян під час воєнних дій.

2.1 Архітектура веб-ресурсу

Архітектура веб-додатків описує взаємодію компонентів системи між собою для досягнення бажаного результату та надання користувачам можливості отримувати потрібну інформацію або послуги. Вона включає в себе розподіл обов'язків між різними частинами додатку та досліджує способи їхньої взаємодії. Якісно розроблена архітектура в поєднанні з вибором сучасних технологій розробки надає можливість забезпечити ефективну роботу, надійність та безпеку веб-додатку.

Розглянувши у попередньому розділі класифікацію інформаційних систем за способом організації, виявилось, що найпоширенішим з підходів до проєктування веб-ресурсів та ідеальним варіантом для власного сайту є клієнт-серверна архітектура.

Ця модель складається з двох компонентів, а саме: клієнта, що зазвичай є веб-браузером, який надає відображення користувацького інтерфейсу та збирає введені дані, а друга частина є сервером, що обробляє отримані запити (HTTP-request) та надає необхідні дані (HTTP-response).

На рисунку 2.1 надано графічне зображення запиту та відповіді від серверної частини, яке пояснює модель відповіді від сервера на запити користувача до системи.

					ІАЛЦ.467200.003 ПЗ	Арк.
						20
Зм.	Арк.	№ докум.	Підпис	Дата		

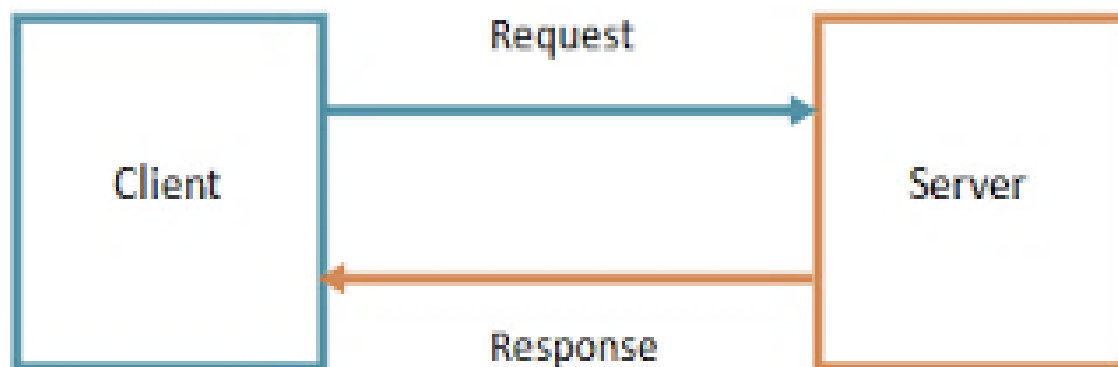


Рисунок 2.1 – Структура запиту та відповіді від сервера.

Зазвичай веб-додатки складаються з двох наборів програм, які взаємодіють між собою та забезпечують функціональність та виконання поставлених задач.

Основні компоненти виглядають так: [8]

- Користувацька частина (Front-End) – це інтерфейс сторінок, на яких працює клієнт у веб-браузері. Для їх створення та реалізації інтерактивних елементів використовуються програмні компоненти, такі як: мова розмітки HTML, стилізація CSS та мова програмування JavaScript.
- Серверна частина (Back-End) – це сам сервер, який насамперед обробляє запити, які отримав від користувачів, обробляє та зберігає ці дані. Для реалізації основної логіки додатку, працює з базами даних та веб-серверами. Такі мови програмування, як Python, JavaScript та PHP з відповідними наборами фреймворків є найпоширенішими рішеннями для створення серверної частини веб-додатків.

У висновку можна сказати, що архітектура веб-додатків відіграє ключову роль у проєктуванні ефективних та надійних програм. Розуміння концепцій та застосування головних компонентів системи надають можливість уникнути проблем з погіршенням продуктивності та доступності готового продукту при використанні користувачами, незалежно від масштабу системи та її призначення.

2.2 Вибір технологій розробки серверної частини

Під час проєктування веб-додатків, незалежно від їх розміру та складності, одним з найважливіших кроків є вибір технологій для реалізації серверної частини. Для цього необхідно розглянути та порівняти існуючі рішення, та обрати мову програмування та базу даних, які будуть оптимальним варіантом для розробки веб-ресурсу.

2.2.1 Огляд мов програмування

На сьогоднішній день питання з вибором мови програмування є дуже актуальним та складним у зв'язку з появою нових технологій на ринку та розвитку вже знайомих нам популярних мов. Це рішення може суттєво вплинути на успішність проєкту, тому необхідно ознайомитись з усіма доступними варіантами та знайти найкраще з них.

У зв'язку з тим, що кожна мова програмування має свої особливості, переваги та недоліки, необхідно докладніше ознайомитись з ними та провести аналіз. На рисунку 2.2 зображено рейтинг мов програмування для серверної розробки веб-застосунків за результатами щорічного опитування ІТ-спеціалістів з України. [9]

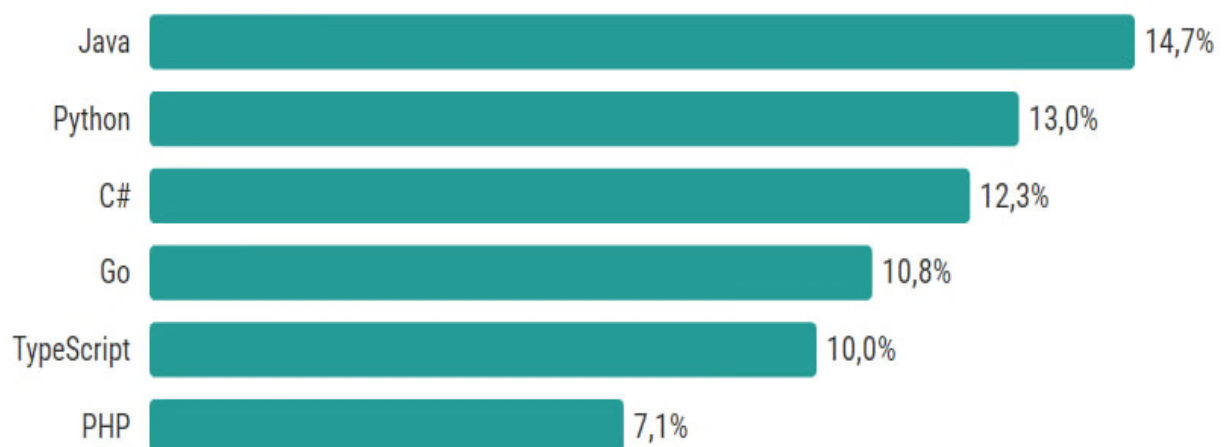


Рисунок 2.2 – Рейтинг мов програмування

Використовуючи вищезгадане дослідження, розглянемо популярні мови програмування та порівняємо недоліки та переваги, щоб обрати оптимальний варіант для розробки веб-ресурсу інформаційної допомоги.

2.2.2 Мова програмування Java

Мова програмування Java є популярною серед мов загального призначення, яка використовується для розробки серверної частини веб-застосунків. Така прихильність зумовлена тим, що Java відома своєю безпечністю та надійністю, які перевірені вже роками та не викликають жодних сумнівів її придатності для створення складних веб-додатків.

Насамперед варто зазначити, що мова програмування Java – це об'єктно-орієнтовна мова, яка була випущена в 1995 році компанією «Sun Microsystems» [10]. Це означає, що вона використовує концепції об'єктів та класів для організації коду, що надає можливість легкого обслуговування та збільшує модульність.

Також, Java – компільована мова, яка може використовуватись на будь-якій платформі, що має віртуальну машину (JVM) і це робить її переносною та кращим рішенням для розробки програмного забезпечення. Це ще більше підтверджується надійністю, масштабованістю та безпечністю мови, тому що вона має ряд функцій, які допомагають запобігати поширеним помилкам програмування; обробляти великі обсяги даних та захищати програмне забезпечення від атак.

Проте, не дивлячись на широке використання та популярність мови, вона має деякі недоліки, що можуть ускладнити та заповільнити роботу. Розглянемо головні мінуси Java: [11]

- Java – це складна мова програмування, яка має неінтуїтивний та багатослівний синтаксис порівняно з іншими мовами, такими як Python або JavaScript.

- Мова вимагає більше пам'яті та процесорних ресурсів, що може зменшувати її ефективність на серверах з обмеженими можливостями.
- Мова програмування Java може займати більше часу на розробку та старт додатків в порівнянні з іншими мовами. Вона вимагає наявності додаткового коду для програмування та сповільнює початок роботи через довгу ініціалізацію віртуальної машини.

Слід також згадати, що Java є однією з найпопулярніших мов через широкий спектр використання. Найчастіше зустрічається при розробці веб-застосунків, мобільних програм, вбудованих систем, ігрових платформ та підприємних програм.

У висновку, можна сказати, що мова програмування є унікальною та потужною для програмування різного рівня та складності. Вона пропонує великий спектр можливостей та вважається важливим інструментом для будь-якого розробника.

2.2.3 Мова програмування Python

Python в наш час є також дуже популярною та поширеною мовою програмування загального використання. Вона виникла в 1990 році за допомогою нідерландського програміста Гвідо ван Россума [12]. Спочатку вона була задумана, як нащадок мови ABC для навчання і саме тому на першому місці стояла лаконічність та простий синтаксис, якими не могли похизуватись більшість мов програмування.

Як і більшість мов програмування, Python має свої переваги та недоліки, які можуть впливати на роботу та проектування. Вона вважається простою у використанні та водночас повноцінною мовою, тому для початку ознайомимось з ключовими аспектами, які роблять Python таким популярним та універсальною, а саме:

					ІАЛЦ.467200.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		24

- Python – мова програмування, яка має читабельний та простий синтаксис, що допомагає писати більш зрозумілий код та є ідеальним варіантом для розробників з різним рівнем знань.
- Велика кількість стандартних бібліотек неабияк полегшують розробку систем різного рівня складності. До цієї категорії також можна віднести популярні фреймворки, які надають готові рішення для програмування, наприклад : Django або Flask.
- Масштабованість мови дозволяє використовувати її для розробки великих корпоративних систем. Це зумовлено наявністю великої кількості інструментів для роботи з паралельністю та асинхронністю в проєктах.
- Мова програмування Python може добре інтегруватись з іншими мовами, такими як: C та C++, що дозволяє використовувати додатковий код для вирішення критичних проблем при розробці.

Це є далеко не всі переваги мови програмування Python, проте для кваліфікованого аналізу слід розглянути й недоліки та ознайомитись зі сферами застосування мови в розробці.

Одже, головними мінусами Python можна виділити декілька пунктів, а саме:

- Невисока швидкість виконання коду робить Python менш ефективним за інші компільовані мови. Адже він інтерпретованою мовою, тобто код виконується без попередньої обробки та компіляції, що може займати більше часу. І до того ж Python є високорівневою мовою, що робить її більше схожою на людські, ніж машинну.
- Виходячи з вищезгаданого означення, Python вимагає більше коду для досягнення тих же результатів, порівняно з іншими мовами програмування.

					ІАЛЦ.467200.003 ПЗ	Арк.
						25
Зм.	Арк.	№ докум.	Підпис	Дата		

- Глобальний інтерпретатор, який використовує Python, може призводити до певних помилок та проблем в керуванні залежностями, управлінням пам'яттю та інших функціях, які ускладнюють роботу з оптимізацією та управлінням великими проєктами.

Незважаючи на всі недоліки, мова програмування Python є однією з найпопулярніших та використовується в різних сферах розробки, наприклад : додатки, пов'язані з машинним навчанням, мобільні застосунки, десктопні програми та штучний інтелект (AI). Велич та універсальність мови також підкреслюється її використанням у розробці великих корпоративних систем, таких як: Google, YouTube, Facebook, Instagram та NASA.

У підсумку до всього згаданого, необхідно зазначити, що дана мова програмування входить у п'ятірку найбільш затребуваних мов у світі , за версією DOU [13]. Тому вважається гарним варіантом для розробки програм, як для початківців, так і для досвідчених програмістів.

2.2.4 Мова програмування C#

C# є потужною мовою програмування з безпечною типізацією, яку розробила компанія Microsoft під керівництвом Андерса Гейлсберга наприкінці 1990-х роках для платформи .NET [14]. Ця мова програмування має багато спільного з Java і тому головні переваги можуть збігатись з раніше зазначеними для другої з них.

А саме, C# також є об'єктно-орієнтовною мовою і це означає, що її код є модульним та легким для повторного використання через концепції об'єктів та класів для організації коду. Такі властивості, як безпечність, сучасність та маштабованість належать мові програмування C#, адже вона також має безліч вбудованих функцій для уникнення помилок, підвищення безпеки коду, що допомагають в розробці ефективних та надійних ресурсів.

Роботу з C# дуже полегшує інтеграція з системою .NET, яка включає в себе велику кількість корисних інструментів, бібліотек та фреймворків для розробки додатків. Найпопулярнішими з них є:

- ASP.NET – це фреймворк для веб-розробки, який допомагає в розробці масштабних та високопродуктивних веб-сервісів.
- Unity – багатоплатформовий інструмент для розробки 2D та 3D ігор, написаний мовою C# [15].
- Xamarin – є фреймворком для крос-платформової розробки мобільних застосунків для iOS, Android та Windows.

Не дивлячись на різноманіття сфер використання мови C#, яке також відстежується з означень фреймворків, недоліки не будуть виключенням і нижче наведені приклади деяких з них:

- Мова C# є залежною від платформи Microsoft .NET і найчастіше використовується саме в середовищі Windows, що може обмежувати розробку. Хоча існують фреймворки, які частково вирішують проблему у певних сферах програмування.
- Складність синтаксису та наявність великої кількості вбудованих функцій, робить мову складною для вивчення початківцям.
- C# є менш продуктивною у порівнянні з мовами нижчого рівня, такими як: C або C++. Це зумовлено додатковими витратами на виконання коду на платформі .NET. Хоча наразі ця проблема ця проблема вирішується удосконаленням середовища, наприклад запровадженням ІТ оптимізації.
- Деякі технології вважаються застарілими, проте все ще використовуються в багатьох проєктах, наприклад Windows Forms або старі версії ASP.NET.

Одже, мова програмування C# дуже відома через свою продуктивність та надійність, вона буде ідеальним варіантом для розробки масштабних проєктів з використання різних інструментів та фреймворків.

					ІАЛЦ.467200.003 ПЗ	Арк.
						27
Зм.	Арк.	№ докум.	Підпис	Дата		

2.2.5 Мова програмування PHP

Завершує наш аналіз мова програмування PHP, яка на сьогоднішній день вважається найбільш популярною для веб-програмування. PHP (англ. PHP: Hypertext Preprocessor) – скриптова мова, яка була створена для генерації HTML-сторінок на стороні вебсервера у 1995 році данським програмістом Расмус Лердорфом [16].

Ця мова програмування дуже відома своєю простотою використання та широкою поширеністю, що робить її ідеальним варіантом для розробки динамічних веб-сайтів.

Для свого проєкту я обрала саме PHP, адже найчастіше його використовують як серверний інструмент для генерації HTML-коду, який згодом інтерпретується веб-браузером [17].

Мова PHP має велику кількість переваг, які допомагають у виборі інструмента для розробки веб-ресурсу. І найголовніші з них наведені нижче:

- Простий та інтуїтивно зрозумілий синтаксис робить мову популярною серед новачків і привертає увагу до коду, який легко читається незалежно від масштабу та складності проєкту.
- PHP є кросплатформовою мовою і сумісна з усіма популярними операційними системами, такими як: Linux, Windows, MacOS.
- Легка інтеграція коду PHP з HTML-документами не викликає складностей та помилок при розробці веб-сайтів і значно полегшує роботу.
- Мова програмування має великий набір інструментів та фреймворків, а саме: Laravel, Symfony і WordPress, які спрощують процес написання веб-додатків.
- Та не менш важливою перевагою PHP є можливість взаємодіяти з різними системами управління базами даних, наприклад MySQL, PostgreSQL або MongoDB.

					ІАЛЦ.467200.003 ПЗ	Арк.
						28
Зм.	Арк.	№ докум.	Підпис	Дата		

Ознайомившись з усіма перевагами та сферами застосування мови PHP, можна подумати, що вона є ідеальною та не буде викликати проблем і помилок. Хоча насправді це не так і недоліки можуть дещо погіршити уявлення про неї. Адже, PHP в минулому мав проблеми з безпекою і неправильне використання коду призводить до вразливостей веб-сайтів. Також мова не виділяється своєю швидкістю та може бути проблемою для високонавантажених веб-додатків, для яких швидкість виконання функцій є важливою. Через довгу історію та широке використання PHP має значну кількість застарілого коду, проте нові версії мови та бази коду вирішують дану проблему і покращують якість розробки проєктів.

В результаті аналізу, можна сказати, що мова програмування PHP залишається ідеальним варіантом для розробки веб-застосунків. Вона не втрачає своєї популярності з роками і значна кількість сайтів в мережі Інтернет створені на основі PHP. Гарними прикладами великих проєктів є Wikipedia та сайт Верховна Ради України.

2.3 Вибір технологій розробки клієнтської частини

Для кожного проєкту важливими умовами успіху є точність та швидкість розробки програмного забезпечення. Вибір технологій для клієнтської частини веб-застосунку є не менш вагомим процесом, тому що він визначає не лише функціональність та продуктивність готового продукту, а й зручність використання системи користувачем та зовнішній вигляд сайту. Головним призначенням клієнтської частини є пряма взаємодія з додатком, надання візуального інтерфейсу та робота з базою даних для обробки та збереження отриманої інформації від користувача.

У даній частині знайдемо оптимальний варіант технології для створення зовнішнього вигляду сайту, що буде відповідати вимогам клієнта та забезпечувати зручність використання.

					ІАЛЦ.467200.003 ПЗ	Арк.
						29
Зм.	Арк.	№ докум.	Підпис	Дата		

2.3.1 HTML, CSS, JavaScript

FrontEnd розробка, тобто клієнтська частина мають стандартні технології для розробки, які поділяються на три категорії:

- HTML (Hypertext Markup Language) – мова розмітки гіпертексту, є основою веб-сторінки, що відповідає за структуру та її відображення в браузері.
- CSS (Cascading Style Sheets) – мова стилів, яка відповідає за зовнішній вигляд сторінок.
- JavaScript – мова програмування, що додає веб-сторінкам інтерактивність та динамічність.

Саме ці інструменти створюють веб-сторінки, які користувач бачить у браузері та може взаємодіяти з ними.

Для покращення ефективності за функціональності веб-застосунку зазвичай використовують фреймворки для стилізації та розмітки, надбудови над кодом HTML та JS-фреймворки, які в свою чергу приймає браузер та відображає контент.

Як вже зазначено, HTML – це мова, яку чудово розуміють браузери, вона вказує з яких елементів та в якому порядку слід реалізовувати сторінки, включаючи шлях до скриптів та стилів. З використанням цього інструменту є можливість створювати списки, таблиці, параграфи, заголовки, посилання та кнопки, додавати зображення на сторінки. Проте HTML має скорочений функціонал і його не вистачить, щоб оформити інтерфейс елементів, адаптувати сторінки до різних розмірів екрану та змусити посилання та кнопки реагувати на дії користувача. Для того, щоб вирішити цю проблему і полегшити роботу розробника, використовуються стилі CSS та мова програмування JavaScript.

CSS – це мова стилів або, як ще її називають, каскадні таблиці стилів [18]. Слугує для опису інтерфейсу веб-застосунку, розташування елементів на сторінках та адаптації контенту для зручного ознайомлення користувачем.

					ІАЛЦ.467200.003 ПЗ	Арк.
						30
Зм.	Арк.	№ докум.	Підпис	Дата		

Використання CSS разом з HTML значно спрощує розробку, адже надає можливість встановлювати властивості, такі як: стилі шрифту, розмір, кольори та фон для кожного елемента на сторінці. Проте, зі збільшенням кількості контенту стає все складніше прописувати стилі за допомогою чистого CSS, адже це займає багато часу та ресурсів розробника. Для вирішення цієї проблеми були створені готові набори з правилами та стилями, які називаються CSS-фреймворки. На даний час, найпопулярніші з них – це Bootstrap, Tailwind, та Materialize.

JavaScript – це мова програмування, яка була створена для того, щоб оживляти сторінки веб-застосунку і надавати їм інтерактивності. Головними завданнями є реагування на дії користувача, наприклад обробка натискання клавіш або кліки мишкою, управління динамічним контентом, який змінюється від часу або наданих даних.

Для розширення можливостей розробки веб-застосунків так використовуються JS-фреймворки, які є готовими наборами коду і надають структуру, інструменти та шаблони для поширених задач.

2.3.2 CSS-фреймворки

У світі веб-розробки CSS-фреймворки стали невід'язною частиною для швидкого та ефективного створення клієнтської частини сайтів без необхідності додавання зайвого коду на нагромадження проєкту. В 2024 році в список найкращих вийшли 2 фреймворки, а саме: Tailwind та Bootstrap [19]. Вони забезпечують готові компоненти та макети CSS, серед яких є поля вводу, кнопки, навігаційне меню, форми, списки та інше, а також попередньо визначені стилі.

Tailwind – відносно новий фреймворк, але не дивлячись на це користується великою популярністю. Головна відмінність в тому, що він пропонує набір дрібнозернистих класів утиліт, які можна комбінувати.

На рисунку 2.3 зображено головну сторінку сайту CSS-фреймворку Tailwind.

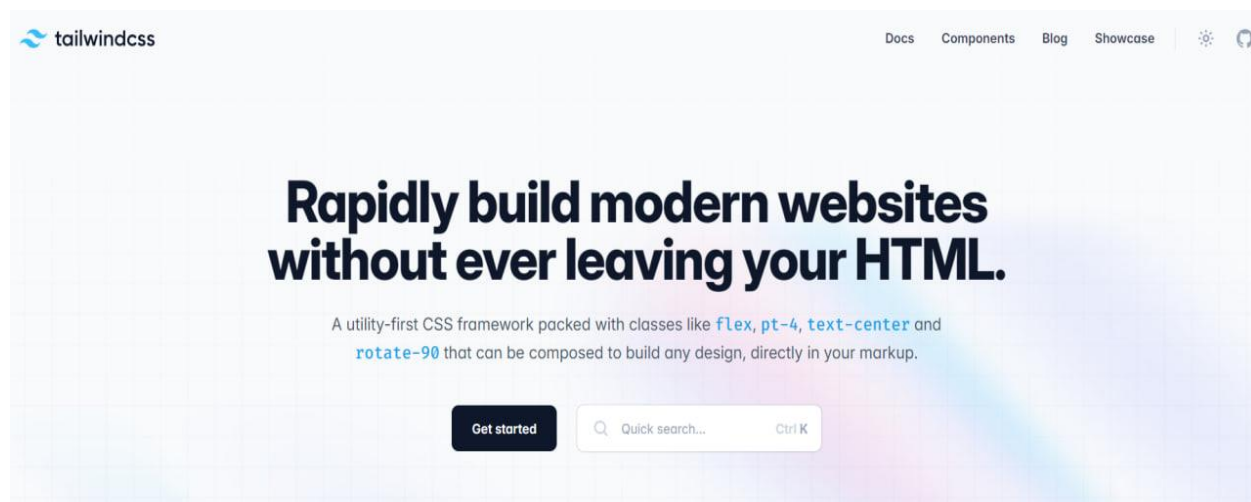


Рисунок 2.3 – Головна сторінка сайту Tailwind

Bootstrap же протягом кількох років є головним інструментом у середовищі CSS. Він надає доступ розробникам до великої кількості готових сіток, стилів та інструментів для полегшення процесу розробки та стилізації веб-сторінок. На рисунку 2.4 зображено головну сторінку сайту Bootstrap.

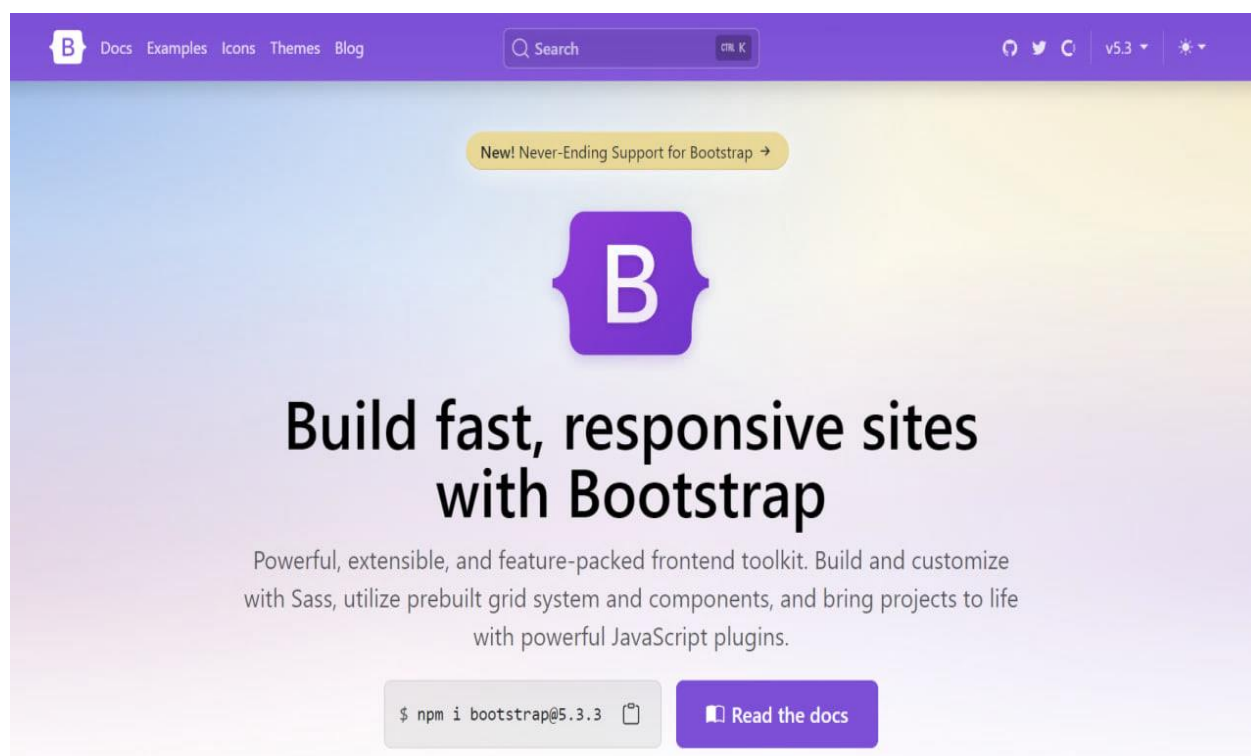


Рисунок 2.4 – Головна сторінка сайту Bootstrap

Зм.	Арк.	№ докум.	Підпис	Дата

2.3.3 JS-фреймворки

Важливу роль в розробці клієнтської частини також відіграють JavaScript фреймворки, хоча програмування можна виконати і без їхнього використання, проте правильно підібране середовище значно полегшить роботу та зменшить час розробки.

Фреймворки JavaScript – це бібліотеки програмування мови JS, які містять попередньо написаний код для використання його у стандартних функціях без перенавантаження системи. Завдяки цим інструментам розробникам доводиться писати меншу кількість коду вручну, що значно підвищує продуктивність та зменшує час програмування. Перелік фреймворків на сьогоднішній день є дуже широким, але серед найпопулярніших протягом одного року можна виділити: React, Angular та Vue.js. На рисунку 2.5 зображено графік кількості пошукових запитів, щодо JS-фреймворків за останній рік по всьому світу за даними Google trends [20].



Рисунку 2.5 – Графік популярності запитів, щодо JS-фреймворків

ReactJS – це бібліотека мови програмування JavaScript з відкритим кодом, створена компанією Facebook та вперше згадана у 2011 році. Після випуску фреймворку у 2013 році, він набув великої популярності завдяки своєму революційному підходу до програмування користувацьких інтерфейсів.

ReactJS базується на компонентах багаторазового користування, що в свою чергу можна назвати блоками коду, які можуть бути повторно застосовані у різних частинах додатку. Більш того, команді розробників ReactJS вдалося пришвидшити оновлення сторінок без зайвого впливу на інші компоненти інтерфейсу. Такий результат досягли через використання абстрактної копії, тобто віртуального DOM дерева, на відміну від інших фреймворків, які працюють з реальним деревом.

Одже, фреймворк ReactJS є потужним та популярним інструментом для створення інтерактивних веб-застосунків, односторінкових додатків (SPA) та мобільних інтерфейсів, незалежно від їхнього масштабу та складності програмування.

Фреймворк Angular також не поступається своєю популярністю та займає друге місце в нашому рейтингу. У 2010 році з'явився в світі програмування завдяки компанії Google та в наш час має велику кількість розробників, які надають перевагу саме йому. Навідміну від ReactJS, який є бібліотекою мови JS, Angular є повноцінним фреймворком, який використовується для розробки інтерфейсів користувача.

Фреймворк використовує компонентну архітектуру та прагне створити структуру для організації проєкту у вигляді організованих частин з чіткими обов'язками, щоб код можна було підтримувати та масштабувати [21]. Таке рішення може сприяти покращенню організації коду програми та його підтримці.

Цей інструмент використовується в багатьох світових компаніях і є ідеальним для створення великих, корпоративних систем, на відміну від ReactJS, який буде чудовим варіантом для менших проєктів, де цінується швидкість роботи.

					ІАЛЦ.467200.003 ПЗ	Арк.
						34
Зм.	Арк.	№ докум.	Підпис	Дата		

Vue.js – це третій прогресивний та популярний фреймворк мови JavaScript, створений для полегшення роботи з односторінковими додатками та інтерфейсами користувача. Він поєднує в собі деякі кращі властивості інструментів React та Angular та доповнює їх простотою використання та легкістю в освоєнні.

Цей фреймворк має широкий вибір плагінів, наприклад Vue Router для роботи з роутингом та Vuex для керування станом застосунку. Більш того Vue.js має велику активну спільноту та багато додаткових розширень і плагінів, що дозволяє полегшити програмування та водночас розширити функціонал додатку. В таблиці 2.1 наведено короткий та швидкий опис найпопулярніших фреймворків у 2024 році - React, Angular та Vue.js.

Таблиця 2.1 – Опис JS- фреймворків

	React	Angular	Vue.js
Компанія	Facebook	Google	Працівник Google
Категорія	Бібліотека JS	Фреймворк JS	Бібліотека JS
Мова	JavaScript, JSX	TypeScript	JavaScript
Архітектура	Заснована на архітектурі DOM	Заснована на архітектурі MVC	Заснована на архітектурі DOM
Тип застосунку	Мобільні застосунки та SPA додатки	Інтерактивні додатки та складні веб- застосунки	Розширені SPA додатки

Кінець таблиці 2.1

	React	Angular	Vue.js
Крива навчання	Легкий для розуміння фреймворк та займає менше часу для вивчення	Складний фреймворк для вивчення, що вимагає розуміння концепцій	Дуже простий для розуміння зі схожим синтаксисом на HTML, CSS і JavaScript
Призначення	Дозволяє створювати компоненти, які організовані в ієрархічну структуру	Дає структурований підхід до розробки, забезпечуючи можливості для маршрутизації, аутентифікації та управлінням стану додатку	Поєднує в собі простоту використання з потужними можливостями, ідеальний для створення невеликих веб-додатків
Популярність	Популярний для розробки веб-додатків	Популярний серед корпоративних розробників та компаній	Популярний серед початківців
Приклади використання	Facebook, Netflix, Twitter, PayPal, Walmart	Google, Forbes, Wix, Microsoft	Alibaba, Baidu, GitLab, Xiaomi

2.4 Вибір системи керування базами даних

На сьогоднішній день існує велика кількість баз даних зі, створеними для них, системами керування. MySQL – це одна з найпопулярніших відкритих систем керування базами даних, яка використовується для зберігання, управління та налаштуванням доступу до структурованих даних. Інформація організована у вигляді пов’язаних між собою таблиць, які працюють за встановленими правилами для регулювання зв’язків між різними полями даних.

База даних MySQL – це клієнт-серверна система, яка має вбудовану багатопотокову бібліотеку, що зв’язується із додатком і допомагає отримати швидкий та простий в управлінні продукт. Це ідеальний інструмент для розробки веб-додатків, адже відомий простотою використання, доступністю та масштабованістю. MySQL підтримує багато великих корпорацій, включаючи Facebook, Twitter, Netflix, Uber, Airbnb та Shotify.

Переваги бази даних MySQL:

- Доступність використання
- Висока продуктивність та швидкість збереження даних
- Можливість масштабованості
- Широка підтримка мов програмування, а саме: PHP, Java, Python, C++
- Наявність ряду вбудованих функцій для зменшення кількості коду

2.5 Хостинг сервіс

В якості хостинг сервісу було обрано платформу InfinityFree, яка надає можливість розміщувати веб-сайти безкоштовно. Надає доступ до використання Cpanel та дозволяє використовувати субдомени.

					ІАЛЦ.467200.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		37

Цей хостинг є ідеальним варіантом для сервісів, які написані за допомогою технологій MySQL та PHP і мають структуру особистих веб-сайтів, невеликих блогів або проєктів з обмеженою кількістю ресурсів. InfinityFree пропонує розробникам та користувачам 5 ГБ дискового простору, 400 баз даних MySQL, необмежену пропускну здатність, відсутність реклами для користувачів веб-сайту та SSL-сертифікат для всіх доменів [22].

Хостинг InfinityFree є найкращим варіантом для проєкту, через його простоту, швидкість налаштування та захищеність від DDoS-атак.

					ІАЛЦ.467200.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		38

ВИСНОВОК ДО РОЗДІЛУ 2

Точне встановлення мети розробки та вивчення характеристик та вимог для інформаційних систем в розділі 1, надає можливість зробити вибір найбільш корисних та придатних інструментів для розробки вебресурсу для підтримки та інформування громадян під час воєнних дій. Для майбутнього проектування серверної частини обрали мову PHP та допоміжні інструменти у вигляді бази даних MySQL та сервіс-хостингу InfinityFree. Це буде ідеальним варіантом для того проєкту, через свою сумісність та простоту використання та вивчення.

Для користувацької частини були обрані стандартні набори технологій, такі як: HTML, CSS та JavaScript, що також є найкращим варіантом для розробки застосунків з невеликою кількістю ресурсів. І для покращення функціональності та полегшення реалізації було прийнято рішення використовувати додаткові бібліотеки та фреймворки.

Таким чином, всі, вище згадані, інструменти та технології для розробки веб-сервісу є обґрунтованими та забезпечать швидку та ефективну роботу, як розробника, так і готового проєкту.

					ІАЛЦ.467200.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		39

РОЗДІЛ 3 ПРОЄКТУВАННЯ ТА РОЗРОБКА ІНФОРМАЦІЙНОГО ВЕБ-РЕСУРСУ

На основі переглянутих та проаналізованих раніше матеріалів дослідження та обраних технологій було розпочато створення веб-сервісу інформаційної підтримки постраждалих від війни з додатковою можливістю реєстрації на семінари. У веб-сервісі також будуть наявні декілька простих і зрозумілих елементів керування, щоб користувач мав змогу зареєструватися на семінар та переглянути корисну та необхідну інформацію, щоб отримати допомогу.

В цьому розділі буде наданий детальний опис документації розробленого веб-сервісу, в результаті чого це може полегшити подальший розвиток системи, а також розуміння роботи застосунку.

3.1 Створення та підключення бази даних

База даних — це система керування інформацією, яка є важливою для більшості сучасних веб-додатків. Вона використовується для зберігання та обробки даних, які створюють привабливу веб-програму для користувачів. Аналізуючи висновки попереднього розділу, для цього проєкту було вирішено використовувати базу даних MySQL. Для того, щоб створити базу даних для майбутнього використання в проєкті, використовуємо веб-додаток phpMyAdmin, який був реалізований для адміністрування СУБД MySQL. Це дає змогу створити базу даних та керувати нею.

Після чого необхідно виконати SQL запити в командному рядку для створення кожного поля в таблиці або скористатись інструментом phpMyAdmin для додавання необхідних елементів. На рисунку 3.1 зображено структуру таблиці “Users” в базі даних.

					ІАЛЦ.467200.003 ПЗ	Арк.
						40
Зм.	Арк.	№ докум.	Підпис	Дата		

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно	Действие
<input type="checkbox"/>	1	id		int(11)	Нет	Нет		AUTO_INCREMENT	Изменить Удалить Ещё
<input type="checkbox"/>	2	username	utf16_general_ci	varchar(255)	Нет	Нет			Изменить Удалить Ещё
<input type="checkbox"/>	3	surname	utf16_general_ci	varchar(255)	Нет	Нет			Изменить Удалить Ещё
<input type="checkbox"/>	4	email	utf16_general_ci	varchar(255)	Нет	Нет			Изменить Удалить Ещё
<input type="checkbox"/>	5	seminar	utf16_general_ci	varchar(255)	Нет	Нет			Изменить Удалить Ещё

Рисунок 3.1 – Структура таблиці “Users”

Цей приклад є набором атрибутів, які використовуються для зберігання введених даних користувачем. Вони надходять в базу даних після заповнення користувачем форм на сторінках веб-ресурсу та зберігаються доки адміністратори не почнуть взаємодіяти з даними та редагувати їх за необхідності. І кожен з них має своє значення, а саме:

- Id- обов’язковий атрибут кожної таблиці є первинним ключем. Він автоматично генерується та є унікальним для кожного користувача. Тип Int слугує для запису цілих чисел, а в цей час AUTO_INCREMENT виконує функцію автоматичного збільшення значення на 1.
- Username – атрибут, що використовується для зберігання ім’я користувача. Тип Varchar позначає текстовий тип змінної довжини та має бути не більше 255 символів.
- Surname – також атрибут з типом Varchar, що зберігає прізвище користувача.
- Email – атрибут зберігає унікальну для кожного користувача адресу електронної пошти та не може бути порожнім.
- Seminar – використовується для зберігання назви семінару, на який зареєстрований користувач.

Для підключення бази даних MySQL до проєкту було використано мову програмування PHP та бібліотеку PDO, що забезпечує легкий та зручний доступ. На рисунку 3.2 наведе код з файлу connect.php для підключення бази даних до проєкту. Для перевірки успішного підключення БД до веб-ресурсу був реалізований код та зображений на рисунку 3.3.

```

<?php

$driver = 'mysql';
$host = 'localhost';
$db_name = 'information-site';
$db_user = 'root';
$db_pass = 'mysql';
$charset = 'utf8';
$options = [PDO::ATTR_ERRMODE => PDO::ERRMODE_EXCEPTION,
            PDO::ATTR_DEFAULT_FETCH_MODE => PDO::FETCH_ASSOC];

try{
    $pdo = new PDO(
        dsn: "$driver:host=$host;dbname=$db_name;charset=$charset", $db_user, $db_pass, $options
    );
}
catch (PDOException $e){
    die("Помилка підключення до бази даних");
}

```

Рисунок 3.2 – Підключення бази даних

```

// Перевірка запиту до БД
5 usages
function dbCheckError($query){
    $errInfo = $query->errorInfo();
    if ($errInfo[0] !== PDO::ERR_NONE){
        echo $errInfo[2];
        exit();
    }
    return true;
}

```

Рисунок 3.3 – Перевірка запиту до БД

3.1.1 Функції взаємодії з базою даних

Для реалізації основних операцій CRUD (Create, Read, Update, Delete) з базою даних було створено функції, які виконують SQL-запити та обробляють результати. В проєкті використовується база даних MySQL.

					ІАЛЦ.467200.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		42

На рисунку 3.4 зображено код з файлу db.php, в якому починається сесія PHP та підключається файл connect.php з підключеною базою даних.

```
session_start();  
require 'connect.php';
```

Рисунок 3.4 – Початок сесії

Функція selectAll виконує запит до БД для отримання всіх даних з таблиці та повертає їх у вигляді масиву.

```
// Запит на отримання даних з однієї таблиці  
5 usages  
function selectAll($table, $params = []){  
    global $pdo;  
    $sql = "SELECT * FROM $table";  
  
    if(!empty($params)){  
        $i = 0;  
        foreach ($params as $key => $value){  
            if (!is_numeric($value)){  
                $value = "".$value."";  
            }  
            if ($i === 0){  
                $sql = $sql . " WHERE $key = $value";  
            }  
            else {  
                $sql = $sql . " AND $key = $value";  
            }  
            $i++;  
        }  
    }  
    $query = $pdo->prepare($sql);  
    $query->execute();  
    dbCheckError($query);  
    return $query->fetchAll();  
}
```

Рисунок 3.5 – Функція selectAll

					ІАЛЦ.467200.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		43

На рисунку 3.6 зображено код функції selectOne, що подібна до попередньої, але отримує один рядок з таблиці на основі наданих параметрів.

```
// Запит на отримання одного рядка з таблиці
2 usages
function selectOne($table, $params = []){
    global $pdo;
    $sql = "SELECT * FROM $table";

    if(!empty($params)){
        $i = 0;
        foreach ($params as $key => $value){
            if (!is_numeric($value)){
                $value = "'".$value."'";
            }
            if ($i === 0){
                $sql = $sql . " WHERE $key = $value";
            }
            else {
                $sql = $sql . " AND $key = $value";
            }
            $i++;
        }
    }
    $query = $pdo->prepare($sql);
    $query->execute();
    dbCheckError($query);
    return $query->fetch();
}
```

Рисунок 3.6 – Функція selectOne

Для внесення даних в таблицю, за використанням атрибутів: ім'я таблиці та асоціативного масиву даних, що необхідно додати, було створено функцію insert. Витягуючи назви стовпців, створює список, розділений комами та надсилає SQL-запит за допомогою методів PDO. На рисунку 3.7 зображено код для функції insert.

```

// Внесення даних в таблицю
1 usage
function insert($table, $params) {
    global $pdo;
    $columns = implode( separator: ", ", array_keys($params));
    $values = implode( separator: ", ", array_map(function ($value) {
        return "'$value'";
    }, array_values($params)));

    $sql = "INSERT INTO $table ($columns) VALUES ($values)";

    $query = $pdo->prepare($sql);
    $query->execute();
    dbCheckError($query);
    return $pdo->lastInsertId();
}

```

Рисунок 3.7 – Функція insert

Для оновлення та видалення рядків з таблиці в базі даних було реалізовано функції update та delete (рис. 3.8 та рис. 3.9). Перша з них оновлює рядок на основі його ідентифікаторів, використовуючи аргументи: ім'я таблиці, ідентифікатор рядка та асоціативний масив, що містить дані для оновлення. А друга видаляє рядок, використовуючи атрибути: ім'я таблиці та ідентифікатор рядка.

```

// Видалення даних з таблиці
no usages
function delete($table, $id){
    global $pdo;

    $sql = "DELETE FROM $table WHERE id = $id";

    $query = $pdo->prepare($sql);
    $query->execute();
    dbCheckError($query);
}

```

Рисунок 3.8 – Функція delete

```

// Оновлення даних в таблиці
no usages
function update($table, $id, $params){
    global $pdo;
    $i = 0;
    $str = '';
    foreach ($params as $key => $value){
        if ($i === 0){
            $str = $str . $key . "= '" . $value . "'";
        }
        else{
            $str = $str . ", " . $key . "= '" . $value . "' ";
        }
        $i++;
    }
    $sql = "UPDATE $table SET $str WHERE id = $id";

    $query = $pdo->prepare($sql);
    $query->execute($params);
    dbCheckError($query);
}

```

Рисунок 3.9 – Функція update

3.2 Опис основних компонентів

3.2.1 Форма реєстрації

Форма реєстрації створена для збору інформації від користувача, що має бажання зареєструватись на один з доступних семінарів. Вона містить поля для вводу даних, таких як ім'я та прізвище користувача, адреса електронної пошти та вибір назви семінару для запису. Перед надсиланням інформації в базу даних вона проходить валідацію на заповненість всіх полів форми, довжину імені, що має бути більша ніж 2 символи та наявності користувача з імейлом в базі даних для обраного семінару. При порушенні цих налаштувань, встановлюється повідомлення про помилку.

					ІАЛЦ.467200.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		46

3.2.2 URL-адреса

Для забезпечення централізованого управління URL-адресами було створено константу “BASE_URL”, що зберігає базову адресу сайту в одному місці. Вона може бути використана в різних частинах коду для посилання на ресурси сайту та спрощує управління адресами через те, що не змушує змінювати кожен рядок коду з посиланням, а лише саму себе. На рисунку 3.11 зображено код для визначення константи “BASE_URL”.

```
<?php  
  
define('BASE_URL', 'http://infohelp.great-site.net/');
```

Рисунок 3.11 – Визначення константи “BASE_URL”

Ідеальним прикладом використання є верхня частина сторінки, а саме Header, тому що він має багато посилань на різні сторінки та зображення, а для того, щоб не змінювати посилання на кожній сторінці, було використано константу, яка змінюється лише в одному файлі, що наведений вище.

```
<header class="container-fluid">  
  <div class="container">  
    <div class="row">  
      <div class="logo col-4">  
        <a href="<?php echo BASE_URL ?>"></a>  
      </div>  
      <nav class="col-8">  
        <ul>  
          <li><a href="<?php echo BASE_URL ?>">Головна</a></li>  
          <li><a href="<?php echo BASE_URL . 'seminar.php'?>">Семінари</a></li>  
          <li><a href="<?php echo BASE_URL . 'about.php'?>">Про нас</a></li>  
        </ul>  
      </nav>  
    </div>  
  </div>  
</header>
```

Рисунок 3.12 – Приклад використання BASE_URL

3.2.3 Конструкція “require”

Конструкція “require” використовується для ефективної організації структури веб-ресурсу та спрощує оновлення коду та його використання. На рисунку 3.13 зображено приклад використання конструкції на головній сторінці веб-застосунку.

```
<!DOCTYPE html>
<html lang="ru">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <link rel="stylesheet" href="assets/css/style.css">
  <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css">
  <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/@fortawesome/fontawesome-free@5.15.4/css/all.min.css">
  <title>Веб-ресурс</title>
</head>
<body>

  <?php require "app/blocks/header.php"?>

  <div class="background-image">
    <div class="banner-info"...>

  <div class="container main" id="section1"...>

  <?php require "app/blocks/footer.php"?>

</body>
</html>
```

Рисунок 3.13 – Приклад використання require

В прикладі використовується конструкція для додавання Header та Footer, які мають бути на кожній сторінці сайту. Це дозволяє редагувати код блоків лише в одному файлі, а зміни будуть застосовані на всіх сторінках, які його включають. Використання конструкції “require” значно полегшує розробку системи та надає можливість уникнути дублювання коду.

					ІАЛЦ.467200.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		49

3.2.4 Форма зворотнього зв'язку

Форма зворотнього зв'язку для веб-ресурсу є важливим компонентом для взаємодії користувача з адміністрацією сайту. Вона дозволяє залишати свої пропозиції, коментарі та питання, що згодом надішле всю заповнену інформацію на пошту адміністратора.

Форма була реалізована з використанням HTML, CSS, JavaScript та фреймворку Bootstrap, а також інтегрований сервіс EmailJS для надсилання повідомлень. Код на мові JS обробляє події форми, виконує валідацію заповнених полів та введених даних та підключає форму зворотнього зв'язку до сервісу EmailJS. Для обробки введених даних отримуємо дані від користувача з таких полів форми, як ім'я, номер телефону (що є не обов'язковим), адреса електронної пошти та текстове повідомлення. Валідація перевіряє поля на непорожні значення для надсилання та виводить помилки в іншому випадку. На рисунках 3.14 та 3.15 зображений код, що працює з формою та передає дані.

```
$(document).ready(function() :void {
  $('#preloading').hide();
  Show usages
  function postData(url :string = "", data :{} = {}) :Promise<Response> - {
    return fetch(url, {method: "POST"...});
  }
  $('#sendMail').on('click', function() :void {
    let to_name = $('#name').val().trim();
    let phone = $('#phone').val().trim();
    let email = $('#email').val().trim();
    let message = $('#message').val().trim();
    let errorMessage = $('#errorMessage');

    errorMessage.hide();
    errorMessage.text('');

    if (!email) {
      errorMessage.text('Введіть Вашу пошту!');
      errorMessage.show();
      return;
    }
    if (!to_name) {
      errorMessage.text('Введіть Ваше ім'я!');
      errorMessage.show();
      return;
    }
    if (!message) {
      errorMessage.text('Введіть Ваше повідомлення!');
      errorMessage.show();
      return;
    }
  }
}
```

Рисунок 3.14 – Код обробки форми зворотнього зв'язку

```

var data :{...} = {
  service_id: 'service_ukgx3sa',
  template_id: 'template_nu3e08m',
  user_id: '_MPuL1JM4p7_vn9ua',
  template_params: {
    to_name,
    phone,
    email,
    message
  }
};
$('#preloading').show();
postData( url: 'https://api.emailjs.com/api/v1.0/email/send', data)
  .then(function() :void {
    $('#preloading').hide();
    $('#mailForm')[0].reset();
    $('#myModal').modal('hide');
  }) Promise<void>
  .catch(function(error) :void {
    $('#preloading').hide();
    errorMessage.text('Oops... ' + JSON.stringify(error));
    errorMessage.show();
  });
});
});

```

Рисунок 3.15 – Код обробки форми зворотнього зв'язку

Сервіс EmailJS дозволяє надсилати електронні листи на адресу адміністратора без використання сервера. Для використання цього API в проєкті необхідно створити обліковий запис на сайті компанії та отримати доступ до сервісу. Створити шаблон листа, який зображений на рисунку 3.16, що буде надсилатися з введеними даними від користувача. Та додати з'єднання з сервісом в проєкт в файлі script.js, який був розглянутий раніше.

```

<p>You got a new message from Helper:</p>
<p>Message sent by {{to_name}},</p>
<p>Phone: {{phone}}</p>
<p>Email: {{email}}</p>
<p style="padding: 12px; border-left: 4px solid #d0d0d0; font-style: italic;">
  {{message}}</p>
<p>Best wishes, Helper!<br><br></p>

```

Рисунок 3.16 – Шаблон листа на сервісі EmailJS

В результаті ми отримуємо надійну та ефективну форму зворотнього зв'язку для взаємодії з користувачами сайту.

					ІАЛЦ.467200.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		51

3.2.5 Доступ до сторінки адміністратора

Для обмеження доступу до сторінки адміністратора сайту було використано змінну `$secret_key`, що додається до посилання на саму сторінку. Запит буде містити значення "key", яке має відповідати зазначеному в коді та надавати доступ до адмін-панелі, а в іншому випадку, якщо параметр відсутній або його значення не збігається, користувач буде потрапляти на головний екран веб-ресурсу. На рисунку 3.17 зображено код контролера в файлі `admin.php`, що створює доступ до сторінки адміністратора лише з використанням параметра "key".

```
<?php
$secret_key = 'supersecretkey';

if (!isset($_GET['key']) || $_GET['key'] !== $secret_key) {
    header( header: 'Location: http://infohelp.great-site.net/ ');
    exit();
}
```

Рисунок 3.17 - Код контролера для адмін-панелі

3.3 Клієнтська частина

Саме Frontend частина веб-ресурсу відіграє важливе значення у забезпечення зручності використання та ефективної взаємодії системи та користувача.

Для візуального зображення клієнтської частини використовувались технології HTML, CSS, Bootstrap та JQuery. Через те, що більшість сторінок та блоків має подібний код, демонстрація не має сенсу. Проте в розділі під номером 4 буде надано доступ до зображень основних сторінок та компонентів веб-ресурсу, що дозволить оцінити інтерфейс системи.

					ІАЛЦ.467200.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		52

ВИСНОВОК ДО РОЗДІЛУ 3

В даному розділі було розглянуто та проаналізовано розробку веб-ресурсу для інформаційної допомоги населенню під час воєнних дій та спроектовано базу даних.

Використання бази даних MySQL та мови програмування PHP у поєднанні з бібліотекою PDO надало змогу створити ефективний механізм для збору та збереження інформації наданої користувачами. Розроблені функції CRUD (створення, читання, оновлення, видалення) забезпечують функціональність та взаємодію з БД.

Для створення зручного та привабливого інтерфейсу було використано основні технології HTML, CSS, Bootstrap та JQuery. Для детального огляду зовнішнього представлення веб-ресурсу створений розділ під номер 4 з візуальним зображенням основних сторінок.

Були розглянуті основні компоненти системи та шляхи їхньої взаємодії для покращення функціональності веб-ресурсу.

Результатом роботи є готовий веб-застосунок з лаконічним дизайном та основними компонентами, що працюють без помилок.

					ІАЛЦ.467200.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		53

РОЗДІЛ 4 АНАЛІЗ ТА ТЕСТУВАННЯ РОЗРОБЛЕНОГО ПРОДУКТУ

Даний розділ є головним представленням результатів розробки веб-застосунку для інформування та підтримки громадян під час воєнних дій. Процес створення системи включав в себе ретельне проектування, вдосконалення функціональності та тестування для забезпечення ефективної та безпечної роботи веб-ресурсу майбутніх користувачів. Розроблений ресурс виступає у ролі єдиного джерела корисної та актуальної інформації для громадян, які потерпають від можливих обстрілів або інших загроз та критичних ситуацій під час збройних конфліктів. Також є порталом для отримання психологічної та інформаційної допомоги у вигляді семінарів з досвідченими фахівцями після реєстрації на сайті.

Програмним забезпеченням можуть користуватись абсолютно всі користувачі мережі Інтернет та отримувати важливу та необхідну інформацію за короткий час при будь-яких умовах. Для отримання доступу до веб-порталу необхідно лише ввести в пошуковий рядок посилання: <http://infohelp.great-site.net/>, так як готовий сайт розташований на хостинг сервісі та має відкритий доступ для користувачів.

Отже, розглянемо функціонал інформаційного веб-застосунку зі сторони клієнта, щоб перевірити реалізацію та функціональність у готовому варіанті.

4.1 Основні сторінки веб-застосунку

На рисунках 4.1 та 4.2 зображено головну сторінку сайту, яка містить баннер-привітання з якірними посиланнями на інші сторінки, розділи з віртуальними картками різних тем критичних ситуацій. Перейшовши на які, можна знайти розгорнуту інформацію та докладніше ознайомитись з нею.

					ІАЛЦ.467200.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		54

Вітаємо на нашому сайті!

Ми для вас підготували корисну інформацію, яка стане в нагоді під час критичних ситуацій і допоможе врятувати життя.

Також на вас чекає можливість взяти участь у семінарах з досвідченими викладачами та наблизитись до вирішення психологічних проблем або ознайомитись з правилами першої медичної допомоги.

Швидка інформативна допомога

Обрати семінар для себе

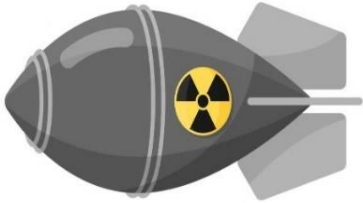
Швидка допомога при загрозі

Рисунок 4.1 - Головна сторінка веб-ресурсу

Швидка допомога при загрозі

Всі пости Обстріли та мінування Окупація Перша допомога


Ядерна загроза



Можливі сценарії ядерної загрози та практичні поради щодо захисту від них.

[Читати більше »](#)


Перша допомога пораненому



Практичні поради та інструкції щодо надання допомоги постраждалим у різних ситуаціях.

[Читати більше »](#)

Окупація міста



Розвідка, безпека та зв'язок, що допоможе реагувати на окупацію та захищати себе.

[Читати більше »](#)

Рисунок 4.2 – Головна сторінка веб-ресурсу

Загальний дизайн сторінки є візуально привабливий та зручний для використання, так як має навігацію по розділах, що відображають пости за обраною темою. Користувач візуально бачить заголовки та невеликий опис кожного блоку та, натискаючи на кнопку «Читати більше», потрапляє на окрему сторінку з розгорнутою інформацією та фактами.

Саме на рисунку 4.3 зображено сторінку одного з блоків, що містить розширений опис тем та зображення.



Що таке ядерна зброя?

Рисунок 4.3 – Сторінка окремого поста

Друга сторінка під назвою «Семінари» зображена на рисунках 4.4 та 4.5 та надає можливість ознайомитись з видами підтримки, які доступні на сайті. Наявна інтерактивна карусель з загальною інформацією про семінари та нижче наведені окремі картки з описом та можливістю перейти на сторінку реєстрації. Вона забезпечена легкою навігацією для ознайомлення з доступною інформацією користувачам, які зацікавлені в підтримці.

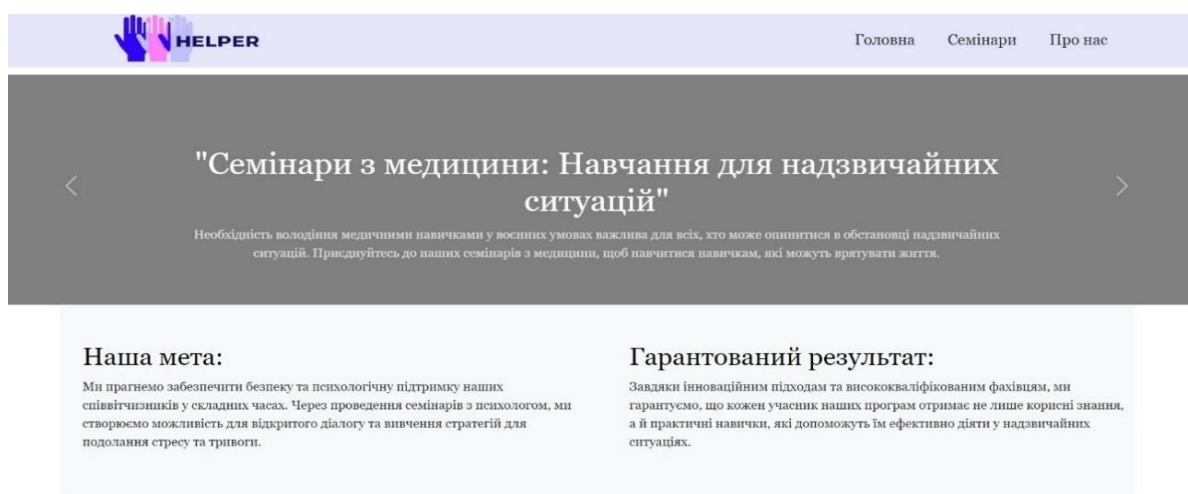


Рисунок 4.4 – Сторінка «Семінари»

					ІАЛЦ.467200.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		56



Семінар з психологом

Семінар психологом, який пропонує цінні інструменти та підтримку для тих, хто зіткнувся з психологічними викликами.

👤 **Спікер:** Логінов Юрій
📅 23 червня 10:00

[Записатись »](#)



Перша медична допомога

Семінар з першої медичної допомоги, де ви отримаєте неціненні навички, що можуть врятувати життя у надзвичайних ситуаціях.

👤 **Спікер:** Куш Марія
📅 20 червня 12:00

[Записатись »](#)



Тривожність під час війни

Спеціально призначеного для тих, хто переживає стрес та тривожність у зв'язку з воєнним конфліктом.

👤 **Спікер:** Чорний Михайло
📅 24 червня 16:00

[Записатись »](#)

[Головна](#) [Семінари](#) [Про нас](#)

Зв'язатись з нами

Телефон: +1234567890

Пошта: example@example.com

© 2024 Company, Inc.



Рисунок 4.5 – Сторінка «Семінари»

На рисунках 4.6 та 4.7 зображена сторінка «Про нас», на якій користувач може знайти інформацію про веб-ресурс та авторів, що його створили. Також вона містить основний опис проєкту та поставлену мету розробників з додатковим блоком форми зворотнього зв'язку для взаємодії з адміністратором сайту.



[Головна](#) [Семінари](#) [Про нас](#)

Проєкт підтримки громадян під час воєнних дій

Вітаємо на нашому веб-ресурсі! Ми - команда фахівців, яка присвятила свою роботу підтримці та інформуванню громадян під час воєнних дій. Наша місія полягає в тому, щоб забезпечити доступ до необхідної інформації, ресурсів та підтримки для тих, хто переживає важливі події та кризові ситуації.



Наша мета - створити безпечне та інформоване середовище для всіх, хто потребує допомоги та підтримки під час воєнних дій.

Рисунок 4.6 – Сторінка «Про нас»

Зм.	Арк.	№ докум.	Підпис	Дата

ІАЛЦ.467200.003 ПЗ

Арк.

57

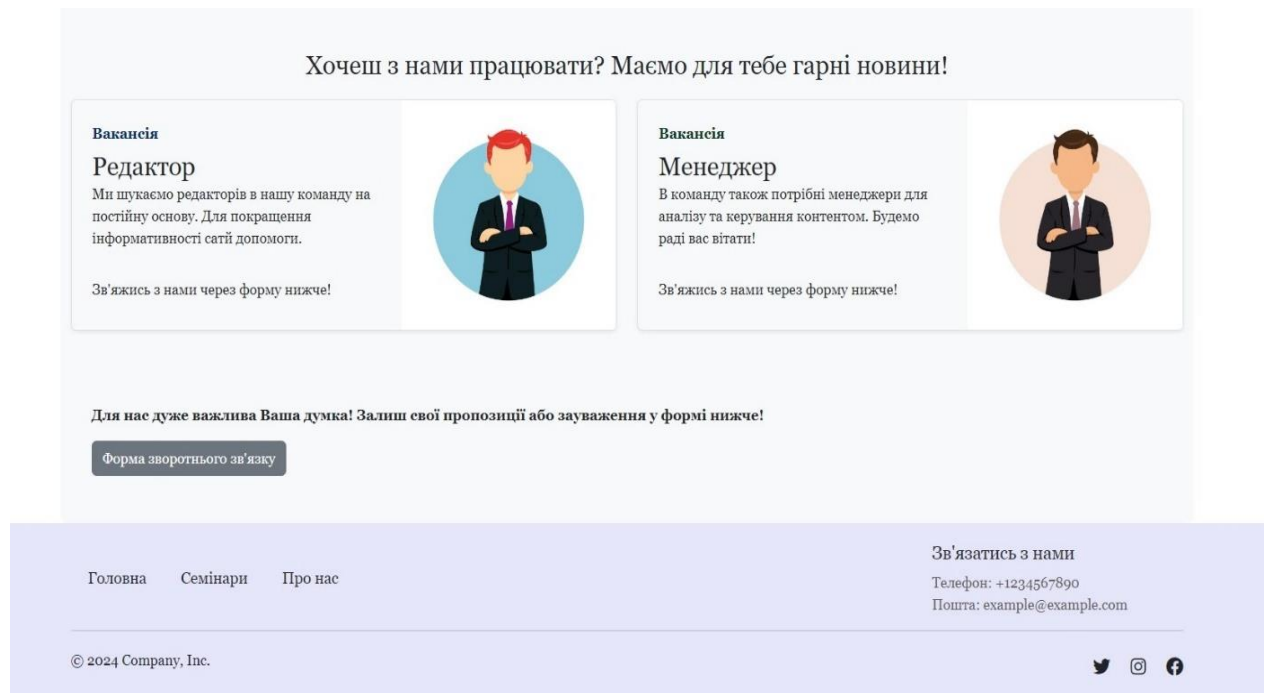


Рисунок 4.7 – Сторінка «Про нас»

Кожна з представлених сторінок має 2 частини, а саме верхню панель, тобто Header та нижню – Footer. Перша з них зображена на рисунку 4.8 та має вигляд навігаційної панелі з посиланнями на головні сторінки веб-ресурсу, які були розглянуті вище та логотип сайту, що є посиланням на головну сторінку.



Рисунок 4.8 – Верхня панель сайту

Нижня панель схожа на верхню, тому що також має навігацію по головним сторінкам, але завжди знаходиться під основним контентом сторінки та зображена на рисунку 4.9 Також має контактну інформацію адміністраторів веб-ресурсу та посилання на соціальні мережі.



Рисунок 4.9 – Нижня панель сайту

4.2 Форма реєстрації

Як ми раніше розглянули, на сторінці «Семінари» користувач має можливість зареєструватися на обраний семінар. Для цього відповідна кнопка перенаправляє на сторінку з формою реєстрації, яка підключена до бази даних, що зберігає всю надану інформацію. Без обов'язкових полів та введені некоректних даних неможливо відправити форму та користувач отримує відповідне повідомлення. Форма реєстрації складається з таких полів (рис. 4.10):

1. Ім'я користувача
2. Прізвище користувача
3. Електронна пошта користувача
4. Поле з вибором семінару

Форма реєстрації на семінар

Ваше ім'я

Ваше прізвище

Email

Ваш Email не буде використовуватись для спаму!

Обрати семінар



Зареєструватись

Рисунок 4.10 – Форма реєстрації

На рисунках 4.11 та 4.12 зображені можливі помилки при введених некоректних даних, або ігноруванні обов'язкових полів.

Зм.	Арк.	№ докум.	Підпис	Дата

Форма реєстрації на семінар

Ім'я має бути більше 2-х символів!

Ваше ім'я

Ю

Ваше прізвище

Бондаренко

Email

uliabondarenko48@gmail.com

Ваш Email не буде використовуватись для спаму!

Обрати семінар



Зареєструватись

Рисунок 4.11 – Помилка в формі реєстрації

Форма реєстрації на семінар

Не всі дані заповненні!

Ваше ім'я

Введіть ваше ім'я...

Ваше прізвище

Введіть ваше прізвище...

Email

Введіть ваш Email...

Ваш Email не буде використовуватись для спаму!

Обрати семінар



Зареєструватись

Рисунок 4.12 – Помилка в формі реєстрації

Зм.	Арк.	№ докум.	Підпис	Дата

ІАЛЦ.467200.003 ПЗ

Арк.

60

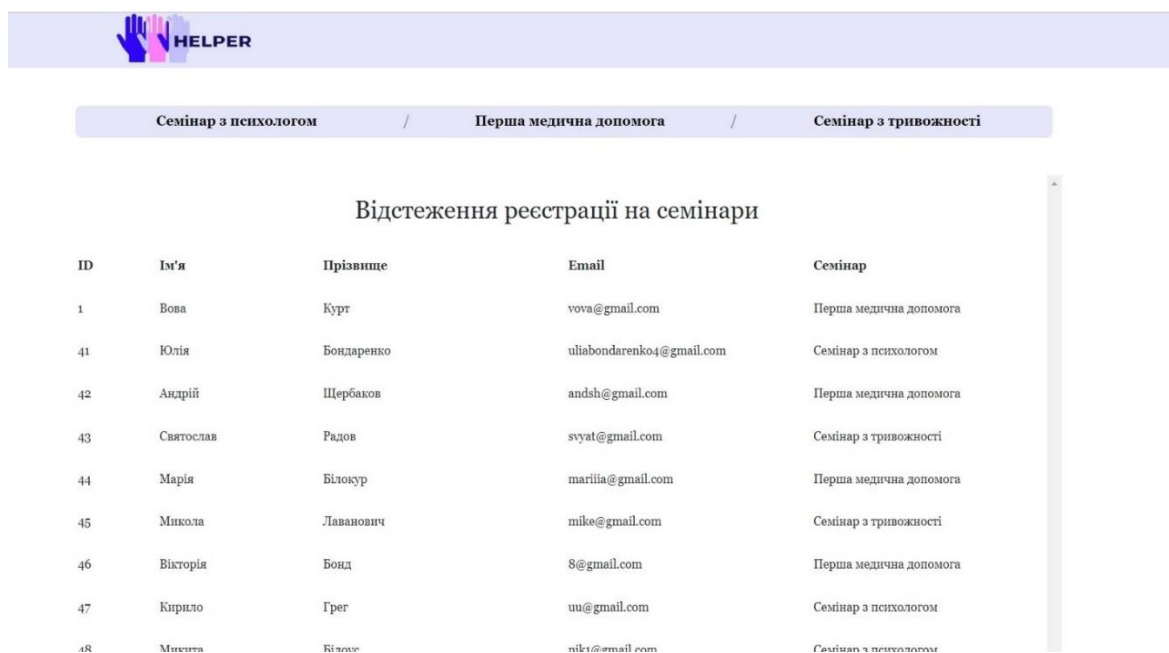
4.3 Сторінка адміністратора

Для зручного відстежування реєстрацій на семінари було реалізовано сторінку адміністратора, яка має доступ лише для однієї особи та відкривається через введення шляху до сторінки в проєкті з додаванням параметру «Ключ», який відомий лише адміністратору та заповнення форми входу. При спробі стороннього користувача потрапити на сторінку та введенням шляху в пошуковий рядок без заданого параметра, він перенаправляється на головну сторінку веб-ресурсу.

На сторінці адміністратора наявна таблиця, що відображає інформацію з бази даних у більш структурованому та лаконічному вигляді з додатковим сортуванням по назві семінару.

В майбутньому при розширенні функціонала сайту є можливість додавати різні аспекти, такі як управління контентом на основних сторінках або проводити аналітику та звітність використання веб-ресурсу.

На рисунку 4.13 зображена сторінка адміністратора, яка також має верхню та нижню частину, але вже без навігації.



ID	Ім'я	Прізвище	Email	Семінар
1	Вова	Курт	vova@gmail.com	Перша медична допомога
41	Юлія	Бондаренко	ulabondarenko4@gmail.com	Семінар з психологом
42	Андрій	Щербаків	andsh@gmail.com	Перша медична допомога
43	Святослав	Радов	svyat@gmail.com	Семінар з тривожності
44	Марія	Білокур	mariaa@gmail.com	Перша медична допомога
45	Микола	Лаванович	mike@gmail.com	Семінар з тривожності
46	Вікторія	Бонд	8@gmail.com	Перша медична допомога
47	Кирило	Грег	uu@gmail.com	Семінар з психологом
48	Микита	Кізілюк	nik1@gmail.com	Семінар з психологом

Рисунок 4.13 – Сторінка адміністратора

4.4 Форма зворотнього зв'язку

Цей функціонал є важливим елементом для веб-ресурсу, тому що надає можливість користувачу надсилати свої відгуки та питання адміністратору сайту на електронну пошту. На рисунку 4.14 зображено модульне вікно форми, що викликається натисканням на кнопку.

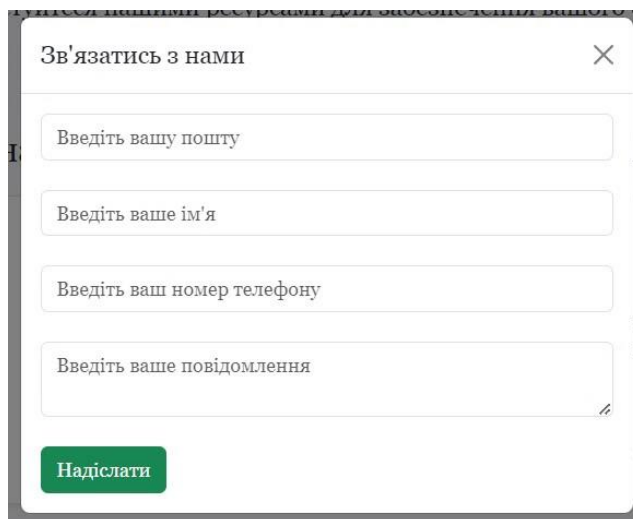


Рисунок 4.14 – Форма зворотнього зв'язку

Користувач вводить в доступні поля свої дані, які також перевіряються валідацією на наявність інформації. Якщо відсутнє ім'я, адреса електронної пошти або повідомлення, користувач отримує помилку з проханням ввести ці дані. Зовнішній вигляд дуже схожий на помилки у формі реєстрації.

Якщо ж всі поля форми пройшли перевірку, введена інформація надсилається на електронну пошту адміна та має такий вигляд, як рисунок 4.15.

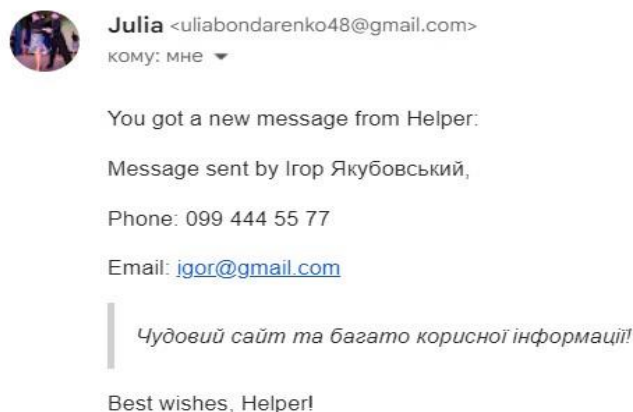


Рисунок 4.15 – Лист з форми зворотнього зв'язку

					ІАЛЦ.467200.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		62

ВИСНОВОК ДО РОЗДІЛУ 4

В даному розділі було представлено реалізацію елементів системи та продемонстровано процес використання веб-ресурсу для інформування та підтримки громадян під час воєнних дій. Було представлено візуальний інтерфейс сторінок сайту та вигляд деяких дій користувача, а саме: реєстрація на семінари та форма зворотнього зв'язку.

Основними перевагами веб-додатку серед існуючих рішень, що були розглянуті раніше, можна вважати інформативність, легкість використання та можливість взаємодії між клієнтом та адміністратором. Результатом проєкту є додаток, що забезпечує оперативний доступ до важливої інформації, яка може допомогти у важкій ситуації та врятувати життя. Додаткові функції, такі як інтерактивні картки, можливість отримання консультацій з фахівцями та форми взаємодії з користувачем, підвищують залученість та корисність ресурсу.

Таким чином, розроблений проєкт доступний для всіх користувачі мережі Інтернет та стане в нагоді під час критичних та небезпечних ситуацій, пропонуючи надійне, ефективне та швидке рішення для реагування на змінювані обставини.

					ІАЛЦ.467200.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		63

ВИСНОВКИ

Ця дипломна робота була присвячена реалізації веб-ресурсу для підтримки та інформування громадян під час воєнних дій. Розроблений застосунок має на меті надавати актуальну та корисну інформацію під час критичних ситуацій та пропонувати підтримку.

У першому розділі було проаналізовано структуру інформаційних систем та розглянуто вже існуючі аналоги для порівняння недоліків та переваг. Це допомогло визначити основні аспекти, які мають бути присутні в майбутньому продукті та сформулювати постановку задачі для реалізації системи. При розробці веб-ресурсу були враховані та вирішені основні недоліки, що допомогло вдосконалити власний проєкт. Зокрема це покращення навігації сторінками сайту, зменшення навантаження уваги користувача великою кількістю інформації та додавання інтерактивних елементів, що створюють взаємодію користувача з порталом.

Другий розділ висвітлює аналіз та вибір технологій розробки веб-застосунку. Було розглянуто найпопулярніші рішення на сьогоднішній день та порівнянні між собою для пошуку ідеального варіанту. Вибір технологій здійснювався, як для клієнтської частини ресурсу, так і для серверної. Це допомогло обрати фреймворки та сервіси, що найкраще інтегруються для створення легкого та зручного веб-сайту.

Третій розділі був призначений розробці ключових компонентів серверної та клієнтської частини. Зосереджено увагу на написанні коду та функціональності веб-ресурсу.

Четвертий розділ надає можливість візуально протестувати готовий веб-ресурс, який був розроблений впродовж написання дипломного проєкту. Були представлені головні сценарії взаємодії користувача з системою та представленні рекомендації користування застосунком.

					ІАЛЦ.467200.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		64

12. Python [Електронний ресурс] – Режим доступу до ресурсу:
<https://uk.wikipedia.org/wiki/Python>
13. Бекенд у 2023-2024 [Електронний ресурс] – Режим доступу до ресурсу:
<https://dou.ua/forums/topic/45581/>
14. C Sharp [Електронний ресурс] – Режим доступу до ресурсу:
https://uk.wikipedia.org/wiki/C_Sharp
15. Unity [Електронний ресурс] – Режим доступу до ресурсу:
[https://uk.wikipedia.org/wiki/Unity_\(%D1%96%D0%B3%D1%80%D0%BE%D0%B2%D0%B8%D0%B9_%D1%80%D1%83%D1%88%D1%96%D0%B9\)](https://uk.wikipedia.org/wiki/Unity_(%D1%96%D0%B3%D1%80%D0%BE%D0%B2%D0%B8%D0%B9_%D1%80%D1%83%D1%88%D1%96%D0%B9))
16. PHP [Електронний ресурс] – Режим доступу до ресурсу:
<https://uk.wikipedia.org/wiki/PHP>
17. Що таке PHP? [Електронний ресурс] – Режим доступу до ресурсу:
<https://freehost.com.ua/ukr/faq/wiki/chto-takoe-php/>
18. CSS [Електронний ресурс] – Режим доступу до ресурсу:
<https://developer.mozilla.org/ru/docs/Web/CSS>
19. Top 5 CSS Frameworks in 2024 [Електронний ресурс] – Режим доступу до ресурсу:
<https://blog.stackademic.com/the-top-5-css-frameworks-in-2024-boosting-web-development-efficiency-e5592992df6e>
20. Google Trends [Електронний ресурс] – Режим доступу до ресурсу:
<https://trends.google.com/trends/explore?date=today%2012-m,today%2012-m,today%2012-m&geo=.,&q=%2Fm%2F01211vxv,%2Fg%2F11c6w0ddw9,%2Fg%2F11c0vmgx5d>
21. What is Angular? [Електронний ресурс] – Режим доступу до ресурсу:
<https://angular.io/guide/what-is-angular>
22. InfinityFree [Електронний ресурс] – Режим доступу до ресурсу:
<https://dash.infinityfree.com/accounts>

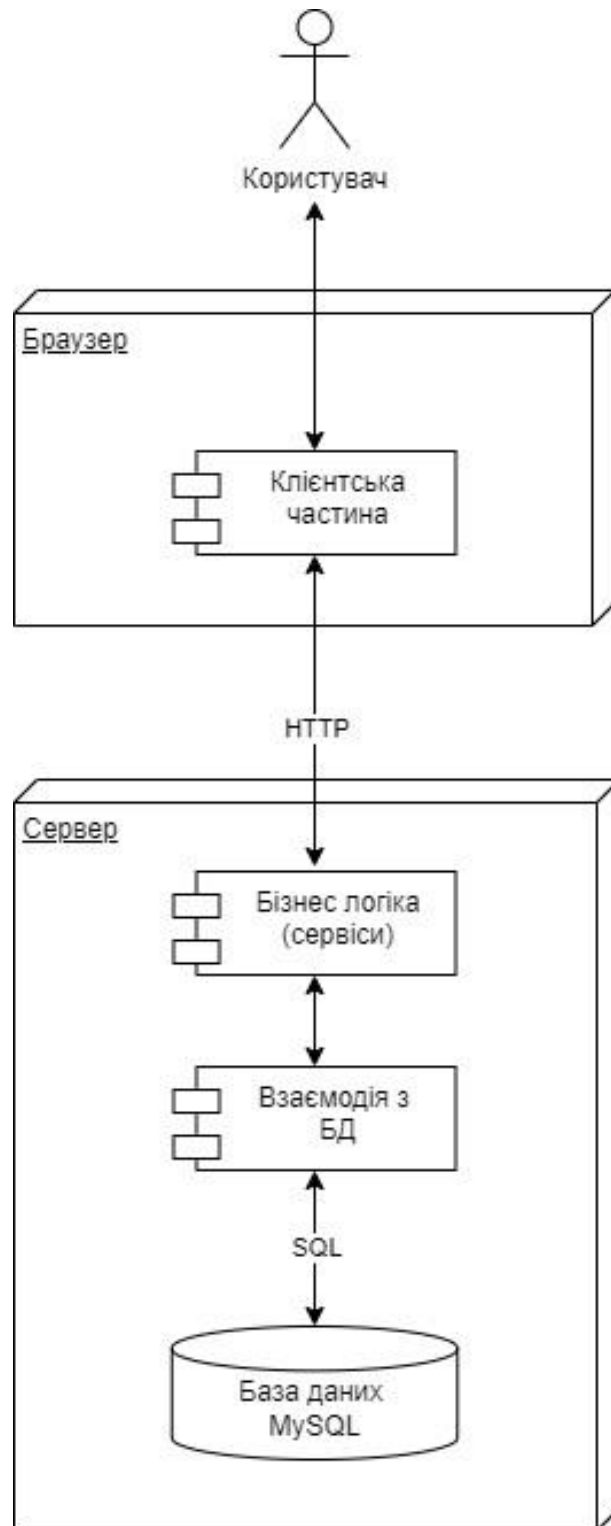
ДОДАТОК 1

Веб-ресурс для підтримки та інформування громадян під час воєнних дій

Структурна схема системи ІАЛЦ.467200.004 Д1

Аркушів 1

Київ 2024 р



					ІАЛЦ.467200.004 Д1		
		№ докум.	Підпис	Дата			
Розробив	Бондаренко Ю.А.				Літ.	Аркуш	Аркушів
Перевірив	Щур В.Ю.					1	1
Н. Контр.	Іваніщев Б.В.				КПІ ім. Ігоря		
Затвердив					Сікорського, ФІОТ, ІО-04		

Веб-ресурс для підтримки та
інформування громадян під час
воєнних дій
Структурна схема системи

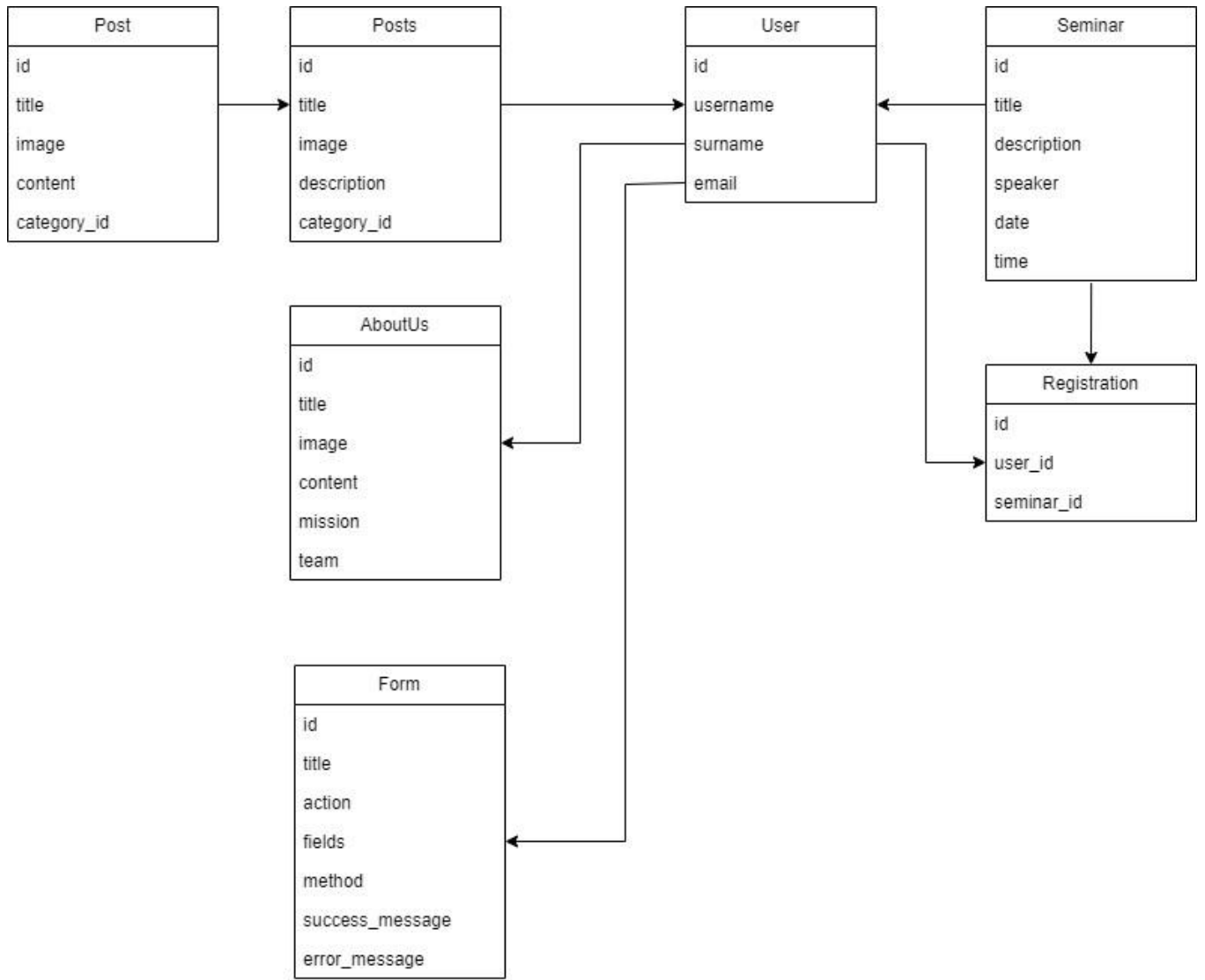
ДОДАТОК 2

Веб-ресурс для підтримки та інформування громадян під час воєнних дій

Діаграма даних
(функціональна схема)
ІАЛЦ.467200.005 Д2

Аркушів 1

Київ 2024 р



					ІАЛЦ.467200.005 Д2		
		№ докум.	Підпис	Дата			
Розробив	Бондаренко Ю.А.				Літ.	Аркуш	Аркушів
Перевірив	Щур В.Ю.					1	1
Н. Контр.	Іваніщев Б.В.				КПІ ім. Ігоря Сікорського, ФІОТ, ІО-04		
Затвердив					Веб-ресурс для підтримки та інформування громадян під час воєнних дій Діаграма даних (функціональна схема)		

ДОДАТОК 3

Веб-ресурс для підтримки та інформування громадян під час воєнних дій

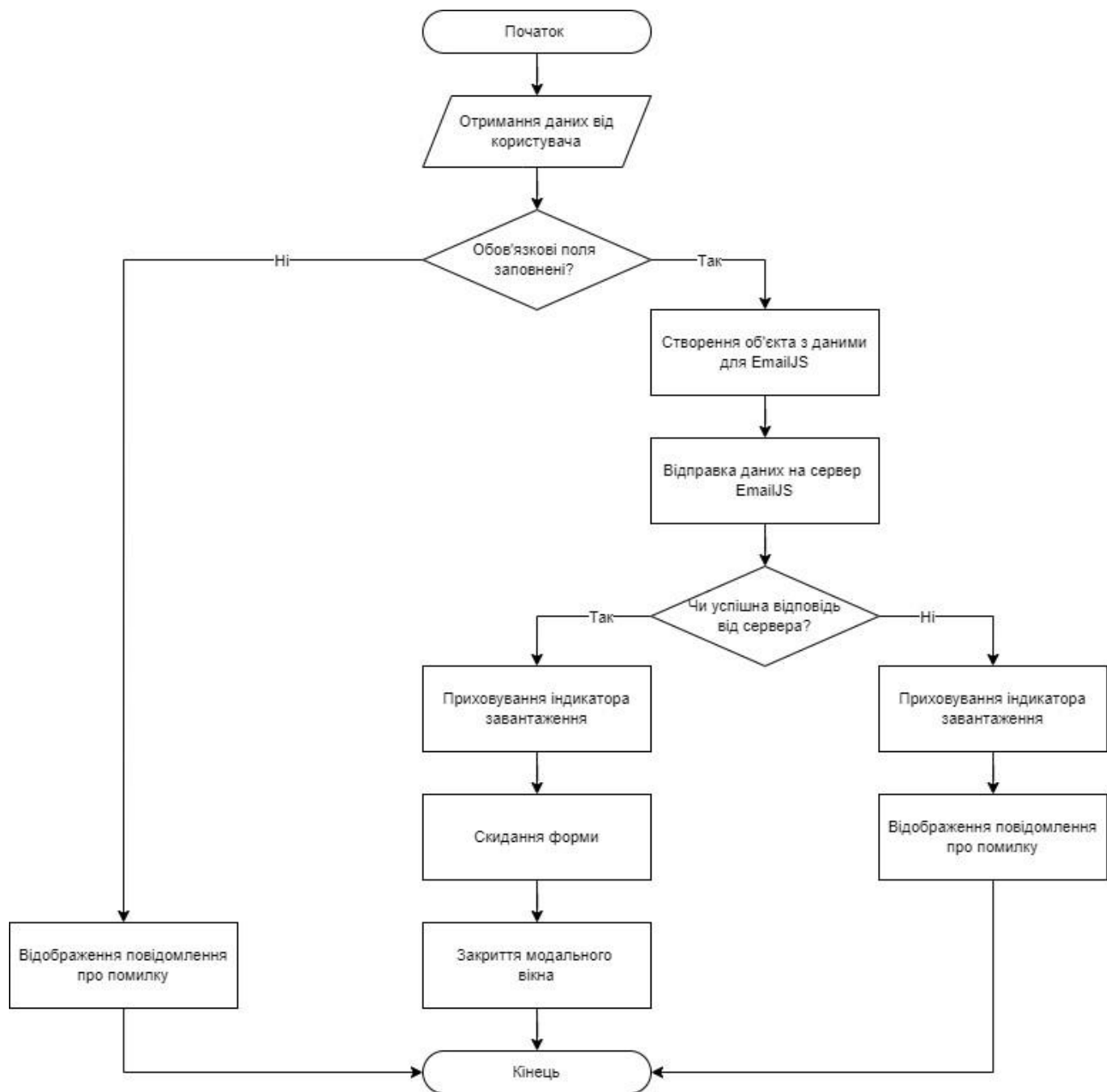
Алгоритм дії форми зворотнього зв'язку

(принципова схема)

ІАЛЦ.467200.006 ДЗ

Аркушів 1

Київ 2024 р



					ІАЛЦ.467200.006 ДЗ		
		№ докум.	Підпис	Дата			
Розробив	Бондаренко Ю.А.				Літ.	Аркуш	Аркушів
Перевірив	Щур В.Ю.					1	1
Н. Контр.	Іваніщев Б.В.				КПІ ім. Ігоря Сікорського, ФІОТ, ІО-04		
Затвердив							
Веб-ресурс для підтримки та інформування громадян під час воєнних дій Алгоритм дії форми зворотнього зв'язку (принципова схема)							

ДОДАТОК 4

Веб-ресурс для підтримки та інформування громадян під час воєнних дій

Текст програмного коду

ІАЛЦ.467200.007 Д4

Аркушів 8

Київ 2024 р

database / connect.php

```
<?php

// Підключення бази даних
$driver = 'mysql';
$host = 'localhost';
$db_name = 'information-site';
$db_user = 'root';
$db_pass = 'mysql';
$charset = 'utf8';
$options = [PDO::ATTR_ERRMODE => PDO::ERRMODE_EXCEPTION,
            PDO::ATTR_DEFAULT_FETCH_MODE => PDO::FETCH_ASSOC];

try{
    $pdo = new PDO(
        "$driver:host=$host;dbname=$db_name;charset=$charset",
        $db_user, $db_pass, $options
    );
}
catch (PDOException $e){
    die("Помилка підключення до бази даних");
}
```

database / db.php

```
<?php
session_start();
require 'connect.php';

// Перевірка запиту до БД
function dbCheckError($query){
    $errInfo = $query->errorInfo();
    if ($errInfo[0] !== PDO::ERR_NONE){
        echo $errInfo[2];
        exit();
    }
    return true;
}

// Запит на отримання даних з однієї таблиці
function selectAll($table, $params = []){
    global $pdo;
    $sql = "SELECT * FROM $table";
    if(!empty($params)){
        $i = 0
```

					ІАЛЦ.467200.007 Д4			
		№ докум.	Підпис	Дата				
Розробив	Бондаренко Ю.А.				Веб-ресурс для підтримки та інформування громадян під час воєнних дій	Літ.	Аркуш	Аркушів
Перевірив	Щур В.Ю.						1	8
Н. Контр.	Іваніщев Б.В.				КПІ ім. Ігоря Сікорського, ФІОТ, ІО-04			
Затвердив								

```

foreach ($params as $key => $value){
    if (!is_numeric($value)){
        $value = "'".$value."'";
    }
    if ($i === 0){
        $sql = $sql . " WHERE $key = $value";
    }
    else {
        $sql = $sql . " AND $key = $value";
    }
    $i++;
}
}
$query = $pdo->prepare($sql);
$query->execute();
dbCheckError($query);
return $query->fetch();
}

// Внесення даних в таблицю
function insert($table, $params) {
    global $pdo;
    $columns = implode(", ", array_keys($params));
    $values = implode(", ", array_map(function ($value) {
        return "'$value'";
    }, array_values($params)));

    $sql = "INSERT INTO $table ($columns) VALUES ($values)";

    $query = $pdo->prepare($sql);
    $query->execute();
    dbCheckError($query);
    return $pdo->lastInsertId();
}

// Оновлення даних в таблиці
function update($table, $id, $params){
    global $pdo;
    $i = 0;
    $str = '';
    foreach ($params as $key => $value){
        if ($i === 0){
            $str = $str . $key . "= '" . $value . "'";
        }
        else{
            $str = $str . ", " . $key . "= '" . $value . "' ";
        }
        $i++;
    }
    $sql = "UPDATE $table SET $str WHERE id = $id";
}

```

					ІАЛЦ.467200.007 Д4	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		2

```

$query = $pdo->prepare($sql);
$query->execute($params);
dbCheckError($query);
}

// Видалення даних з таблиці
function delete($table, $id){
    global $pdo;

    $sql = "DELETE FROM $table WHERE id = $id";

    $query = $pdo->prepare($sql);
    $query->execute();
    dbCheckError($query);
}

```

controllers / users.php

```

<?php
include ("app/database/db.php");

$errorMsg = '';

$users = selectAll('users');

// Код для форми реєстрації на семінар
if($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['button-
reg'])) {
    // Отримання даних з форми
    $login = trim($_POST['login']);
    $name = trim($_POST['surname']);
    $email = trim($_POST['mail']);
    $type = trim($_POST['seminar']);

    if($login === '' || $name === '' || $email === '' || $type ===
''){
        $errorMsg = "Не всі дані заповненні!";
    }elseif (mb_strlen($login, 'UTF8') < 2) {
        $errorMsg = "Ім'я має бути більше 2-х символів!";
    }else{
        $existence = selectOne('users', ['email' => $email, 'seminar'
=> $type]);
        if($existence && $existence['email'] === $email &&
$existence['seminar'] === $type){
            $errorMsg = "Користувач з таким Email на цей семінар вже
zareestrovаний!";
        }else{
            // Запис даних користувача в базу даних
            $post = [
                'username' => $login,

```

Зм.	Арк.	№ докум.	Підпис	Дата

```

        'surname' => $name,
        'email' => $email,
        'seminar' => $type
    ];

    $id = insert('users', $post);
    $user = selectOne('users', ['id' => $id]);

    header('Location: http://localhost/information-site/');

    }
}
}else{
    $login = '';
    $name = '';
    $email = '';
}
}

```

controllers / admin.php

```

<?php
$secret_key = 'supersecretkey';

if (!isset($_GET['key']) || $_GET['key'] !== $secret_key) {
    header('Location: http://infohelp.great-site.net/');
    exit
}

```

blocks / header.php

```

<header class="container-fluid">
    <div class="container">
        <div class="row">
            <div class="logo col-4">
                <a href="<?php echo BASE_URL ?>"></a>
            </div>
            <nav class="col-8">
                <ul>
                    <li><a href="<?php echo BASE_URL ?>">Головна</a></li>
                    <li><a href="<?php echo BASE_URL .
'seminar.php'?>">Семінари</a></li>
                    <li><a href="<?php echo BASE_URL . 'about.php'?>">Про
нас</a></li>
                </ul>
            </nav>
        </div>
    </div>
</header>

```

						ІАЛЦ.467200.007 Д4	Арк. 4
Зм.	Арк.	№ докум.	Підпис	Дата			

blocks / footer.php

```
<footer class="container-fluid">
  <div class="container">
    <div class="footer-top">
      <div class="footer-links">
        <ul>
          <li><a href="<?php echo BASE_URL .
'about.php'?>">Про нас</a></li>
          <li><a href="<?php echo BASE_URL .
'seminar.php'?>">Семінари</a></li>
          <li><a href="<?php echo BASE_URL
?>">Головна</a></li>
        </ul>
      </div>
      <div class="contact-info">
        <h5>Зв'язатись з нами</h5>
        <p>Телефон: +1234567890</p>
        <p>Пошта: example@example.com</p>
      </div>
    </div>
    <hr>
    <div class="footer-bottom">
      <div class="copyright">
        <p>© 2024 Company, Inc.</p>
      </div>
      <div class="social-links">
        <a href="https://x.com/?lang=ru"><i
class="fab fa-twitter"></i></a>
        <a href="https://www.instagram.com/"><i
class="fab fa-instagram"></i></a>
        <a
href="https://www.facebook.com/?locale=uk_UA"><i class="fab fa-
facebook"></i></a>
      </div>
    </div>
  </div>
</footer>
```

admin / index.php

```
<?php

include '../app/database/db.php';
include '../path.php';
include '../app/controllers/admin.php';

$users = selectAll('users');

?>
```

						ІАЛЦ.467200.007 Д4	Арк.
							5
Зм.	Арк.	№ докум.	Підпис	Дата			

```

<!DOCTYPE html>
<html lang="ru">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <link rel="stylesheet" href="../assets/css/admin.css">
  <link rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/boots
trap.min.css">
  <title>Кабінет адміна</title>
</head>
<body>

  <?php require "../app/blocks/header-admin.php" ?>

  <div class="container my-5" id="tags"...>

<div class="container">
  <div class="row">

    <div class="posts col-12">
      <div class="row title-table">
        <h2>Відстеження реєстрації на семінари</h2>
        <div class="col-1">ID</div>
        <div class="col-2">Ім'я</div>
        <div class="col-3">Прізвище</div>
        <div class="col-3">Email</div>
        <div class="col-3">Семінар</div>
      </div>
      <?php foreach ($users as $user): ?>
        <div class="row post">
          <div class="col-1"><?=$user['id'];?></div>
          <div class="col-2"><?=$user['username'];?></div>
          <div class="col-3"><?=$user['surname'];?></div>
          <div class="col-3"><?=$user['email'];?></div>
          <div class="col-3"><?=$user['seminar'];?></div>
        </div>
      <?php endforeach; ?>
    </div>
  </div>
</div>

  <?php require "../app/blocks/footer-admin.php" ?>

</body>
</html>

```

assets / js / script.js

```
$(document).ready(function() {
```

									ІАЛЦ.467200.007 Д4	Арк.
										6
Зм.	Арк.	№ докум.	Підпис	Дата						

```

$('#preloading').hide();
function postData(url = "", data = {}) {
    return fetch(url, {
        method: "POST",
        mode: "cors",
        cache: "no-cache",
        credentials: "same-origin",
        headers: {
            "Content-Type": "application/json",
        },
        redirect: "follow",
        refererPolicy: "no-referrer",
        body: JSON.stringify(data),
    });
}
$('#sendMail').on('click', function() {
    let to_name = $('#name').val().trim();
    let phone = $('#phone').val().trim();
    let email = $('#email').val().trim();
    let message = $('#message').val().trim();
    let errorMessage = $('#errorMessage');

    errorMessage.hide();
    errorMessage.text('');

    if (!email) {
        errorMessage.text('Введіть Вашу пошту!');
        errorMessage.show();
        return;
    }
    if (!to_name) {
        errorMessage.text('Введіть Ваше ім`я!');
        errorMessage.show();
        return;
    }
    if (!message) {
        errorMessage.text('Введіть Ваше повідомлення!');
        errorMessage.show();
        return;
    }
    var data = {
        service_id: 'service_ukgx3sa',
        template_id: 'template_nu3e08m',
        user_id: '_MPuL1JM4p7_vn9ua',
        template_params: {
            to_name,
            phone,
            email,
            message
        }
    };
});

```

Зм.	Арк.	№ докум.	Підпис	Дата

ІАЛЦ.467200.007 Д4

Арк.

7

```
        $('#preloading').show();
        postData('https://api.emailjs.com/api/v1.0/email/send',
data)
            .then(function() {
                $('#preloading').hide();
                $('#mailForm')[0].reset();
                $('#myModal').modal('hide');
            })
            .catch(function(error) {
                $('#preloading').hide();
                errorMessage.text('Oops... ' +
JSON.stringify(error));
                errorMessage.show();
            });
    });
});
```

					ІАЛЦ.467200.007 Д4	Арк.
						8
Зм.	Арк.	№ докум.	Підпис	Дата		