

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**ВСП «ЕКОНОМІКО-ПРАВНИЧИЙ ФАХОВИЙ КОЛЕДЖ**  
**ЗАПОРІЗЬКОГО НАЦІОНАЛЬНОГО УНІВЕРСИТЕТУ»**

Циклова комісія математичних дисциплін та інформаційних технологій

**КВАЛІФІКАЦІЙНА РОБОТА**

на тему: «ПРОЄКТУВАННЯ ТА РОЗРОБКА ВЕБСАЙТУ ЗАПОРІЗЬКОЇ  
МІСЬКОЇ МОЛОДІЖНОЇ РАДИ»

Виконав:	здобувач освіти 4 курсу, групи К121-20
Спеціальність	121 Інженерія програмного забезпечення (шифр і спеціальність)
	Денис ШУТЬ (ім'я та ПРІЗВИЩЕ)
Керівник	Юлія БОРИСОВСЬКА (ім'я та ПРІЗВИЩЕ)
Рецензент	асистент кафедри програмних засобів НУ «Запорізька політехніка», к.т.н. Жанна КАМІНСЬКА (посада, вчене звання, науковий ступінь, ім'я та ПРІЗВИЩЕ)

Запоріжжя  
2024

# ЗАПОРІЗЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

ВСП «Економіко-правничий фаховий коледж ЗНУ»

Освітньо-кваліфікаційний рівень фаховий молодший бакалавр

Спеціальність 121 – Інженерія програмного забезпечення  
(шифр і назва)

## ЗАТВЕРДЖУЮ

Голова циклової комісії  
математичних дисциплін та  
інформаційних технологій

Т.М. Смолянкова  
(підпис)

“ 14 ” червня 2024 р.

## ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ

ШУТЮ Денису Володимировичу  
(ПРИЗВИЩЕ, ім'я та по- батькові)

1. Тема роботи «Проектування та розробка вебсайту Запорізької міської молодіжної ради

Керівник роботи викладач першої категорії БОРИСОВСЬКА Юлія Олександрівна  
(науковий ступінь, вчене звання, прізвище, ім'я, по батькові)

Затверджені наказом ВСПЕПФК ЗНУ від « 30 » листопада 2023 р. № 2004-с

2. Строк подання студентом роботи 01.06.2024

3. Вихідні дані до роботи 1. Постановка задачі.

2. Перелік літератури.

4. Зміст розрахунково- пояснювальної записки (перелік питань, які потрібно розробити)

1. Сучасні інформаційні системи.

2. Розробка проекту інформаційної системи.

3. Програмна реалізація інформаційної системи.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)

презентація до захисту

## 6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		Завдання видав	Завдання прийняв

7. Дата видачі завдання 01.12.2023

## КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1.	Розробка плану роботи:		
	Постановка задачі	Грудень 2023	виконано
2.	Збір вихідних даних, обробка методичних та теоретичних джерел		
		Січень 2024	виконано
3.	Розробка першого розділу:		
	Огляд інформаційних систем та технологій для їхнього створення	Лютий 2024	виконано
4.	Розробка другого розділу:		
	Розробка проєкту інформаційної системи та вибір технологій	Квітень 2024	виконано
5.	Розробка третього розділу:		
	Розробка застосунку	Травень 2024	виконано
6.	Оформлення і нормоконтроль кваліфікаційної роботи та перевірка на плагіат	Червень 2024	виконано
7.	Захист кваліфікаційної роботи	21.06.2024	виконано

Здобувач освіти \_\_\_\_\_  
(підпис)

Денис ШУТЬ \_\_\_\_\_  
(ім'я ПРІЗВИЩЕ)

Керівник роботи \_\_\_\_\_  
(підпис)

Юлія БОРИСОВСЬКА \_\_\_\_\_  
(ім'я ПРІЗВИЩЕ)

### Нормоконтроль пройдено

Нормоконтролер \_\_\_\_\_  
(підпис)

Юлія БОРИСОВСЬКА \_\_\_\_\_  
(ім'я ПРІЗВИЩЕ)

## РЕФЕРАТ

Кваліфікаційна робота: 44 сторінки, 1 таблиця, 18 ілюстрацій.

Об'єктом дослідження є основні методи та інструменти розробки вебсайту для молодіжної ради міста Запоріжжя. Предметом дослідження є розробка вебсайту для потреб молодіжної ради. Мета дослідження полягає у створенні функціональної інформаційної системи для молоді та громадян міста і країни, яка відповідає потребам молодіжної ради.

У рамках кваліфікаційної роботи були досліджені теоретичні аспекти систем управління та представлені приклади сучасних систем. У кваліфікаційній роботі викладено теоретичні відомості про конструктор сайтів Weblium, мову розробки HTML, а також методи та засоби розробки інформаційного сайту. В рамках кваліфікаційної роботи, використовуючи можливості середовища розробки Weblium, був створений та налаштований інформаційний сайт, що дозволяє користувачам легко та зручно дізнаватися новини молодіжної політики, нові можливості, зручно комунікувати з представниками молодіжної ради та слідкувати за життям в місті.

Особлива увага в роботі була приділена забезпеченню зручності навігації та інтуїтивно зрозумілому інтерфейсу. Практичне значення роботи полягає у впровадженні сучасних технологій, які допомагають користувачеві ознайомитись з діяльністю молодіжної ради міста Запоріжжя. Важливо зазначити, що застосування платформи Weblium дозволило значно скоротити час розробки та знизити витрати на створення сайту.

В процесі розробки були використані такі технології та інструменти: Weblium та мова програмування HTML. Весь процес розробки відбувався з використанням платформи Weblium, а розроблений вебсайт пристосований для використання на вебсайті молодіжної ради.

ІНФОРМАЦІЙНІ СИСТЕМИ, WEBLIUM, HTML, ВЕБСАЙТ.

## SUMMARY

Qualification Work: 44 pages, 1 table, 18 illustrations.

The object of the study is the main methods and tools for developing information systems for the Youth Council of the city of Zaporizhzhia. The subject of the study is the development of an information system specifically for the needs of the Youth Council. The purpose of the study is to create a functional information system for informing the youth and citizens of the city and country that meets the needs of the Youth Council.

Within the framework of the qualification work, the theoretical aspects of management systems were studied and examples of modern systems were presented. The qualification work details theoretical information about the Weblium site builder, the HTML development language, as well as methods and tools for developing an information site. Based on the theoretical knowledge obtained, a website was created to inform the youth and others. Within the qualification work, using the capabilities of the Weblium development environment, an information site was created and configured, allowing users to easily and conveniently learn about youth policy news, new opportunities, communicate comfortably with representatives of the Youth Council, and follow the life in the city.

Special attention was paid to ensuring easy navigation and an intuitive interface, which significantly simplifies the process of selecting and purchasing goods. The practical significance of the work lies in the implementation of modern technologies that help users quickly find the desired product and make a purchase without extra effort. It is important to note that the use of the Weblium platform significantly reduced the development time and costs associated with creating the site, which is an important factor for a non-profit organization.

During the development process, the following technologies and tools were used: Weblium and the HTML programming language. The entire development process took place using the Weblium platform, and the developed application is adapted for use on the Youth Council's website.

INFORMATION SYSTEMS, WEBLIUM, HTML, WEBSITE.

## ЗМІСТ

Завдання на кваліфікаційну роботу студенту .....	2
Реферат .....	4
Summary .....	5
Вступ.....	8
1 Сучасні інформаційні системи.....	10
1.1 Поняття інформаційної системи.....	10
1.2 Призначення інформаційних систем.....	12
1.3 Класифікація інформаційних систем .....	13
1.4 Складові інформаційної системи.....	14
1.5 Засоби створення інформаційної системи.....	15
2 Розробка проєкту інформаційної системи.....	17
2.1 Технічне завдання .....	17
2.2 Етапи створення інформаційної системи .....	18
2.3 Діаграма прецедентів .....	20
2.4 Проєктування інтерфейсу.....	22
3 Програмна реалізація інформаційної системи .....	27
3.1 База даних застосунку.....	27
3.2 Створення Rest-full сервера.....	29
3.3 Створення клієнта .....	31
Висновки .....	43
Перелік використаних джерел .....	44

## ВСТУП

Сьогоднішній світ неможливо уявити без інформаційних технологій. Вони проникають у всі сфери нашого життя, включаючи молодь та молодіжно політику.

Мета кваліфікаційної роботи полягає в розробці інформаційного сайту Запорізької міської молодіжної ради, а саме для того, щоб зручно було відслідкувати діяльність молодіжних депутатів та задля зв'язку з ними. Для досягнення поставленої мети, були сформульовані такі завдання:

1. Визначити поняття інформаційної системи та переваги її використання для інформування та зв'язку.
2. Проаналізувати наявні інформаційні системи зі схожою тематикою.
3. Описати інструменти, за допомогою яких така система може бути реалізована.
4. Розробити проєкт інформаційної системи для Запорізької міської молодіжної ради.
5. Зробити програмну реалізацію інформаційної системи Запорізької міської молодіжної ради.

Об'єкт дослідження – основні методи та засоби розробки інформаційних систем. Предмет дослідження – інформаційна система для Запорізької міської молодіжної ради. Розробка програмного застосунку відбувалася в інтегрованому середовищі розробки Weblium.

Кваліфікаційна робота складається зі вступу, трьох розділів, висновків та переліку використаних джерел. Перший розділ містить основні відомості про інформаційні системи, а також технології, які можуть бути використані для реалізації сучасної інформаційної системи для обміну інформацією та комунікації. Другий розділ містить технічне завдання, технічні вимоги до



програмного продукту, діаграму варіантів використання, ER-діаграму бази даних, опис структур в базі даних, етапи розробки інформаційної системи та макети застосунку. Третій розділ містить особливості реалізації кожної частини застосунку – бази даних, сервера та клієнта, а також надана схема API та опис застосунку.

# 1 СУЧАСНІ ІНФОРМАЦІЙНІ СИСТЕМИ

## 1.1 Поняття інформаційної системи

Інформаційна система – це комплексний механізм для зберігання, пошуку та обробки інформації. Вона складається з різних інструментів та процесів, які допомагають взаємодіяти з даними. Залежно від рівня автоматизації, інформаційні системи можуть бути ручними, автоматизованими або автоматичними [1].

Автоматизована інформаційна система – це сукупність апаратних та програмних засобів, які дозволяють зберігати, організовувати пошук і обробку інформації. Вони забезпечують автоматизацію роботи з даними у різних сферах людської діяльності [2].

Інформаційна система – це комплексний механізм для зберігання, пошуку та обробки інформації. Вона складається з різних інструментів та процесів, які допомагають взаємодіяти з даними. Залежно від рівня автоматизації, інформаційні системи можуть бути ручними, автоматизованими або автоматичними.

На теперішній час можна інформаційні системи можна поділити на такі типи [2]:

1. «Ручні» інформаційні системи. Такі системи використовують паперові документи та фізичні засоби зберігання даних. Всі процеси, включаючи обробку та пошук інформації, виконуються вручну.

2. Автоматизовані інформаційні системи. В цих системах для обробки інформації залучають комп'ютери та програмне забезпечення для зберігання та обробки інформації. Люди все ще беруть участь у деяких етапах, але основна робота виконується автоматизовано.

3. Автоматичні інформаційні системи. Це системи, які є повністю автоматизованими, в них всі процеси виконуються без втручання людини. Такі системи використовують алгоритми штучного інтелекту, машинного навчання та інші передові технології для автономної роботи з даними.

Переваги використання інформаційних систем :

- підвищення ефективності – автоматизація процесів дозволяє значно зменшити час на виконання завдань та підвищити продуктивність роботи з інформацією;

- зменшення помилок – використання програмного забезпечення знижує ризик людських помилок при обробці та зберіганні інформації;

- зручний доступ до даних – інформаційні системи забезпечують швидкий доступ до необхідної інформації, що підвищує оперативність прийняття рішень та опрацювання даних;

- покращення комунікації – інформаційні системи сприяють покращенню обміну інформацією в усіх сферах суспільного життя;

- аналітика та прогнозування – використання інформаційних систем дозволяє проводити детальний аналіз даних та будувати прогнози на основі отриманої інформації.

Інформаційні системи є невід'ємною частиною сучасного світу, вони охоплюють всі сфери людської діяльності – від бізнесу та медицини до освіти та розваг. Розуміння їхньої структури та функцій є ключовим для ефективного використання цих систем у різних галузях.

## 1.2 Призначення інформаційних систем

Інформаційні системи в міській молодіжній раді міста Запоріжжя мають на меті покращити зв'язок між молодими людьми та органами місцевого самоврядування. Основні завдання цих систем включають [2]:

- моніторинг: інформаційні системи допомагають збирати, аналізувати та відображати дані про потреби та проблеми молоді. наприклад, вони можуть слідувати за статистикою щодо зайнятості, освіти, життєвих умов та інших аспектів, що стосуються молодіжної політики;

- комунікація: інформаційні системи надають засоби для спілкування між молодими людьми та міськими органами. це може включати веб-портали, додатки для смартфонів, соціальні мережі та інші канали зв'язку;

- участь: інформаційні системи сприяють активній участі молоді в прийнятті рішень. вони можуть допомагати організовувати опитування, голосування та інші форми збору думок та пропозицій;

- інформаційна підтримка: інформаційні системи надають доступ до актуальної інформації про різні можливості, програми та заходи для молоді, що можуть бути корисні для їхнього розвитку та самореалізації;

- моніторинг виконання рішень: інформаційні системи можуть допомагати контролювати виконання рішень, прийнятих в молодіжній раді, та відстежувати їхній вплив на життя молоді;

- звітність: інформаційні системи допомагають збирати та аналізувати дані про роботу молодіжної ради та надавати звіти громаді та органам влади.

Загалом, інформаційні системи для комунікації в молодіжній раді міста Запоріжжя сприяють покращенню якості молодіжної політики та збільшенню впливу молоді на прийняття рішень.

### 1.3 Класифікація інформаційних систем

Інформаційні системи можна класифікувати за різними критеріями, такими як функціональне призначення, спосіб взаємодії з користувачами, область застосування тощо. Найбільш загальна класифікація включає чотири типи інформаційних систем [2]:

- операційні інформаційні системи: ці системи призначені для забезпечення щоденних операційних потреб підприємства або організації. вони включають системи обліку, контролю, управління запасами, виробництва тощо;

- підтримка прийняття рішень: ці системи допомагають керівництву в прийнятті стратегічних рішень, аналізуючи великі обсяги даних. вони включають системи підтримки прийняття рішень, системи бізнес-аналітики, системи управління ресурсами підприємства тощо;

- електронна комерція: ці системи забезпечують проведення електронних торгів та взаємодію між бізнес-партнерами через мережу інтернет. вони включають системи електронної комерції, системи управління відносинами з клієнтами, системи управління ланцюжками постачання тощо;

- інформаційні системи для підтримки виробництва: ці системи використовуються для автоматизації інженерних та технологічних процесів, управління виробництвом та контролю якості. вони включають системи автоматизації виробництва, системи комп'ютерного-інтегрованого проектування, системи управління якістю тощо.

## 1.4 Складові інформаційної системи

Основні компоненти інформаційної системи включають [2]:

1. Апаратні засоби, а саме: комп'ютери, сервери, мережеве обладнання та інші пристрої, що забезпечують фізичну основу для функціонування інформаційної системи.

2. Програмні засоби – програмне забезпечення, включаючи операційні системи, бази даних, прикладні програми та інші інструменти, що забезпечують збереження, обробку та передачу інформації.

3. Дані, тобто всі форми інформації, які зберігаються та обробляються в інформаційній системі. Це можуть бути текстові файли, зображення, відео, аудіо, бази даних тощо.

4. Люди, а саме: користувачі системи, включаючи кінцевих користувачів, адміністраторів, аналітиків, розробників та інших осіб, що взаємодіють з системою.

5. Процедури – це сукупність правил та інструкцій, які визначають, як саме повинна функціонувати система. Це можуть бути бізнес-процеси, методики збирання та обробки даних, політики безпеки тощо.

6. Мережеві ресурси такі, як Інтернет та інші мережі, які забезпечують зв'язок між компонентами інформаційної системи та доступ до зовнішніх ресурсів.

7. Процеси, це послідовність дій, які виконуються в ІС для обробки даних. Процеси можуть бути автоматизованими (наприклад, робота програми для обробки даних) або ручними (наприклад, введення даних вручну).

8. Безпека, а саме – це забезпечення захисту від несанкціонованого доступу до інформації та збереження конфіденційності, цілісності та доступності даних. Безпека може бути забезпечена за допомогою різних методів, таких як шифрування, аутентифікація користувачів, відповідальність за доступ до даних тощо.

## 1.5 Засоби створення інформаційної системи

Для створення інформаційної системи міської молодіжної ради міста Запоріжжя можуть бути використані різні засоби та технології. Зазначимо деякі з них [3].

Веб-портал. Створення спеціального веб-порталу для молодіжної ради, де можна розміщувати інформацію про роботу ради, програми та події для молоді. Веб-портал може містити інтерактивні функції, такі як опитування та форуми для обговорення питань молодіжної політики.

Мобільний додаток. Розробка мобільного додатка для молодіжної ради, який дозволить молоді швидко отримувати інформацію про події та програми, а також брати участь у голосуваннях та опитуваннях.

Система електронного документообігу. Впровадження системи електронного документообігу для забезпечення ефективного обміну документами та інформацією між учасниками молодіжної ради.

Соціальні мережі. Використання соціальних мереж для залучення молоді до роботи молодіжної ради та комунікації з ними. Соціальні мережі можуть бути використані як канал для розміщення інформації, організації дискусій та голосувань.

Система електронного голосування. Впровадження системи електронного голосування для прийняття рішень та збору думок молоді.

Аналітичні та звітні інструменти. Використання аналітичних та звітних інструментів для аналізу даних про роботу молодіжної ради та її вплив на життя молоді.

Електронна пошта та SMS-повідомлення. Використання електронної пошти та SMS-повідомлень для сповіщення молоді про важливі події та рішення.

Конференції та вебінари. Організація онлайн-конференцій та вебінарів для обговорення питань молодіжної політики та взаємодії з молоддю. Wix: Wix є

одним з найпопулярніших конструкторів веб-сайтів. Він має велику бібліотеку шаблонів, які можна налаштувати під власні потреби. Wix також має інтуїтивний інтерфейс для редагування сторінок та додавання функцій, таких як контактні форми або галереї.

На ринку існує багато популярних конструкторів вебсайтів, кожен з яких має свої особливості та переваги [4]:

- Squarespace – це інший популярний конструктор веб-сайтів, який пропонує елегантні та сучасні шаблони. Він також має інтуїтивний інтерфейс та ряд інструментів для редагування та додавання функцій.

- WordPress.com – це хостингова платформа, яка також має свій власний конструктор веб-сайтів. Вона має велику кількість шаблонів та різноманітні плагіни для розширення функціональності сайту.

- Webflow – це інструмент для створення веб-сайтів з візуальним редактором та можливістю роботи зі структурою сайту на рівні коду.

- Google Sites – це безкоштовний інструмент для створення веб-сайтів, який пропонує простий інтерфейс та інтеграцію з іншими сервісами Google.

- Weblium – пропонує зручний інтерфейс для створення веб-сайтів без необхідності знань кодування. Він має багато шаблонів та інструментів для налаштування сторінок, додавання функцій та організації контенту. Крім того, Weblium пропонує оптимізацію для мобільних пристроїв та інші корисні функції для створення сучасних та ефективних веб-сайтів.

Ці платформи дозволяють користувачам створювати професійні вебсайти з мінімальними зусиллями, забезпечуючи широкий спектр інструментів та можливостей для налаштування та оптимізації.



## **2 РОЗРОБКА ПРОЄКТУ ІНФОРМАЦІЙНОЇ СИСТЕМИ**

### **2.1 Технічне завдання**

#### **2.1.1 Найменування і область застосування**

Програмний продукт, що розробляється, отримує найменування: «Вебпортал для Запорізької міської молодіжної ради».

Програма призначена для моніторингу, комунікації та доступу до інформації для молоді міста Запоріжжя.

#### **2.1.2 Призначення розробки**

Даний проєкт призначений для вирішення наступних завдань:

- моніторинг заходів та новин;
- комунікація з молодіжними лідерами;
- відображення повної інформації про певну особу та її контакти;
- створення та надання пропозиції по власним проєктам.

#### **2.1.3 Технічні вимоги до програмного продукту**

Зазначимо технічні вимоги до розробки:

- забезпечення навігації по системі без перезавантаження сторінок, отримання та відправлення даних з сервера;

- при архівації категорій забезпечити архівацію всіх даних у цій категорії;
- при видаленні категорій забезпечити видалення всіх даних у цій категорії;
- можливість редагування назв категорій та повідомлень, а також їх архівація при перегляді;
- зміна відображення зображень між актуальними та архівованими. додавання інформації про дату архівації та користувача, який її виконав;
- редагування повної інформації про категорії та повідомлення;
- стилізація всіх сторінок сайту в єдиному стилі відповідно до принципів матеріального дизайну.

## **2.2 Етапи створення інформаційної системи**

Основними етапами розробки інформаційної системи Міської молодіжної ради є [5]:

1. Визначення цілей створення системи: цей етап передбачає ретельний аналіз потреб та очікувань користувачів системи, а також визначення основних цілей та завдань, які система повинна вирішувати. Складання технічного завдання та вимог до додатку є важливою складовою цього етапу, оскільки вони визначають напрямки подальшої роботи.

2. Створення макетів сторінок застосунку: на цьому етапі розробляються концепції та дизайн сторінок, що включають в себе графічні елементи, панелі інструментів та інші складові частини інтерфейсу, які допоможуть забезпечити зручну та ефективну взаємодію з користувачем.

3. Проектування бази даних: цей етап передбачає розробку структури бази даних, включаючи визначення таблиць, полів, зв'язків між ними та інших

важливих аспектів, що дозволяють ефективно зберігати та організувати інформацію.

4. Вибір технологій для реалізації застосунку: на цьому етапі важливо визначити набір інструментів, мов програмування та фреймворків, які найбільш підходять для реалізації функціональності застосунку з урахуванням його потреб та вимог.

5. Реалізація серверної та клієнтської частин застосунку: цей етап передбачає фактичну розробку програмного забезпечення, включаючи реалізацію серверної та клієнтської частин застосунку згідно з визначеними технічними вимогами.

6. Тестування застосунку: на цьому етапі проводиться обширне тестування розробленого застосунку з метою виявлення та виправлення можливих помилок, а також перевірка відповідності вимогам та очікуванням користувачів.

7. Деплой застосунку: після успішного завершення тестування та виправлення всіх виявлених недоліків, застосунок готується до розгортання (деплою) на віддаленому сервері, де він стає доступним для користувачів. Зважаючи на важливість кожного етапу та необхідність докладних підходів, процес розробки автоматизованої інформаційної системи для Міської молодіжної ради включає такі проміжні етапи:

8. Аналіз потреб користувачів та дослідження ринку: перед складанням технічного завдання та визначенням цілей системи важливо провести аналіз потреб молодіжної громади, її проблем та очікувань. Дослідження ринку також допоможе виявити конкурентні переваги та знайти оптимальні рішення для реалізації системи.

9. Спілкування з зацікавленими сторонами: цей етап передбачає активне спілкування з потенційними користувачами системи, стейкхолдерами та іншими зацікавленими сторонами для збору відгуків, пропозицій та врахування їхніх потреб у процесі розробки.

10. Проектування користувацького досвіду (UX/UI): на цьому етапі розробляються детальні сценарії взаємодії користувачів з системою, а також створюються макети інтерфейсу з урахуванням принципів зручності та естетики.

11. Архітектурне проектування системи: перед фактичною розробкою програмного забезпечення важливо розробити архітектуру системи, включаючи розподіл функціональності між різними компонентами, вибір технологій та побудову взаємозв'язків між ними.

12. Створення технічної документації: цей етап включає підготовку документації, яка описує архітектуру системи, методи реалізації, особливості інтеграції та інші технічні аспекти, що сприяють зрозумінню та підтримці проекту.

13. Розробка прототипу: перед фактичною реалізацією системи може бути корисно створити прототип або MVP (Minimum Viable Product), який дозволить швидко перевірити концепції та збирати відгуки від користувачів.

### **2.3 Діаграма прецедентів**

На сьогодні для цілей моделювання предметної області на ринку програмних продуктів представлений широкий спектр CASE-засобів. Найбільш популярними CASE-засобами є RationalRose, BPwin, Silverrun, ProcessAnalyst. Моделювання предметної області в цих засобах має багато спільного. Проте важливим є комплексність підходу і використання єдиної уніфікованої нотації, не тільки на етапі моделювання предметної області, але і на наступних етапах розробки програмної системи, як це має місце в CASE RationalRose [6].

Діаграми прецедентів складають модель прецедентів (варіантів використання, use-case). Прецедент – це функціональність системи, що дозволяє користувачеві отримати якийсь значущий для нього, відчутний і вимірний

результат. Кожен прецедент відповідає окремому сервісу, що надається моделюючою системою у відповідь на запит користувача (див. табл. 2.1). Визначає спосіб використання цієї системи. Варіанти використання найчастіше застосовуються для специфікації зовнішніх вимог до проєктованої системи або для специфікації функціонального поведінки вже існуючої системи [6].

Таблиця 2.1 – Таблиця акторів та прецедентів вебзастосунку

Актор	Прецедент
Керівник сайтом	Зайти на сайт Ввійти в панель адміністратору Подивитися всі сторінки Створювати, редагувати та видаляти новини Створювати, редагувати та архівувати категорії, позиції Подивитися аналітику
Менеджер	Зайти на сайт Ввійти в панель адміністратору Подивитися всі сторінки Створювати, редагувати Подивитися аналітику

На діаграмі прецедентів (див. рис. 2.1) відображенні основні принципи роботи вебресурсу.

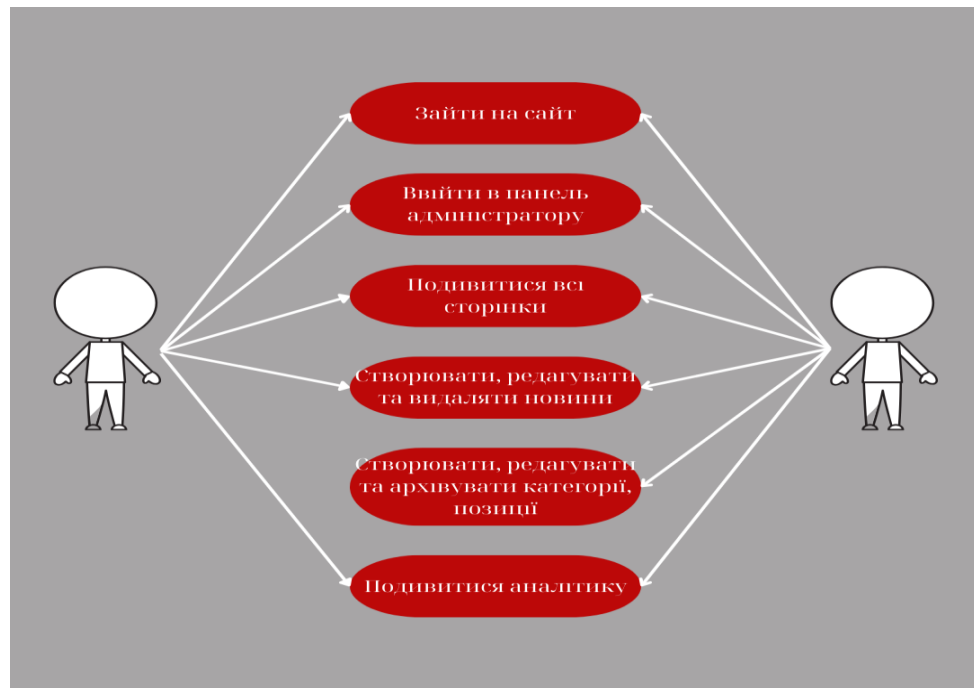


Рисунок 2.1 – Діаграма прецедентів

## 2.4 Проектування інтерфейсу

На основі поставленого технічного завдання та визначених потреб користувачів Міської молодіжної ради було спроектовано інтерфейс застосунку, який має за мету забезпечити зручний та ефективний доступ до інформації та функціональності системи. Важливим аспектом розробки інтерфейсу було забезпечення його легкості використання, сприйняття та інтуїтивного розуміння для різних категорій користувачів.

Розроблений макет містить такі сторінки:

1. Сторінка новин (див. рис. 2.2).

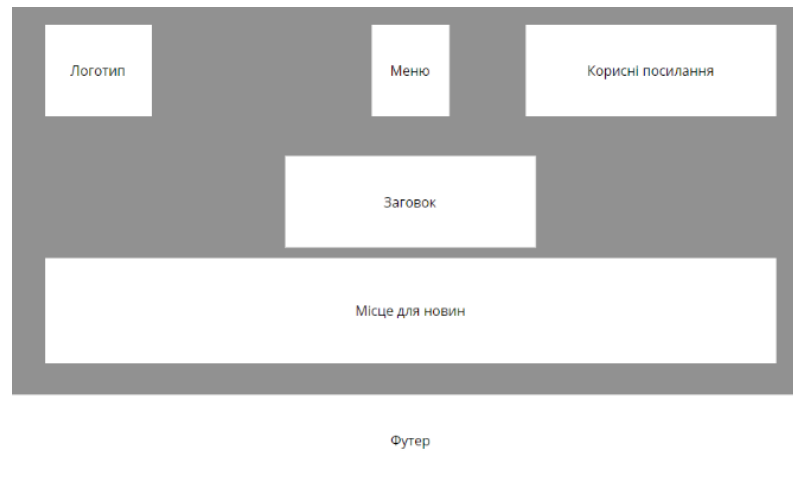


Рисунок 2.2 – Макет сторінки Новини

Ця сторінка призначена для публікації останніх новин та подій, що стосуються роботи Міської молодіжної ради. Користувачі зможуть отримувати актуальну інформацію про проведені заходи, проекти та інші активності.

2. Про нас (див. рис. 2.3).

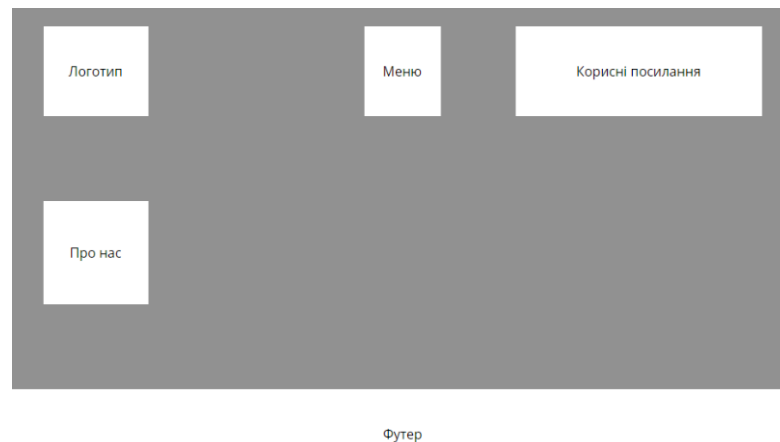


Рисунок 2.3 – Макет сторінки Про нас

Сторінка "Про нас" містить загальну інформацію про Міську Молодіжну раду, її місію, цілі та завдання. Тут також може бути наведена історія створення та розвитку ради, інформація про її структуру та робочі органи.

### 3. Виконавчий комітет (див. рис. 2.4).

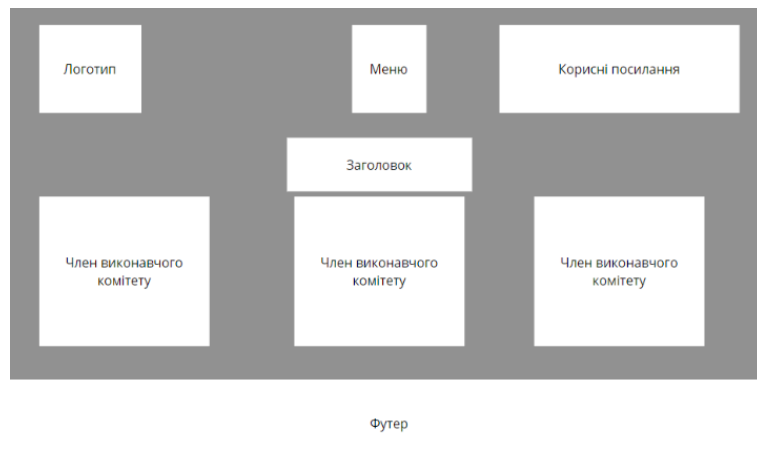


Рисунок 2.4 – Макет сторінки Виконавчий комітет

На цій сторінці користувачі зможуть знайти інформацію про склад виконавчого комітету Міської молодіжної ради, включаючи імена членів комітету, їхні посади та контактні дані.

### 4. Члени молодіжної ради (див. рис. 2.5).

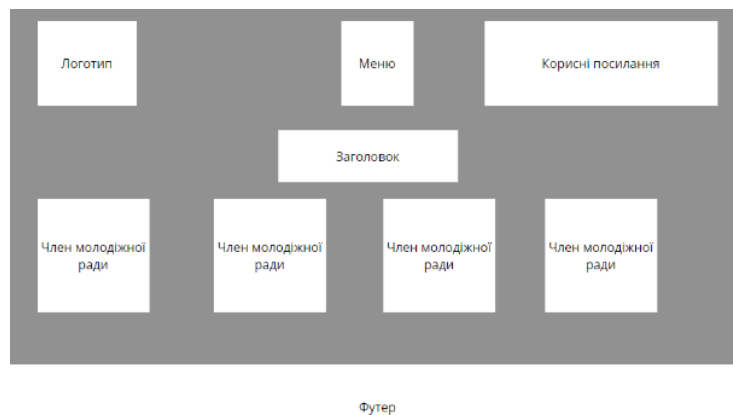


Рисунок 2.5 – Макет сторінки Члени молодіжної ради



Ця сторінка містить профілі та інформацію про членів Міської молодіжної ради. Користувачі можуть дізнатися про біографічні дані, обрані посади та ролі у раді.

5. Контактні дані (див. рис. 2.6).

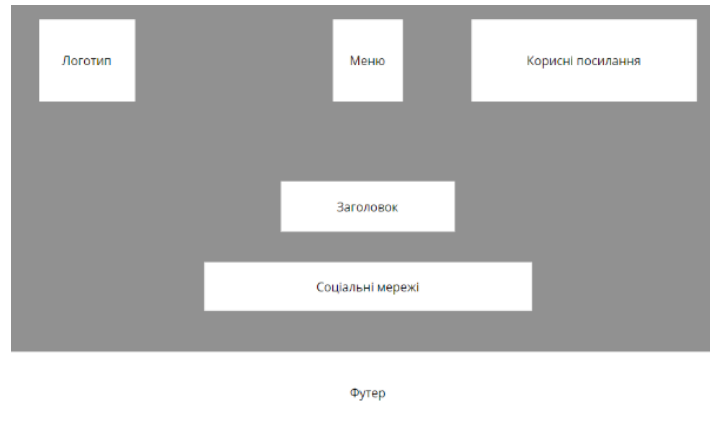


Рисунок 2.6 – Макет сторінки Контактні дані

Сторінка "Контактні дані" містить інформацію про способи зв'язку з Міською Молодіжною радою. Сюди включаються адреса, телефонні номери, електронна пошта та інші контактні дані для звернення та отримання додаткової інформації.

6. Головна сторінка (див. рис. 2.7).

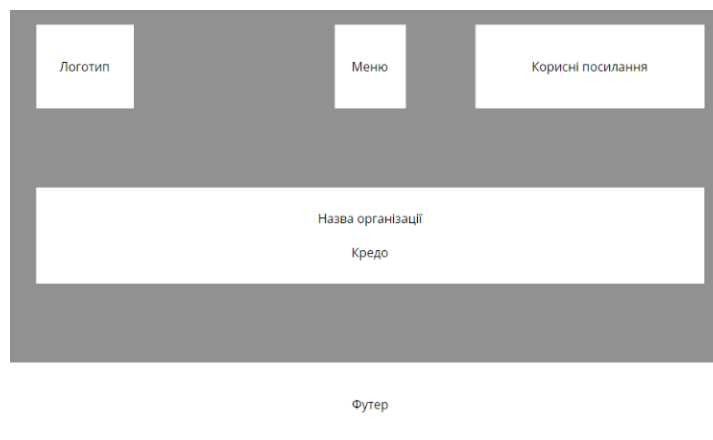


Рисунок 2.7 – Макет сторінки Головна

Головна сторінка є центральною точкою входу до системи та містить загальну інформацію та швидкі посилання на найважливіші розділи, включаючи новини, профілі членів, інформацію про виконавчий комітет та інші ключові розділи.

## 3 ПРОГРАМНА РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ

На етапі проектування інформаційної системи було вирішено створити сайт міської молодіжної ради засобами середовища розробки Weblium.

### 3.1 База даних застосунку

Weblium використовує різні типи баз даних, включаючи реляційні та NoSQL, забезпечує масштабованість і безпеку через хмарні рішення, кешування, шифрування, контроль доступу, а також підтримує автоматичні резервні копії та відновлення після аварій для ефективного управління даними [7]. Зазначимо основні аспекти використання баз даних в середовищі проектування вебсайтів Weblium.

#### 1. Типи баз даних:

- Реляційні бази даних. Weblium може використовувати реляційні бази даних, такі як MySQL або PostgreSQL, для зберігання структурованих даних. Реляційні бази даних дозволяють організувати дані в таблиці, де інформація пов'язана між собою через відносини.

- NoSQL бази даних. Додаток також може використовувати NoSQL бази даних, такі як MongoDB, для зберігання неструктурованих або напівструктурованих даних. Це дозволяє забезпечити гнучкість у зберіганні великих обсягів різномірних даних.

#### 2. Архітектура та масштабованість:

- Хмарні бази даних. Weblium ймовірно використовує хмарні бази даних для забезпечення високої доступності та масштабованості. Використання

хмарних рішень, таких як Amazon RDS або Google Cloud SQL, дозволяє динамічно масштабувати ресурси баз даних відповідно до потреб користувачів.

- Кешування – для покращення швидкості доступу до даних додаток може використовувати системи кешування, такі як Redis або Memcached. Це знижує навантаження на основну базу даних і забезпечує швидший доступ до часто запитуваних даних.

### 3. Забезпечення безпеки:

- Шифрування даних. Важливим аспектом є забезпечення безпеки даних користувачів. Це може включати шифрування даних як у стані спокою, так і під час передачі. Для цього можуть використовуватися протоколи SSL/TLS для захищеного з'єднання.

- Контроль доступу. Встановлення ролей і прав доступу до баз даних дозволяє контролювати, хто має доступ до яких даних. Це допомагає забезпечити конфіденційність та цілісність інформації.

### 4. Резервне копіювання та відновлення:

- Автоматичні резервні копії. Регулярне резервне копіювання баз даних є критично важливим для запобігання втраті даних у випадку збою системи. Це може включати автоматичні резервні копії з можливістю відновлення до певного моменту часу.

- План відновлення після аварій. Механізми відновлення після аварій дозволяють швидко відновити роботу системи з мінімальними втратами даних у разі критичних збоїв.

### Переваги використання баз даних у Weblium

- Ефективне управління даними. Бази даних дозволяють ефективно зберігати та організувати великі обсяги даних, забезпечуючи їх швидкий доступ і обробку.

- Гнучкість та масштабованість. Використання сучасних технологій баз даних дозволяє масштабувати систему відповідно до зростаючих потреб

користувачів. Безпека: Інтеграція передових методів безпеки допомагає захистити дані користувачів від несанкціонованого доступу та втрат.

### 3.2 Створення Rest-full сервера

RESTful сервер є важливим компонентом архітектури сучасних веб-застосунків, таких як платформа для створення веб-сайтів Weblium. REST (Representational State Transfer) – це архітектурний стиль, який використовує HTTP-протокол для зв'язку між клієнтом і сервером [8]. Нижче описано, як RESTful сервер працює у контексті Weblium і які його основні характеристики.

#### 1. Ресурси.

Ресурсно-орієнтована архітектура: у RESTful сервері кожен елемент даних представляється як ресурс. У Weblium це можуть бути ресурси, такі як "сайт", "сторінка", "користувач", "шаблон" тощо. Кожен ресурс ідентифікується унікальним URL.

#### 2. Методи HTTP.

- GET. Для отримання інформації про ресурси. Наприклад, запит GET до `/api/sites` може повернути список усіх сайтів користувача.
- POST. Для створення нових ресурсів. Наприклад, POST-запит до `/api/sites` може створити новий сайт.
- PUT. Для оновлення існуючих ресурсів. Наприклад, PUT-запит до `/api/sites/{site_id}` оновлює дані сайту з конкретним ID.
- DELETE. Для видалення ресурсів. Наприклад, DELETE-запит до `/api/sites/{site_id}` видаляє сайт з вказаним ID.

#### 3. Статуси HTTP.

- 200 OK: запит успішно оброблений.
- 201 Created: ресурс успішно створений.

- 204 No Content: запит успішно оброблений, але відповідь не містить контенту.

- 400 Bad Request: невірний запит.
- 401 Unauthorized: необхідна авторизація.
- 404 Not Found: ресурс не знайдено.
- 500 Internal Server Error: помилка на сервері.

#### 4. Безпека.

- Аутентифікація та авторизація. RESTful сервер Weblium використовує методи аутентифікації (наприклад, OAuth 2.0) для забезпечення доступу лише авторизованих користувачів до ресурсів.

- SSL/TLS. Використання HTTPS для захищеного з'єднання між клієнтом і сервером, що забезпечує шифрування даних під час передачі.

#### 5. Формат даних.

JSON – найпоширеніший формат для обміну даними між клієнтом і сервером у Weblium. JSON є легким для читання та обробки форматом, що забезпечує швидку передачу даних.

#### 6. Масштабованість і продуктивність.

- Кешування: використання кешування для підвищення швидкості обробки запитів та зменшення навантаження на сервер. HTTP-заголовки, такі як ETag і Cache-Control, можуть використовуватись для управління кешуванням.

- Масштабованість: RESTful архітектура дозволяє легко масштабувати серверні ресурси для обробки зростаючого числа запитів.

#### Переваги використання RESTful сервера в Weblium:

- Легкість інтеграції: стандартні методи і формати даних дозволяють легко інтегрувати RESTful сервер Weblium з іншими сервісами та застосунками.

- Гнучкість: RESTful архітектура дозволяє легко розширювати функціонал сервера та додавати нові ресурси та методи без значних змін у коді.

- Зрозумілість: чітка організація ресурсів і використання стандартних HTTP-методів роблять API зрозумілим для розробників.
- Підтримка масштабованості: RESTful сервери можуть обробляти великий обсяг запитів, забезпечуючи високу продуктивність і доступність.

### 3.3 Створення клієнта

#### 3.3.1 Налаштування застосунку

На платформі для створення вебсайтів Weblium процес налаштування клієнтського застосунку включає кілька ключових етапів. Нижче наведено детальний опис цих етапів для успішного налаштування клієнтського застосунку на платформі Weblium.

1. Реєстрація:
    - почніть з реєстрації на платформі weblium:
    - створіть обліковий запис, вказавши основну інформацію, таку як ім'я, електронна адреса та пароль.
  2. Вхід у систему: після реєстрації увійдіть до системи, використовуючи свої облікові дані.
  3. Створення нового проєкту
    - стартовий екран: після входу у систему ви потрапите на стартовий екран, де можете створити новий проєкт;
    - вибір шаблону: Weblium пропонує широкий вибір шаблонів для різних типів сайтів, оберіть шаблон, що відповідає вашим потребам.
    - назва проєкту: після вибору шаблону задайте назву своєму проєкту.
- Етапи налаштування дизайну сайту включають:

1. Редагування шаблону: налаштуйте обраний шаблон під свої потреби, змінюючи текст, зображення, кольори та інші елементи дизайну.
2. Додавання та редагування блоків: додавайте нові блоки на сторінки сайту, такі як галереї, форми зворотного зв'язку, кнопки соціальних мереж тощо.
3. Перегляд в реальному часі: переглядайте зміни в реальному часі, щоб оцінити вигляд сайту.

#### Налаштування функціоналу:

1. Інтеграція з іншими сервісами: підтримка інтеграції з різними сервісами, такими як Google Analytics, CRM-системи, сервіси email-маркетингу.
2. SEO-налаштування: налаштуйте параметри для пошукової оптимізації (SEO), такі як метатеги, заголовки, описи та ключові слова для кожної сторінки сайту.
3. Форми зворотного зв'язку: додавайте та налаштовуйте форми зворотного зв'язку для відвідувачів сайту.

#### Налаштування домену та публікація сайту:

1. Вибір домену: підключіть власний домен або скористайтесь доменом, запропонованим Weblium.
2. SSL сертифікат: налаштуйте SSL сертифікат для безпечного з'єднання через HTTPS.
3. Публікація сайту: опублікуйте свій сайт, щоб зробити його доступним для відвідувачів.

#### Переваги налаштування застосунку на платформі Weblium:

- інтуїтивно зрозумілий інтерфейс – простий та зручний інтерфейс робить процес налаштування сайту легким навіть для користувачів без технічних знань;
- гнучкість та адаптивність – платформа дозволяє створювати адаптивні сайти, які виглядають добре на будь-яких пристроях;



- широкі можливості кастомізації – вибір шаблонів та блоків дає можливість створювати унікальні сайти, що відповідають індивідуальним потребам користувача;
- безпека та підтримка – Weblium забезпечує високий рівень безпеки даних користувачів та надає технічну підтримку.

### 3.3.2 Розробка сторінок застосунку

Розробка сторінок застосунку на платформі Weblium.

Розробка сторінок застосунку на платформі Weblium є багатоетапним процесом, що включає створення нового проекту, налаштування структури та контенту, інтеграцію з іншими сервісами, тестування та публікацію сайту. Weblium надає широкий спектр інструментів та можливостей для створення професійних веб-сайтів з унікальним дизайном та функціоналом. Розглянемо детальніше кожен етап цього процесу.

Крок 1: Створення нового проекту

1. Реєстрація та вхід у систему (див. рис. 3.1).

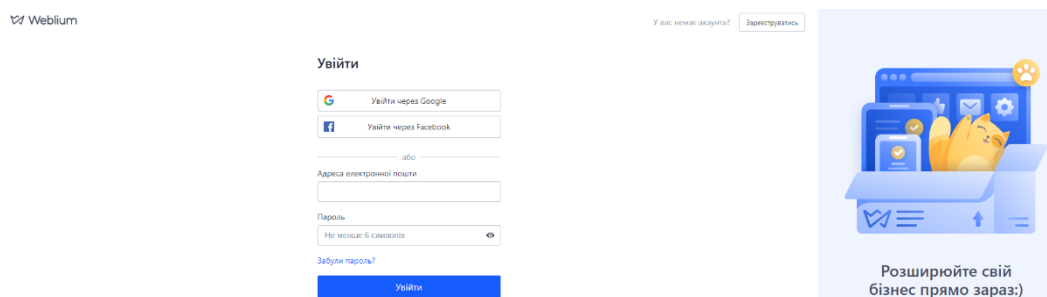


Рисунок 3.1 – Вхід в систему

- Реєстрація: перший крок — це реєстрація на платформі Weblium. Необхідно створити обліковий запис, вказавши основну інформацію, таку як ім'я,

електронну адресу та пароль. Після успішної реєстрації користувач отримує доступ до всіх можливостей платформи.

- Вхід у систему: після реєстрації можна увійти до системи, використовуючи свої облікові дані. Вхід відкриває доступ до створення та редагування проектів, а також до інших функцій платформи.

## 2. Стартовий екран (див. рис. 3.2).

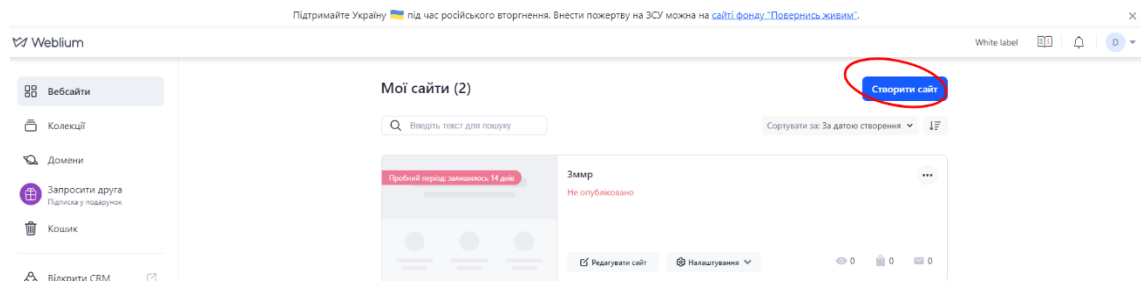


Рисунок 3.2 – Стартова сторінка

- Після входу у систему користувач потрапляє на стартовий екран. Тут можна створити новий проект, скориставшись кнопкою "Створити новий сайт". Цей етап дозволяє розпочати роботу над новим сайтом з чистого аркуша або за допомогою готових шаблонів.

## 3. Вибір шаблону (див. рис. 3.3).

- Weblium пропонує широкий вибір шаблонів для різних типів сайтів, таких як бізнес-сайти, портфоліо, блоги, інтернет-магазини тощо. Користувач може обрати шаблон, який найбільше відповідає його потребам та задуму сайту. Кожен шаблон містить попередньо налаштовані блоки та дизайни, які можна змінювати та налаштовувати.

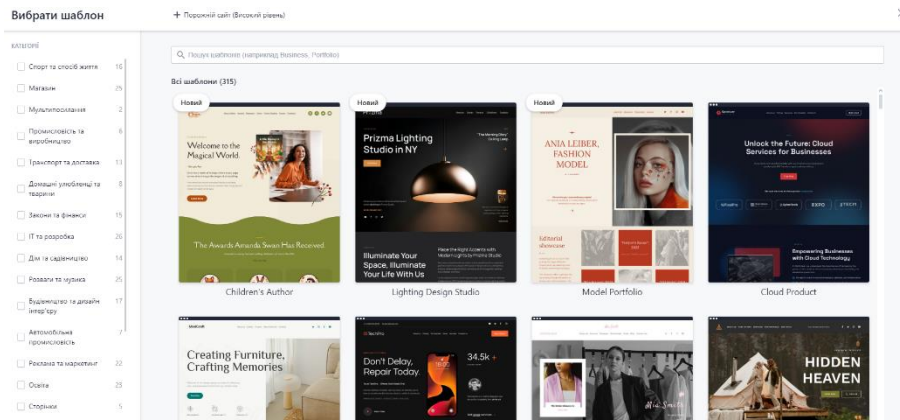


Рисунок 3.3 – Шаблони

#### 4. Назва проекту.

- Після вибору шаблону користувач задає назву своєму проекту. Це допоможе організувати роботу та легко ідентифікувати проект серед інших. Назва проекту може відображати суть або тему сайту.

#### Крок 2: Налаштування структури сайту

##### 1. Головна сторінка (див. рис. 3.4).

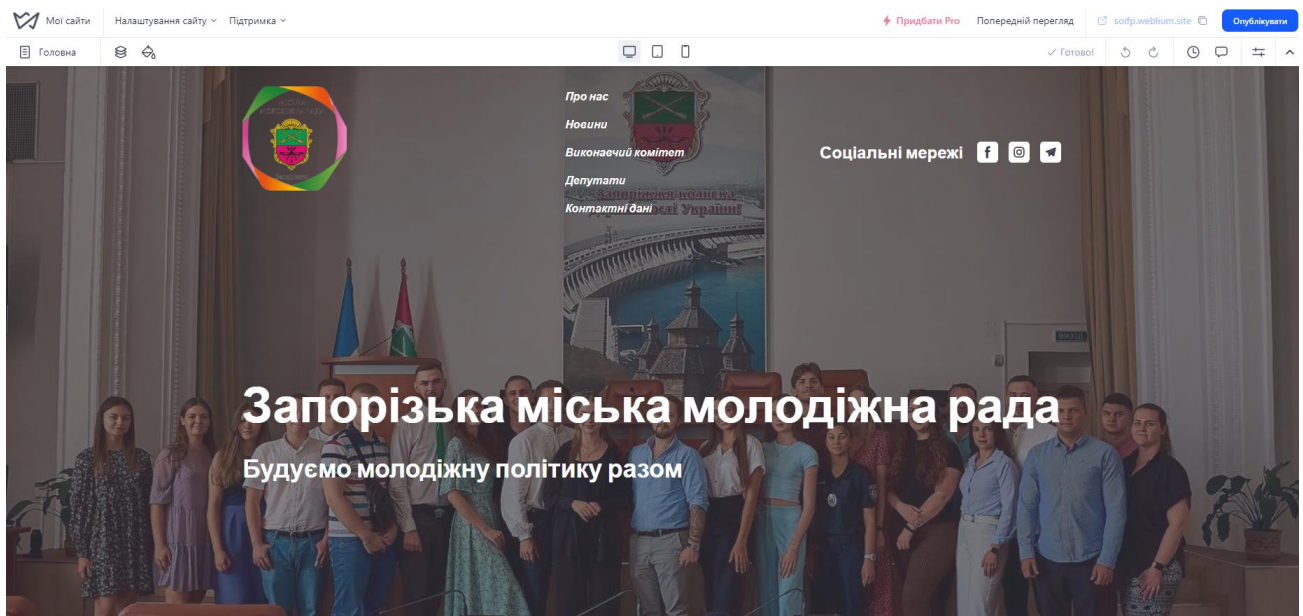


Рисунок 3.4 – Головна сторінка вебсайту

- Головна сторінка є обличчям сайту, тому важливо правильно налаштувати її структуру. Вона повинна містити основні елементи, такі як заголовок, основні секції (про нас, послуги, відгуки, контактну інформацію тощо). На цій сторінці часто розміщують ключову інформацію, яка має привернути увагу відвідувачів.

## 2. Додавання нових сторінок (див. рис. 3.5).

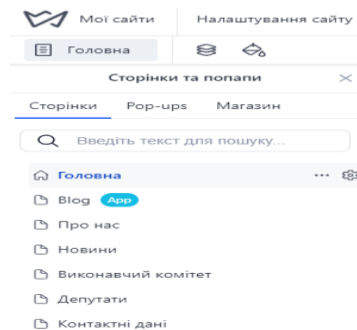


Рисунок 3.5 – Головна сторінка вебсайту

- Для створення повноцінного сайту необхідно додати нові сторінки. Weblium дозволяє легко додавати сторінки, такі як "Про нас", "Послуги", "Контакти", "Блог" та інші. Кожна сторінка може мати свій унікальний контент та дизайн, що робить сайт більш інформативним та корисним для відвідувачів.

## 3. Організація меню навігації (див. рис. 3.6).

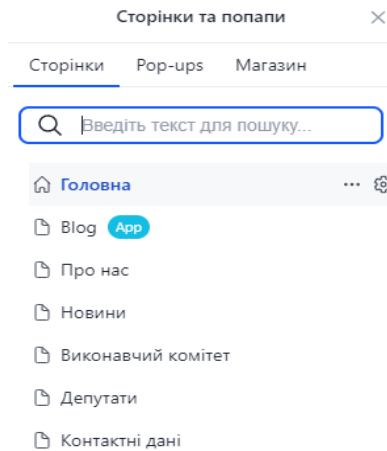


Рисунок 3.6 – Меню навігації

- Навігаційне меню є важливим елементом, що забезпечує зручність користування сайтом. Користувач може налаштувати меню навігації, додаючи до нього посилання на всі основні сторінки сайту. Це допоможе відвідувачам легко знайти потрібну інформацію, покращуючи їхній досвід користування.

Крок 3: Редагування та налаштування сторінок

1. Редагування контенту (див. рис. 3.7).

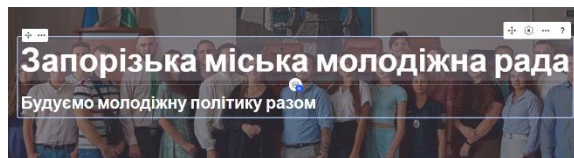


Рисунок 3.7 – Редагування контенту

- **Текстовий контент:** додавання та редагування текстового контенту є одним з основних завдань. Користувач може додавати заголовки, абзаци, цитати та інші текстові елементи, що допоможуть передати необхідну інформацію відвідувачам. Важливо використовувати відповідний стиль та форматування для покращення читабельності та візуального сприйняття контенту.

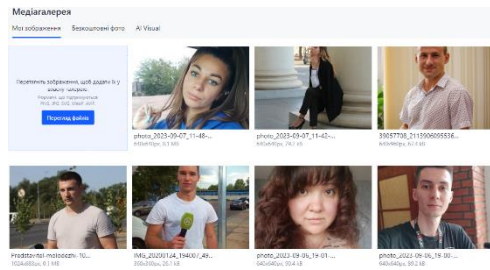


Рисунок 3.8 – Додавання зображень

- Зображення та медіа (див. рис. 3.8): важливою частиною контенту є зображення та інші медіа-файли. Weblium дозволяє завантажувати власні зображення або використовувати бібліотеку зображень, надану платформою. Також можна додавати відео та інші медіа-елементи, щоб зробити сторінки більш привабливими та інформативними. Медіа-контент може значно покращити візуальну привабливість сайту та залучити більше відвідувачів.

## 2. Налаштування стилю (див. рис. 3.9).



Рисунок 3.9 – Налаштування

- Шрифти та кольори: Користувач може налаштувати шрифти та кольорову схему сайту відповідно до свого бренду або особистих уподобань. Це дозволяє створити унікальний стиль, що підкреслює індивідуальність сайту. Правильний вибір кольорів та шрифтів може значно покращити візуальне сприйняття сайту та зробити його більш професійним.

- Блоки та секції: Weblium надає можливість додавати різноманітні блоки та секції, такі як галереї, каруселі, форми зворотного зв'язку, кнопки тощо. Кожен блок можна налаштувати індивідуально, що дозволяє створювати унікальний дизайн для кожної сторінки (див. рис. 3.10).

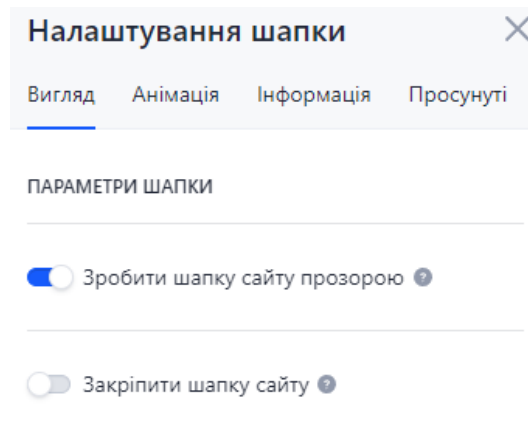


Рисунок 3.10 – Налаштування блоків та секцій

Це дає можливість створювати складні та багатofункціональні сторінки без необхідності кодування.

### 3. Перегляд в реальному часі

- Функція попереднього перегляду в реальному часі дозволяє користувачам бачити зміни, які вони вносять, одразу на сторінці. Це допомагає оцінити вигляд сайту та зробити необхідні коригування ще до публікації. Завдяки цій функції можна миттєво бачити результати своїх дій та виправляти помилки в процесі роботи.

#### Крок 4: Додавання інтерактивного функціоналу

##### 1. Форми зворотного зв'язку

- Створення та налаштування форм зворотного зв'язку є важливим елементом інтерактивності сайту. Це дозволяє відвідувачам залишати свої коментарі, запитання або зв'язуватися з власником сайту безпосередньо через веб-сторінку. Форми можуть бути різного типу: контактні форми, форми підписки на розсилку, опитування та інші. Налаштування форм дозволяє збирати необхідну інформацію від відвідувачів і автоматично відправляти її на електронну пошту або в CRM-систему.

##### 2. Інтеграції з сервісами

- Weblium підтримує інтеграцію з різними сервісами, такими як Google Analytics, CRM-системи, сервіси email-маркетингу тощо. Це дозволяє розширити функціонал сайту та забезпечити зручність його використання. Наприклад, інтеграція з Google Analytics дозволяє відстежувати статистику відвідувань, аналізувати поведінку користувачів та оптимізувати контент для покращення результатів. Інтеграція з CRM-системами дозволяє автоматично зберігати контактні дані відвідувачів і керувати взаємодією з клієнтами.

### 3. SEO-налаштування

- Налаштування параметрів для пошукової оптимізації (SEO) є важливим кроком для покращення видимості сайту у пошукових системах. Користувач може налаштувати метатеги, заголовки, описи та ключові слова для кожної сторінки, що допоможе залучити більше відвідувачів. SEO-налаштування також включають оптимізацію зображень, створення дружніх URL-адрес та налаштування швидкості завантаження сторінок. Це забезпечує кращу індексацію сайту пошуковими системами та підвищує його рейтинг у результатах пошуку.

#### Крок 5: Тестування та публікація сайту

##### 1. Перевірка працездатності

- Перш ніж опублікувати сайт, важливо перевірити всі посилання, форми та інтерактивні елементи на працездатність. Це дозволить уникнути можливих помилок і забезпечити позитивний досвід для відвідувачів. Також варто протестувати сайт на різних пристроях та браузерах, щоб переконатися, що він коректно відображається та працює в будь-яких умовах.

##### 2. Підключення домену

- Користувач може вибрати та підключити власний домен або скористатися доменом, запропонованим Weblium. Підключення власного домену допоможе зробити сайт більш професійним та впізнаваним. Процес підключення



домену на Weblium є простим і зручним, з детальними інструкціями для користувачів.

### 3. SSL сертифікат

- Налаштування SSL сертифікату забезпечує безпечне з'єднання через HTTPS, що є важливим для захисту даних користувачів та покращення довіри до сайту. Наявність SSL сертифікату є важливою вимогою для багатьох пошукових систем, що також позитивно впливає на SEO.

### 4. Публікація сайту

- Після завершення всіх налаштувань сайт можна опублікувати, зробивши його доступним для відвідувачів. Публікація займає кілька хвилин, після чого сайт стає доступним в Інтернеті. Weblium надає можливість швидкого внесення змін після публікації, що дозволяє постійно оновлювати контент та покращувати сайт.

### Переваги використання Weblium для розробки сторінок

- Інтуїтивно зрозумілий інтерфейс: Простий та зручний інтерфейс Weblium дозволяє швидко створювати і редагувати сторінки навіть користувачам без технічних знань. Це значно спрощує процес розробки сайту та робить його доступним для широкого кола користувачів.

- Гнучкість та адаптивність: Сайти, створені на платформі Weblium, автоматично адаптуються до різних пристроїв і розмірів екранів, забезпечуючи гарний вигляд і функціональність на будь-якому пристрої. Це гарантує зручність користування сайтом для відвідувачів, незалежно від того, який пристрій вони використовують.

- Широкі можливості налаштування: Weblium пропонує великий вибір шаблонів та блоків, що дозволяє створювати унікальні дизайни. Користувачі можуть налаштовувати всі елементи сайту відповідно до своїх потреб і вподобань, створюючи індивідуальний стиль та функціонал.

- Інтеграції та підтримка: Платформа підтримує інтеграцію з популярними сервісами, що розширює функціональність сайту. Також Weblium надає технічну підтримку, що допомагає вирішувати будь-які питання та проблеми, що можуть виникнути під час роботи. Це забезпечує надійність та безперебійність роботи сайту.

## ВИСНОВКИ

Відповідно до технічного завдання, можливостей сучасних інформаційних систем та аналізу доступних інструментів, було реалізовано вебсайт молодіжної ради міста Запоріжжя. Для розробки була використана платформа Weblium, а саме її функціонал для створення автоматизованих інформаційних систем.

У ході реалізації проєкту були виконані такі кроки:

1. Визначено поняття інформаційної системи та переваги її використання для оптимізації діяльності молодіжної ради.

2. Проведено аналіз існуючих інформаційних в схожій предметній області, щоб визначити оптимальні підходи та рекомендації.

3. Описані інструменти, які можуть бути використані для реалізації інформаційної системи, особлива увага була приділена їхнім можливостям та перевагам.

4. Розроблено та реалізовано проєкт інформаційної системи, що відповідає специфіці потреб молодіжної ради, відповідно до розроблених макетів.

5. Проведено тестування розробленого вебсайту, перевірку його функціональності та надійності.

Результатом кваліфікаційної роботи є розроблений вебсайт, який відповідає визначеним технічним вимогам та вирішує поставлені завдання для оптимізації роботи молодіжної ради.

Реалізована інформаційна системи дозволяє об'єднати декілька інструментів ведення обліку роботи молодіжної ради в єдину систему. Вона допомагає організації ефективно взаємодіяти з містянами, знаходити нових діячів, будувати аналітику та планувати роботу на наступний місяць або рік. В результаті, ця системи призводять до збільшення комунікації та покращення роботи місцевого органу самоврядування.

## ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Запорожець О. О. Інформаційні системи, їх види; апаратне та програмне забезпечення інформаційної системи. URL: <https://www.kievoit.ippo.kubg.edu.ua/kievoit/2013/95/95.html> (дата звернення: 29.04.2024).
2. Марченко Д.М., Яровенко Г.М. Автоматизовані інформаційні системи управління – ресурс підвищення ефективності внутрішнього контролю підприємств. URL: [https://essuir.sumdu.edu.ua/bitstream-download/123456789/52016/5/Marchenko\\_internal\\_control.pdf](https://essuir.sumdu.edu.ua/bitstream-download/123456789/52016/5/Marchenko_internal_control.pdf) (дата звернення: 29.04.2024).
3. Класифікація інформаційних систем. URL: <https://ua5.org/isystem/374-klasifikacija-nformacijnikh-sistem.html> (дата звернення: 03.05.2024).
4. Складові частини інформаційних систем: характеристика. URL: [https://osvita.ua/vnz/reports/econom\\_pidpr/19058/](https://osvita.ua/vnz/reports/econom_pidpr/19058/) (дата звернення: 08.05.2024).
5. Діаграми Прецедентів (Use Case UML Diagram). URL: <https://lvivqaclub.blogspot.com/2008/10/use-case-uml-diagram.html> (дата звернення: 18.05.2024).
6. Титенко, С. В. FreshKnowledge – система управління Вебконтентом на семантичному рівні. К. : Просвіта, 2017. 352 с.
7. Заводський І. Основи візуального програмування. Вид. група ВНУ, 2011. 411 с.
8. Огляд конструктора сайтів Weblium. URL: <https://adwservice.com.ua/uk/ogljad-weblium> <https://adwservice.com.ua/uk/ogljad-weblium> (дата звернення: 18.05.2024).

**Декларація**  
**академічної доброчесності**  
**здобувача освіти ВСП «Економіко-правничого фахового коледжу ЗНУ»**

Я, Шуть Денис Володимирович, здобувач освіти 4 курсу, спеціальності/освітньо-професійної програми Інженерія програмного забезпечення/Розробка програмного забезпечення, групи K121-20, адреса електронної пошти denisshut5665@gmail.com.

- підтверджую, що написана мною кваліфікаційна робота на тему «Проектування та розробка вебсайту запорізької міської молодіжної ради» відповідає вимогам академічної доброчесності та не містить порушень, що визначені у ст. 42 Закону України «Про освіту», зі змістом яких ознайомлений/ознайомлена;

- заявляю, що надана мною для перевірки електронна версія роботи є ідентичною її друкованій версії;

- згоден/згодна на перевірку моєї роботи на відповідність критеріям академічної доброчесності у будь-який спосіб, у тому числі за допомогою інтернет-системи, а також на архівування моєї роботи в базі даних цієї системи.

Дата \_\_\_\_\_ Підпис \_\_\_\_\_ Денис ШУТЬ

Дата 01.06.2024 Підпис  Юлія БОРИСОВСЬКА