

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Комп'ютерних наук і технологій

(повне найменування факультету)

Кафедра системного аналізу та обчислювальної математики

(повне найменування кафедри)

Пояснювальна записка

до дипломного проекту (роботи)

Бакалавра

(ступінь вищої освіти)

на тему Розробка клієнт-серверної інформаційної системи
громадської організації

(назва теми)

Виконав: студент 4 курсу, групи КНТз-810сп

Спеціальності 124 – Системний аналіз
(код і найменування спеціальності)

Освітня програма (спеціалізація)

«Інтелектуальні технології та прийняття рішень в
складних системах»

КОЗИРЬ Д.В.

(ПРІЗВИЩЕ та ініціали)

Керівник РЯБЕНКО А. Є.
(ПРІЗВИЩЕ та ініціали)

Рецензент ЛОЗОВСЬКА Л.І.
(ПРІЗВИЩЕ та ініціали)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет _____ Комп'ютерних наук і технологій _____
Кафедра _____ Системного аналізу та обчислювальної математики _____
Ступінь вищої освіти _____ Бакалавр _____
Спеціальність _____ 124 – Системний аналіз _____
(код і найменування)
Освітня програма (спеціалізація) «Інтелектуальні технології та прийняття рішень в складних системах» _____
(назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮ

В. о. завідувач кафедри _____
_____ **Е. В. Терещенко** _____
« _____ » _____ 20 _____ року

З А В Д А Н Н Я
НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА(КИ)

_____ **КОЗИРЬ Денис Володимирович** _____

(ПРИЗВИЩЕ, ім'я, по батькові)

1. Тема проєкту (роботи) _____ Розробка клієнт-серверної інформаційної системи громадської організації _____

керівник проєкту (роботи) _____ **РЯБЕНКО А.Є., д.ф.-м.н., професор.,** _____
(науковий ступінь, вчене звання, ПРИЗВИЩЕ, ім'я, по батькові)

затверджені наказом закладу вищої освіти від « 22 » лютого 2023 року № 407

2. Строк подання студентом проєкту (роботи) _____ 12 червня 2023 _____

3. Вихідні дані до проєкту (роботи) _____ Дані про потреби громадської організації в інформаційній системі, технічні вимоги до системи, а також вимоги користувачів до зручності та функціональності. _____

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) _____ Опис методів та інструментів для розробки клієнт-серверної інформаційної системи, загальний опис функціоналу системи, аналіз користувацького досвіду та використання веб-технологій для покращення ефективності роботи громадської організації. _____

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, кількість слайдів, плакатів)

_____ Макети інтерфейсу системи та взаємодії між різними елементами системи, схеми взаємодії клієнта та сервера. _____

6. Консультанти розділів проєкту (роботи)

Розділ	ПРИЗВИЩЕ, ініціали та посада консультанта	Підпис, дата	
		завдання видав	приймав виконане завдання
1	РЯБЕНКО А.Є., д.ф.-м.н., професор.		
2	РЯБЕНКО А.Є., д.ф.-м.н., професор.		
3	РЯБЕНКО А.Є., д.ф.-м.н., професор.		
4	РЯБЕНКО А.Є., д.ф.-м.н., професор.		
Нормоконтроль	ШИРОКОРАД Д.В., к.ф.-м.н., ст. викл.		

7. Дата видачі завдання « 22 » лютого 2023 року.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проєкту (роботи)	Строк виконання етапів проєкту (роботи)	Примітка
1	Сформулювати мету на основні завдання дипломної роботи	11.09.2023 – 18.02.2023	
2	Визначення проблеми, завдань та методів дослідження	18.02.2023 – 22.02.2023	
3	Аналіз потреб та систем в громадських організаціях	22.02.2023 – 02.03.2023	
4	Створення клієнт-серверної інформаційної системи	02.03.2023 – 17.05.2023	
5	Підводження підсумків	17.05.2023 – 12.05.2023	
6	Оформлення пояснювальної записки	12.05.2023 – 07.06.2023	
7	Попередній захист дипломної роботи та отримання рецензій	07.06.2023 – 12.06.2023	
8	Захист дипломної роботи	14.06.2023	

Студент(ка)

_____ КОЗИРЬ Д.В.
(підпис) (Ім'я ПРИЗВИЩЕ)

Керівник проєкту (роботи)

_____ РЯБЕНКО А.Є.
(підпис) (Ім'я ПРИЗВИЩЕ)

РЕФЕРАТ

Обсяг: 85 сторінок, 13 рисунків, 19 джерел

Об'єкт дослідження — Підвищення ефективності роботи громадської організації за допомогою цифрових технологій.

Мета дослідження — Визначити, як розробка клієнт-серверної інформаційної системи може вплинути на діяльність громадської організації, поліпшити комунікацію з аудиторією та вдосконалити процеси взаємодії.

Метод дослідження — Аналіз потреб організації та її користувачів, проектування, розробка та тестування інформаційної системи.

Ціль дослідження створити клієнт-серверну інформаційну систему, що дозволить громадській організації ефективніше виконувати свої ключові завдання: поширення новин, здійснення запису на конференції та консультування, надання можливості скачувати файли та запису до лав організації. З урахуванням специфіки роботи організації та потреб її користувачів, система буде містити інструменти для поширення конференцій, новин організації, ознайомлення з науковою діяльністю та залучення до нових членів до громадської організації.

ГРОМАДСЬКА ОРГАНІЗАЦІЯ, ІНФОРМАЦІЙНА СИСТЕМА, КЛІЄНТ-СЕРВЕРНА АРХІТЕКТУРА, НАУКОВА ДІЯЛЬНІСТЬ, ЗАЛУЧЕННЯ ЛЮДЕЙ ДО ЛАВ, ПОШИРЕННЯ КОНФЕРЕНЦІЇ, HTML, CSS, JQUERY, AJAX, BOOTSTRAP, JS.

ЗМІСТ

Завдання	Error! Bookmark not defined.
Реферат.....	3
Вступ.....	6
Розділ 1: Основи клієнт-серверних систем для громадських організацій	8
1.1 Розгляд концепції клієнт-серверної архітектури.....	8
1.2 Визначення основних компонентів клієнт-серверної системи.....	11
1.3 Аналіз основних технологій та інструментів, які використовуються для розробки клієнт-серверних систем	14
1.4 Роль та значення клієнт-серверних систем для громадських організацій	18
Розділ 2: Аналіз потреб та систем в громадських організаціях.....	23
2.1 Аналіз потреб громадської організації	23
2.2 Визначення вимог до системи.....	27
2.3 Аналіз наявних аналогів та їх порівняльний аналіз.....	30
Розділ 3 Розробка та створення клієнт-серверної інформаційної системи громадської організації.....	45
3.1 Визначення цілей та проектування дизайну	45
3.2 Вибір технологічного стеку для розробки.....	53
3.3 Розробка бази даних і серверної частини системи	55
3.4 Розробка клієнтської частини системи	58
3.5 Тестування системи	60
Розділ 4 Підсумки створення клієнт-серверної інформаційної системи.....	62
4.1 Результати роботи	62
4.2 Можливі напрямки подальшого розвитку системи.....	63
Перелік посилань.....	65
Додаток А Приклад коду клієнтської частини.....	68
Додаток Б Приклад коду серверної частини.....	83

ВСТУП

В сучасному цифровому віці, інтернет-сайти відіграють надзвичайно важливу роль у спілкуванні, розповсюдженні інформації та представленні організацій та бізнесів у світі онлайн. Зростання значення веб-присутності призвело до збільшення вимог до якості та ефективності веб-сайтів. У такому контексті використання системного аналізу стає ключовим інструментом для успішного створення та розвитку сайтів. Цей підхід дозволяє глибоко дослідити потреби користувачів, оптимізувати бізнес-процеси, забезпечити ефективну взаємодію та досягти поставлених цілей. Системний аналіз визначається як методологія дослідження, проектування та управління складними системами. У контексті створення сайтів, цей підхід орієнтований на розуміння веб-сайту як складної системи, що включає в себе елементи технологій, дизайну, інформації та взаємодії з користувачами. Використання системного аналізу дозволяє проникнути глибоко у сутність веб-сайту, виявити його ключові складові та зв'язки між ними. Це дозволяє розробникам створювати гармонійні та функціональні сайти, що задовольняють потреби користувачів та досягають бізнес-цілей організацій.

Актуальність застосування системного аналізу в створенні сайтів пояснюється рядом факторів. По-перше, зростає конкуренція в онлайн-середовищі, тому ефективність та якість сайту стають ключовими факторами успіху. Системний аналіз допомагає зрозуміти потреби та очікування користувачів, виявити проблеми та встановити оптимальні рішення. По-друге, змінюються вимоги та технології веб-розробки, що вимагає системного підходу до проектування та інтеграції компонентів сайту. Системний аналіз дозволяє вибрати найкращі технології та інструменти, а також забезпечити взаємодію між ними для досягнення максимальної продуктивності та якості.

Застосування системного аналізу у створенні сайтів є актуальним для будь-яких видів веб-проектів. Він допомагає виявити потреби різних зацікавлених сторін, встановити мету та цілі проекту, розробити оптимальну структуру сайту, забезпечити ефективну навігацію та взаємодію з користувачами, а також використовувати кращі практики у сфері безпеки та захисту даних. В результаті, веб-сайти, створені з використанням системного аналізу, мають вищу якість, ефективність та забезпечують успішну взаємодію з користувачами.

У сучасному світі, громадські організації мають унікальну можливість використовувати інтернет та веб-сайти як потужний інструмент для залучення нових членів, проведення консультацій, організації конференцій та поширення наукових робіт. Наявність веб-присутності для громадських організацій стає дедалі більш важливою, оскільки це дозволяє їм ефективно спілкуватися зі широким загалом, розповсюджувати інформацію, сприяти залученню нових учасників та забезпечувати взаємодію з зацікавленими сторонами.

Веб-сайт громадської організації створює обличчя організації в онлайн-середовищі, допомагає залучати нових членів, проводити консультації, організовувати події, а також поширювати наукові роботи. Він надає потенційним учасникам можливість ознайомитися з метою, цілями та завданнями організації, пропонуючи інформацію про членство та вигоди. Отримання відгуків і пропозицій від аудиторії забезпечується через різні онлайн-інструменти, такі як анкети або форуми. Інформація про події та можливості реєстрації доступна на сайті, роблячи процес участі в заходах простим і зручним. Крім того, сайт дозволяє організації демонструвати свої наукові досягнення, що допомагає забезпечити їх поширення та визнання у науковому співтоваристві.

Розділ 1 ОСНОВИ КЛІЄНТ-СЕРВЕРНИХ СИСТЕМ ДЛЯ ГРОМАДСЬКИХ ОРГАНІЗАЦІЙ

1.1 Розгляд концепції клієнт-серверної архітектури

Клієнт-серверна архітектура почала свій шлях у 1960-х роках, коли було виявлено необхідність розділення функцій обчислення та зберігання даних між різними комп'ютерами. У цій моделі, комп'ютер, що ініціює запит, виступає як клієнт, тоді як комп'ютер, що надає послугу, виступає як сервер. Початково, клієнт-серверні системи використовувалися для спільного використання ресурсів, таких як принтери, але згодом їх застосування поширилося на обробку даних та забезпечення доступу до великих обсягів інформації.

У 1980-х роках, зі зростанням обсягів даних та потреби у швидкому доступі до них, клієнт-серверна архітектура стала широко застосовуватись у сфері баз даних. Це дозволяло розподілити завдання між клієнтами та серверами, зберігати дані централізовано та забезпечувати керування та безпеку даних. З розвитком Інтернету та з'явленням веб-сайтів, клієнт-серверна архітектура стала ще більш поширеною. Вона дозволяла користувачам отримувати доступ до веб-сторінок (клієнт) з веб-серверів, що зберігають та надають ці сторінки (сервер). Це відкрило безліч можливостей для спілкування, комунікації та обміну інформацією в Інтернеті.

Клієнт-серверна архітектура є однією з основних архітектурних моделей у сфері комп'ютерних мереж та розподілених систем. У цій моделі взаємодія між комп'ютерними пристроями відбувається за принципом "клієнт-сервер", де клієнт (клієнтська програма або пристрій) запитує та отримує послуги, а сервер (серверна програма або пристрій) надає ці послуги.

Основні складові клієнт-серверної архітектури:

1. Клієнт: Це програма або пристрій, який ініціює запити до сервера та отримує відповіді. Клієнти можуть бути розташовані на різних комп'ютерах або пристроях, які підключені до мережі. Вони можуть виконувати різні функції, відображати інформацію, взаємодіяти з користувачем тощо.

2. Сервер: Це програма або пристрій, який надає послуги клієнтам. Він обробляє запити, здійснює необхідні обчислення та передає результати клієнтам. Сервер може мати більшу обчислювальну потужність, ресурси та доступ до даних, ніж клієнт.

3. Мережа: Клієнти та сервери підключені до мережі, яка забезпечує комунікацію між ними. Це може бути локальна мережа (LAN), широкосмугова мережа (WAN) або Інтернет.

Ключові принципи функціонування клієнт-серверної архітектури:

1. Розподіленість: Клієнти та сервери можуть бути розташовані на різних фізичних пристроях або в різних мережах. Це дозволяє розподілити завдання та навантаження між різними компонентами системи та забезпечити більшу масштабованість.

2. Запит-відповідь: Клієнт ініціює запит до сервера, який обробляє цей запит та повертає відповідь з результатами. Запит-відповідь є основним механізмом взаємодії між клієнтом та сервером.

3. Централізоване керування: Сервер зазвичай виконує централізоване керування ресурсами, даними та безпекою. Він встановлює правила доступу, контролює виконання запитів та забезпечує консистентність даних.

Скальованість: Клієнт-серверна архітектура дозволяє гнучко масштабувати систему шляхом додавання або видалення клієнтів та серверів. Це дозволяє відповідати зростаючим потребам та навантаженню.

Протоколи є основою для комунікації між клієнтами та серверами, вони визначають правила та процедури, які дозволяють передавати дані та

забезпечувати взаємодію між ними. Протокол HTTP використовується для обміну гіпертекстовими документами в Інтернеті і є одним з найбільш поширених протоколів у сфері веб-розробки.

Протокол HTTP заснований на моделі "запит-відповідь", де клієнт ініціює запит до сервера, а сервер відповідає на цей запит. Запити та відповіді передаються у вигляді повідомлень, які містять заголовки та необов'язкові дані. Основні елементи протоколу HTTP включають методи запитів, коди статусу, URL-адреси та версії протоколу. URL-адреси (Uniform Resource Locators) ідентифікують ресурси, з якими клієнт хоче взаємодіяти. Вони складаються з протоколу, доменного імені та шляху до ресурсу. Наприклад, "http://www.example.com/about" є URL-адресою для сторінки "about" на веб-сайті "www.example.com". Версії протоколу HTTP (наприклад, HTTP/1.1) вказують на специфікацію та можливості протоколу, які використовуються в комунікації між клієнтом та сервером. Кожна нова версія протоколу може включати поліпшення, додаткові функції та покращення продуктивності.

Протокол HTTP є ключовим елементом клієнт-серверної архітектури, зокрема для веб-додатків. Він визначає правила та процедури для ефективного обміну даними між клієнтами та серверами. Протокол HTTP є важливим інструментом для забезпечення доступу до інформації, комунікації та розробки веб-додатків у сучасному світі. Використання правильних методів запитів, кодів статусу, URL-адрес та версій протоколу HTTP допомагає покращити продуктивність та ефективність клієнт-серверної взаємодії.

Клієнт-серверна архітектура є важливим концептом у сфері розробки програмного забезпечення та розподілених систем. Вона дозволяє розділити функції та завдання між різними компонентами системи та забезпечує ефективну комунікацію та спілкування між ними.

1.2 Визначення основних компонентів клієнт-серверної системи

Клієнт і сервер є ключовими компонентами цієї архітектури, які взаємодіють між собою для забезпечення функціональності та обміну даними. Тож розглянемо їх більш детально:

Клієнт, також відомий як "клієнтська частина", ініціює комунікацію з сервером та здійснює запити на отримання даних або виконання операцій. Клієнт виступає як інтерфейс між користувачем та сервером, забезпечуючи зручну та ефективну взаємодію з системою. Однією з головних функцій клієнта є надсилання запитів до сервера. Ці запити можуть включати запити на отримання певних даних, створення нових записів, виконання операцій або зміну стану системи. Клієнт формує запити, включаючи необхідну інформацію та параметри, і надсилає їх до сервера за допомогою відповідного протоколу комунікації, такого як HTTP. Крім того, клієнт відповідає за обробку результатів, отриманих від сервера. Він може аналізувати отримані дані, відображати їх на екрані для користувача або виконувати додаткові дії на основі цих даних. Клієнт може також здійснювати локальне зберігання даних для покращення продуктивності та забезпечення офлайн-режиму роботи.

Для взаємодії з клієнт-серверними системами клієнти можуть використовувати:

1. Веб-браузери: Веб-браузери є одними з найпоширеніших типів клієнтських систем. Вони дозволяють користувачам отримувати доступ до веб-сторінок та взаємодіяти з веб-додатками через Інтернет. Веб-браузери, такі як Google Chrome, Mozilla Firefox, Microsoft Edge, Safari, надають інтерфейс для відображення веб-сторінок та виконання скриптів.

2. Десктопні програми: Десктопні програми, також відомі як стільничні додатки або нативні додатки, є програмними засобами, що встановлюються безпосередньо на локальний комп'ютер користувача.

Вони можуть надавати більшу функціональність та швидкодію, оскільки працюють безпосередньо на операційній системі комп'ютера. Приклади десктопних програм включають Microsoft Word, Figma, Discord, Adobe Photoshop, тощо.

3. Мобільні додатки: Мобільні додатки призначені для використання на мобільних пристроях, таких як смартфони або планшети. Вони розробляються під певну операційну систему, таку як iOS для Apple-пристроїв або Android для пристроїв, що працюють на базі Google. Мобільні додатки надають можливість користувачам взаємодіяти зі специфічними функціями та ресурсами пристрою, а також отримувати сповіщення та доступ до Інтернету. Приклади мобільних додатків включають Facebook, YouTube, Telegram, аплікація від компанії АТБ “Скануй купуй”, тощо.

4. Клієнтські додатки для IoT: За останні роки стали популярними клієнтські додатки для Інтернету речей (IoT). Ці додатки взаємодіють зі смарт-пристроями, датчиками або іншими пристроями, що підключені до Інтернету, і надають можливість керувати ними або отримувати дані з них. Приклади таких додатків включають додатки для керування освітленням вдома, термостатами, системами безпеки тощо.

Роль сервера в клієнт-серверній архітектурі є надважливою, оскільки він приймає запити від клієнтів, обробляє їх та повертає відповідь. Сервер відіграє ключову роль у наданні послуг, зберіганні та обробці даних, а також управлінні функціями системи. Основні функції сервера включають приймання запитів від клієнтів, обробку цих запитів та надання відповіді. Сервер взаємодіє з базою даних та іншими ресурсами, зберігає та обробляє дані, необхідні для системи. Він також відповідає за управління функціями системи, контроль доступу та забезпечення безпеки.

Сервер є централізованою точкою взаємодії між клієнтами та різними компонентами системи. Він виконує важливу роль у забезпеченні ефективної комунікації, обробці запитів та наданні результатів клієнтам.

Від правильної роботи сервера залежить продуктивність, надійність та функціональність клієнт-серверної системи.

Нижче розглянемо види серверів, що використовуються в клієнт-серверних системах:

1. Веб-сервери: Веб-сервери є основою для надання веб-сторінок та веб-додатків через Інтернет. Вони обробляють запити веб-браузерів клієнтів та надають їм необхідні файли (HTML, CSS, JavaScript, зображення тощо) для відображення сторінок у браузері користувача. Веб-сервери використовують такі протоколи, як HTTP і HTTPS, для комунікації з клієнтами.

2. Сервери баз даних: Сервери баз даних забезпечують зберігання, керування та доступ до даних. Вони дозволяють створювати, змінювати та видаляти дані, а також виконувати складні запити до бази даних. Деякі популярні сервери баз даних включають MySQL, PostgreSQL, Oracle, Microsoft SQL Server та MongoDB.

3. Сервери електронної пошти: Сервери електронної пошти (наприклад, Microsoft Exchange або Postfix) забезпечують обробку, надсилання та отримання електронних листів. Вони забезпечують комунікацію між різними поштовими клієнтами та надають засоби для управління поштовими скриньками, фільтрації спаму та інших функцій, пов'язаних з електронною поштою.

4. Файлові сервери: Файлові сервери надають централізований доступ до файлів та папок для користувачів у мережі. Вони дозволяють зберігати, керувати та ділитися файлами, забезпечують контроль доступу та можливість одночасного редагування файлів декількома користувачами. Деякі приклади файлових серверів включають Windows File Server, Network Attached Storage (NAS) або cloud-провайдери, такі як Dropbox, Google Drive та OneDrive.

5. Додаткові сервери: Крім вищезазначених типів, існують інші види серверів, що виконують специфічні функції для конкретних потреб.

Наприклад, сервери відео- або аудіо-потоків, сервери спільної роботи (collaboration), сервери віртуальної реальності (VR), сервери масштабованої обробки даних та багато інших.

Ці різноманітні види серверів виконують різні функції у клієнт-серверній архітектурі. Вони забезпечують надійне зберігання, обробку та передачу даних, комунікацію між клієнтами та іншими компонентами системи. Вибір певного типу сервера залежить від конкретних потреб проекту та вимог до системи.

1.3 Аналіз основних технологій та інструментів, які використовуються для розробки клієнт-серверних систем

Ефективна розробка клієнт-серверних систем є багатоаспектним процесом, який включає у себе вибір правильних технологій та інструментів. Кожна компонента системи, від клієнтського інтерфейсу до серверної логіки та бази даних, вимагає уважного підходу до вибору мов програмування, фреймворків, бібліотек та інших інструментів. Розуміння того, як ці технології взаємодіють та як вони можуть бути оптимально використані для вирішення конкретних завдань, є ключем до створення надійної, ефективної та масштабованої клієнт-серверної системи. Тому в даному розділі ми поглиблено розглядаємо різні технології, що застосовуються при розробці клієнт-серверних систем, їх особливості, переваги та недоліки.

Вибір правильного набору технологій може значно покращити продуктивність розробки, забезпечуючи легкість написання коду, швидкість виконання, безпеку даних та зручність користувачів. Зокрема, різні мови програмування та фреймворки надають різні можливості для створення серверної логіки, розробки інтерфейсу користувача, взаємодії з

базами даних та інтеграції з іншими сервісами. Також важливим елементом є вибір протоколів мережевої взаємодії, які дозволяють клієнту та серверу ефективно обмінюватися даними. Залежно від конкретних завдань та вимог, можуть бути вибрані різні протоколи, такі як HTTP, WebSocket, gRPC тощо.

Не менш важливим етапом є тестування та розгортання системи. Для цього також існують численні інструменти та підходи, від мануального тестування до автоматизованого.

Клієнтська частина веб-додатку, відома також як "front-end", створюється з метою взаємодії з користувачами і представлення даних, отриманих від сервера.

1. JavaScript є основною мовою програмування для розробки клієнтської частини веб-додатків. Він використовується для створення інтерактивних веб-сторінок, що можуть відгукуватися на дії користувача без перезавантаження сторінки.

2. React, Angular, та Vue.js є відомими JavaScript фреймворками, що надають структуру для розробки веб-додатків. Вони забезпечують набори інструментів і бібліотек, що спрощують розробку складних веб-додатків.

3. React, розроблений Facebook, дозволяє розробникам створювати великі веб-додатки, які можуть змінювати дані, не перезавантажуючи сторінку. Це дуже швидкий, гнучкий та простий у використанні JavaScript фреймворк.

4. Angular, випущений Google, це повноцінний фреймворк, який включає в себе шаблони декларативних шаблонів, залежний від ін'єкцій, кінцеву до кінця інструменти для тестування та інше.

5. Vue.js – це прогресивний JavaScript фреймворк, який зосереджений на декларативному рендерингу даних та складному складанні компонентів.

Серверна частина інформаційної системи, відома також як back-end, є ключовою складовою клієнт-серверної архітектури. Вона відповідає за

обробку запитів від клієнтської частини, взаємодію з базами даних, проведення необхідних обчислень, та генерацію відповідей до клієнта. Серверна частина складається з таких складових як:

1. Мов програмування, які відіграють важливу роль у розробці серверних додатків. Java є мовою високого рівня з сильними рисами безпеки, широко використовується для розробки надійних інтернет-додатків. Python є мовою із зрозумілим синтаксисом, що підтримує кілька парадигм програмування, включаючи об'єктно-орієнтовану, процедурну та функціональну. PHP є популярною мовою програмування, зорієнтованою на розробку веб-додатків.

2. Фреймворків, таких як Express.js, Django та Mod, які важливі для прискорення розробки серверної частини. Express.js є мінімалістичним і гнучким веб-фреймворком для Node.js, який надає набір корисних функцій для веб та мобільних додатків. Django є високорівневим Python веб-фреймворком, що спрощує розробку складних веб-додатків. Mod – це зручний фреймворк, що забезпечує високий рівень налаштування.

3. Систем управління контентом (Content Management Frameworks – CMF), таких як Modx і Tomcat, вони пропонують розробникам платформи для швидкого створення веб-додатків. Modx надає гнучкість і контроль над усіма аспектами веб-додатку. Tomcat, у свою чергу, є відкритим Java сервером, що реалізує специфікації Java Servlet, JavaServer Pages (JSP) та Java Expression Language (Java EL).

4. Баз даних, які є основою для зберігання, обробки та аналізу даних в інформаційних системах. SQL – це мова стандартного запиту, яка використовується для взаємодії з реляційними базами даних, тоді як MongoDB – це NoSQL система бази даних, що використовує гнучку схему даних і забезпечує високу продуктивність для великих даних.

Тестування та забезпечення якості коду є критично важливими етапами в процесі розробки клієнт-серверних систем. Вони допомагають забезпечити стабільність, надійність та продуктивність системи, а також

виявити та виправити помилки ще до впровадження системи.

1. Мануальне тестування є традиційним методом тестування, який полягає в перевірці роботи системи людиною. Це означає, що тестувальник вручну перевіряє функціональність системи, її інтерфейс, робить спроби викликати помилки або вади системи. Мануальне тестування є доречним для дослідницького тестування, тестування користувацького інтерфейсу та сценаріїв, які важко автоматизувати.

2. Автоматизоване тестування включає використання спеціальних інструментів і програм, які автоматично виконують певні тестові сценарії та аналізують результати. Це зазвичай швидше, ніж мануальне тестування, і може бути ефективно використано для регресійного тестування, випробування продуктивності, навантаження та стресу.

3. Unit тести фокусуються на перевірці ізольованих частин коду – модулів або функцій, для того, щоб впевнитися, що вони коректно виконують свої функції. Це важливий елемент підходу test-driven development (TDD).

4. Integration tests (інтеграційні тести) перевіряють взаємодію між різними модулями системи. Вони забезпечують, що різні частини системи можуть ефективно співпрацювати разом.

Розробка клієнт-серверних систем – це складний процес, який вимагає глибокого розуміння різних технологій та інструментів. Вибір правильних технологій може значно вплинути на продуктивність, надійність, безпеку та зручність використання кінцевого продукту. Тому глибокий аналіз доступних технологій та інструментів є критично важливим для успішної реалізації проекту.

У даному розділі ми розглянули основні технології та інструменти, які часто використовуються для розробки клієнт-серверних систем, включаючи різні мови програмування, фреймворки, системи керування базами даних та інструменти тестування. Кожен з цих інструментів має свої переваги та недоліки, тому важливо ретельно вивчити їх та зрозуміти, які з

них найкраще підходять для конкретного проекту. Використання правильних технологій та інструментів може значно спростити процес розробки, покращити якість продукту та знизити загальні витрати на проект. В подальшому дослідженні та практиці ми будемо використовувати ці знання для розробки оптимальної клієнт-серверної системи для громадської організації.

1.4 Роль та значення клієнт-серверних систем для громадських організацій

Громадська організація – це добровільне об'єднання громадян, засноване на спільних інтересах і цілях, які не мають на меті отримання прибутку. Вони є важливим елементом громадянського суспільства, оскільки вони займаються різноманітними сферами діяльності, включаючи, але не обмежуючись, благодійністю, освітою, охороною довкілля, здоров'ям, правами людини, соціальними послугами та багатьма іншими. Вони відіграють важливу роль в суспільстві, оскільки вони допомагають виявляти і розв'язувати різноманітні соціальні проблеми, захищати інтереси громадян та сприяти демократії. Громадські організації часто служать мостом між громадянами та урядом, надаючи канал для громадянського участі та впливу на політичні процеси.

Основними характеристиками громадських організацій є їхня некомерційність, добровільність, самоуправління, та відсутність членства в урядових структурах. Вони можуть включати все, від маленьких місцевих груп до великих міжнародних організацій. Діяльність громадських організацій відзначається високим рівнем залучення громадян та волонтерства.

Наукові громадські організації є підтипом громадських організацій,

які спеціалізуються на певному секторі науки, технології або дослідженнях. Їхня діяльність може бути широкою і включати підтримку та проведення наукових досліджень, популяризацію науки, розвиток наукової політики, надання грантів та стипендій для науковців, організацію наукових конференцій та інших заходів. Подібні громадські організації можуть відігравати важливу роль в науковому співтоваристві та суспільстві в цілому. Вони можуть сприяти розвитку наукових досліджень, забезпечуючи фінансування та ресурси, створювати мережі співпраці між науковцями та іншими зацікавленими сторонами, підвищувати обізнаність суспільства про науку і її значення, і допомагати формувати наукову політику та прийняття рішень.

Клієнт-серверні системи, такі як веб-сайти та онлайн-платформи, є особливо важливими для наукових громадських організацій. Вони можуть використовувати ці системи для публікації наукових досліджень, спілкування з членами, організації конференцій та воркшопів, залучення волонтерів та фінансування, і для розробки віртуальних мереж співпраці.

Наукові громадські організації, відповідаючи на виклики сучасної ери інформації, все більше використовують клієнто-орієнтовані системи. Ці системи відіграють роль потужного інструменту для поширення інформації, збору даних, взаємодії з аудиторією та забезпечення оперативного та ефективного управління ресурсами організації.

Перш за все, важливість клієнто-орієнтованих систем для наукових громадських організацій полягає в їхній здатності до швидкого та ефективного поширення важливої інформації. Вони дозволяють організаціям демонструвати свої наукові дослідження, публікації, результати проектів та інші релевантні матеріали, забезпечуючи їх доступність для широкої аудиторії.

Другим важливим аспектом є можливість збору даних. Клієнто-орієнтовані системи дозволяють організаціям використовувати різні методи збору даних, включаючи онлайн-опитування, форми зворотного

зв'язку, аналіз відвідуваності та інтерактивності користувачів. Ці дані є невід'ємною частиною наукового дослідження, а також дозволяють організаціям краще зрозуміти свою аудиторію, покращити комунікаційні стратегії та оптимізувати ресурси.

Також, використання клієнто-орієнтованих систем дозволяє науковим громадським організаціям встановлювати зв'язок з аудиторією. Чи то за допомогою форм зворотного зв'язку, форумів, соціальних медіа, або інших засобів взаємодії, організації отримують можливість взаємодіяти з відвідувачами, враховувати їхні потреби та відповіді, а також створювати спільноту зацікавлених сторін.

Нарешті, використання клієнто-орієнтованих систем є ключовим для ефективного управління ресурсами організації. Вони дозволяють автоматизувати багато процесів, від реєстрації учасників на конференції до збору та аналізу даних для наукових досліджень. Автоматизація звільняє ресурси для інших важливих задач та сприяє більшій продуктивності.

Аналізуючи роль та значення клієнт-серверних систем в діяльності наукових громадських організацій, необхідно розглянути конкретні приклади їхнього застосування, що надасть нам більш детального та всеосяжного розуміння цієї теми.

- Платформи для співпраці: Деякі наукові громадські організації використовують клієнт-серверні системи для створення платформ, що сприяють колаборації та обміну знаннями. Прикладом може бути веб-сайт організації, на якому вчені, студенти та інші зацікавлені сторони можуть обмінюватися ідеями, публікувати свої роботи та отримувати зворотний зв'язок від колег. Системи такого роду важливі для розвитку наукової спільноти та підтримки інновацій.

- Реєстрація та управління заходами: Для організацій, що регулярно проводять конференції, воркшопи або семінари, клієнт-серверні системи можуть бути використані для управління реєстрацією учасників, плануванням заходів та наданням інформації учасникам. Це значно

полегшує процеси планування та управління, а також забезпечує учасникам зручний доступ до необхідної інформації.

- Дистанційне навчання: З появою пандемії COVID-19 дистанційне навчання стало нормою для багатьох освітніх установ та організацій. А конкретно для нашої країни повномасштабна війна з Росією остаточно закріпило дистанційне навчання у багатьох населених пунктів і їх установ. Використання клієнт-серверних систем дозволило організаціям перевести свої курси в онлайн-формат, забезпечивши навчання безперервним незалежно від обставин. Студенти та дослідники мають можливість вчитися, спілкуватися та виконувати наукові дослідження, не виходячи з дому.

- Аналіз даних та наукова візуалізація: Деякі наукові громадські організації використовують клієнт-серверні системи для аналізу великих обсягів даних та їх візуалізації. Наприклад, системи можуть автоматично обробляти дані з експериментів, алгоритми машинного навчання можуть використовуватися для виявлення шаблонів або зв'язків в даних. Результати потім можуть бути візуалізовані за допомогою інтерактивних діаграм, що дозволяє вченим більш ефективно та наочно представляти свої дослідження.

Після глибокого аналізу можемо висловити впевненість у тому, що клієнт-серверні системи стали не лише важливим інструментом сучасності, але й мають значний потенціал впливати на майбутнє наукових громадських організацій. Використання цих систем може стати платформою для більш ефективного поширення знань та інформації, яке прямує від індивідуальних дослідників до всього наукового співтовариства. Це прискорює процес наукового відкриття, оскільки дослідники мають можливість швидко ділитися новими даними, отримувати зворотний зв'язок та підтримувати діалог із колегами по всьому світу. Більше того, використання клієнт-серверних систем дозволяє громадським організаціям більше залучати громадськість. Демократизація знань, яку підтримують ці

системи, означає, що більше людей може мати доступ до наукових досліджень і, відповідно, може брати участь у наукових дискусіях та прийнятті рішень. Це може привести до більшого громадського розуміння науки та вищого рівня довіри до наукової спільноти. Крім того, клієнт-серверні системи відкривають нові можливості для глобалізації наукових громадських організацій. З допомогою цих систем, організації можуть здійснювати свою діяльність без географічних обмежень, співпрацюючи з дослідниками, волонтерами та донорами з усього світу. Це дозволяє їм збільшувати свій вплив та підтримувати більш широкий діапазон досліджень.

Отже, з урахуванням вищезазначеного, ми можемо стверджувати, що ефективне використання клієнт-серверних систем може суттєво вплинути на розвиток і еволюцію наукових громадських організацій, надавши їм більше можливостей для розвитку науки та зміцнення їхнього впливу на суспільство.

РОЗДІЛ 2 АНАЛІЗ ПОТРЕБ ТА СИСТЕМ В ГРОМАДСЬКИХ ОРГАНІЗАЦІЯХ

2.1 Аналіз потреб громадської організації

Сучасний світ неперервно еволюціонує, що нерідко призводить до змін у соціальному та політичному ландшафті, в тому числі й у структурі громадських організацій. Стійкість та продуктивність цих організацій взаємопов'язані з їх здатністю відповідати на виклики, адаптуватися до нових обставин та забезпечувати своєму цільовому аудиторією потрібні послуги.

Аналіз потреб відіграє центральну роль у плануванні та управлінні громадськими організаціями. Він служить інструментом для ідентифікації та оцінки обставин, в яких працює організація, її ставлення до основних стейкхолдерів та спільнот, з якими вона взаємодіє. Це допомагає організаціям розуміти, чого вони хочуть досягти, які ресурси вони мають для досягнення своїх цілей, та як вони можуть використовувати ці ресурси найефективніше. Без аналізу потреб громадські організації можуть ризикувати витратою часу, ресурсів та енергії на розвиток програм та проектів, які не задовольняють реальні потреби їх цільових аудиторій. Крім того, невдалий аналіз потреб може призвести до непорозумінь зі стейкхолдерами та недосягнення запланованих цілей.

Наукова громадська організація повинна ретельно вивчати та аналізувати потреби своєї аудиторії та спільноти. Це означає розуміння їхніх цілей, бажань, інтересів, проблем і ставлення до різних питань, що мають відношення до області діяльності організації. Важливо відзначити, що потреби можуть суттєво відрізнятися в залежності від конкретних обставин, в яких знаходиться цільова аудиторія.

Для візуального зрозуміння, як аналіз потреб працює на практиці, розглянемо декілька прикладів, специфічних для наукових громадських

організацій.

- "Асоціація Фізиків": Ця організація, яка об'єднує фізиків з різних областей, може використовувати аналіз потреб для з'ясування, які області науки їх члени вважають найбільш пріоритетними для досліджень. Для цього можуть бути використані онлайн-опитування, обговорення на наукових конференціях, а також аналіз актуальних наукових тенденцій у світовій спільноті фізиків.

- "Союз Хіміків": Наукова громадська організація, що зосереджена на хімії, може провести аналіз потреб, щоб визначити, які освітні ресурси та матеріали є найбільш корисними та потрібними для їхніх членів. Це може бути зроблено через аналіз відгуків про навчальні матеріали, що вже використовуються, а також через обговорення з членами організації про їх поточні потреби в області навчання та розвитку.

- "Біологічний Союз": Організація, що працює в галузі біології, може провести аналіз потреб для визначення, які області біологічних досліджень потребують більше уваги та фінансування. Засновуючись на результатах аналізу, організація може вирішити сконцентрувати свої ресурси на підтримці конкретних проектів або напрямків досліджень.

Всі ці приклади ілюструють, як наукові громадські організації можуть ефективно використовувати аналіз потреб для кращого визначення своїх цілей, планування діяльності та прийняття рішень. Однак важливо зазначити, що аналіз потреб - це процес, який вимагає часу, ресурсів та відповідних навичок.

Аналіз потреб включає у себе ряд послідовних етапів, починаючи з ідентифікації ключових проблем і викликів, з якими стикається громадська організація "Системні дослідження", а саме:

Публікація наукових робіт, проектів та грантів є важливою частиною діяльності громадської організації "Системні дослідження" з різних причин:

- Наукова відкритість і доступність: Публікація своїх наукових

робіт забезпечує відкритий доступ до набутих знань. Відповідно до даних Європейської Комісії, відкритий доступ може прискорити науковий прогрес, покращивши розповсюдження та доступність наукових досліджень [1].

- Залучення нових членів та партнерів: Публікації можуть слугувати рекламою діяльності організації. Згідно з дослідженнями Science-Metrix, більш ніж 60% наукових статей, опублікованих у 2016 році, були доступні в відкритому доступі, що вказує на зростання цієї практики. Відповідно, публікації можуть привернути увагу потенційних членів та партнерів.

- Фінансування та гранти: Публікація результатів грантів та проектів може підтвердити ефективність використання ресурсів, що може сприяти отриманню подальшого фінансування. Крім того, деякі грантоотримувачі вимагають публікації результатів досліджень.

- Співпраця і взаємодія: Публікації створюють платформу для спілкування та обміну думками з іншими дослідниками та організаціями. Вони сприяють співпраці та можуть викликати ідеї для нових проектів або досліджень.

- Поліпшення репутації: Видання наукових робіт та результатів досліджень також відіграє значну роль у формуванні репутації організації. Це демонструє професіоналізм, серйозність та відданість науковій діяльності, що в свою чергу може підвищити довіру спільноти та потенційних спонсорів. Статистика показує, що організації, які публікують свої дослідження, мають більшу вірогідність отримати фінансування та підтримку [2, 3].

- Вплив на політику та практику: Нарешті, публікації можуть впливати на політику та практику в області, де працює організація. Наприклад, публікації, що просувають нові ідеї або дослідження, можуть викликати зміни в політиці або практиці [4].

Залучення нових членів в громадську організацію "Системні

дослідження" є важливим аспектом ефективного виконання місії організації та забезпечення її стійкого розвитку. Використання дієвих стратегій залучення нових членів допомагає організації забезпечити її зростання, розширити спектр впливу та підвищити ефективність виконання її завдань.

- По-перше, нові члени організації збагачують її свіжими ідеями, новими перспективами та додатковими навиками та знаннями, які можуть доповнити існуючий пул компетенцій. Це важливо для організації, що займається науковими дослідженнями, оскільки інновації та креативність є ключовими факторами в успішності наукових зусиль. Згідно зі статистикою, у галузі науки та техніки, зокрема, робота в команді з різномірним складом може значно підвищити продуктивність та інноваційність (Hong & Page, 2004) [5].

- По-друге, нові члени можуть розширити мережу зв'язків організації, що сприяє її видимості, авторитету та можливостям для співпраці. Це особливо важливо у контексті набуття фінансування, співпраці з іншими дослідницькими установами та взаємодії з громадськістю [6].

- По-третє, зростаюча база членів також забезпечує більшу силу та вплив організації, що дозволяє їй ефективніше впливати на політику, формувати громадську думку та досягати своїх стратегічних цілей.

Однак, варто пам'ятати, що залучення нових членів – це не лише про збільшення кількості. Ключове значення має якість нових членів, їх внесок у роботу організації та їх відповідність цінностям та місії організації [7].

Конференції та консультації відіграють важливу роль у житті багатьох громадських організацій, включаючи "Системні дослідження". Вони забезпечують платформу для обміну знаннями, обговорення та вироблення стратегій та підтримують створення та розширення професійних мереж.

Конференції можуть бути особливо корисними для громадських

організацій, оскільки вони збирають в одному місці дослідників, практиків, політиків та інших зацікавлених сторін. За даними статистики, 81% учасників конференцій вважають, що вони важливі для виконання їхньої роботи (Bennett, 2014) [8]. Це може пояснюватися тим, що на конференціях можна отримати останні дані, установити нові контакти та зрозуміти головні тренди у галузі.

Консультації, з іншого боку, важливі для підтримки відносин зі стейкхолдерами, зокрема з цільовими групами, донорами, партнерами та волонтерами. Вони дозволяють громадським організаціям зрозуміти потреби та очікування цих груп, отримати відгук на свою діяльність та визначити, як вони можуть її покращити. За даними дослідження, проведеного Bordenave (2013) [9], організації, що проводять регулярні консультації зі своїми стейкхолдерами, мають вищий рівень задоволення їхніми послугами та вищий рівень внутрішньої мотивації своїх співробітників.

2.2 Визначення вимог до системи

В рамках даної системи існує необхідність визначити функціональні вимоги, які забезпечать реалізацію таких основних функцій, як контактна форма для відправки даних на електронну пошту громадської організації, публікація наукових робіт, грантів та проектів, проведення консультацій та конференцій, а також публікація новин організації.

1. Контактна форма:

- Система повинна містити контактну форму, яка дозволяє користувачеві ввести свої дані.
- Форма повинна містити поля для введення інформації, такої як ім'я, електронна адреса, телефон.

- Після заповнення форми, система повинна надсилати введені дані на електронну пошту громадської організації.

- Після успішної відправки форми, користувач повинен отримати підтвердження про надіслання його даних.

2. Публікація наукових робіт, грантів та проектів:

- Система повинна мати можливість публікації наукових робіт, грантів та проектів на веб-сайті організації.

- Адміністратор системи повинен мати можливість додавати, редагувати та видаляти наукові роботи, гранти та проекти.

- Користувачі повинні мати можливість завантажити публікації наукових робіт, грантів та проектів з веб-сайту організації, для ознайомлення з ними.

3. Проведення консультацій та конференцій:

- Система повинна мати можливість організації консультацій та конференцій.

- Користувачі повинні мати можливість реєстрації на консультації та конференції через веб-сайт організації.

- Система повинна надсилати підтвердження реєстрації на електронну пошту користувача.

- Адміністратор системи повинен мати можливість, переглядати та змінювати інформацію про консультації та конференції.

4. Публікація новин організації:

- Система повинна мати можливість публікації новин організації на веб-сайті.

- Адміністратор системи повинен мати можливість додавати, редагувати та видаляти новини.

- Користувачі повинні мати можливість переглядати новини на веб-сайті організації.

- Система повинна підтримувати можливість прикріплення зображень, відео або інших мультимедійних матеріалів до новин.

В рамках системи, крім функціональних вимог, існують нефункціональні вимоги, які є ключовими для забезпечення ефективності, продуктивності та надійності. Нижче наведено деякі нефункціональні вимоги до системи:

1. Надійність:

- Система повинна бути стабільною та надійною, здатною працювати без відмов і витримувати велику кількість одночасних запитів.
- Система повинна мати механізми відновлення та резервного копіювання даних, щоб уникнути втрати інформації.

2. Масштабованість:

- Система повинна бути масштабованою, здатною ефективно працювати зі зростанням обсягу даних та кількості користувачів.
- Система повинна мати гнучкість для легкої розширюваності та інтеграції з іншими системами.

3. Безпека:

- Система повинна забезпечувати високий рівень безпеки для захисту конфіденційної інформації користувачів.

4. Час відклику:

- Система повинна мати швидкий час відклику, забезпечуючи швидку обробку запитів користувачів.
- Система повинна бути оптимізованою для ефективного використання ресурсів та мінімізації затримок.

Аналіз вимог різних стейкхолдерів організації "Системні дослідження" є важливим кроком у розробці системи, оскільки дозволяє врахувати потреби та очікування різних груп користувачів. Основні стейкхолдери організації включають членів організації, відвідувачів веб-сайту та науковців. Нижче наведено деякі вимоги, що вони можуть мати:

1. Члени організації:

- Можливість отримання інформації про активності та новини організації.

- Можливість подати заявку на вступ до організації через веб-сайт.
- Можливість взаємодії з іншими членами організації шляхом конференцій та консультацій.

2. Відвідувачі веб-сайту:

- Зручний та легкий використовувати інтерфейс веб-сайту.
- Можливість швидкого пошуку та доступу до інформації про наукові роботи, гранти та проекти.
- Можливість заповнити контактну форму для зв'язку з організацією та отримання відповіді.

- Можливість записатись на консультацію чи конференцію.

- Наявність розділу з новинами та оновленнями організації.

3. Науковці:

- Можливість публікації своїх наукових робіт та досліджень на веб-сайті організації.

- Можливість отримання інформації про гранти, конференції та інші наукові заходи.

- Можливість отримати консультацію з експертами організації або інших науковців.

2.3 Аналіз наявних аналогів та їх порівняльний аналіз

У цьому розділі ми проведемо аналіз двох веб-сайтів громадських організацій. Зокрема, наша увага буде зосереджена на дизайні цих веб-сайтів і швидкості їхнього завантаження. Дизайн включає в себе естетичну привабливість, зручність навігації, оригінальність, а також наскільки він відповідає цілям та місії організації.

Швидкість завантаження сайту важлива для забезпечення хорошого користувацького досвіду. Чим швидше сайт завантажується, тим вище

ймовірність того, що користувачи залишаться на ньому. Велика кількість досліджень показує, що затримка в завантаженні веб-сайту може призвести до зниження задоволеності користувача, зменшення кількості переглядів сторінок і, в кінцевому рахунку, до втрати потенційних клієнтів або членів організації.

Для аналізу швидкості завантаження ми скористаємось такими інструментами, як Google PageSpeed Insights, GTmetrix, які дозволяють оцінити швидкість завантаження веб-сайту і надають рекомендації щодо його оптимізації.

Хоча ці інструменти дають важливу інформацію про швидкість завантаження веб-сайту, вони мають лише допоміжний характер у загальному контексті оцінки ефективності веб-сайту. Хороші показники швидкості завантаження від цих інструментів зазвичай включають час завантаження сторінки менше 2–3 секунд. Ці значення вважаються прийнятними для сучасних веб-сайтів і можуть бути гарним початковим показником ефективності веб-сайту.

Проте, слід пам'ятати, що основою успіху веб-сайту є дизайн користувацького інтерфейсу (User Interface, UI) і досвід користувача (User Experience, UX). Дизайн користувацького інтерфейсу включає в себе не лише естетику, але й структуру та навігацію сайту, які повинні бути інтуїтивно зрозумілими і зручними для користувачів. UX, у свою чергу, включає в себе широкий спектр взаємодій користувача з продуктом або службою, включаючи емоційні та психологічні аспекти.

Ефективність веб-сайту в значній мірі залежить від того, наскільки вдало реалізовані UI та UX. Якщо користувачі можуть легко знайти потрібну їм інформацію, зручно навігуватися по сайту і отримують позитивні емоції від взаємодії з сайтом, це може значно збільшити задоволеність користувачів, сприяти їхньому залученню та збільшити їхню лояльність до організації.

Наша мета — виявити найефективніші практики та потенційні

можливості для поліпшення, які можна використати при створенні сайту для громадської організації "Системні дослідження".

1. Веб-сайт громадської організації "Соборність":

<http://conference-ukraine.com.ua/ua/training/>

QR-code з посиланням (Рисунок 2.1):



Рисунок 2.1 – QR-code з посиланням на веб-сайт громадської організації «Соборність»

Трохи про організацію:

Громадська організація "Соборність" – це місцева, незалежна, неприбуткова та некомерційна структура, створена на принципах добровільності та спільності інтересів, світоглядних позицій та моральних якостей. Основна мета організації – сприяння розвитку науки, освіти та культури, що веде до збагачення духовного потенціалу світу та відродження моральних засад суспільства. Соборність займається розширенням громадської участі в різноманітних освітніх, наукових, культурних програмах, здійснює підтримку обміну досвідом та знаннями у сферах науки, освіти та культури, сприяє розвитку видавничої справи та інформаційної інфраструктури. Організація також докладає зусиль до формування ідеологічних засад патріотизму та національної самосвідомості, залучає науковців до своєї діяльності, пропагує пріоритет духовних, моральних засад у житті суспільства та бере участь у розробці та реалізації регіональних програм, спрямованих на відродження духовності, культури, науки.

Аналіз дизайну сайту:

1. Сайт не має безпечного підключення.

При завантаженні сторінки даного сайту, ми помічаємо відсутність безпечного підключення, оскільки веб-адреса починається з "http" замість "https". Це вказує на відсутність шифрування та захисту передачі даних між користувачем і сервером.

Важливість безпечного підключення на веб-сайтах не може бути недооцінена. Шифрування даних за допомогою протоколу HTTPS гарантує конфіденційність та цілісність передаваних даних. Відсутність безпечного підключення може впливати на сприйняття користувачів, створюючи сумніви щодо безпеки їхньої особистої інформації та довіреності самого сайту. Користувачі, які приділяють увагу кібербезпеці, можуть вважати веб-сайти без безпечного підключення ризикованими та небезпечними для передачі чутливої інформації. Відсутність зеленого замку або індикації безпечного з'єднання в браузері може викликати в користувачів відчуття незахищеності та обурення. Це може мати негативний вплив на їхню готовність спілкуватися з організацією чи виконувати операції на цьому сайті.

Крім того, відсутність безпечного підключення може відкривати можливості для різних кібератак. Наприклад, атаки "людина посередник" (man-in-the-middle) можуть дозволити зловмисникам перехоплювати та змінювати дані, які передаються між клієнтом і сервером. Це може включати чутливі дані, такі як паролі, фінансові відомості та іншу особисту інформацію.

Отже, відсутність безпечного підключення в даному веб-сайті має певні негативні наслідки. Вона може підірвати довіру користувачів до сайту, викликати сумніви щодо безпеки їхньої особистої інформації та створити потенційні ризики для їхньої безпеки. Рекомендується впровадження безпечного підключення на веб-сайтах, щоб забезпечити конфіденційність, цілісність та довіреність передаваних даних, а також підвищити безпеку та сприйняття сайту користувачами.

2. Недостатня структурованість головної сторінки

На головній сторінці вказаного веб-сайту замало інформації для користувача, який вперше відвідує сайт. Це може призвести до того, що користувачі можуть відчувати збентеження або невпевненість, що у свою чергу призведе до втрати інтересу і відмови від подальшого використання сайту. Це особливо важливо для громадських організацій, де головна мета полягає у тому, щоб залучити та утримати відвідувачів, надаючи їм важливу інформацію та заохочуючи до участі в їх ініціативах.

Одним із найвдаліших рішень було б розподілити весь важливий контент на головній сторінці, з посиланнями на додаткові сторінки, де користувачі можуть отримати більше інформації. Наприклад, на головній сторінці можна було б розмістити інформацію про організацію, її місію та цінності, а також надати посилання на детальніші сторінки з інформацією про партнерів, заходи, отримані сертифікати та програми стажувань.

Це не лише допомогло б розмістити більше контенту на головній сторінці, але й зробило б її менш завантаженою та більш структурованою, що сприяло б кращому навігуванню користувачів. Такий підхід до дизайну користувацького інтерфейсу може в значній мірі поліпшити користувацький досвід, збільшити задоволеність від використання сайту та збільшити кількість повернень користувачів на сайт.

В різних дослідженнях показано, що вміст та організація головної сторінки веб-сайту мають важливе значення для відвідувачів. За даними дослідження Google, 48% користувачів вважають дизайн головної сторінки за ключовий фактор, який впливає на їхнє сприйняття якості сайту (Google, 2020) [10]. Окрім того, дослідження UXCam з'ясувало, що 86% користувачів очікує отримати інформацію про компанію і її пропозиції вже на головній сторінці (UXCam, 2021) [11]. В іншому дослідженні, проведеному Nielsen Norman Group, встановлено, що користувацький досвід на веб-сайті зростає на 124%, коли веб-сайт має добре структуровану навігацію та логічно розміщені секції з інформацією (Nielsen Norman Group,

2022) [12].

Ці дослідження підкреслюють важливість правильної організації та представлення контенту на головній сторінці веб-сайту громадської організації.

3. Не до кінця оформлений "header" веб-сайту

У дизайні веб-сайтів одним з найважливіших аспектів є належне використання "header" або верхнього колонтитула. Оформлення хедера на сайті здійснює критичну функцію – воно впливає на перше враження користувача про сайт та мотивує його до подальшого перегляду вмісту. Відповідно, невдале оформлення хедера може призвести до небажаного результату, у тому числі до втрати потенційних відвідувачів сайту.

Веб-сайт, що розглядається, не має в хедері елемента "call to action" (СТА). СТА – це критичний компонент веб-дизайну, який стимулює користувача до певної дії, як-от реєстрації, завантаження матеріалу, подачі заявки тощо. Наявність чіткого та виділеного СТА може значно підвищити конверсію на сайті, збільшуючи кількість користувачів, які виконують бажану дію.

Також відзначимо, що простір в хедері на даному сайті використовується не економно. У веб-дизайні кожен піксель на екрані важливий, особливо в області, яка завжди видима користувачам. Багато вільного простору в хедері може створити враження порожнечі або незавершеності дизайну. Кращим рішенням було б структурувати цей простір, розмістивши там додаткову інформацію, меню або СТА.

Є численні дослідження, які підтверджують важливість СТА та ефективного використання простору в хедері. Згідно з дослідженням компанії Small Business Trends, більше 70% малих бізнес-веб-сайтів не включають в свою домашню сторінку жодного СТА [13]. Це є серйозною помилкою, адже дослідження WordStream виявило, що СТА на домашній сторінці може збільшити конверсію на 80% [14].

Крім того, дослідження UXPin показало, що використання більшого

білого простору навколо елементів може покращити увагу користувача на 20%. Однак це не означає, що велика кількість незаповненого простору в хедері буде ефективною. Більше того, "пустий" простір може бути використаний для збільшення розуміння користувачем вмісту та покращення загального враження від сайту [15].

Отже, оформлення хедера вимагає обміркованого підходу, який бере до уваги потреби користувача та сприяє досягненню цілей сайту.

4. Відсутність розділу “Новини організації”

Розділ новин є основним каналом комунікації між організацією та її аудиторією. Він служить місцем для розповсюдження актуальних та важливих повідомлень, включаючи оголошення про майбутні заходи, зміни в політиці організації, досягнення та успіхи, нові проекти та ініціативи, а також інформацію про важливі теми та тренди в галузі.

Відсутність розділу новин може мати негативний вплив на відносини з аудиторією та загальний імідж організації. Це може створити враження недостатньої прозорості та відкритості, що може погіршити довіру аудиторії до організації. Більше того, це може обмежити можливості організації взаємодіяти з аудиторією, збирати зворотній зв'язок та сприяти взаємозрозумінню.

Крім того, відсутність розділу новин може позбавити організацію можливості використовувати вміст для підвищення видимості в пошукових системах. Постійно оновлюваний вміст є важливим фактором ранжування для пошукових систем, і регулярні новини можуть допомогти збільшити органічний трафік на веб-сайт.

Перейдемо до швидкості завантаження.

1. Посилання на Google PageSpeed Insights (Рисунок 2.2):

https://pagespeed.web.dev/analysis/http-conference-ukraine-com-ua-ua-training/s4w2fe7w9n?form_factor=desktop



Рисунок 2.2 – Посилання на Google PageSpeed Insights веб-сайту громадської організації «Соборність»

2. Посилання на Gtmetrix (Рисунок 2.3):

<https://gtmetrix.com/reports/conference-ukraine.com.ua/ErY3ZmWy/>



Рисунок 2.3 – Посилання на Gtmetrix веб-сайту громадської організації «Соборність»

Як показують дослідження, затримка відвантаження веб-сайту на одну секунду може знизити перегляди сторінки на 11%, призвести до 7% втрати конверсій і 16% зниження задоволення користувачів. Все це робить швидкість завантаження ключовим параметром при оцінці ефективності веб-сайту [16].

Проведена оцінка сайту громадської організації "Соборність" за допомогою Google PageSpeed Insights та GTmetrix показала високі результати. Це може бути пояснено простотою його верстки та відсутністю об'ємного контенту, такого як великі графічні файли або відео, що можуть

сповільнити завантаження сторінки.

Хоча простота верстки та відсутність "важкого" контенту, безумовно, сприяють швидкості завантаження, вони також можуть вказувати на відсутність інтерактивних або динамічних елементів, що можуть зробити сайт більш привабливим та корисним для користувачів. Примітивна верстка може обмежити можливості сайту викликати емоційну відповідь у відвідувачів або пропонувати персоналізований користувацький досвід.

Незважаючи на високу швидкість завантаження, важливо пам'ятати, що успішний веб-сайт потребує більшої стратегії, яка включає в себе як технічні аспекти, так і зміст та дизайн. Швидкість завантаження слід розглядати в контексті більшої картини, яка включає в себе також якість контенту, зручність навігації, доступність та зрозумілість для користувачів різних категорій.

Висновки стосовно сайту громадської організації “Соборність”

Наводячи підсумки аналізу веб-сайту громадської організації "Системні дослідження", можна відмітити, що незважаючи на відмінні показники швидкості завантаження, що можуть сприяти позитивному користувацькому досвіду, наявність ряду критичних недоліків ставить під загрозу його здатність ефективно виконувати свою місію.

Серед найбільш очевидних проблем – обмежений зміст на головній сторінці. Сайт не надає достатньо інформації для користувачів, що прийшли з цікавістю або потребою дізнатись більше про діяльність організації. Інформація про активності, місію та цінності організації, проекти та події, партнерство – всі ці важливі аспекти повинні бути ясно видимі та доступні для користувачів прямо з головної сторінки.

Структура навігації також вимагає поліпшення. Відсутність СТА в хедері сайту може спричинити пропуск можливостей для залучення користувачів. Відповідно оформлений СТА може стимулювати відвідувачів до певної дії, що буде відповідати цілям організації.

Крім того, хедер має схожу проблему з використанням простору.

Наразі він використовується не ефективно, і в ньому є велика кількість вільного простору. Більше інформації, а також розуміння того, як краще структурувати цей простір, може покращити користувацький досвід та ефективність сайту.

Відсутність розділу з новинами є ще одним серйозним недоліком. Новини можуть виступати як посилання на важливі події та зміни в організації, а також допомагають підтримувати зв'язок з користувачами.

В цілому, хоча сайт має високу швидкість завантаження, його потенціал щодо забезпечення інформації та залучення користувачів є обмеженим. Розробка більш структурованого та змістовного дизайну може значно покращити його ефективність.

2. Веб-сайт громадської організації “Громадська Рада освітян і науковців України”:

<https://gronu.org.ua/>

QR-code з посиланням (Рисунок 2.4):



Рисунок 2.4 – QR-code з посиланням на веб-сайт громадської організації «Громадська Рада освітян і науковців України»

Трохи про організацію:

Всеукраїнська громадська організація "Громадська Рада освітян і науковців України" (ГРОНУ), заснована 2005 року, призвана підвищити авторитет педагогів і вчених, акцентувати роль освіти і науки в розвитку суспільства, та захищати інтереси своїх членів. ГРОНУ захищає соціальні та законодавчі права освітян та науковців, сприяє підвищенню їх

матеріального становища, підтримує молодих спеціалістів та талановитих студентів. Організація працює над демократизацією та гуманізацією освіти, впровадженням інновацій, вдосконаленням педагогічної освіти та впливом на якість освіти. ГРОНУ також займається громадським моніторингом за дотриманням моралі та законодавства в освіті та науці.

Аналіз дизайну сайту:

1. Недолік у структуризації головної сторінки.

Розглядаючи структуру головної сторінки цього веб-сайту громадської організації, можна помітити що вона страждає від значного недостатку інформації. Дана проблема є досить критичною, адже головна сторінка, фактично, служить візитівкою організації в онлайн-середовищі. Це місце, де потенційні відвідувачі, донори або партнери можуть швидко отримати загальне уявлення про місію, цілі та діяльність організації.

На даному сайті єдиний видимий контент на головній сторінці - це розділ "Про нас". Це, безумовно, важливий розділ, але для більшої ефективності потрібно включати більше різноманітного та важливого контенту прямо на головну сторінку. Такий контент може включати інформацію про актуальні проекти, найближчі події, новини організації та інше.

Одним з найбільш вдалих рішень було б розподілити весь важливий контент на головній сторінці, з посиланнями на детальніші сторінки, що містять більше інформації. Таким чином, користувачам надається можливість отримати загальний огляд, а потім зануритися глибше у теми, які їх найбільше цікавлять.

Варто відмітити, що на цьому сайті є посилання на останні новини на кожній сторінці, що є відмінною функцією. Такий вбудований модуль новин допомагає утримувати відвідувачів в курсі останніх подій та розвитку організації.

Однак, загалом, головна сторінка цього веб-сайту не реалізує свій повний потенціал. Більший обсяг контенту, правильно структурований та

презентований, значно покращив би користувацький досвід та загальну ефективність сайту. Це, в свою чергу, може сприяти збільшенню залучення користувачів, конверсій та, в кінцевому рахунку, підтримки громадської організації.

2. Відсутність “мети” сайту

Під час оцінки веб-сайту громадської організації одним з ключових аспектів, на який слід звернути увагу, є його цільове призначення. Що веб-сайт намагається досягти? Яку роль він відіграє у стратегії організації? Ці питання є важливими, оскільки вони допомагають формувати структуру та контент веб-сайту, а також впливають на те, як організація залучає та взаємодіє з своїм аудиторією.

На цьому сайті ми бачимо, що цілі та цільове призначення веб-сайту не ясні. Очевидно, що веб-сайт має інформаційний характер, що дозволяє відвідувачам дізнатися про організацію. Однак, відсутність яскраво вираженого виклику до дії (СТА), який стимулює відвідувачів до певного дійства або взаємодії, робить сайт нескільки тривіальним і безцільовим.

Виклики до дії важливі, оскільки вони прямують користувачів до певного дійства, яке організація вважає важливим. Це може бути реєстрація на захід, підписка на розсилку новин, внесення пожертвування, або, наприклад, підписка на добровольчу діяльність. Виклики до дії не тільки поліпшують користувацький досвід, але і допомагають організації досягти своїх цілей [17].

Відсутність такого виклику до дії на даному сайті має негативний вплив на його смислове наповнення. Сайт має обмежену функціональність та відчутний дефіцит цілеспрямованості. Всупереч тому, що сайт надає інформацію про організацію, він не змушує відвідувачів до активного взаємодії або дії, що уповільнює його здатність досягти будь-яких значущих цілей [18, 19].

Ця ситуація стає ще більш проблематичною, коли ми розглядаємо сайт в контексті ролі, яку він має виконувати для громадської організації.

Громадські організації, як правило, залежать від активної участі та підтримки громадськості. Вони мають не тільки інформувати, але й стимулювати дію. Така дія може бути такою простою, як розповсюдження інформації в соціальних мережах, або такою складною, як участь у волонтерських програмах або внесення пожертвуваль. Якщо веб-сайт не мотивує відвідувачів до таких дій, він втрачає свою важливість.

Таким чином, цей сайт має велику можливість для покращення відносно цілей його використання. На основі аналізу варто підкреслити важливість мати виразну мету на сайті, яка визначатиме його смислове наповнення та діяльність. Організація повинна чітко вказати, чого вона очікує від відвідувачів, і як вони можуть допомогти їй у досягненні її цілей.

3. Недостатність контентного наповнення

Важливим аспектом веб-дизайну, який потребує покращення на цьому сайті, є його контентна наповненість. Як відомо, контент є королем в цифровому світі. Він відіграє важливу роль у привертанні уваги користувачів, виклику зацікавленості, а також у формуванні позитивного враження про бренд або організацію. У контексті даного сайту, контент обмежується лише новинами організації і коротким описом "про нас", що є недостатнім для повноцінного взаємодію з аудиторією.

- По-перше, необхідно збільшити різноманітність контенту. Це може включати більше відео, зображень, інфографік, кейсів, свідчень, відгуків або інших форм вмісту, які допоможуть зробити сайт більш привабливим та інформативним. Для організацій, як ця, контент може також включати огляди проектів, інформацію про волонтерство, матеріали для навчання, деталі про партнерство та інше.

- По-друге, контент повинен бути орієнтований на дію. Це означає, що кожна сторінка або розділ сайту повинен містити відповідний виклик до дії (СТА), що направляє відвідувачів на наступний крок в їхньому взаємодії з сайтом. Наприклад, новини можуть мати посилання на проекти, пов'язані з темою новини, а сторінка "Про нас" може містити посилання на

сторінку для залучення волонтерів або пожертв.

- По-третє, вміст повинен бути систематично оновлюваним, щоб він залишався актуальним та цікавим для аудиторії. Це вимагає створення і реалізації ефективної стратегії контенту, яка включає регулярне планування, створення, редагування, публікацію та перегляд контенту.

Враховуючи вищезазначені відмінності, можна зробити висновок, що наявний веб-сайт має суттєвий потенціал для покращення в частині контенту. Ці зміни не тільки збільшать взаємодію користувачів з сайтом, але й покращать їх загальний досвід, а також зроблять сайт більш ефективним у досягненні його основної мети – повідомлення про діяльність організації, залучення волонтерів та підтримки проектів.

Перейдемо до швидкості завантаження.

1. Посилання на Google PageSpeed Insights (Рисунок 2.5):

https://pagespeed.web.dev/analysis/https-gronu-org-ua/kcxwn6p72t?form_factor=desktop



Рисунок 2.5 – Посилання на Google PageSpeed Insights веб-сайту громадської організації «Громадська Рада освітян і науковців України»

2. Посилання на Gtmetrix (Рисунок 2.6):

<https://gtmetrix.com/reports/gronu.org.ua/mPcXPkqM/>



Рисунок 2.6 – Посилання на Gtmetrix веб-сайту громадської організації «Громадська Рада освітян і науковців України»

В контексті розглянутого веб-сайту, результати аналізу за допомогою сервісів Google PageSpeed Insights та GTmetrix, швидкості завантаження демонструють високий показник. Це, безумовно, позитивна властивість, яка сприяє поліпшенню користувацького досвіду. Це досягається за рахунок простоти верстки веб-сайту, мінімальної кількості великих медіа-елементів, обмеження використання скриптів та відсутності важких фреймворків.

Висновки стосовно сайту громадської організації “Громадська Рада освітян і науковців України”

РОЗДІЛ 3 РОЗРОБКА ТА СТВОРЕННЯ КЛІЄНТ-СЕРВЕРНОЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ ГРОМАДСЬКОЇ ОРГАНІЗАЦІЇ

3.1 Визначення цілей та проектування дизайну

У цьому розділі ми переходимо до головної фази нашого проекту – створення клієнт-серверної інформаційної системи громадської організації. Використання клієнт-серверної моделі дозволяє нам створювати динамічний, інтерактивний веб-сайт, який може надавати різноманітні послуги та відповідати на потреби користувачів в режимі реального часу.

Одним з перших кроків на цьому шляху є визначення основних цілей нашої системи. Ці цілі повинні відображати основні інтереси та потреби організації, а також її ключову аудиторію. За умови розглядуваної громадської організації, можливі цілі можуть включати:

- Ознайомлення відвідувачів з основними активностями організації. Це може включати інформацію про проведення конференцій, консультацій та інших подій.
- Залучення нових членів до організації.
- Надання актуальної та повної інформації про діяльність організації, її місію, цілі та досягнення.
- Забезпечення зв'язку між членами організації, її партнерами та громадськістю в цілому.
- Ознайомлення відвідувачів з науковими публікаціями організації, проектів та грантів.

Після визначення цих цілей, ми можемо приступити до розробки функціональних вимог до нашої системи, включаючи структуру сайту, дизайн інтерфейсу, технології розробки та інше.

Основою будь-якої ефективної інформаційної системи є її структура. Вона визначає, як інформація організована і як користувачі можуть навігуватися по системі. Відповідно до цілей, які були визначені раніше, ми

запропонуємо наступну структуру сайту:

- Головна сторінка: Це перша сторінка, яку бачить користувач, коли відвідує сайт. Головна сторінка слугує вступом до сайту і вона має дати користувачам швидкий огляд того, чим займається організація, її місію, основні діяльності і як їх можна досягнути.

- Сторінка "Про нас": Тут буде докладний опис організації, її місії, цілей, історії, членів команди та інших ключових деталей.

- Сторінка "Наукова діяльність": Ця сторінка буде включати деталі про наукові проекти, гранти та публікації, які організація виконує або фінансує. Це може включати описи проектів, результати, публікації, інформацію про гранти та інше.

- Сторінка "Конференції": На цій сторінці будуть розміщені деталі про всі конференції, які організація проводить або планує провести.

- Сторінка "Консультавання": Тут будуть наведені деталі про консультації, які надає організація.

- Сторінка "Новини": Остання, але не менш важлива, ця сторінка буде слугувати платформою для останніх новин та оновлень від організації. Це може бути інформація про нові проекти, найновіші досягнення, події тощо.

Для забезпечення професійного та естетично привабливого вигляду сайту, використовується графічний редактор Figma. Figma є потужним інструментом, який дозволяє дизайнерам створювати високоякісні макети та прототипи веб-сторінок з великою точністю.

При проектуванні дизайну в Figma, дизайнер зосереджується на створенні елементів інтерфейсу, виборі кольорової палітри, розташуванні контенту та інших деталей, що визначають зовнішній вигляд сайту. Графічний редактор Figma надає широкі можливості для маніпулювання об'єктами, додавання ефектів та взаємодії з елементами інтерфейсу. В процесі роботи з Figma, дизайнер може створювати макети різних сторінок сайту, включаючи головну сторінку, сторінки з контентом, форми,

навігаційні елементи та інші. Дизайнер враховує принципи доброго UX-дизайну, забезпечуючи зручну навігацію, читабельність контенту та зручність взаємодії для користувачів.

Використання Figma дозволяє дизайнеру взаємодіяти з командою розробників та іншими зацікавленими сторонами, ділитися макетами та отримувати зворотній зв'язок. Це сприяє ефективному спілкуванню та спільному розумінню щодо вигляду та функціональності майбутнього сайту.

Головна сторінка веб-сайту, безсумнівно, є його хребтом. Вона надає користувачам загальний огляд організації і прямує їх до детальнішої інформації, доступної на інших сторінках. Це перший контакт, який користувач має з сайтом, і від цього залежить враження, яке вони отримають.

З точки зору дизайну, головна сторінка служить як візуальний стандарт для всього сайту. Вона визначає загальний стиль, колірну схему, типографію та інші дизайнерські елементи, які будуть використовуватися на всьому сайті. Прийняття рішень щодо дизайну на цьому етапі полегшує розробку інших сторінок, оскільки ми вже будемо мати чіткий напрямок, відповідно до якого працювати. Крім того, головна сторінка є ідеальним місцем для встановлення основного сценарію навігації. Вона дозволяє встановити ієрархію інформації, де важливіші частини вмісту або основні діяльності є більш видимими і легко доступними.

Останнє, але не менш важливе, головна сторінка може включати важливі функції, такі як вхід для членів організації, динамічні елементи, такі як слайдери новин або майбутніх подій, і так далі. Все це сприяє забезпеченню корисного та зручного досвіду користування для відвідувачів сайту.

З усіма цими важливими ролями, які головна сторінка грає, її планування та дизайн вимагають особливої уваги та ретельності. Основна мета полягає в тому, щоб забезпечити, щоб користувачі можуть легко знайти

ту інформацію, яку вони шукають, та стимулювати їх до дії, такої як реєстрація на конференцію, зв'язок з організацією або вступ до членів. Це всі дуже важливі речі, які необхідно врахувати при створенні ефективної головної сторінки для громадської організації.

З огляду на розглянуті вище аспекти, ми переходимо до етапу планування і проектування головної сторінки веб-сайту громадської організації. Це критичний етап, який вимагає уважної розробки і реалізації дизайну та структури вмісту.

Центральним елементом головної сторінки буде назва організації, поміщена на видному місці, таким чином забезпечуючи відразу впізнаваність та встановлення зв'язку. Це супроводжується відразу ж зверненням до користувача з пропозицією приєднатися до організації. Заклик до дії чітко формулюється, а їх яскравий дизайн забезпечує видимість (Рисунок 3.1).

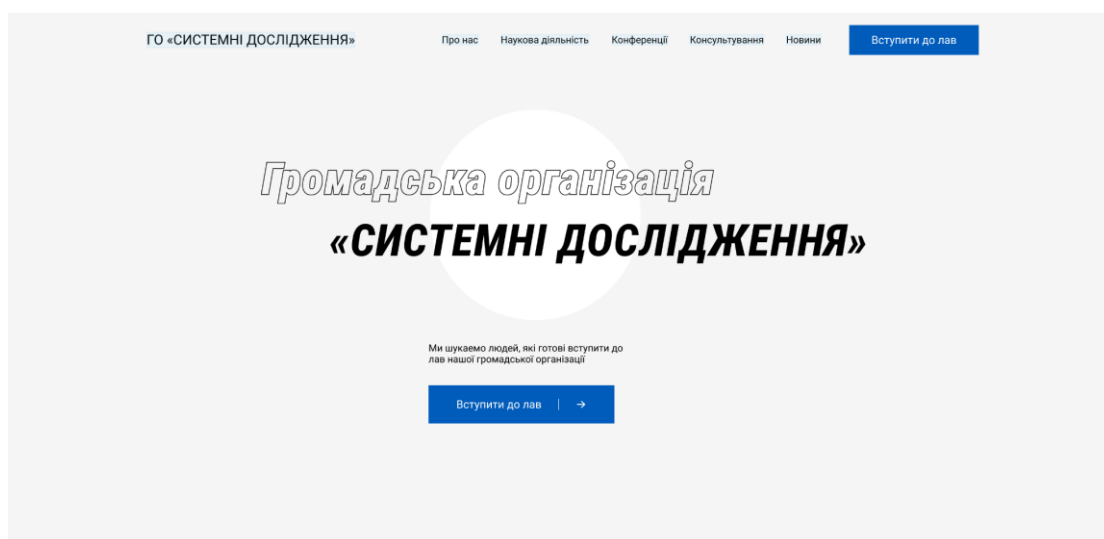


Рисунок 3.1 – Перший блок головної сторінки

Наступним блоком буде розділ, присвячений останнім конференціям організації. Він буде містити короткі анотації до двох останніх конференцій та гіперпосилання для доступу до повного переліку. Це допоможе користувачам швидко ознайомитися з активностями організації і

спонукатиме їх до подальшого дослідження (Рисунок 3.2).

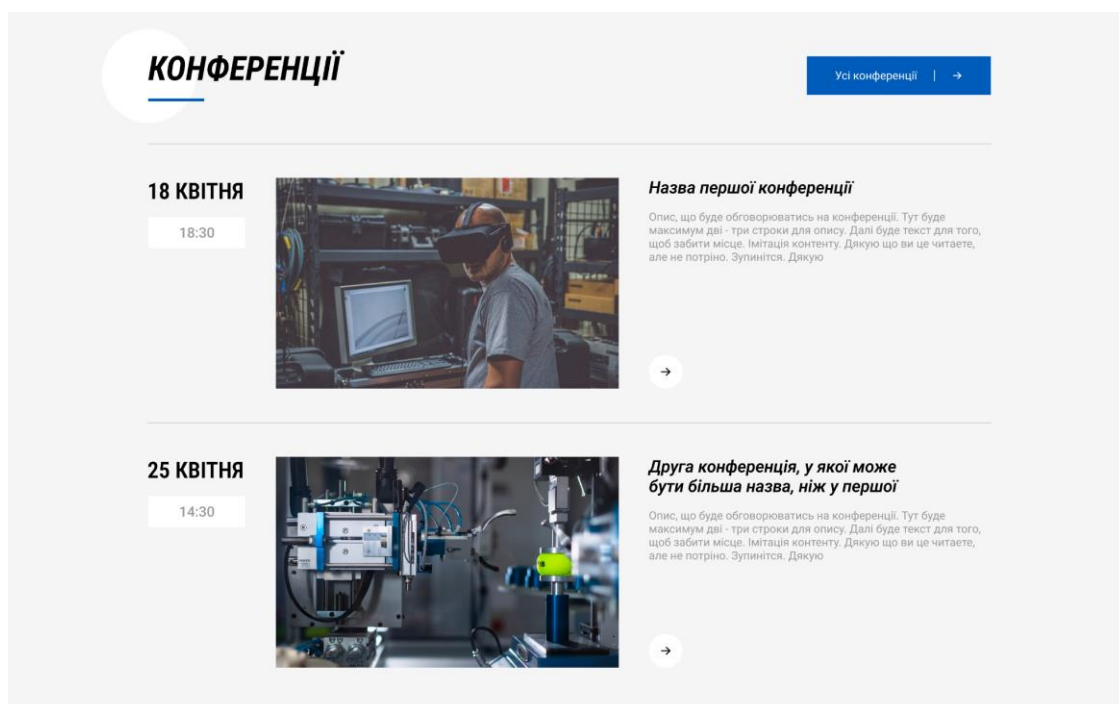


Рисунок 3.2 – Блок з конференціями на головній сторінці

Після цього йде розділ про консультування. Тут коротко описуються послуги консультування, які надає організація, а також надаються посилання для отримання додаткової інформації на окремій сторінці (Рисунок 3.3).

КОНСУЛЬТУВАННЯ



Курси та стажування

В цьому розділі ви зможете ознайомитись з інформацією по проходженню курсів та стажуванню у го. Чи якийсь інший текст.

Дізнатися більше | →

Для шкіл

Тут буде інформація для шкіл, щодо взаємного розвитку та сумісної наукової діяльності. Чи інший текст.

Дізнатися більше | →



Рисунок 3.3 – Блок “Консультавання” с посиланнями до сторінки консультавань

Частину з новинами, яка іде наступною, заповнюють останні важливі оголошення або події, що відбулися в організації. Це допомагає утримувати користувачів в курсі останніх новин, причому знову ж таки є посилання для перегляду всіх новин (Рисунок 3.4).

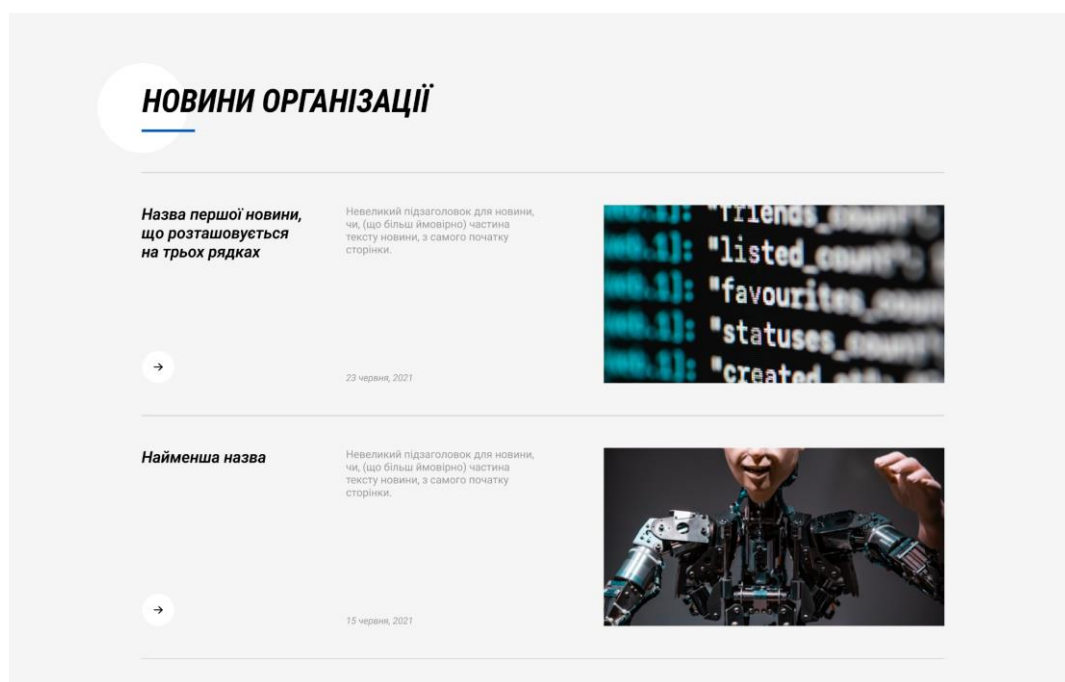


Рисунок 3.4 – Блок з новинами організації на головній сторінці

Накінець, головна сторінка завершується блоком з контактною інформацією, яка надає деталі для зв'язку з організацією. Це включає в себе адресу, номер телефону, електронну пошту та можливо, соціальні медіа, які організація використовує (Рисунок 3.5).

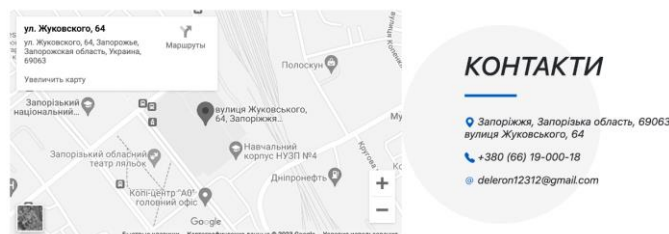


Рисунок 3.5 – Блок з контактами на головній сторінці

Така структура головної сторінки відображає цілі та місію організації, водночас надаючи користувачам легкий доступ до ключової інформації. Продуманий дизайн і ефективне розташування вмісту дозволять головній сторінці виконувати свою роль в якості "хребта" веб-сайту, ведучи користувачів до необхідної їм інформації і спонукаючи до дій, що відповідають цілям організації.

Вихідячи з дизайну, викладеного на головній сторінці, ми розробляємо концепцію дизайну для наступних сторінок веб-сайту. Оскільки ми вже визначили ключові елементи і загальну структуру на головній сторінці, використання цього як основи для решти сайту дозволить збільшити ефективність і швидкість розробки.

Хедер сайту, що містить основне навігаційне меню, буде стандартним для всіх сторінок. Він включатиме посилання на всі ключові розділи веб-сайту, що забезпечує легкий доступ до інформації. Центральне місце в хедері займає прямий заклик до дії – приєднатися до організації, таким чином постійно нагадуючи користувачу про цю можливість (Рисунок 3.6).

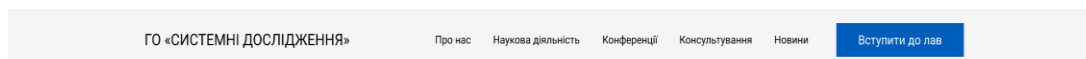


Рисунок 3.6 – Хедер сайту, який повторюється на кожній сторінці

Однак сторінка "Наукова діяльність" вимагає індивідуального підходу. Вона відрізняється від інших, оскільки це місце, де користувачі можуть ознайомитися з науковими досягненнями організації. Ця сторінка міститиме файли публікацій, деталі проектів і грантів, які користувачі можуть переглянути або завантажити. Вона буде містити як текстовий, так і візуальний контент, що забезпечує повне розуміння діяльності організації. Ця сторінка вимагає особливого вигляду, щоб привернути увагу до важливості наукової роботи, яку проводить організація (Рисунок 3.7).

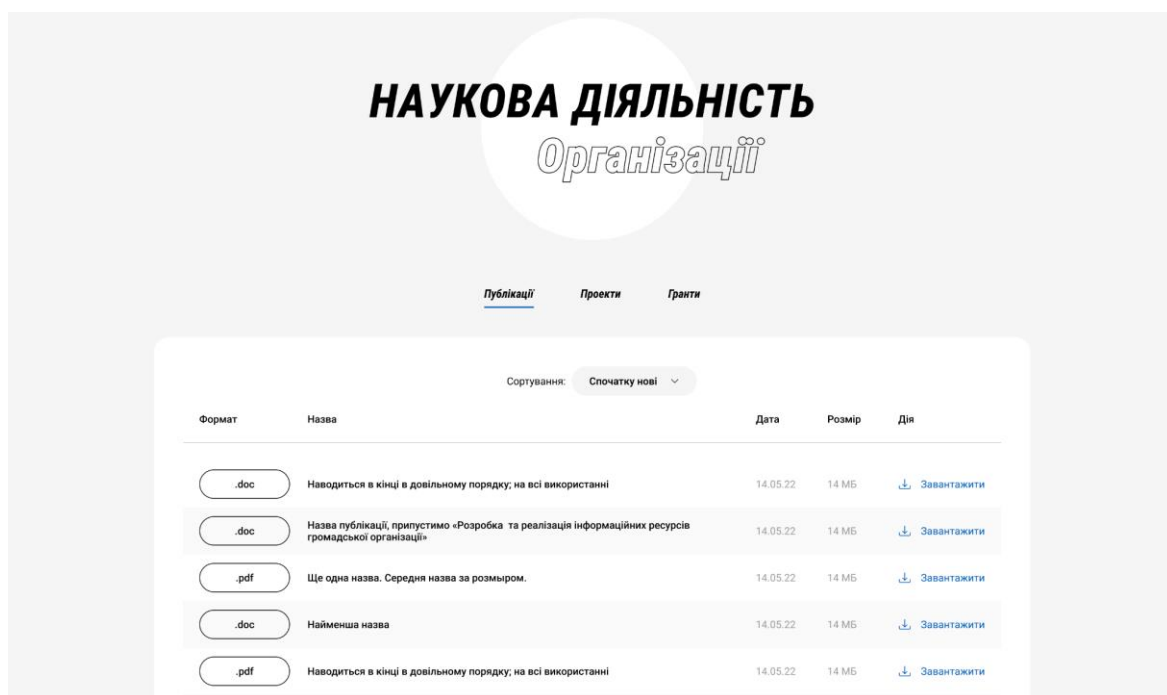


Рисунок 3.7 – Сторінка “Наукова діяльність” де будуть знаходитись документи, які можна завантажити та ознайомитись

Таким чином, наш підхід до дизайну веб-сайту ґрунтується на стратегії використання повторюваних елементів для забезпечення

консистентності, водночас розробляючи унікальні рішення для окремих ключових сторінок, що відображають специфіку їх вмісту.

3.2 Вибір технологічного стеку для розробки

Для розробки серверної частини нашої клієнт-серверної інформаційної системи ми обрали наступні ключові технології та інструменти:

- PHP: PHP є відкритим високорівневим мовою програмування, що активно використовується в веб-розробці. Ця мова забезпечує гнучкість і швидкість розробки, що дозволяє нам прискорити процес розробки та отримати більш стабільний продукт.

- MySQL: MySQL є однією з найпопулярніших відкритих систем керування базами даних. MySQL відрізняється високою швидкістю, масштабованістю, гнучкістю та надійністю, що робить його вибором №1 для багатьох веб-додатків.

- MODX: MODX є відкритим системою управління контентом (CMS), який надає велику гнучкість в процесі розробки та управління сайтами. Він використовує PHP для обробки динамічного контенту, та MySQL для зберігання даних, і дозволяє легко створювати і керувати контентом на сайті.

- DNS: Система доменних імен (DNS) перетворює зрозумілі для людини доменні імена, такі як "example.com", на числові IP-адреси, такі як "192.0.2.1", що використовуються для маршрутизації трафіку в Інтернеті. Це дуже важливо для забезпечення доступу до нашого сайту з усього світу.

- Apache: Apache є лідером серед веб-серверів, який дозволяє розмістити наш сайт в Інтернеті. Він стабільний, надійний, гнучкий і добре підтримується, що робить його ідеальним вибором для нашого проекту.

Для розробки клієнтської частини нашої клієнт-серверної інформаційної системи ми обрали наступні ключові технології та інструменти:

- HTML: HTML (HyperText Markup Language) є основним мовою розмітки для створення веб-сторінок. HTML дозволяє нам структурувати інформацію на веб-сторінках і є фундаментом для будь-якого веб-сайту.

- CSS: CSS (Cascading Style Sheets) використовується для стилізації HTML-документів. За допомогою CSS, ми можемо контролювати колір тексту, шрифти, відступи, розмір тексту, та багато іншого, що дозволяє нам створювати ефективні та привабливі сторінки.

- JavaScript (JS): JavaScript дозволяє нам створювати динамічний контент, обробляти події, створювати анімацію, та все інше, що залучає взаємодію користувача. Воно необхідне для створення багатофункціонального веб-сайту.

- Ajax: Ajax (Asynchronous JavaScript and XML) є набором технологій веб-розробки, який використовується для створення асинхронних веб-додатків. За допомогою Ajax, ми можемо виконувати запити на сервер без перезавантаження сторінки, що забезпечує плавну та швидку взаємодію користувача.

- Bootstrap: Bootstrap є відкритим фреймворком для розробки веб-сайтів. Він містить HTML- і CSS-шаблони дизайну для типографіки, форм, кнопок, навігації та інших компонентів інтерфейсу, а також додаткові JavaScript-розширення.

- JQuery: JQuery є швидкою, невеликою і багатофункціональною бібліотекою JavaScript. Вона робить речі, такі як обхід та маніпуляція HTML документів, обробка подій, анімація та Ajax, набагато простішими за допомогою простого API, яке працює на безлічі браузерів.

Вибір цих технологій та інструментів надає нам можливість розробити масштабовану, стабільну, гнучку та швидку інформаційну систему, яка буде задовольняти потреби організації та її користувачів.

3.3 Розробка бази даних і серверної частини системи

Для підтримки функціональності нашого веб-сайту, який включає новини, реєстрацію на конференції, консультації та скачування файлів, ми створюємо потужну базу даних за допомогою MySQL, реляційної системи управління базами даних (RDBMS).

В основу нашої бази даних покладено кілька ключових таблиць:

- Table_News: Ця таблиця створена для зберігання інформації про всі новини, що публікуються на веб-сайті. Вона включає поля для ідентифікатора новини (News_ID), заголовка новини (News_Title), тексту новини (News_Content), дати публікації (Date_Published).

- Table_Conferences: Для управління реєстрацією на конференції, ми створюємо таблицю Conferences, яка містить поля для ідентифікатора конференції (Conference_ID), назви конференції (Conference_Title), дати проведення (Conference_Date), та детального опису конференції (Conference_Description).

- Table_Consultations: Для обробки запису на консультації, нам потрібна таблиця Consultations. Вона включає поля для ідентифікатора консультації (Consultation_ID), назви консультації (Consultation_Title), дати консультації (Consultation_Date), місця проведення, опису консультації (Consultation_Description).

- Table_Files: Для обробки скачування файлів ми створюємо таблицю Files, яка містить поля для ідентифікатора файлу (File_ID), назви файлу (File_Name), типу файлу (File_Type) та шляху до файлу на сервері (File_Path).

- Table_Members: Для запису користувачів, що мають бажання вступити долав. Вона включає Ім'я користувача (Name_ID), телефон користувача (Phone_ID), та e-mail користувача (Email-ID)

Крім цих основних таблиць, ми можемо включати додаткові таблиці

для авторів, користувачів, консультантів та іншої специфічної для проекту інформації. Кожна таблиця має визначені індекси для підвищення швидкості пошуку та обмеження для забезпечення цілісності даних. Цей підхід до структурування нашої бази даних дає нам гнучкість та надійність, необхідні для підтримки широкого спектра функціональності нашого веб-сайту.

Розробка серверної частини веб-сайту для нашої громадської організації включає використання PHP, широко використовуваної та надійної мови програмування відкритого коду, спеціально призначеної для веб-розробки. PHP надає змогу розробити потужний серверний код, який може взаємодіяти з базою даних, обробляти запити від користувача і повертати відповіді в форматі, який розуміє веб-браузер.

- Новини: Ми використовуємо PHP для динамічного отримання та відображення новин з нашої бази даних MySQL. Кожен раз, коли користувач відвідує сторінку новин, PHP-скрипт виконує запит SELECT до таблиці News в базі даних, отримує останні новини та відображає їх на веб-сторінці.

- Реєстрація на конференції: Коли користувач заповнює форму реєстрації на конференцію, введені дані передаються на сервер через HTTP-запит POST. PHP-скрипт на сервері отримує ці дані, проводить валідацію, і якщо всі дані вірні, виконує INSERT запит до таблиці Conferences в базі даних, зберігаючи нову реєстрацію.

- Реєстрація на консультації: Аналогічно процесу реєстрації на конференції, форма реєстрації на консультації обробляється PHP-скриптом, який валідує вхідні дані та зберігає нову реєстрацію в базі даних.

- Скачування файлів: Щодо завантаження файлів, PHP використовується для обробки запиту на завантаження, перевірки дозволу користувача на завантаження вибраного файлу, та служить файл для завантаження користувачем.

- Використання PHP як основного серверного мови для нашого

проекту дозволило нам ефективно виконувати серверну логіку, яка потрібна для взаємодії з користувачами та управління даними на веб-сайті нашої організації.

- За допомогою PHP ми створюємо серверні скрипти, які приймають дані від користувача.

MODX являє собою гнучку та потужну систему управління контентом, яку ми обрали для цього проекту, спираючись на її можливості адаптації та здатність задовольняти унікальні потреби нашого веб-сайту для громадської організації.

- Управління контентом: Основним завданням будь-якої CMS є управління контентом, і MODX виконує це завдання з високою ефективністю. Новини, події, консультації та документи для скачування – все це може бути легко додано, оновлено або видалено через інтуїтивно зрозумілий інтерфейс адміністратора MODX.

- Реєстрація на конференції та консультування: Через використання засобів MODX для управління формами та користувацькими полями, ми можемо створювати та кастомізувати форми реєстрації для наших конференцій і консультацій. Коли користувач заповнює форму, дані надсилаються на сервер, де вони обробляються та зберігаються.

- Скачування файлів: MODX надає нам можливість управляти файлами, які доступні для скачування, дозволяючи нам завантажувати, організовувати та контролювати доступ до цих файлів через нашу CMS.

- Реєстрація до лав: Форма вступу до лав може бути легко створена та налаштована з використанням MODX. Ми можемо визначити необхідні поля (ім'я, телефон, електронна пошта), та налаштувати спосіб обробки даних форми на сервері.

На завершення, MODX дозволяє нам створювати рішення, яке відповідає специфічним потребам нашого сайту, надаючи нам гнучкість, якої часто не знаходиш у стандартних CMS. Даний факт дозволяє нам створити продуктивний, безпечний та дружній до користувача веб-сайт,

який ефективно виконує свою місію як інформаційний ресурс для нашої громадської організації.

DNS (Domain Name System) та Apache є ключовими елементами для розгортання нашого веб-сайту, забезпечуючи що веб-сайт доступний для відвідувачів та що його контент слугує запланованому призначенню.

Апаратна та програмна конфігурація DNS та Apache була виконана таким чином, щоб максимально використати ресурси нашого сервера, забезпечуючи швидку і надійну роботу сайту. Налаштування DNS було здійснено так, щоб користувачі могли легко знайти наш веб-сайт, а конфігурація Apache була оптимізована для обробки великого навантаження та слугування великої кількості HTTP-запитів від відвідувачів нашого веб-сайту.

3.4 Розробка клієнтської частини системи

Створення впізнаваної, інтуїтивно зрозумілої та естетично привабливої веб-сторінки вимагає вмілого використання технологій розмітки та стилізації, а також скриптових мов. У нашому випадку основою стало поєднання HTML, CSS, JavaScript, jQuery, Ajax та Bootstrap:

- HTML (HyperText Markup Language) є основою будь-якої веб-сторінки. Він визначає структуру та семантику вмісту веб-сторінки. Для нашого сайту ми використали HTML для створення різних структурних елементів: заголовків, абзаців, списків, посилань, форм для запису на конференції та консультування, а також областей для скачування файлів. Всі ці елементи були організовані таким чином, щоб користувачам було зручно навігуватися по веб-сайту.

- CSS (Cascading Style Sheets) відповідає за візуальний вигляд веб-

сторінки. Ми використали CSS для задання колірив, шрифтів, розміру тексту, відступів, меж та інших стилів, що створюють загальний "вигляд і відчуття" нашого веб-сайту. Це включає в себе стилізацію кнопок, форм, меню навігації, і багато іншого, щоб забезпечити однорідний та професійний вигляд нашого сайту.

- JavaScript дозволяє додавати взаємодію та динаміку на веб-сторінку. Ми використали JavaScript для створення взаємодії з користувачем, таких як відповідь на кліки по кнопкам, валідація форм, обробка даних форми перед відправкою на сервер, та інше.

- Bootstrap є найпопулярнішим HTML, CSS, і JS фреймворком для розробки респонсивних та мобільно-орієнтованих веб-проектів. У нашому проекті, ми використали Bootstrap для створення адаптивної верстки сайту, що дозволяє коректно відображати його на різних пристроях і розмірах екранів, включаючи комп'ютери, планшети та мобільні пристрої.

- Ми використали Ajax для реалізації функціоналу запису на конференції та консультування. Коли користувач заповнює форму запису, Ajax відправляє дані на сервер без перезавантаження сторінки, забезпечуючи плавний та ефективний користувацький досвід. Це також є корисним при відображенні динамічної інформації, наприклад, при оновленні списку новин.

Таким чином, налаштування структури та стилізації веб-сторінки за допомогою HTML, CSS, JavaScript, jQuery Ajax та Bootstrap. Використання цих технологій дозволило нам створити адаптивний, функціональний та візуально привабливий веб-сайт для нашої громадської організації.

3.5 Тестування системи

Тестування є віддільною і вкрай важливою частиною розробки інформаційного сайту. Воно дає можливість перевірити, чи відповідає функціонал веб-сайту всім вимогам та очікуванням. У нашому проекті, ми проводили ретельне мануальне тестування сайту громадської організації, перевіряючи кожен аспект його функціональності, включаючи перегляд новин, можливість запису на конференції, запису на консультування, скачування файлів, та процес запису до лав організації.

- Тестування Перегляду Новин – Основна мета була перевірити, чи коректно відображається контент, чи є можливість перегляду повного тексту новини після кліку на прев'ю, та чи добре працює пагінація на сторінці новин. Було проведено ретельне тестування на різних розмірах екрану, щоб впевнитися, що сторінка коректно адаптується до різних розмірів екранів.

- Тестування Запису на Конференції і Консультування – Ми перевіряли, чи правильно працює форма реєстрації, чи відправляються введені дані на сервер, та чи отримує користувач відповідне повідомлення про успішну реєстрацію або про помилку, якщо дані були введені некоректно. Також ми тестували чи коректно працює вибір дати та часу конференції або консультування.

- Тестування Скачування Файлів – Основними критеріями було визначено можливість скачування файлу, коректне відображення назви та розміру файлу, та роботу посилань на скачування.

- Тестування Запису до Лав Організації – Цей процес потребував перевірки всіх полів вводу (ім'я, телефон, та імейл), а також перевірки на стороні сервера. Також було важливо перевірити, чи коректно дані зберігаються в базі даних.

У результаті мануального тестування, ми виявили та виправили

декілька проблем, що покращило стабільність та зручність використання сайту.

РОЗДІЛ 4 ПІДСУМКИ СТВОРЕННЯ КЛІЄНТ-СЕРВЕРНОЇ ІНФОРМАЦІОННОЇ СИСТЕМИ

4.1 Результати роботи

У підсумку всіх зусиль, ми успішно розробили функціональну клієнт-серверну інформаційну систему для громадської організації. Ця система, яка була створена за допомогою сучасних технологій та підходів до розробки, надає організації зручний, інтуїтивно зрозумілий інтерфейс для взаємодії з їх аудиторією та вирішення їх ключових проблем.

Центральним елементом системи є веб-сайт, який об'єднує у собі інформаційний портал для поширення новин та оголошень, платформу для реєстрації та участі у конференціях та консультаціях, механізм для скачування важливих документів, а також форму для залучення нових членів до лав організації.

Використання технологій HTML, CSS, JavaScript, jQuery, Bootstrap, Ajax на клієнтській стороні, а також PHP, MySQL, MODX на серверній стороні, забезпечило нам гнучкість і ефективність у процесі розробки. Всі ці технології були інтегровані в одну систему, що дозволило нам реалізувати потрібну функціональність, забезпечити надійність і швидкість роботи системи, а також зробити її простою і зручною для користувачів.

Мануальне тестування гарантувало нам високу якість кінцевого продукту, дозволило виявити та виправити виявлені проблеми перед тим, як система була введена в експлуатацію. Завдяки цьому, ми змогли запобігти багатьом потенційним проблемам, що могли би виникнути під час роботи системи.

У підсумку, наша розроблена клієнт-серверна інформаційна система є важливим активом для громадської організації. Вона сприяє ефективному вирішенню їх ключових завдань, зокрема поширенню інформації, координації заходів та взаємодії з членами організації та широкою

аудиторією. Даний проект став своєрідним мостом між громадською організацією та її цільовою аудиторією, дозволяючи обмінюватися важливою інформацією, координувати діяльність та розвивати ефективну взаємодію.

4.2 Можливі напрямки подальшого розвитку системи

На даний час, інформаційна система, яку ми розробили для громадської організації, забезпечує ряд основних функцій, які важливі для ефективного функціонування організації. Однак, завдяки гнучким технологіям та підходам, які були використані під час розробки, у нас є можливість для подальшого розвитку та удосконалення системи.

Однією з ключових областей, яка може бути вдосконалена, є робота з науковими роботами, грантами та публікаціями. Наразі, користувачі можуть переглядати ці ресурси, але можливості по сортуванню та фільтрації є досить обмеженими. В подальшому, ми плануємо розширити ці можливості, щоб користувачі могли легко знайти саме ту інформацію, яка їм потрібна. Специфічні удосконалення можуть включати розширення параметрів сортування для наукових робіт, грантів та публікацій, включаючи можливість сортувати за датою публікації, авторами, областю дослідження, рівнем впливу, та іншими критеріями. За допомогою розширення фільтрації, користувачі зможуть вибирати лише ті документи, які відповідають певним критеріям, наприклад, тільки ті наукові роботи або гранти, які відносяться до певної області дослідження, фінансуються певним джерелом або були опубліковані в певний період часу.

Або процес реєстрації на сайті. Наразі він доволі простий і вимагає введення базової інформації, такої як ім'я, телефон, та електронна пошта користувача, які записуються у базу даних. Однак, для подальшого

вдосконалення користувацького досвіду та надання більш персоналізованих послуг, можливе введення додаткових параметрів реєстрації. Наприклад, ми можемо надати користувачам опціональну можливість вказувати свої особисті інтереси, професійну сферу, а також вибирати, що саме вони хочуть отримувати від нашого сайту – від новин та оголошень до спеціальних інформаційних бюлетенів і пропозицій.

Чи, припустим підписка на новини. Оформлення підписки може бути реалізовано таким чином, що користувачам буде запропоновано заповнити просту форму з власною електронною адресою. Вони також зможуть вибрати від яких категорій новин вони хотіли б отримувати сповіщення: оголошення про нові події, оновлення блогу, спеціальні пропозиції тощо. У свою чергу, це допоможе організації покращити свої комунікації з користувачами, а також збільшити енгажованість та лояльність спільноти. Крім того, використання системи підписки на новини дозволить зібрати базу даних електронних адрес користувачів для подальшого аналізу та націленого маркетингу.

Всі ці удосконалення зможуть значно поліпшити користувацький досвід, роблячи систему більш корисною та ефективною для цільової аудиторії організації. Безумовно, використання гнучких технологій в процесі розробки інформаційної системи надає нам великі переваги, дозволяючи нам адаптуватися до змінюваних потреб організації та її користувачів. Завдяки цьому, ми можемо продовжувати поліпшувати систему, роблячи її все більш ефективною засобом підтримки діяльності громадської організації.

ПЕРЕЛІК ПОСИЛАНЬ

1. **Björk, B. C.** Open Access to the Scientific Journal Literature: Situation [Електронний ресурс] / B. C. Björk, P. Welling, M. Laakso, P. Majlender, T. Hedlund, G. Guðnason // Plos one. –2009. – I. 5. – No. 6. – P. e11273. – Режим доступу:

<https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0011273>

2. **Siler, K.** Measuring the effectiveness of scientific gatekeeping [Електронний ресурс] / K. Siler, C. J. Lee, L. Bero // Proceedings of the National Academy of Sciences. –2015. – Vol. 112. – No. 2. – P. 360–365. – Режим доступу: <https://www.pnas.org/doi/full/10.1073/pnas.1418218112>

3. **Bryson, J. M.** Working with evaluation stakeholders: A rationale, step-wise approach and toolkit [Електронний ресурс] / J. M. Bryson, M. Q. Patton, R. A. Bowman // Evaluation and Program Planning. –2011. – Vol. 34. – No. 1. – P. 1–12. – Режим доступу: <https://www.sciencedirect.com/science/article/abs/pii/S0149718910000637?via%3Dihub>

4. **Neely, A.** Performance measurement system design: a literature review and research agenda [Електронний ресурс] / A. Neely, M. Gregory, K. Platts // International Journal of Operations & Production Management. – 2005. – Vol. 25. – No. 12. – P. 1228–1263. – Режим доступу: <https://www.emerald.com/insight/content/doi/10.1108/01443570510633639/full/html>

5. **Hong, L.** Групи різноманітних розв'язувачів проблем можуть перевершувати групи розв'язувачів проблем з високою здатністю [Текст] / L. Hong, S. E. Page // Збірник Національної академії наук. –2004. – Vol. 101. – No. 46. – P. 16385–16389."

6. **Jackson, S. E.** Недавні дослідження командної та організаційної різноманітності: аналіз SWOT та наслідки [Текст] / S. E. Jackson, A. Joshi,

N. L. Erhardt // Журнал управління. – 2003. – Vol. 29. – No. 6. – P. 801–830.

7. **McLeod, P. L.** Етнічна різноманітність та творчість в невеликих групах [Текст] / P. L. McLeod, S. A. Lobel, T. H. Cox // Дослідження невеликих груп. – 1996. – Vol. 27. – No. 2. – P. 248–264.

8. **Bennett, D.** The Importance of Conferences for the Academic Profession [Електронний ресурс] / D. Bennett // Higher Education Research & Development. – 2014. – Vol. 33. – No. 5. – P. 1034–1045. – Режим доступу: DOI: <https://doi.org/10.1080/07294360.2014.890572>.

9. **Bordenave, G.** The Consultation as a Communication Process: Implications for NGOs [Електронний ресурс] / G. Bordenave // Journal of Nonprofit & Public Sector Marketing. – 2013. – Vol. 25. – No. 1. – P. 80-107. – Режим доступу: <https://doi.org/10.1080/10495142.2013.759651>.

10. Google. The Importance of Homepage Design [Електронний ресурс]. – 2020. – Режим доступу: <https://research.google.com/homepage-design>.

11. UXCam. Why Users Expect to Find Company Information on the Homepage [Електронний ресурс]. – 2021. – Режим доступу: <https://uxcam.com/blog/company-information-homepage>.

12. Nielsen Norman Group. The Impact of Well-Structured Web Design on User Experience [Електронний ресурс]. – 2022. – Режим доступу: <https://www.nngroup.com/articles/impact-of-well-structured-web-design>.

13. **Smith, N.** 70% of Small Business Websites Ignore Call to Action [Електронний ресурс] / N. Smith. – 2020. – Режим доступу: <https://smallbiztrends.com/2020/07/call-to-action-statistics.html>.

14. **Chaffey, D.** Creating effective calls-to-action [Електронний ресурс] / D. Chaffey. – 2021. – Режим доступу: <https://www.wordstream.com/blog/ws/2014/07/09/call-to-action>.

15. **Wroblewski, L.** White Space in UI Design [Електронний ресурс] / L. Wroblewski. – 2019. – Режим доступу: <https://www.uxpin.com/studio/blog/white-space-in-ui-design>.

16. Kissmetrics blog. How Loading Time Affects Your Bottom Line [Электронный ресурс]. – Режим доступа: <https://neilpatel.com/blog/loading-time/>.
17. **Krug, S.** Don't Make Me Think, Revisited: A Common Sense Approach to Web Usability [Текст] / S. Krug. – New Riders, 2014. – 216 pp.
18. **Nielsen, J.** Designing Web Usability: The Practice of Simplicity [Текст] / J. Nielsen. – New Riders, 1999. – 300 pp.
19. Brinck, T. Wood. Usability for the Web: Designing Web Sites That Work [Текст] / T. Brinck, D. Gergle, S. D. Wood. – Morgan Kaufmann, 2001. – 496 pp.

ДОДАТОК А

Приклад коду клієнтської частини

```

<!doctype html>
<html lang="en">
<head>
<meta charset="utf-8" />
<meta name="viewport" content="width=device-width,initial-scale=1,shrink-to-
fit=no" />
<meta name="description" content="Bootstrap + webpack Boilerplate" />
<meta name="author" content="Alex Antsiferov" />
<base href="/">
<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap-
icons@1.3.0/font/bootstrap-icons.css">
<title>Головна - ZDNU</title>
<link rel="icon" href="icon.ico">
<link href="/css/0.css" rel="stylesheet">
<link as="style" href="css/0.css" rel="preload">
<link as="script" href="/js/main.f55867c988ad729adb02.js" rel="preload">
<link as="script" href="/js/runtime.f55867c988ad729adb02.js" rel="preload">
<link as="script" href="/js/vendor.f55867c988ad729adb02.js" rel="preload">
<link rel="preconnect" href="https://fonts.googleapis.com">
<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
<link
href="https://fonts.googleapis.com/css2?family=Roboto+Condensed:wght@700&dis
play=swap" rel="stylesheet">
<link rel="stylesheet" href="/assets/components/ajaxform/css/default.css"
type="text/css" />
</head>

```

```

<body class="bg-light pt-0">
<nav class="container navbar navbar-expand-lg navbar-light bg-light">
<div class="px-md-0 py-md-3 container-fluid"><a class="navbar-brand text-
uppercase" href="/">ГО “СИСТЕМНІ ДОСЛІДЖЕННЯ”</a> <button
class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-
target="#navbarNavAltMarkup" aria-controls="navbarNavAltMarkup" aria-
expanded="false"
aria-label="Toggle navigation"><span class="navbar-toggler-
icon"></span></button>
<div class="collapse navbar-collapse show" id="navbarBasic">
<ul class="navbar-nav ms-auto mb-2 mb-xl-0"><li class="nav-item"><a href="/pro-
nas.html" class="nav-link mx-2 first" >Про нас</a></li><li class="nav-item"><a
href="/naukova-diyalnist/" class="nav-link mx-2 " >Наукова діяльність</a></li><li
class="nav-item"><a href="/konferenczii/" class="nav-link mx-2 "
>Конференції</a></li><li class="nav-item"><a href="/konsultuvannya/"
class="nav-link mx-2 " >Консультування</a></li><li class="nav-item"><a
href="/novini/" class="nav-link mx-2 me-md-3 me-0" >Новини</a></li></ul>

<a href="#" data-bs-toggle="modal" data-bs-whatever="1" data-bs-type="3" data-bs-
target="#exampleModal" class="btn btn-primary rounded-0 px-3 py-2">Вступити
до лав</a>

<!--<ul class="navbar-nav ms-auto mb-2 mb-xl-0">-->
<!-- <li class="nav-item"><a class="nav-link active me-4" aria-current="page"
href="#">Про нас</a></li>-->
<!-- <li class="nav-item"><a class="nav-link me-4" href="#">Наукова
діяльність</a></li>-->
<!-- <li class="nav-item"><a class="nav-link me-4"
href="#">Конференції</a></li>-->
<!-- <li class="nav-item"><a class="nav-link me-4"
href="#">Консультування</a></li>-->

```

```

<!-- <li class="nav-item"><a class="nav-link" href="#">Новини</a></li>-->
<!--</ul>-->
</div>
</div>
</nav>
<main class="main">
<section class="container">
<div class="main_section d-flex flex-column align-items-center mt-5">
<div class="my-auto italic">
<h1 class="hero_title3 text-stroke">Громадська організація</h1>
<h1 class="hero_title3 mt-3 ms-5">«СИСТЕМНІ ДОСЛІДЖЕННЯ»</h1>
</div>
</div>
<div class="d-flex mx-auto flex-column my-4">
<div class="h7 w-25 mx-auto mt-3">Ми шукаємо людей, які готові вступити до
лав нашої громадської організації <a href="#" data-bs-toggle="modal" data-bs-
whatever="1" data-bs-type=3 data-bs-target="#exampleModal" class="mt-4 btn btn-
primary rounded-0 px-45 py-25">Вступити до лав <i class="h321 px-3"></i> <i
class="my-auto bi-arrow-right"></i></a></div>
</div>
</section>
<section class="container">
<div class="row py-5">
<div class="col-md-2"></div>
<div class="col-md-7">
<h3 class="h1 text-uppercase italic semibold">КОНФЕРЕНЦІЇ ЯКІ МИ
ПРОВОДИМО</h3>
<p class="mt-3">Невеличкий опис розділу, на який буде вести посилання. Та не
тільки. Більш детільний опис розділу. Що саме буде в ньому.</p>
</div>

```

```

<div class="col-md-3 d-flex ms-auto">
<div class="my-auto ms-auto"><a href="#" class="my-auto mx-auto btn btn-outline-
primary rounded-0 px-45 py-2">Перейти <i class="h321 px-3"></i> <i class="my-
auto bi-arrow-right"></i></a></div>

</div>
</div>
<div id="carouselWithCaptions" class="carousel slide" data-bs-ride="carousel">
<ol class="carousel-indicators">
<li data-bs-target="#carouselWithCaptions" data-bs-slide-to="0"
class="active"></li>
<li data-bs-target="#carouselWithCaptions" data-bs-slide-to="1"></li>
<li data-bs-target="#carouselWithCaptions" data-bs-slide-to="2"></li>
</ol>
<div class="carousel-inner">
<div class="carousel-item active">
<div class="carousel-caption d-none d-sm-block">
<h5>First slide label</h5>
<p>Nulla vitae elit libero, a pharetra augue mollis interdum.</p>
</div>
</div>
<div class="carousel-item">
<div class="carousel-caption d-none d-sm-block">
<h5>Second slide label</h5>
<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
</div>
</div>

```

```

<div class="carousel-item">
<div class="carousel-caption d-none d-sm-block">
<h5>Third slide label</h5>
<p>Praesent commodo cursus magna, vel scelerisque nisl consectetur.</p>
</div>
</div>
</div><a class="carousel-control-prev" href="#carouselWithCaptions" role="button"
data-bs-slide="prev"><span class="carousel-control-prev-icon" aria-
hidden="true"></span> <span class="visually-hidden">Previous</span> </a><a
class="carousel-control-next"
href="#carouselWithCaptions" role="button" data-bs-slide="next"><span
class="carousel-control-next-icon" aria-hidden="true"></span> <span
class="visually-hidden">Next</span></a></div>
</section>
<section class="container py-5">
<div class="d-flex">
<div class="me-auto">
<div class="d-flex ellipsis">
<h3 class="mt-4 under text-uppercase ms-2 ps-5 h2 bold
italic">КОНФЕРЕНЦІЇ</h3>
</div>

</div>
<div class="ms-auto d-flex">
<div class="my-auto"><a href="konferenczii/" class="my-auto mx-auto btn btn-
primary rounded-0 px-45 py-2">Подивитися усі конференції <i class="h321 px-
3"></i> <i class="my-auto bi-arrow-right"></i></a></div>
</div>
<!--<div class="ms-auto d-flex">-->

```

```

<!-- <div class="my-auto"><a href="#" data-bs-toggle="modal" data-bs-
whatever="1" data-bs-type=3 data-bs-target="#exampleModal" class="my-auto mx-
auto btn btn-primary rounded-0 px-45 py-2">Вступити до лав <i class="h321 px-
3"></i> <i class="my-auto bi-arrow-right"></i></a></div>-->

<!--</div>-->

</div>

<div class="py-4"><div class="d-flex border-top py-4">
<div class="col-md-2">
<div class="h5 bold text-uppercase pb-2">18 квітня</div>
<div class="bg-white w-50 p-2 text-center text-gray">18:30</div>
</div>
<div class="col-md-5"></div>
<div class="d-flex flex-column">
<h4 class="italic">Конференція на тему дипломування</h4>
<p class="text-gray"></p>
<a href="konferenczii/konferencziya-na-temu-diplomuvannya.html" class="d-flex
mt-auto text-black bg-white rounded-circle link-c"><i class="bi-arrow-right circle-m
h5"></i></a>
</div>
</div>

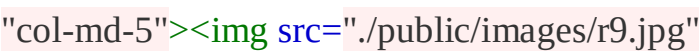
<div class="d-flex border-top py-4">
<div class="col-md-2">
<div class="h5 bold text-uppercase pb-2">18 квітня</div>
<div class="bg-white w-50 p-2 text-center text-gray">18:30</div>
</div>
<div class="col-md-5"></div>
<div class="d-flex flex-column">
<h4 class="italic">Зустріч з представником EOS Data Analytics</h4>

```

24 січня о 14:00 відбулась зустріч з випускником нашої кафедри Слободяником Олександром, представником компанії EOS Data Analytics, посада python developer. Він розповів про деякі задачі, над якими працює компанія: аналіз полів, з якими супутниками має справу, про вегетаційні індекси та про вирубку лісів.

<konferenczii/zustrich-z-predstavnikom-eos-data-analytics.html> *bi-arrow-right circle-m h5*

Консультавання

 *img w-100 img-fluid alt=""*

Курси та стажування

```

<p class="text-gray">В цьому розділі ви зможете ознайомитись з інформацією
по проходженню курсів та стажуванню у го. Чи якийсь інший текст.</p>
<div class="my-auto"><a href="konsultuvannya/#kursy" class="my-auto mx-auto
btn btn-primary rounded-0 px-45 py-2">Дізнатися більше <i class="h321 px-
3"></i> <i class="my-auto bi-arrow-right"></i></a></div>
</div>
</div>
</div>
<div class="d-flex py-4">
<div class="col-md-5 d-flex my-auto flex-column">
<div class="col-md-7 ms-auto">
<h4 class="italic h3">Для шкіл</h4>
<p class="ms-auto text-gray">Тут буде інформація для шкіл, щодо взаємного
розвитку та сумісної наукової діяльності. Чи інший текст.</p>
<div class="my-auto d-flex"><a href="konsultuvannya/#shkil" class="my-auto btn
btn-primary rounded-0 px-45 py-2">Дізнатися більше <i class="h321 px-3"></i> <i
class="my-auto bi-arrow-right"></i></a></div>
</div>
</div>
<div class="ms-auto"></div>
</div>
</div>
</div>
</section>
<section class="container py-5">
<div class="d-flex">
<div class="me-auto">
<div class="d-flex ellipsis">
<h3 class="mt-4 under text-uppercase ms-2 ps-5 h2 bold italic">НОВИНИ
ОРГАНІЗАЦІЇ</h3>

```

```

</div>
</div>
<div class="ms-auto d-flex">
<div class="my-auto"><a href="novini/" class="my-auto mx-auto btn btn-primary
rounded-0 px-45 py-2">Подивитися усі новини <i class="h321 px-3"></i> <i
class="my-auto bi-arrow-right"></i></a></div>
</div>
</div>
<div class="py-4"><div class="d-flex border-top py-4">
<div class="col-md-3 d-flex flex-column">
<div class="h5 italic text-uppercase pb-2">Новина номер один</div>
<a href="/novini/novina-nomer-odin.html" class="d-flex mt-auto text-black bg-white
rounded-circle link-c"><i class="bi-arrow-right circle-m h5"></i></a>
</div>
<div class="col-md-4 d-flex flex-column">
<p class="text-gray col-9">Припустимо новина номер один, буде розповідати
про новий дослід який був зроблений нашою громадською організацією.</p>
<p class="text-gray mt-auto">20 Червня 2022</p>
</div>
<div class="col-md-5 ms-auto d-flex flex-column">

</div>
</div>
<div class="d-flex border-top py-4">
<div class="col-md-3 d-flex flex-column">
<div class="h5 italic text-uppercase pb-2">Зустріч з представником EOS Data
Analytics</div>

```

```

<a href="/novini/zustrich-z-predstavnikom-eos-data-analytics.html" class="d-flex mt-
auto text-black bg-white rounded-circle link-c"><i class="bi-arrow-right circle-m
h5"></i></a>

</div>

<div class="col-md-4 d-flex flex-column">

<p class="text-gray col-9">24 січня о 14:00 відбулась зустріч з випускником
нашої кафедри Слободяником Олександром, представником компанії EOS
Data&#8230;</p>

<p class="text-gray mt-auto">20 Червня 2022</p>

</div>

<div class="col-md-5 ms-auto d-flex flex-column">



</div>

</div>

</div>

<div class="ms-auto d-flex mb-5 pb-5">

<div class="mx-auto"><a href="novini/" class="my-auto mx-auto btn btn-primary
rounded-0 px-45 py-2">Подивитися усі новини <i class="h321 px-3"></i> <i
class="my-auto bi-arrow-right"></i></a></div>

</div>

</section>

<section class="contact bg-white">

<div class="d-flex container py-6">

<div class="col-md-6 d-flex"><iframe class="ms-auto"
src="https://www.google.com/maps/embed?pb=!1m18!1m12!1m3!1d2679.10940001
63785!2d35.18208685052358!3d47.81809744489068!2m3!1f0!2f0!3f0!3m2!1i1024!
2i768!4f13.1!3m3!1m2!1s0x40dc5e0a946bd1af%3A0xf38c477de51a80ee!2z0YPQu
y4g0JbRg9C60L7QstGB0LrQvtCz0L4sIDY0LCDQl9Cw0L_QvtGA0L7QttGM0LU
sINCX0LDQv9C-

```

0YDQvtC20YHQtCw0Y8g0L7QsdC70LDRgdGC0YwsINCj0LrRgNCw0LjQvdC
wLCA2OTA2Mw!5e0!3m2!1sru!2snl!4v1654693654617!5m2!1sru!2snl"

width="600" height="350" style="border:0; filter: grayscale(100);"
allowfullscreen="" loading="lazy" referrerpolicy="no-referrer-when-
downgrade"></iframe></div>

<div class="col-md-6 d-flex loail">

<div class="my-auto ps-6">

<h3 class="h1 text-uppercase italic under">Контакти</h3>

<ul class="mt-5 list-unstyled italic col-md-8">

<li class="my-2 py-1"><i class="text-primary bi-geo-alt-fill"></i> Запоріжжя,
Запорізька область, 69063 вулиця Жуковського, 64

<li class="my-2 py-1"><i class="text-primary bi-telephone-fill"></i> +380 (66) 19-
000-18

<li class="my-2 py-1"><i class="text-primary bi-at"></i>
deleron12312@gmail.com

</div>

</div>

<div></div>

</div>

</section>

<footer class="bg-dark text-white py-5 pt-3">

<div class="container d-flex py-5">

<div class="col-md-5 italic">

<h2 class="h3 text-antistroke">Громадська організація</h2>

<h2 class="h3 ms-3">СИСТЕМНІ ДОСЛІДЖЕННЯ</h2>

</div>

<div class="col-md-7 justify-content-between d-flex">

<div>

<p class="mb-4 h5 italic">ГОЛОВНЕ</p>

```
<nav class="nav foo-nav flex-column">  
<a href="/pro-nas.html" class="text-white nav-link" >Про нас</a><a  
href="/konferenczii/" class="text-white nav-link" >Конференції</a><a  
href="/konsultuvannya/" class="text-white nav-link" >Консультування</a><a  
href="/novini/" class="text-white nav-link" >Новини</a>  
</div>  
  
<div>  
<p class="h5 mb-4 italic">НАУКОВА ДІЯЛЬНІСТЬ</p>  
<nav class="nav foo-nav flex-column"><a class="text-white nav-link"  
href="#">Публікації</a> <a class="text-white nav-link" href="#">Проекти</a> <a  
class="text-white nav-link" href="#">Гранти</a></nav>  
</div>  
  
<div>  
<p class="mb-4 h5 italic">МИ У СОЦ. МЕРЕЖАХ</p>  
</div>  
</div>  
  
</footer>  
  
<div class="modal fade" id="exampleModal" tabindex="-1" aria-  
labelledby="exampleModalLabel" aria-hidden="true">  
<div class="modal-dialog modal-dialog-centered modal-dialog-scrollable">  
<div class="modal-content rounded-0">  
<button type="button" class="p-3 ms-auto btn-close" data-bs-dismiss="modal" aria-  
label="Close"></button>  
<div class="modal-body col-8 pb-5 pt-0 mx-auto">  
<h5 class="modal-title mb-3 italic bold ls1" id="exampleModalLabel">Запис на  
конференцію</h5>  
Напишіть ваш Е-mail адресс, ми вишлемо вам посилання на конференцію  
<form action="" method="post" class="modal-form ajax_form af_example">
```

```

<input type="hidden" class="bg-tight p-3 px-4 text-dark rounded-0 border-0 form-
control" value="" name="type" placeholder="Default Input">
<input type="hidden" class="bg-tight p-3 px-4 text-dark rounded-0 border-0 form-
control" value="" name="id" placeholder="Default Input">
<div class="my-3"><input type="text" required class="bg-tight p-3 px-4 text-dark
rounded-0 border-0 form-control" value="" name="name" placeholder="Ваше
ім'я"></div>
<div class="my-3"><input type="email" required class="bg-tight p-3 px-4 rounded-0
border-0 form-control" name="email" value="" placeholder="Ваш E-mail"></div>
<div class="my-3"><input type="tel" class="bg-tight p-3 px-4 rounded-0 border-0
form-control" name="tel" value="" placeholder="Ваш телефон"></div>
<button type="submit" class="btn rounded-0 w-100 py-3 btn-primary">Записатись
на конференцію</button>
<input type="hidden" name="af_action"
value="1899fbbbfb69bc468f062d038ac6cbbc" />
</form>
</div>
</div>
</div>
</div>
</div>
</main>
<script src="/js/runtime.f55867c988ad729adb02.js"></script>
<script src="/js/vendor.f55867c988ad729adb02.js"></script>
<script src="/js/main.f55867c988ad729adb02.js"></script>
<script src="https://code.jquery.com/jquery-3.6.0.min.js" integrity="sha256-
/xUj+3OJU5yExlq6GSYGSHk7tPXikynS7ogEvDej/m4="
crossorigin="anonymous"></script>
<script>
const exampleModal = document.getElementById('exampleModal')
exampleModal.addEventListener('show.bs.modal', event => {

```

```

// Button that triggered the modal
const button = event.relatedTarget
// Extract info from data-bs-* attributes
const recipient = button.getAttribute('data-bs-whatever')
const type = button.getAttribute('data-bs-type')
console.log({type,recipient})
if (type){
const modalTypeInput = exampleModal.querySelector('.modal-body
input[name=type]')
modalTypeInput.value = type
}
// If necessary, you could initiate an AJAX request here
// and then do the updating in a callback.
//
// Update the modal's content.
const modalTitle = exampleModal.querySelector('.modal-title')
const tel = exampleModal.querySelector('.modal-body input[name=tel]')
console.log(tel)
let title
if (type == 1){
title = "Запис на конференцію"
modalTitle.textContent = title;
tel.style.display="none"
}
else if (type == 2){
title = "Запис на курс"
modalTitle.textContent = title;
tel.style.display="none"
}
else if (type == 3){

```

```

title = "Вступити до лав"
modalTitle.textContent = title;
tel.style.display="block"
}
const modalBodyInput = exampleModal.querySelector('.modal-body
input[name=id]')
modalBodyInput.value = recipient
})
</script>
<script src="/assets/components/ajaxform/js/default.js"></script>
<script
type="text/javascript">AjaxForm.initialize({ "assetsUrl":"\assets\components\ajaxf
orm\","actionUrl":"\assets\components\ajaxform\action.php", "closeMessage":"clo
se all", "formSelector":"form.ajax_form", "pageId":1});</script>
</body>
</html>

```

ДОДАТОК Б

Приклад коду серверної частини

```

<?php
use ModxPro\PdoTools\Fetch;
use MODX\Revolution\modSnippet;

/** @var array $scriptProperties */
/** @var \MODX\Revolution\modX $modx */
if (isset($parents) && $parents === "") {
    $scriptProperties['parents'] = $modx->resource->id;
}
if (!empty($returnIds)) {
    $scriptProperties['return'] = $return = 'ids';
} elseif (!isset($return)) {
    $scriptProperties['return'] = $return = 'chunks';
}

// Adding extra parameters into special place so we can put them in a results
/** @var modSnippet $snippet */
$additionalPlaceholders = $properties = [];
if (isset($this) && $this instanceof modSnippet && $this->get('properties')) {
    $properties = $this->get('properties');
} elseif ($snippet = $modx->getObject(modSnippet::class, ['name' =>
'pdoResources'])) {
    $properties = $snippet->get('properties');
}
if (!empty($properties)) {
    foreach ($scriptProperties as $k => $v) {
        if (!isset($properties[$k])) {
            $additionalPlaceholders[$k] = $v;
        }
    }
}

```

```

    }
}
}
$scriptProperties['additionalPlaceholders'] = $additionalPlaceholders;
$modx->services['pdotools_config'] = $scriptProperties;
$pdoFetch = $modx->services->get(Fetch::class);
$pdoFetch->addTime('pdoTools loaded');
$output = $pdoFetch->run();
if ($modx->user->isAuthenticated('mgr') && !empty($showLog)) {
    $modx->setPlaceholder('pdoResourcesLog', print_r($pdoFetch->getTime(), true));
}
// Return output
if (!empty($returnIds)) {
    if (!empty($toPlaceholder)) {
        $modx->setPlaceholder($toPlaceholder, $output);
    } else {
        return $output;
    }
} elseif ($return === 'data') {
    return $output;
} elseif (!empty($toSeparatePlaceholders)) {
    $modx->setPlaceholders($output, $toSeparatePlaceholders);
} else {
    if (!empty($tplWrapper) && (!empty($wrapIfEmpty) || !empty($output))) {
        $output = $pdoFetch->getChunk($tplWrapper,
array_merge($additionalPlaceholders, ['output' => $output]),
        $pdoFetch->config['fastMode']);
    }
    if (!empty($toPlaceholder)) {
        $modx->setPlaceholder($toPlaceholder, $output);
    }
}

```

```
    } else {  
        return $output;  
    }  
}
```