

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Тернопільський національний економічний університет
Факультет комп'ютерних інформаційних технологій
Кафедра комп'ютерних наук

БУДЕНЧУК Сергій Сергійович

Веб-орієнтована програмна система
"Екоуніверситет" / Web-based software system
"Ecouiversity"

напрямок підготовки: 6.050103 - Програмна інженерія
фахове спрямування - Програмне забезпечення систем

Бакалаврська дипломна робота

Виконав студент групи ПЗС-41
С. С. Буденчук

Науковий керівник:
викладач ОЛІЙНИК І.С.

Бакалаврську дипломну роботу
допущено до захисту:

"__" _____ 20__ р.

Завідувач кафедри

_____ **А. В. Пукас**

ТЕРНОПІЛЬ - 2016

РЕЗЮМЕ

Дипломна робота містить 75 сторінок, 13 таблиць, 74 рисунки, список використаних джерел із 15 найменувань, 7 додатків.

Метою дипломної роботи є розробка системи інформування користувачів про новини та заходи економічної діяльності вищого навчального закладу

Об'єктом дослідження є процес публікації новини та оголошення заходів у концепції «Екоуніверситет».

Предметом дослідження є діяльність вищого навчального закладу у концепції «Екоуніверситет»

Одержані результати полягають у розробці веб-орієнтованої системи «Екоуніверситет»

Ключові слова: новина, публікація, захід, ініціатива, університет, подання статті.

RESUME

This thesis contains 75 pages, 13 tables, 74 figures, list of references from 15 names, 7 applications.

The aim of the thesis is to develop a system to inform users about news and events economic activity of higher education

The object of the research is the process of publishing news and announcements measures in the concept of "Ekouniversytet".

The subject of the study is the work of higher education in the concept of "Ekouniversytet"

The results are in developing web-based system "Ekouniversytet"

Keywords: news, publication, event, initiative, university, submitting articles.

ЗМІСТ

ВСТУП.....	10
РОЗДІЛ I АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ	12
1.1 Коротка характеристика об'єкту управління.....	12
1.2 Опис предметної області.....	13
1.3 Огляд і аналіз існуючих аналогів	16
1.4 Специфікація вимог	29
Висновки до першого розділу	36
РОЗДІЛ II ПРОЕКТУВАННЯ	37
2.1. Розробка архітектури програмної системи.....	37
2.2 Проектування структури бази даних.....	43
Висновки до другого розділу.....	48
РОЗДІЛ III ПРОГРАМНА РЕАЛІЗАЦІЯ.....	49
3.1 Програмна реалізація проекту	49
3.1.1 Обґрунтування вибору мови програмування.....	49
3.1.2 Організація інтерфейсу з користувачем.....	50
3.1.3 Вимоги до апаратного забезпечення	55
3.2 Програмна реалізація бази даних	55
Висновки до третього розділу	57
РОЗДІЛ IV ТЕСТУВАННЯ ТА ДОСЛІДНА ЕКСПЛУАТАЦІЯ	58
4.1 Тестування	58
4.1.1 Функціональне тестування	58
4.1.2 Тестування сумісності.....	61
4.1.3 Тестування продуктивності	62
4.1.3 Тестування GUI	63
4.2 Розгортання програмного продукту	64
4.3 Інструкція користувача	64
4.3.1 Додавання матеріалу на сайт	65
4.3.2 Реєстрація нового користувача	70
Висновки до четвертого розділу	74
ВИСНОВОК	75

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	76
ДОДАТОК А	78
ДОДАТОК Б	79
ДОДАТОК В	80
ДОДАТОК Г	81
ДОДАТОК Д	83
ДОДАТОК Е	84
ДОДАТОК Є	86

ВСТУП

Актуальність дипломної роботи полягає в тому, що інформування студентів про усі заходи та новини на території студентського містечка та за його межами буде проводитися завжди, але за допомогою сторонніх сервісів та сайтів, не призначених для цієї діяльності. При користуванні сторонніми сайтами новини з часом губляться та втрачають свою актуальність, а користуватися пошуком не досить зручно та практично. Розробка Інтернет - сервісу який допоможе колективу навчального закладу інформувати своїх користувачів про останні новини на території університету, значно спростить та вдосконалить публічне оголошення заходів. Проте основним недоліком багатьох аналогів систем оповіщення є неефективність викладення інформації, адже саме цей критерій визначає зацікавленість людей у користуванні сайтом.

Тому розробка веб-сайту, який надає можливість швидкої публікації новин та заходів зі збереженням актуальності, а саме розподілом на критерії та винесенням головних подій у обране є досить актуальною задачею.

Основною метою роботи є аналіз уже існуючих інтернет-сайтів, які дозволяють інформувати користувачів про останні події та нововведення, ефективно публікувати новини та найбільш інформативним чином оголошувати заходи у своїй галузі. На основі отриманих даних спроектувати систему, яка використає досвід вже створених сервісів та додасть суттєві покращення у процес публікації новин.

Надалі, спираючись на мету, можна оголосити основні задачі, які ставляться до розроблюваного продукту.

1. Детально дослідити предметну область, з якою буде працювати інтернет-сервіс.
2. Проаналізувати уже існуючі аналоги, які дозволяють публікувати новини та заходи.

3. Винести у необхідні критерії проектування такі пункти: збереження актуальність статей, точність та зручність подання інформації і надання зручного доступу усім користувачам.
4. Вибір найбільш актуальних технологій для реалізації визначеного функціоналу
5. Розробка програмного продукту згідно задач представлених раніше.

Об'єкт дослідження – процес публікації новин та оголошення заходів у концепції «Екоуніверситет»

Головна цінність створюваного сервісу полягатиме у збереженні актуальності новин та зручність подання для користувачів.

РОЗДІЛ І АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Коротка характеристика об'єкту управління

Сьогодні широко розповсюдженою темою є ЕКОжиття вищого навчального закладу, а саме проведення заходів для покращення екологічного стану навколишнього середовища.

Планування та проведення заходів можуть здійснювати як керівники студентського самоврядування так і безпосередньо студенти.

Найбільш зручним та швидким методом розповсюдження ЕКОновин є Інтернет, оскільки є можливість донести інформацію досить швидко та у зрозумілому і зручному форматі.

В цілому життєвий цикл новини, починаючи від ідеї та закінчуючи публікацією можна зобразити у вигляді простої схеми (рис. 1.1).

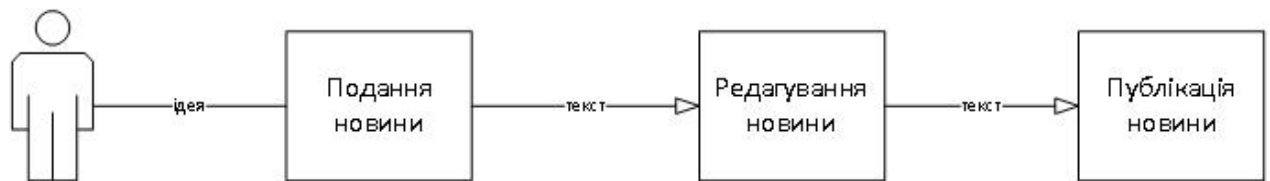


Рисунок 1.1 – Життєвий цикл новини

Відповідно для потрапляння тексту новини до читача необхідною умовою є проходження через усі попередні етапи.

ЕКО-концепція включає: інформаційний портал, журнал новин та подій, персонал який відповідає за наповнення та редагування контенту і персонал який управляє виключно форм факторами наповнення статей.

Через незручність використання паперового методу донесення новин до студентів є доцільним розробка та впровадження веб орієнтованої моделі розповсюдження інформації.

1.2 Опис предметної області

Для належної організації роботи Інтернет-сервісу «Екоуніверситет» необхідне виконання таких основних бізнес-процесів (рис. 1.2):

- процес створення новини;
- процесу редагування новини;
- процесу публікації новини;



Рисунок 1.2 – Діаграма ієрархії процесів розроблюваного програмного продукту

Розглянемо детальніше кожен з вище представлених бізнес-процесів. На рисунку 1.3 бачимо діаграму функцій процесу створення новини.



Рисунок 1.3 – Діаграма дерева функцій створення новини

Першим етапом роботи з Інтернет-сервісом є створення новини для подальшого подання на перевірку. Даний етап використання функціоналу сайту характеризується поетапним заповненням відповідних полів.

Таблиця 1.1

Характеристика бізнес-процесу створення новини

Назва характеристики	Значення характеристики
Ім'я бізнес-процесу	Подання новини
Основні учасники	Зареєстрований користувач, Адміністратор
Вхідна подія	Формування новини
Вхідні документи	Специфікація до оформлення новини
Вихідна подія	Вивід повідомлення про успішне подання новини
Вихідні документи	Сформований текст новини
Клієнт бізнес-процесу	Зареєстрований користувач, Адміністратор

На рисунку 1.4 зображено діаграму функцій процесу «Редагування новини»

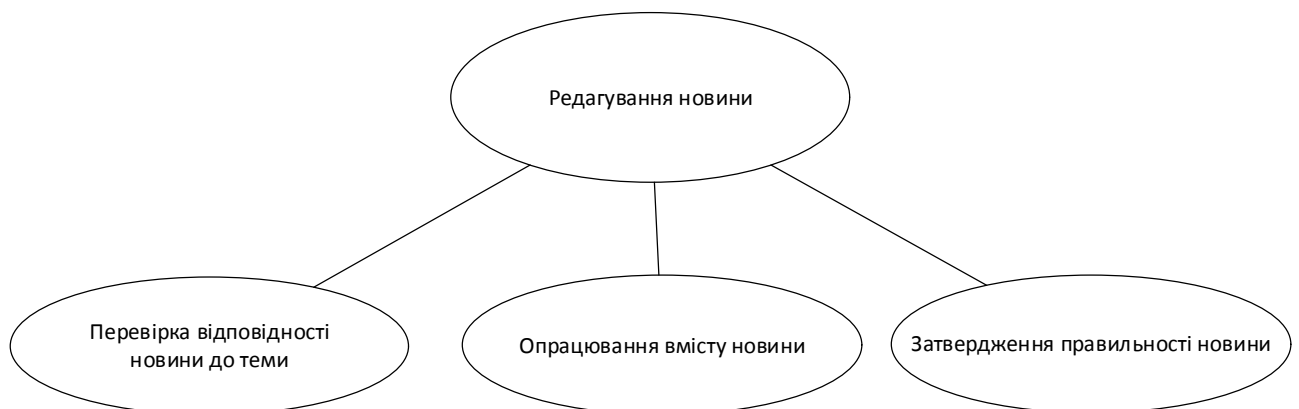


Рисунок 1.4 – Діаграма функцій процесу редагування новини

На рисунку 1.5 зображено процес редагування новини при поданні на сайт. Для подальшого опрацювання новини рецензентом необхідно пройти

етап подання новини на сайт, після чого відбувається перевірка, згідно усіх критеріїв.

Таблиця 1.2

Характеристика бізнес-процесу редагування новини

Назва характеристики	Значення характеристики
Ім'я бізнес-процесу	Редагування новини
Основні учасники	Адміністратор
Вхідна подія	Редагування статті
Вхідні документи	Специфікація до оформлення новини
Вихідна подія	Відредагована новина
Вихідні документи	Відформатована та готова до публікації новина
Клієнт бізнес-процесу	Адміністратор

Кінцевим етапом процесів є подання новини та подальша публікація на сайті. Для доступу до цієї частини процесу необхідно пройти усі попередні етапи.

Детально процес публікації новини зображений на рисунку 1.5



Рисунок 1.5 – Діаграма функцій процесу публікації новини

Процес публікації новини включає в себе:

- перевірку – кінцевий огляд та порівняння новини зі специфікацією;
- збереження – затвердження відповідності статті та збереження на сервер;
- публікацію – додання новини безпосередньо на сайт;
- Функцію редагування контенту статті (необов'язково).

Таблиця 1.3

Характеристика бізнес-процесу публікації новини

Назва характеристики	Значення характеристики
Ім'я бізнес-процесу	Публікація новини
Основні учасники	Адміністратор
Вхідна подія	Перевірка новини
Вхідні документи	Специфікація до оформлення новини
Вихідна подія	Повідомлення про успішну публікацію новини на сайті
Вихідні документи	-
Клієнт бізнес-процесу	Адміністратор

1.3 Огляд і аналіз існуючих аналогів

На етапі проектування веб додатку доцільно оглянути та дослідити усі існуючі аналоги проектів, які безпосередньо або опосередковано включають функціонал концепції еко університету. Для аналізу виберемо наступні веб-сайти: Green Campus Partnership від університету Penn University (режим доступу: <http://www.upenn.edu/sustainability/>), Green Choices - університет розвитку еко-концепції у Маямі (режим доступу: <http://www.greenchoices.org/>) та веб ресурс Еко – Львів (режим доступу: <http://phil.nullable.eu/>).

Розглянемо детальніше усі наведені вище аналоги та їх функціонал. Для початку оберемо сайт Green Campus Partnership.

На рисунку 1.6 зображено головну сторінку сайту від представників Green Campus Partnership. На головній сторінці відображаються останні новини у вигляді статей із зображенням, також представлена навігація по сторінках сайту у вигляді горизонтального списку з усіх категорій сайту. На сайті відсутня можливість реєстрації та авторизації, оскільки це сайт новин. У верхньому блоці сайту розміщена панель пошуку по новинах.

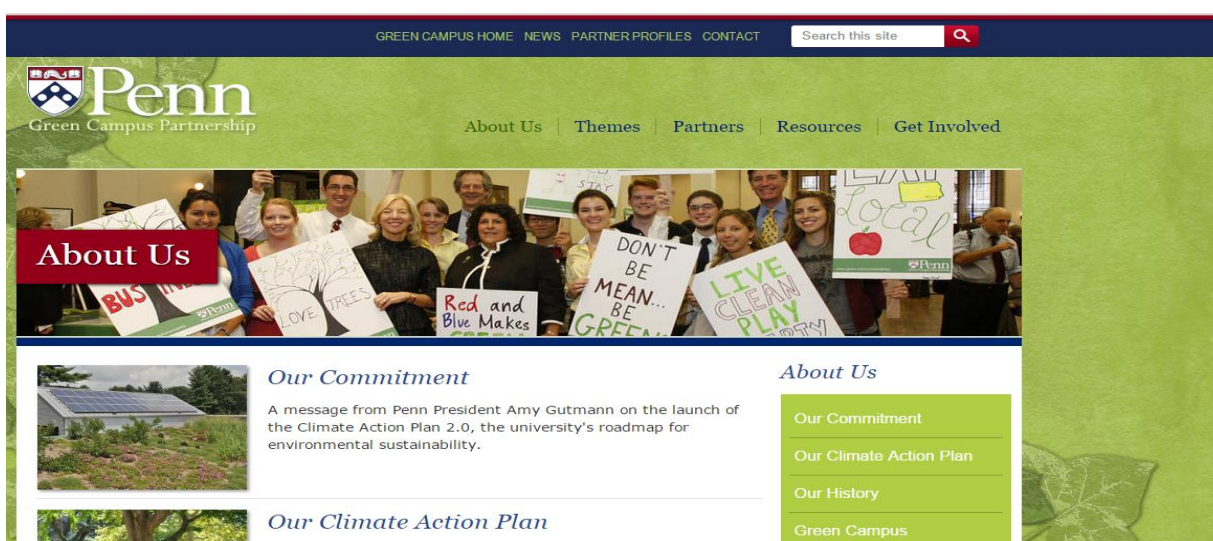


Рисунок 1.6 – Головне меню сайту Green Campus Partnership

При натисканні на кнопку «Про нас» з'явиться вікно, яке зображене на рисунку 1.7. В цьому вікні представлено список категорій з інформацією про сайт та університет. Категорії поділяються на: «Історичний аспект університету», «Діяльність університету» та «Колектив еко концепції».



Рисунок 1.7 – Сторінка з інформацією

На рисунку 1.8 зображено результат вибору однієї із категорій розглянутого веб-ресурсу.



Рисунок 1.8 – Сторінка із вибраною статтею

При натисненні на пункт меню «Themes» відкривається сторінка з списком усіх публікованих на сайті новин, які розділені на категорії та сортовані за темами (рис. 1.9).

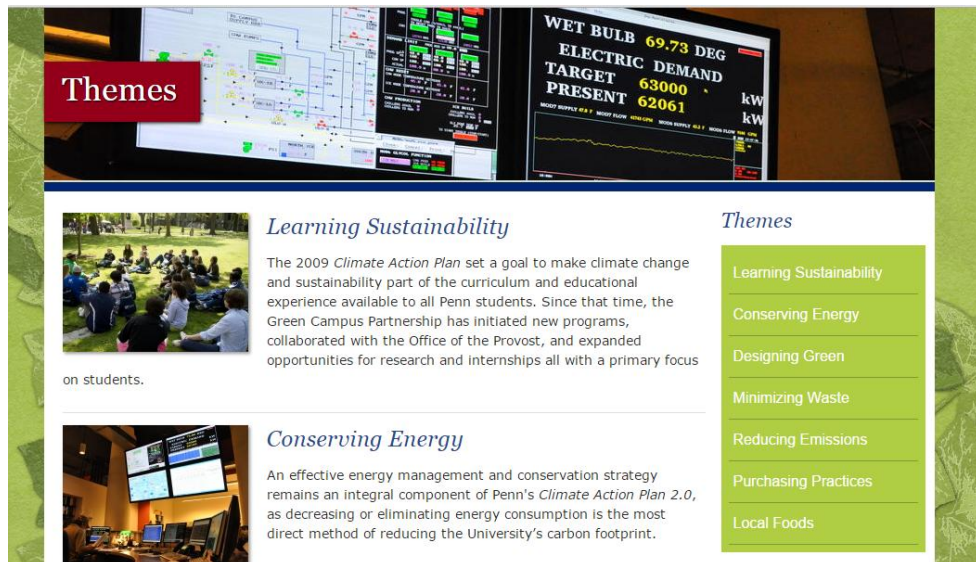


Рисунок 1.9 – Сторінка з темами сайту

При натисканні на пункт меню «Partners» відкриється сторінка з переліком усіх партнерів даного відділу університету (рис. 1.10).



Рисунок 1.10 – Сторінка з партнерами проекту

При натисканні на пункт меню «Resources» відбувається перехід на сторінку з інформаційним контентом, а саме: актуальні питання, карта студентського містечка, відеозаписи та критерії для вступу в університет (рис. 1.11)



Рисунок 1.11 – Сторінка Resources

У верхній частині сторінки розміщений блок пошуку по сайту за допомогою ключових слів (рис. 1.12)

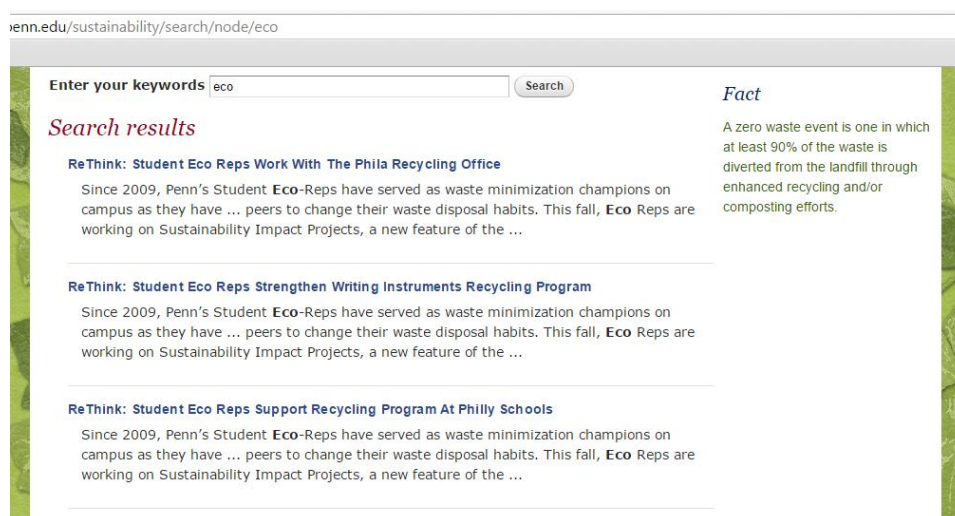


Рисунок 1.12 – Сторінка пошуку

Далі розглянемо наступний аналог – сайт «Green Choices» (режим доступу: <http://www.greenchoices.org>), головна сторінка сайту зображена на рисунку 1.13. Головна сторінка якого містить: банер сайту у вигляді зображення, навігаційне меню представлене горизонтальним списком усіх

сторінок сайту, головне вікно із останньою актуальною новиною та меню пошуку по сайту за допомогою ключових слів.



Рисунок 1.13 – Головна сторінка сайту Green Choices

При наведенні курсором миші на пункт меню «Green Living» відкривається спливаюче контекстне меню із списком запропонованих еко – рішень для різних сфер життя (рис. 1.14).

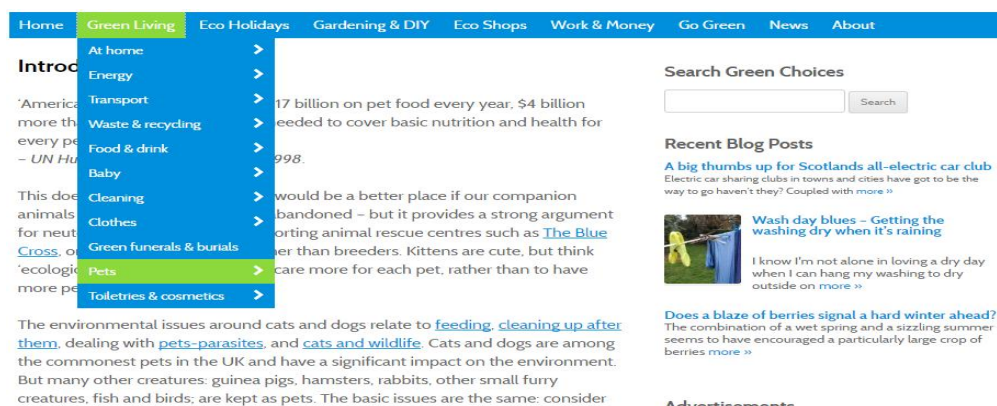


Рисунок 1.14 – Контекстне меню «Green Living»

При виборі будь-якої з категорій відбувається перехід на відповідну сторінку з статтею (рис. 1.15).

ng/energy/wood-for-fuel

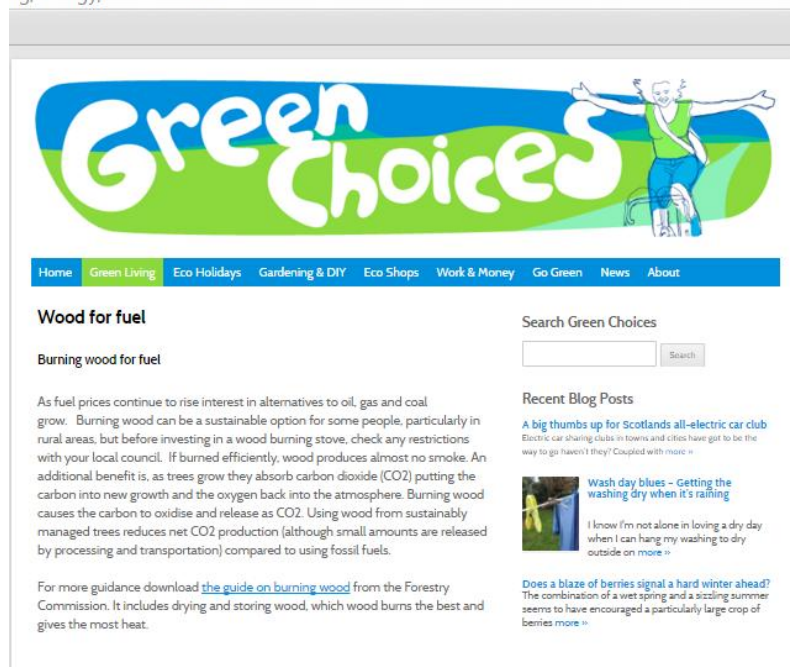


Рисунок 1.15 – Сторінка зі статтею

Усі інші пункти меню також представлені у вигляді спливаючих списків. На рисунку 1.16 зображений пункт меню «Eco Holidays».

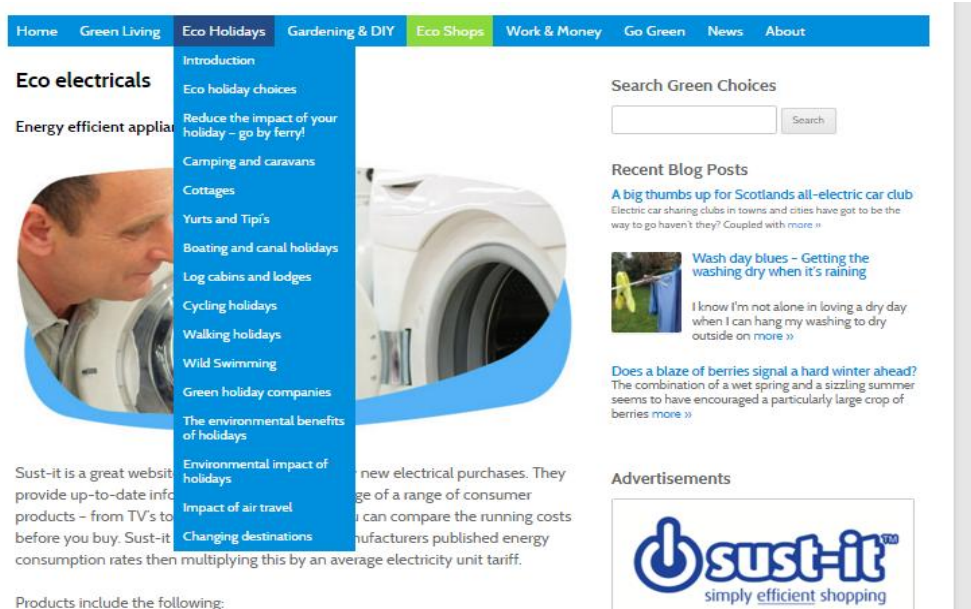


Рисунок 1.16 – Спливаюче меню «Eco Holidays»

На рисунку 1.17 зображено перелік пунктів із спливаючого меню «Gardening & DIY»

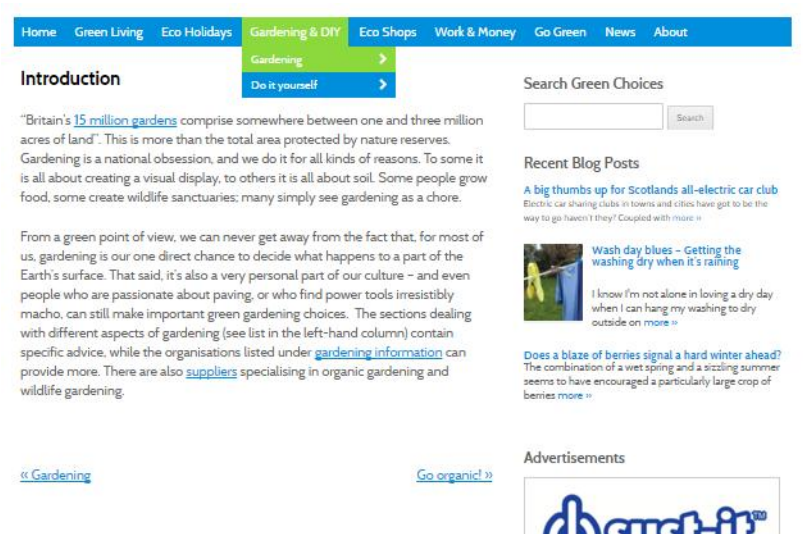


Рисунок 1.17 – Спливаюче меню «Gardening & DIY»

При натисканні на пункт меню «Eco shops» відбувається перехід на сторінку із списком усіх магазинів, які мають в наявності екологічні товари (рис. 1.18).

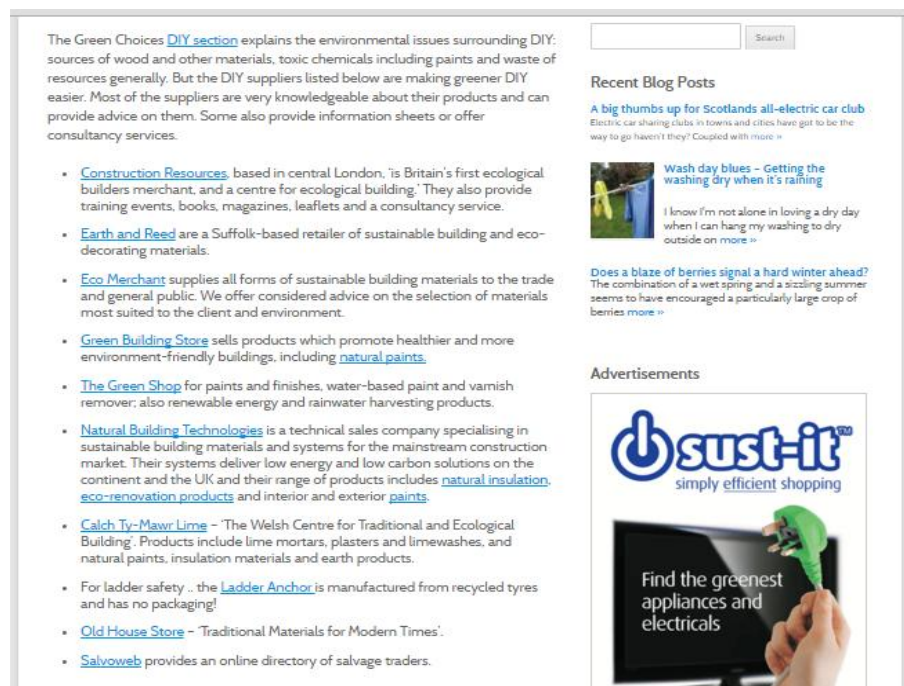


Рисунок 1.18 – Сторінка «Eco shops»

На сайті представлений пункт меню, при переході на який відображається сторінка з усіма актуальними еконовинами (рис. 1.19).



Рисунок 1.19 – Список еко новин

Праворуч від основного блоку сайту розміщене вікно пошуку новин та статей з використанням ключових слів (рис. 1.20).

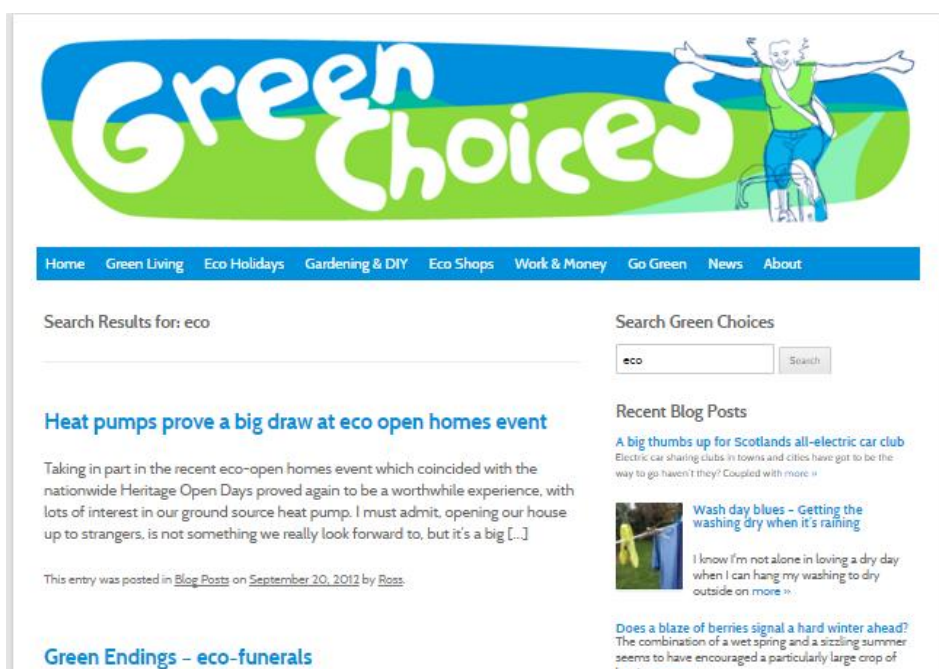


Рисунок 1.20 – Пошук матеріалів

Також користувачеві доступно меню зі списком соціальних мереж даної спільноти (рис. 1.21).



Рисунок 1.21 – Футер із посиланням на соціальні мережі

Для забезпечення зворотного зв'язку на сайті доступна форма зворотного зв'язку, яка зображена на рисунку 1.22. На даній формі користувачу запропоновано ввести таку інформацію:

- коментар;
- ім'я;
- свою веб-сторінку;
- електронну адресу.

Your email address will not be published. Required fields are marked *

Comment

Name *

Email *

Website

Рисунок 1.22 – Форма зворотного зв'язку

Далі розглянемо третій сайт-аналог, а саме –«Еко-Львів» (режим доступу: <http://phil.nullable.eu/>). Головна сторінка сайту зображена на рисунку 1.23. Головна сторінка містить навігацію по сайту, список останніх новин, та інформацію про сайт.



Рисунок 1.23 – Головна сторінка сайту

У верхній та лівій частині сайту представлено вертикальне меню навігації (рис. 1.24).

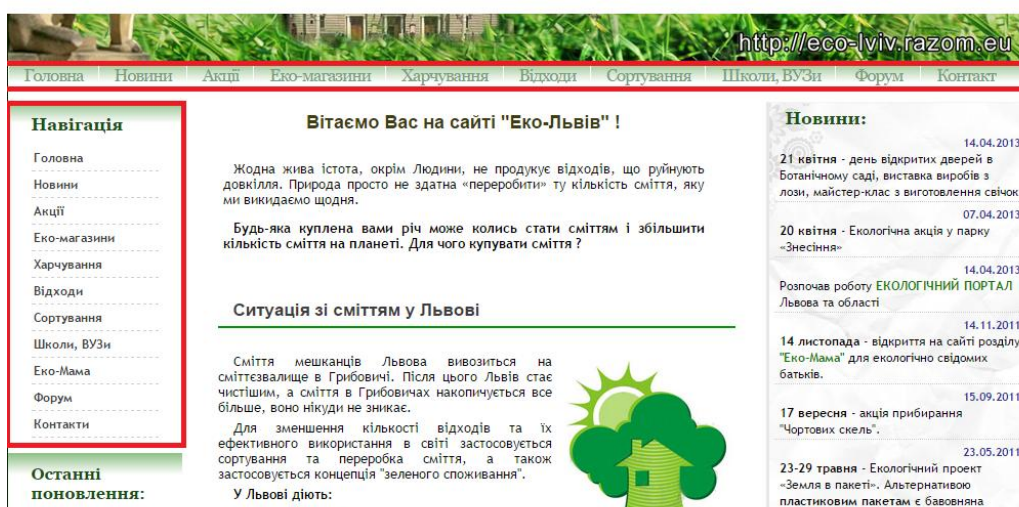


Рисунок 1.24 – Меню навігації по сайту

Усі меню навігації розділено на такі категорії:

- головна;
- новини;
- акції;
- еко-магазин;

- харчування;
- відходи;
- сортування;
- школи, ВУЗи;
- форум;
- контакти.

При натисканні на кнопку «Новини» відбувається перехід на сторінку останніх новин, опублікованих на сайті (рис. 1.25).

Навігація

- Головна
- Новини**
- Акції
- Еко-магазини
- Харчування
- Відходи
- Сортування
- Школи, ВУЗи
- Форум
- Контакти

Останні поновлення:

Конопля
Конопля - рослина з найбільшим екологічним та економічним потенціалом. З неї роблять тканину, папір, олію, будують конопляні будинки. ... читати далі ...

ТАНГРАМ

Новини, оголошення

Для підписки на розсилку новин пришліть запит на e-mail: [eco-lviv\(a\)meta.ua](mailto:eco-lviv(a)meta.ua)
Інші екологічні новини Львова ... тут ...

ЕКОЛОГІЧНИЙ ПОРТАЛ Львова та області

Розпочав роботу ЕКОЛОГІЧНИЙ ПОРТАЛ Львова та області
Метою створення порталу є пошук однодумців, висвітлення екологічних проблем та обмін ідеями щодо їх вирішення, об'єднання зусиль для вирішення конкретних екологічних проблем, розміщення ваших ідей щодо покращення екологічної ситуації, можливість знайти меценатів та спонсорів для еволюційних заходів та можливість для бізнесу зробити свій внесок у добрі справи.
Кожен з вас може розміщати тут екологічну інформацію.

7promeniv.com.ua

14.04.2013

21 квітня - день відкритих дверей в Ботанічному саді, виставка виробів з лози

21 квітня з 14:00 до 18:00 год - день відкритих дверей в Ботанічному саді (вул. Черемшини, 44 - в районі "Погулянки"). В програмі:
- Екскурсії по Ботанічному саду: Дендрофлора Ботанічного саду, Рослини - ефемероїди
- Виставка мистецьких виробів з лози

Акції:

- 02.10.2010
28 квітня - Екологічна акція «Зробимо Україну чистою! 2012»
- 25.04.2012
21 квітня - Екологічна акція у парку «Знесіння» ... читати далі ...
- 25.04.2012
7 березня - акція "Збережімо первоцвіти" з майстер-класом з оригамі ... читати далі ...
- 02.10.2010
Прибирання парку "Знесіння" ... читати далі ...
- 10.09.2010
Конкурс на кращий малюнок для екороботи триває. ... читати далі ...
- 27.08.2010
Нагороджений переможець конкурсу на кращий малюнок для екороботи. ... читати далі ...
- 26.08.2010
Наші друзі-екологи назбирали в Карпатах мішок пластикових пляшок. ... читати далі ...
- 17.04.2010
Прибирання парку "Цитадель" з сортуванням сміття.... читати далі ...

Рисунок 1.25 – Сторінка з новинами

При натисканні на кнопку «Акції» відбувається відображення сторінки з усіма заходами, які були проведені (рис. 1.26).

Рисунок 1.26 – Сторінка із заходами

Для обговорення останніх новин та зручного спілкування користувачів сайту між собою є форум. При натисканні на кнопку «Форум» відбувається перехід на окремий веб-ресурс (рис. 1.27).

Рисунок 1.27 – Форум

На усіх вище розглянутих ресурсах відсутня можливість реєстрації, авторизації користувача на сайті та подання тексту новини безпосередньо з веб-ресурсу. Ідентифікація користувача на сайт є необхідною для визначення автора який відправив текст новини, тому було вирішено додати цей

функціонал. У таблиці 1.4 приведена порівняльна характеристика трьох веб-сайтів.

Порівняльна характеристика програмних продуктів представляє собою детальне порівняння усіх наведених вище програм.

Таблиця 1.4

Порівняльна характеристика програмних продуктів

Фірма-розробник	Не вказано	Не вказано	Не вказано
Назва програмного продукту	Green Campus Partnership	Green Choices	Еко – Львів
Версії продукту	1.0	1.0	1.0
Функціональність	- Профіль користувача - Пошук - Перегляд поточні новини	- Профіль користувача - Пошук - Форум	- Профіль автора - Пошук - Перегляд новин онлайн,

Продовження таблиці 1.4

Інтерфейс користувача	Добре	Відмінно	Задовільно
Допомога користувачеві	Відсутня	Відсутня	Відсутня
Читабельність	Присутня	Присутня	Присутня

1.4 Специфікація вимог

При роботі з системою передбачаються три типи користувачів: користувач, гість та адміністратор. Кожному з них відведений доступ до

різного функціоналу системи. Є ряд функцій, які спільні для обох типів, зокрема: авторизація, перегляд новин та пошук новин.

Відношення між користувачами та варіантами використання зображено на діаграмі 1.28.

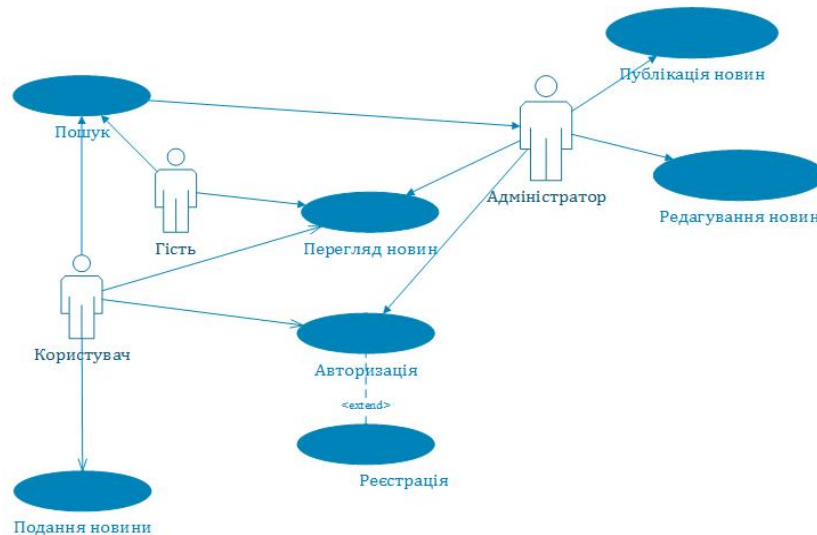


Рисунок 1.28 – Діаграма варіантів використання веб-орієнтованої системи «Екоуніверситет»

Розглянемо більш детально кожний із варіантів використання. Потрапивши на сайт можна без будь яких обмежень переглядати новини, але не можливо створити свою статтю та відправити її на перевірку і в подальшому на публікацію. Для використання цього функціоналу необхідно пройти «Авторизацію». У таблиці 1.5 представлено опис варіанту використання «Авторизація».

Таблиця 1.5

Варіант використання «Авторизація»

Контекст використання	Метою функції є ідентифікація користувача та надання або ненадання йому відповідних прав
Дійові особи	Користувач або Адміністратор
Передумова	-

Тригер	Натиснення кнопки «Увійти»
Сценарій	1 – Надання даних потрібних для авторизації 2 – Натиснення кнопки підтвердження авторизації
Пост умова	повідомлення про успішність/неуспішність авторизації

Логічний і концептуальний опис функціональних можливостей системи для сценарію «Авторизація», відображено на ескізі екранної форми, яка представлена на рисунку 1.29.

Рисунок 1.29 – Кадр сценарію «Авторизація»

Далі розглянемо варіант використання «Реєстрація». На сайті передбачений розподіл прав доступу між користувачами, для отримання додаткового функціоналу сайту, а саме можливості подання новини необхідно мати відповідні права доступу. При реєстрації автоматично присвоюється ідентифікатор «користувач», в подальшому надання прав доступу регулює адміністратор. У таблиці 1.6 представлено опис варіанту використання «Реєстрація».

Таблиця 1.6

Варіант використання «Реєстрація»

Контекст використання	Метою функції є додавання нового користувача у базу даних
Дійові особи	Гість
Передумова	-
Тригер	Натиснення кнопки «Реєстрація»
Сценарій	1- Надання даних потрібних для реєстрації 2- Натиснення кнопки «Зареєструватися»
Пост умова	Внесення у базу даних

Логічний і концептуальний опис функціональних можливостей системи для сценарію «Реєстрація», відображено на ескізі екранної форми, яка представлена на рисунку 1.30.

Реєстрація користувача

* Обов'язкове поле

Ім'я *

Логін *

Пароль *

Підтвердити пароль *

Адреса електронної пошти: *

Підтвердити адресу електронної пошти: *

Рисунок 1.30 - Кадр сценарію «Реєстрація»

Варіант використання «Подання новини», буде доступним лише авторизованому користувачу. Основна ціль процесу полягає у створенні новини та відправленні її на перевірку адміністратору.

У таблиці 1.7 представлено опис варіанту використання «Подання новини».

Таблиця 1.7

Варіант використання «Подання новини»

Контекст використання	Метою функції подання новини на перевірку та публікацію новини
Дійові особи	Користувач
Передумова	Успішна авторизація
Тригер	Вибір функції в меню
Сценарій	1 – Додавання контенту новини 2 – Натиснення кнопки відправити
Пост умова	Toast про успішність/неуспішність надсилання

Логічний і концептуальний опис функціональних можливостей системи для сценарію «Подання новини», відображено на ескізі екранної форми, яка представлена на рисунку 1.31.

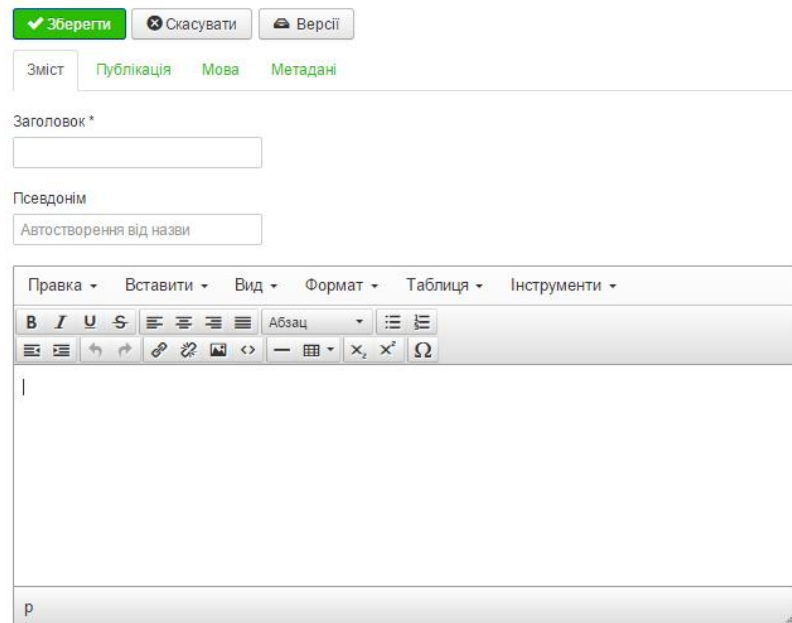


Рисунок 1.31 - Кадр сценарію «Подання новини»

Розглянемо варіант використання «Публікація статті». Публікація статті доступна лише адміністратору. Основна ціль процесу полягає у публікуванні статті на сайті для загального перегляду.

У таблиці 1.8 представлено опис варіанту використання «Подання статті».

Таблиця 1.8

Варіант використання «Публікація статті»

Контекст використання	Метою функції є публікація статті для загального перегляду..
Дійові особи	Адміністратор
Передумова	Стаття пройшла перевірку
Тригер	Натиснення кнопки «Публікувати новину»
Сценарій	<ol style="list-style-type: none"> 1. Додавання статті у відповідну категорію 2. Вибір пункту меню «Опубліковано»

Пост умова	Повідомлення про успішність/неуспішність збереження
------------	---

Логічний і концептуальний опис функціональних можливостей системи для сценарію «Публікація статті», відображено на ескізі екранної форми, яка представлена на рисунку 1.32.

Рисунок 1.32 – Кадр для сценарію «Публікація статті»

Нефункціональні вимоги показують як має виглядати система [13], а функціональні – що система повинна виконувати.[16]

Специфікація функціональних вимог представлена у таблиці 1.9

Таблиця 1.9

Специфікація функціональних вимог

Ідентифікатор вимог	Назва (варіанту вимоги)	Атрибути вимог		
		Пріоритет	Складність	Контакт
1	Реєстрація	Обов'язкове	низька	Гість
2	Вхід	Обов'язкове	низька	Гість, адміністратор
3	Створення статті	Обов'язкове	низька	Адміністратор, користувач
4	Публікація	Обов'язкове	низька	Адміністратор

	статті			
5	Пошук статті	Необов'язкове	Низька	Гість, користувач, адміністратор

При проектуванні програмного продукту необхідно чітко встановити функціональні та нефункціональні вимоги до системи. Список нефункціональних вимог представлений у таблиці 1.10

Таблиця 1.10

Специфікація нефункціональних вимог

Ідентифікатор вимоги	Назва вимоги	Атрибути вимог		
		Пріоритет	Складність	Контакт
1. Застосовність				
1.1	Основні вимоги застосовності нової системи відносно інших систем	Опційна	Низька	Адміністратор, користувач

Продовження таблиці 1.10

1.2	Вимоги по відповідальності стандартам графічного інтерфейсу користувача	Рекомендована	Низька	Адміністратор, користувач
2. Надійність				
2.1	Доступність	Обов'язкова	Середня	Адміністратор
2.2	Середній час безвідмовної роботи	Обов'язкова	Середня	Адміністратор
2.3	Точність	Обов'язкова	Середня	Адміністратор
3. Робочі характеристики				

3.1	Використання ресурсів	Рекомендована	Середня	Адміністратор
4. Проектні обмеження				
4.1	Вимоги до технології програмування	Рекомендована	Середня	Адміністратор

При роботі з обраною предметною областю необхідно знати усі терміни які вживаються, для цього було складено глосарій. Глосарій наведений у додатку А.

Висновки до першого розділу

У даному розділі було проаналізовано предметну область та усі існуючі аналоги проектованої системи. Також згідно теми було наведено коротку характеристику об'єкта автоматизації, приведений детальний огляд аналогів та описані усі вимоги до створюваної системи.

РОЗДІЛ II ПРОЕКТУВАННЯ

2.1. Розробка архітектури програмної системи

Оскільки новини та заходи повинні швидко долати шлях від автора до читача, виникає необхідність створення системи, яка буде виконувати публікацію матеріалів у реальному часі. Для цього рішення ідеально підходить клієнт-серверна архітектура, так як гарантує швидкість та гнучкість подання матеріалів користувачам веб-сервісу.

Структурна модель додатку є досить складною. Система матиме велику кількість класів що забезпечуватимуть можливість її функціонування. Більшість модулів системи відповідатимуть за інтерфейс або стосуватимуться особливостей забезпечення роботи з базою даних MySQL. Тому їх відображення на даному етапі є непотрібним, оскільки воно зробить діаграму класів занадто громіздкою та важкою для читання та розуміння.

Веб-додаток на етапі проектування буде використовувати 3 основні компоненти: веб-браузер, веб-сервер та базу даних. На рисунку 2.1 зображено діаграму компонентів, яка відображає залежності між компонентами програмного забезпечення.

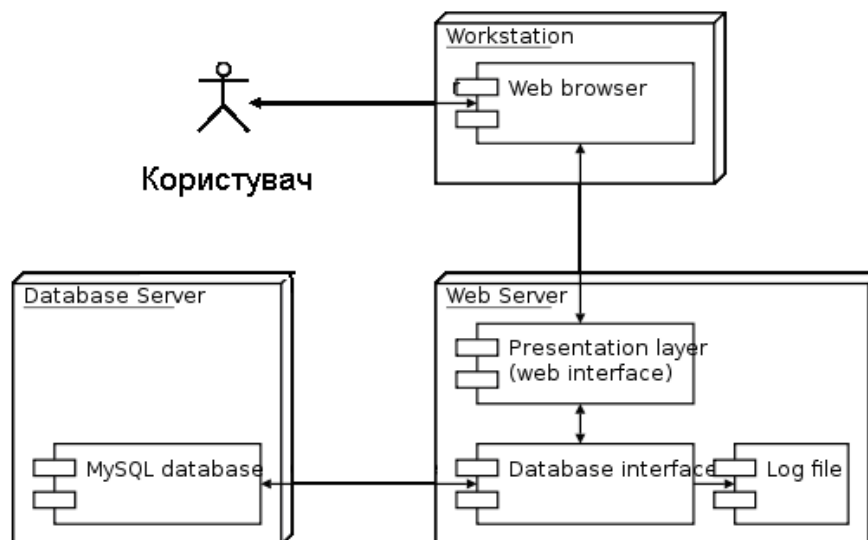


Рисунок 2.1 - Діаграма компонентів

При потраплянні неавторизованого користувача на сторінку сайту йому буде запропоновано зареєструватися. Процес додавання незареєстрованого користувача у базу даних для подальшої можливості авторизації на сайті відображено на рисунку 2.2.

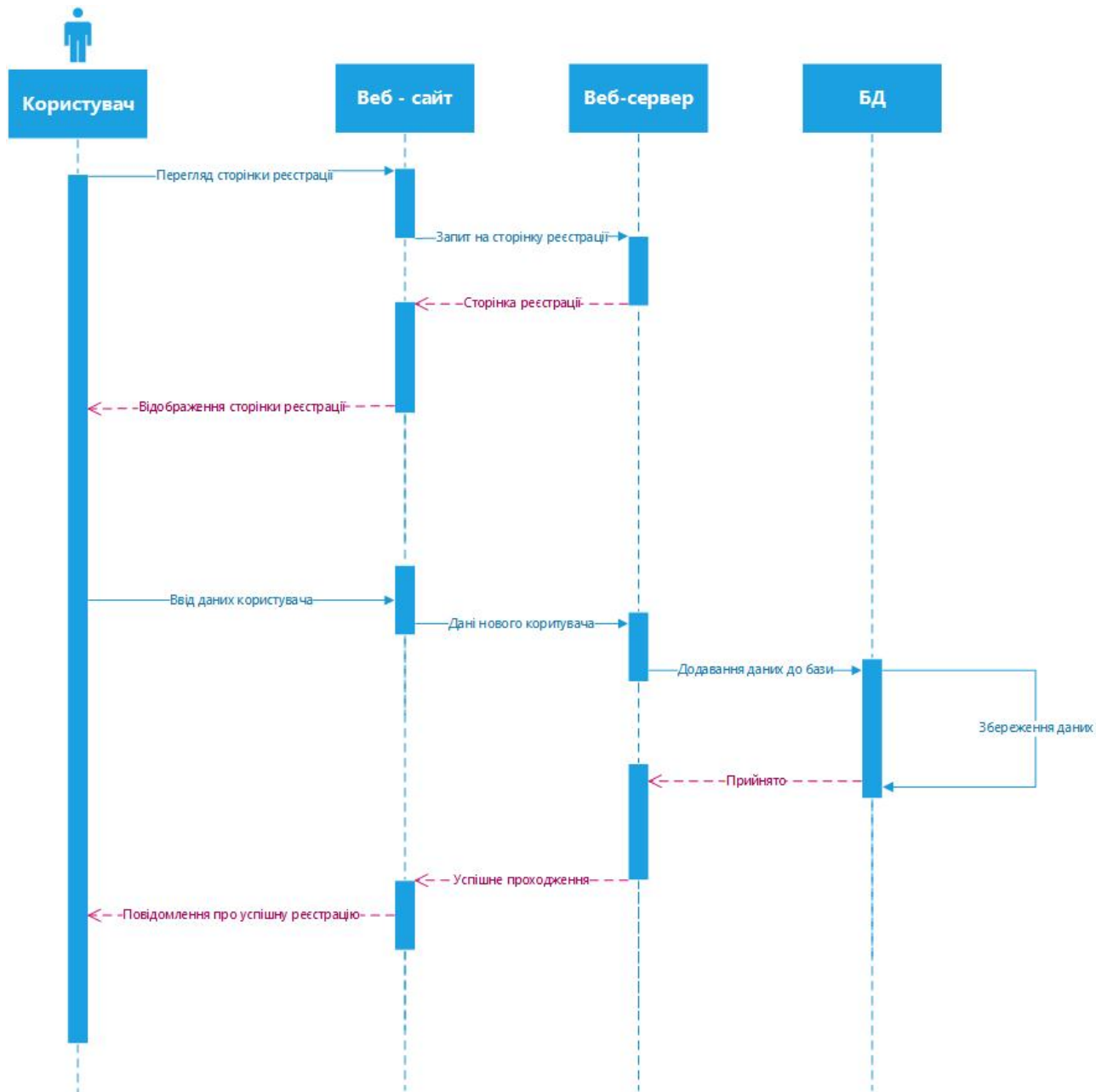


Рисунок 2.2 – Діаграма послідовностей для функції «Реєстрація»

При кожній дії користувача формуються відповідні запити, які переходять між необхідними етапами у структурі сайту. Так, посередником між

веб сервером і користувачем є веб сайт, між базою даних та веб сайтом – веб сервер.

З часом наповнюваність сайту новинами буде зростати, тому для зручного користування функціоналом необхідно додати блок пошуку. На рисунку 2.3 зображена діаграма послідовності, яка відображає взаємодії об'єктів, впорядкованих за часом для функцій «Пошук новин».

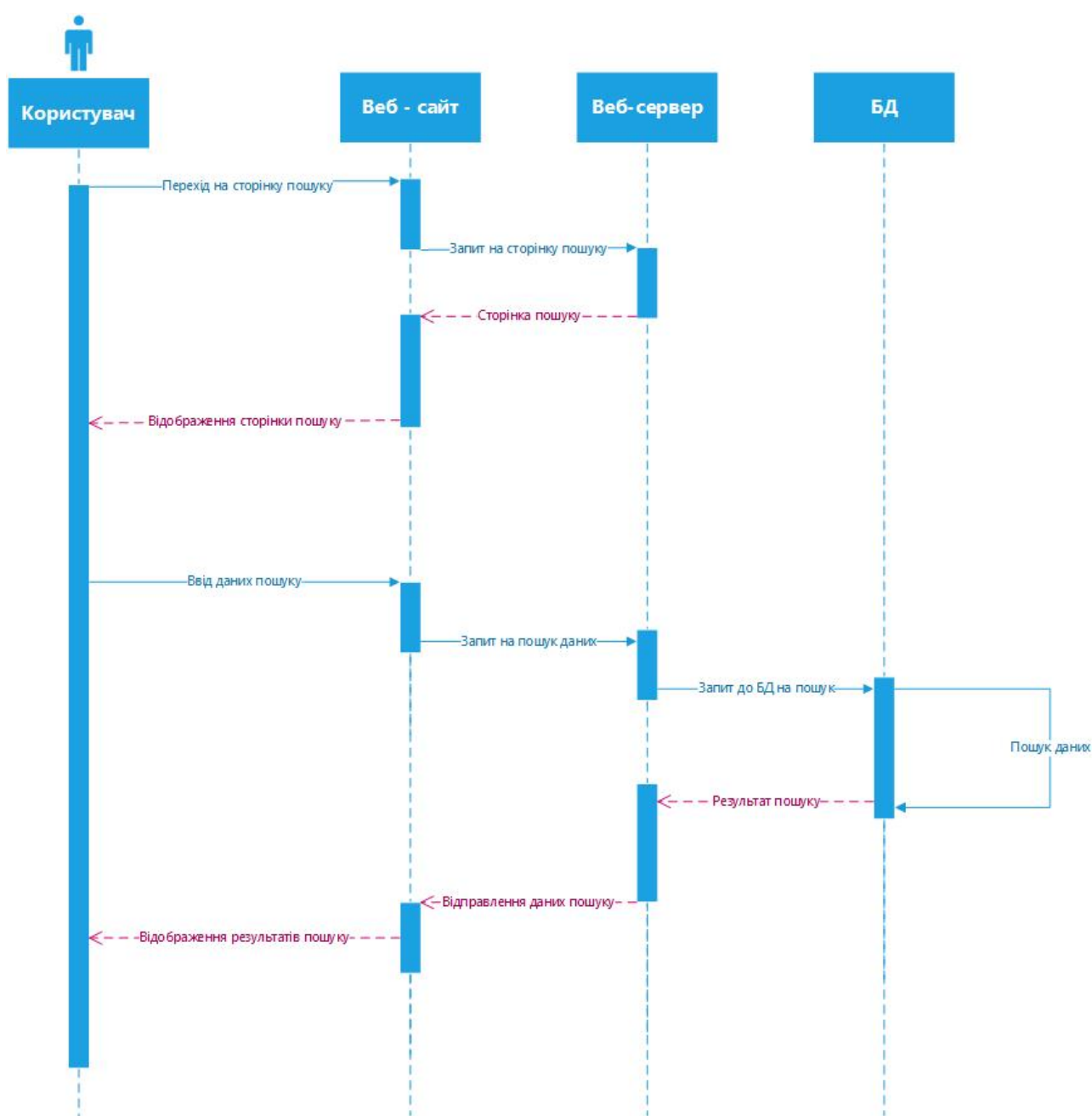


Рисунок 2.3 – Діаграма послідовності для функції «Пошук новин»

На сайті була реалізована функція, яка відповідає за права користування функціоналом для кожного користувача. Відповідно до статусу користувача у системі можна використовувати різну кількість функціональних компонентів, наприклад, звичайний користувач не може редагувати текст не своєї новини на відміну від адміністратора.

Подання новин на сайт є ключовою функцією проєктованого веб додатку. Дана можливість представлена пунктом «В мене ідея» в головному меню сайту.

Діаграму послідовностей для процесу «Подання новини» наведено у додатку Б.

При користуванні сайтом, система може перебувати у багатьох станах. Наприклад, для реєстрації користувача, при введенні даних (логіну та паролю) система перебуває у стані очікування введення усієї інформації. Діаграма станів системи для реєстрації користувача зображена на рисунку 2.4.



Рисунок 2.4 – Діаграма станів для процесу «Реєстрація»

Для пошуку новин система буде знаходитися у стані очікування критеріїв пошуку, а після введення усіх даних перейде у стан обробки даних. Детально цей процес зображено на рисунку 2.5.



Рисунок 2.5 – Діаграма станів для процесу «Пошук»

При додаванні новини необхідно ввести усі дані, у процесі чого система буде перебувати у стані очікування. Після натискання кнопки «Зберегти» переходить у стан опрацювання даних. Пройшовши два попередніх стани, система відобразить результат обробки.

Детальніше даний процес представлено за допомогою діаграми на рисунку 2.6.

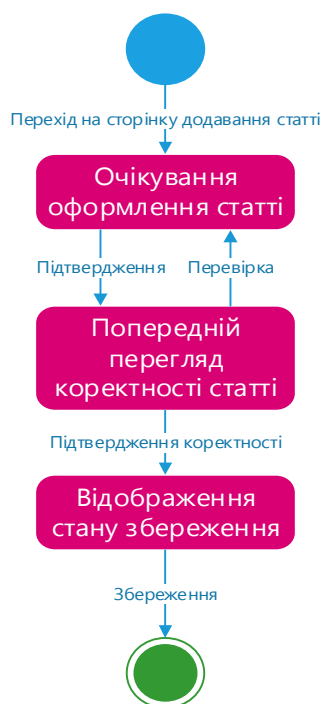


Рисунок 2.6 – Діаграма станів для процесу «Подання новини»

Далі будуть відображені діаграми діяльності. Модель діяльності в UML являє собою поведінку системи як певні роботи, котрі можуть виконувати як система, так і актор, причому послідовність робіт може залежати від прийняття певних рішень залежно від умов, що склалися.

Діаграми діяльності для функцій «Реєстрація користувача», «Пошук статей» та «Додавання статті» наведені у додатку В.

2.2 Проектування структури бази даних

Зважаючи на те, що розроблювана система повинна бути інтегрованою із веб-орієнтованим продуктом, потрібно розробити рекомендаційну структуру бази даних, яка буде містити дані які будуть необхідні для роботи веб-сайту, зокрема: збереження відомостей про користувача, відомості про тип користувача, збереження статті та усіх відомостей про неї. Інтеграція бази даних необхідна для оптимізації роботи веб сервісу з критично важливими даними які зберігаються у ході функціонального використання додатку. У якості РСУБД для цього проекту буде використовуватися MySQL.

Оскільки система являє собою веб орієнтований додаток, вона буде розміщена на віддаленому сервері. Усі дані будуть зберігатися безпосередньо на сервері та підтягуватися локально із подальшим відображенням користувачеві, тому необхідно мати постійний зв'язок серверу із базою даних та клієнту із сервером. Для цього рішення було обрано локальне розміщення бази даних на сервері із веб сайтом.

Для планування бази даних необхідно визначитися з ідентифікаторами, які представляють собою назви об'єктів у базі [14]

На етапі експлуатації системи для функціонування веб сайту будуть використовуватися наступні об'єкти та дані про них:

Таблиця 2.1

Список ідентифікаторів

Об'єкт	Властивість	Тип	Розмірність	Ідентифікатор
Користувач	• Ім'я	Текст	20	User_name
	• Прізвище	Текст	20	User_s_name
	• Логін	Текст	15	User_login
	• Пароль	Текст	15	Password
Матеріал	• Назва	Текст	100	M_name
	• Контент	Текст	255	Content
	• Дата створення	Дата		Create_date
Тип користувача	• Назва	Текст	15	Type_name
	• Права	Текст	15	Permissions
Категорія	• Назва	Текст	15	Category_name

Діаграма класів — статичне представлення структури моделі. Відображає статичні (декларативні) елементи: класи, типи даних, їх зміст та відношення. Діаграма класів структури функціонування веб-сайту зображена на рисунку 2.7.

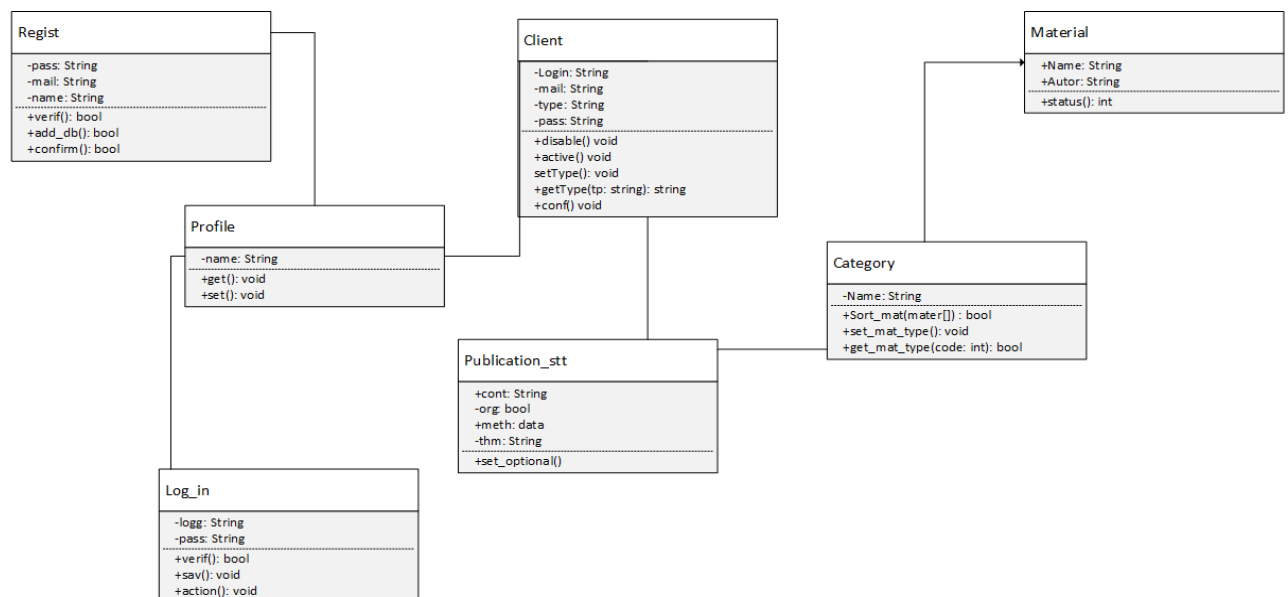


Рисунок 2.7 – Діаграма класів функціонального аспекту системи

Діаграми потоків даних (DFD) є найефективнішим інструментом при проектуванні запланованої системи. При побудові такого типу діаграм враховують процеси та сутності які взаємодіють з цими процесами.

DFD – є досить зручним та наглядним інструментом при проектуванні систем, оскільки дозволяє побачити потоки даних у зручному та доступному форматі.

На рисунку 2.8 зображено модель потоків даних для процесу «Публікація новини».



Рисунок 2.8 – Модель DFD процесу «Публікація новини»

На рисунку 2.9 зображено діаграму декомпозиції другого рівня для процесу «Публікація новини». Даний тип діаграми відображає розбиття головного процесу на поетапні модулі.

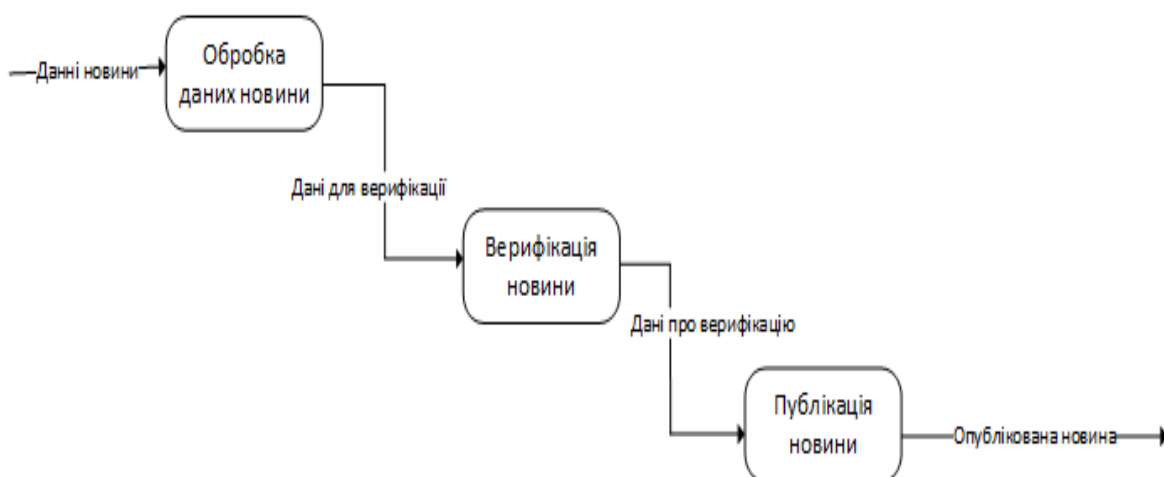


Рисунок 2.9 – Модель DFD процесу «Процес публікації»

На рисунку 2.10 зображено діаграму декомпозиції потоків даних процесу «Обробка даних новини»

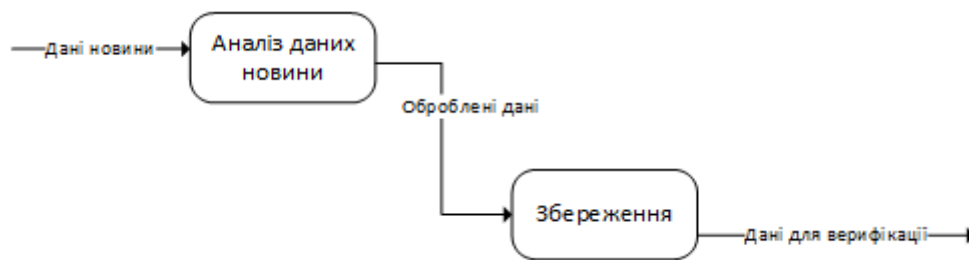


Рисунок 2.10 – Модель DFD процесу «Обробка даних новини»

На рисунку 2.11 зображено діаграму декомпозиції потоків даних процесу «Верифікація новини»



Рисунок 2.11 – Модель DFD процесу «Верифікація новини»

На рисунку 2.12 зображено діаграму декомпозиції потоків даних процесу «Публікація новини».



Рисунок 2.12 – Модель DFD процесу «Публікація новини»

Після опису вхідної та вихідної інформації, яка обробляється в рамках функцій предметної області розроблюваної програмної системи можна

зобразити глобальну модель даних у вигляді логічної моделі ERD на рисунку 2.13. [11]

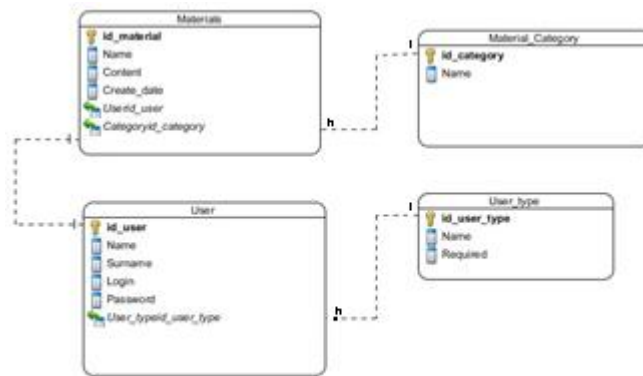


Рисунок 2.13 – Логічна модель ERD

Після побудови логічної моделі та аналізу даних можна побудувати фізичну модель ERD, яка зображена на рисунку 2.14.

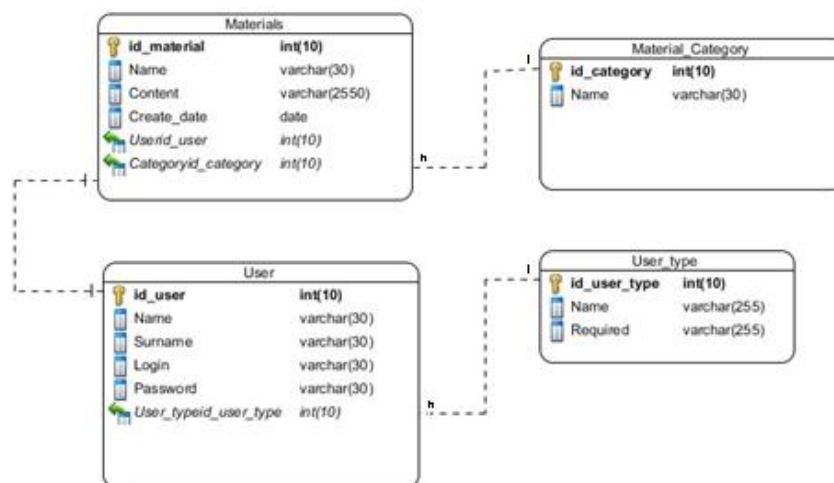


Рисунок 2.14 – Фізична модель ERD

Результатом фізичного проектування бази даних буде DDL код, який представлений у додатку Г.

Висновки до другого розділу

У даному розділі сформовано усі можливі варіанти використання системи за допомогою UML діаграм, також приведено діаграми станів та активностей проектованої системи.

Були сформовані необхідні вимоги до бази даних та створені ER діаграми для простоти організації БД в подальшому.

РОЗДІЛ III ПРОГРАМНА РЕАЛІЗАЦІЯ

3.1 Програмна реалізація проекту

3.1.1 Обґрунтування вибору мови програмування

Усі сучасні веб сайти відрізняються простотою доступу до даних та ефективністю донесення інформації до людини, дані критерії будуть ключовими при проектуванні веб-орієнтованої системи «Екоуніверситет».

Першочергові задачі, поставлені при проектуванні та налаштуванні сайту:

- Зручність експлуатації – згідно першої евристики Нільсена, а саме «Необхідність інформувати користувача про стан системи», потрібно розробити функціонал з найбільшою прозорістю та простотою орієнтації.
- Приємний та зрозумілий дизайн – на даному етапі розвитку веб технологій усі сайти характеризуються динамікою та наповненістю. Адже статичний сайт без рухомих або динамічних елементів є досить нудним, тому необхідно розробити систему, керуючись сучасними вимогами.
- Стабільність – здатність системи витримувати критичні навантаження та працювати при “стресових” умовах.
- Гнучкість – при проектуванні необхідно створити максимальний набір елементів управління на конфігурацій для підлаштування сайту під будь який матеріал.

Перечисливши усі необхідні критерії, можна обрати набір інструментів для виконання поставлених задач. Оскільки система є веб-орієнтованою, буде використовуватися мова програмування PHP та мова для написання стилів CSS.

PHP – мова програмування, яка широко розповсюджена при написанні веб орієнтованих додатків. Виконання коду відбувається у фоновому режимі, а користувач бачить лише сформований HTML. [1]

При правильному поєднанні можна добитися максимального унісону стабільності та естетичності.

Для вирішення поставлених задач було обрано систему керування вмістом Joomla.

Joomla – система керування вмістом, яка використовується для створення та управління веб-орієнтованими додатками. Головними перевагами даного програмного продукту є:

- простота експлуатації;
- можливість створення системи під будь які поставлені цілі;
- ефективність використання;
- максимальний набір конфігурацій;
- модель розповсюдження GNU.

У процесі використання сайту виникає необхідність зберігати та обробляти різного роду дані. Для вирішення цієї задачі було обрано реляційну модель даних, а саме СУБД MySQL.

Головними перевагами MySQL є різноманітність програм для управління та наявність цих програм у спеціалізованих локальних серверах. Вибір програми для управління базою даних був між MySQL Workbench та PhpMyAdmin. В подальшому робота буде здійснюватися у PhpMyAdmin. [2]

3.1.2 Організація інтерфейсу з користувачем

Початковим етапом створення проекту є заповнення та збереження категорій у адміністративній панелі.

При створенні сайту спочатку буде формуватися сама структура, а потім розроблятися шаблон.

Для початку необхідно натиснути на вкладку «Матеріали», обрати пункт «Менеджер категорій» та натиснути на «Створити нову категорію», після чого система запропонує обрати назву категорії та унікальний ідентифікатор (рис 3.1).

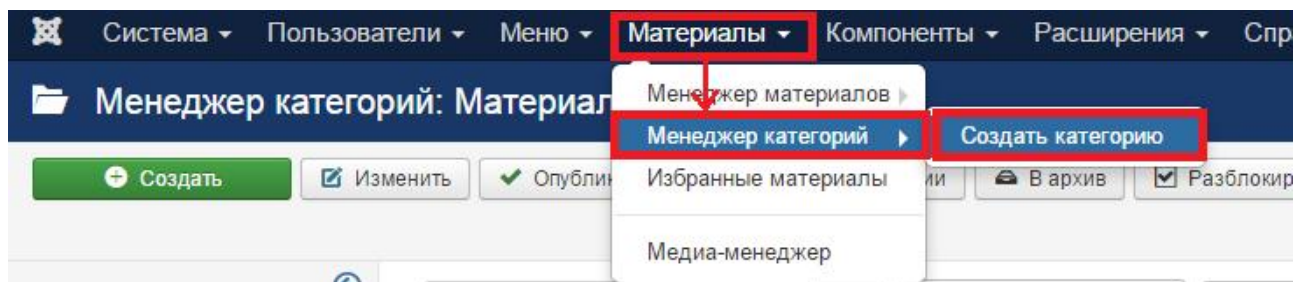


Рисунок 3.1 – Створення категорій новин

В результаті отримаємо список категорій для додавання статей та створення пунктів меню (рис. 3.2).

☰	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Концепція (Алиас: kontseptsiya)
☰	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Волонтери проекту (Алиас: volonteri-proektu)
☰	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Студентські ініціативи (Алиас: studentski-initsiativi)
☰	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Наукова діяльність (Алиас: naukova-diyalnist)
☰	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Міжнародна діяльність (Алиас: mizhnarodna-diyalnist)
☰	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Контакти (Алиас: kontakti)
☰	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Енерго університет (Алиас: energo-universitet)
☰	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Ідея! (Алиас: ideya)

Рисунок 3.2 – список готових категорій

Далі необхідно створити меню для зручного переходу між категоріями сайту. В адміністративній панелі необхідно обрати вкладку «Меню», потім пункт «Створити меню», після чого система запропонує ввести назву меню та тип меню (рис. 3.3).

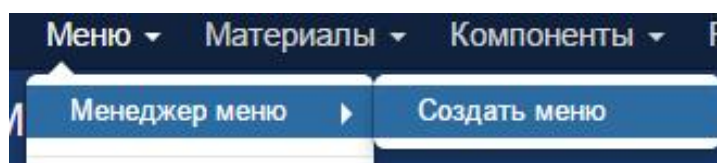


Рисунок 3.3 – Менеджер меню

При проведенні цих операцій сформованого меню для відображення на сайті не буде, необхідно додати модуль та присвоїти йому позицію (рис. 3.4).

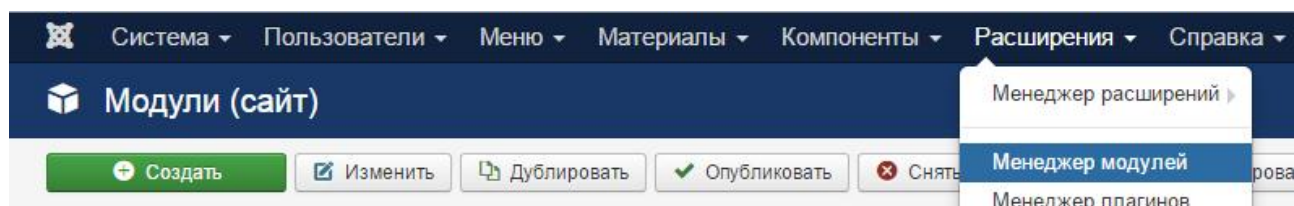


Рисунок 3.4 – Додавання модуля

Далі потрібно вибрати тип модуля (рис. 3.5).

Выбор типа модуля:

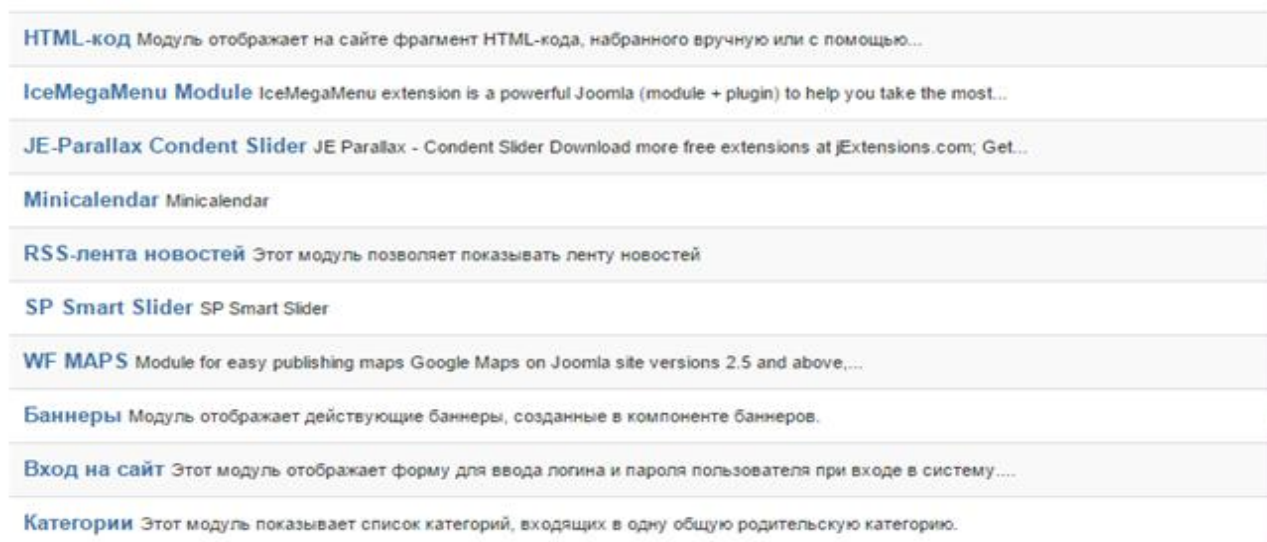


Рисунок 3.5 – Список доступних модулів

Необхідно обрати пункт «Меню», після чого система запропонує ввести назву та зробити прив'язку до уже створеного меню (рис 3.6). Дизайном реалізованого нами функціоналу потрібно займатися після введення на сайт усіх необхідних пунктів та категорій.

Рисунок 3.6 – Створення модуля «Меню»

Для додавання новини на сайт необхідно перейти в адміністративній панелі на вкладку «Матеріали», натиснути «Менеджер матеріалів» та обрати «Створити матеріал». Після даних дій система запропонує ввести заголовок, автора та сам текст статті з попереднім форматуванням (рис 3.7).

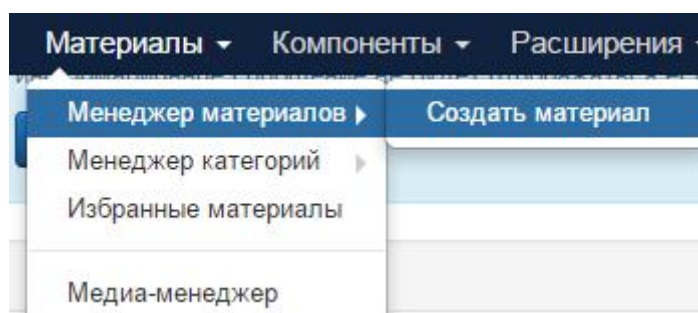


Рисунок 3.7 – Додавання статті

Перелічені вище дії дозволяють формувати основний функціонал сайту, але не редагувати зовнішній вигляд, для цього необхідно використовувати мову CSS.

Переходимо за шляхом: C:\OpenServer\domains\eco\templates\protostar\css та знаходимо там файл Style.css, який відповідає за оформлення сайту. У файлі додаємо частину коду представлену у додатку Д.

Після оформлення стилю сайту, а саме вигляд головного меню необхідно додати код для формування вигляду сторінок сайту [5].

```

body {
  -webkit-font-smoothing: antialiased;
  -moz-osx-font-smoothing: grayscale;
}
body.site {
  background-image: url (/images/back.jpg);
  background-size: cover;
  border-top: 3px solid #0088cc;
  padding: 20px;
  background-color: #f4f6f7;
}

```

Після створення вигляду сторінок додамо на сайт динамічних елементів та сформуємо кінцевий вигляд головного меню. Код формування головного меню наведений у додатку Е.

В результаті отримаємо сформоване головне меню сайту (рис 3.8).

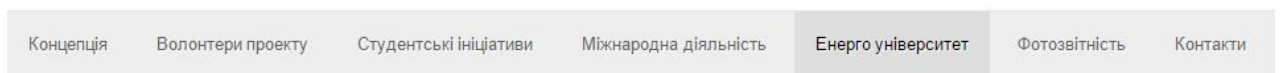


Рисунок 3.8 – Головне меню сайту

Під час роботи виникла необхідність додавання нових позицій на головну сторінку сайту, для цього [3]:

1. Відкриваємо файл `index.php` та дописуємо такий код:

```
<?php if ($this->countModules('menu_position') && $hideByView == false && $hideByEdit == false): ?>
```

```

  <div id="menu_position">
  <jdoc:include type="modules" name="menu_position" style="html5nosize" />
  </div>

```

```
><?php endif; ?>
```

2. Знаходимо файл `templateDetails.xml` та в кінці дописуємо: [6]

```

<positions>
  ...
  <position>top-a</position>
  <position>top-b</position>
  <position>top-c</position>
  <position>menu_position</position>
  ...
</positions>

```

Далі необхідно внести зміни у файл `index.php`, який знаходиться в папці увімкненого шаблону. [4] Лістинг коду наведений у додатку Є.

3.1.3 Вимоги до апаратного забезпечення

- Персональний комп'ютер;
- Встановлений інтернет-браузер;
- Доступ до мережі Інтернет;

3.2 Програмна реалізація бази даних

Після проектування бази даних за допомогою діаграм відбувається безпосередньо реалізація [8]. Для створення БД було обрано програму `PhpMyAdmin`, та СУБД `MySQL`.

Для створення таблиць використовуються SQL запити, приклад для створення таблиці `users` наведений нижче.

```
CREATE TABLE `db1`.`users` (
  `id` INT NOT NULL COMMENT '',
  `name` VARCHAR(400) NULL COMMENT '',
  `username` VARCHAR(150) NULL COMMENT '',
  `email` VARCHAR(100) NULL COMMENT '',
  `password` VARCHAR(100) NULL COMMENT '',
  PRIMARY KEY (`id`) COMMENT '');
```

В результаті отримаємо таблицю в базі даних яка відповідатиме за збереження даних про користувача (рис 3.9). Вона міститиме такі дані:

`Id` – Унікальний ідентифікатор користувача

`Name` – Ім'я користувача на сайті

`Username` – Ім'я користувача у системі

`Email` – електронна пошта користувача

password – Пароль користувача

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Дополнительно	Действие
1	id	int(11)			Нет	Нет	AUTO_INCREMENT	
2	name	varchar(400)	utf8_unicode_ci		Нет			
3	username	varchar(150)	utf8_unicode_ci		Нет			
4	email	varchar(100)	utf8_unicode_ci		Нет			
5	password	varchar(100)	utf8_unicode_ci		Нет			

Рисунок 3.9 – Таблица «users»

Далі для роботи з сайтом додамо користувача «Super User», для цього виконаємо наступний запит [9]:

```
INSERT INTO `db1`.`users` (`id`, `name`, `username`, `email`, `password`)
VALUES ('182', 'Super User', 'ecoadmin', 'sergiubudenchuk@gmail.com', '123456');
```

Результат даного запиту зображено на рисунку 3.10.

	id	name	username	email	password
	182	Super User	ecoadmin	sergiubudenchuk@gmail.com	123456

Рисунок 3.10 – Сформована таблиця users із полем «Super User»

Відповідним алгоритмом формуємо усі таблиці описані у розділі проектування бази даних.

На рисунку 3.11 зображено результат виконання DDL коду для виведення усіх категорій доданих адміністратором.

```
SELECT * FROM `udsat_content` LIMIT 0, 50
```
















<input type="checkbox"/>				4	66	Концепція	kontsepsiya
<input type="checkbox"/>				5	67	Наукова діяльність	naukova-diyalnist
<input type="checkbox"/>				6	68	Енерго університет	energo-universitet
<input type="checkbox"/>				7	69	Міжнародна діяльність	mizhnarodna-diyalnist
<input type="checkbox"/>				8	70	Посадка дерев	posadka-derev

Рисунок 3.11 – Таблиця сформованого sql запиту

Висновки до третього розділу

У даному розділі було детально описано програмну реалізацію проекту. Наведено фрагменти коду та зображення тих інструментів, які використовувались при розробці даної системи.

Програмно реалізовано базу даних та описано дії, які виконувались відносно конкретних таблиць.

РОЗДІЛ IV ТЕСТУВАННЯ ТА ДОСЛІДНА ЕКСПЛУАТАЦІЯ

4.1 Тестування

Тестування – ключовий етап розробки програмного продукту та подальшого впровадження і використання [10]. Основною метою процесу тестування є:

- перевірка на наявність помилок;
- перевірка сумісності з усіма платформами;
- відповідність усім вимогам якими керувалися проектувальники та розробники.

На етапі тестування можна виділити безліч підходів та етапів процесу, але необхідно враховувати затрати часу ресурсів.

Процес тестування може надати об'єктивну оцінку що-до якості програмного продукту, ризику відмови, яка є важливою, як для розробників так і для користувачів.

Доцільність виконання тестування присутнє ще на етапі незакінченого функціоналу, наприклад якщо є реалізована основна або запланована термінами написана функціональність.

При виконанні завдання тестування є не лише доцільним, але й необхідним етапом.

Під час проведення тестування програмного продукту були використані такі види тестування:[15]

- функціональне тестування;
- тестування сумісності;
- тестування продуктивності;
- тестування GUI;
- тестування безпеки;

4.1.1 Функціональне тестування

В процесі функціонального тестування буде виконано перевірку усіх функцій на відповідність до запланованого у технічному завданні показника. В ході роботи виконане ручне та автоматизоване тестування.

Автоматизоване тестування – виконане за допомогою плагіну для Mozilla Firefox під назвою Selenium Програмний продукт є досить зручним та опціональним для проведення тестів суто веб орієнтованих програмних продуктів.

На рисунку 4.1 представлено результат тестування функції «Авторизація» та подання тексту новини на перевірку адміністратору.

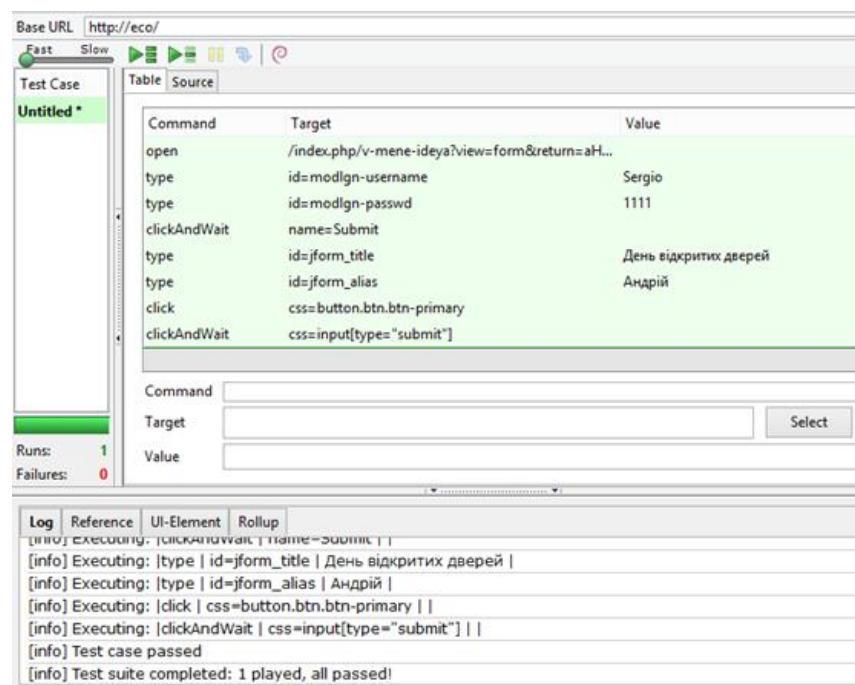


Рисунок 4.1 – Тестування функції подання статті у Selenium

На рисунку 4.2 представлено test case для функції подання новини авторизованим користувачем.

Test1		
open	/index.php/component/users/?view=login	
type	id=modlgn-passwd	1111
clickAndWait	name=Submit	
clickAndWait	link=В мене ідея!	
type	id=jform_title	Ghbdsn
type	id=jform_alias	sad
click	css=button.btn.btn-primary	
clickAndWait	css=input[type="submit"]	

Рисунок 4.2 – Test case для подання новини користувачем

При виконанні роботи було спроектовано тести для функціонального тестування та тестування безпеки, детально описано кожен тестовий випадок, кроки та очікуваний результат. Спроектовано 5 функціональних тестових випадків. Результати приведено у таблиці 4.1.

Таблиця 4.1

Функціональні тестові випадки

id	Назва	Кроки	Очікуваний результат	Статус
1	Авторизація	<ul style="list-style-type: none"> Заповнення форми авторизації Натиснення кнопки «Вхід» 	Повідомлення системи про успішну авторизацію Доступ функціоналу «В мене ідея»	Виконано
2	Реєстрація	<ul style="list-style-type: none"> Перехід на сторінку реєстрації Заповнення форм реєстрації Натиснення кнопки «Зареєструватися» 	Повідомлення системи про успішну реєстрацію Додавання користувача у систему	Виконано
3	Перегляд новин	<ul style="list-style-type: none"> Вибір категорії з новиною 	Відкриття сторінки з новиною	Виконано
4	Подання тексту новини	<ul style="list-style-type: none"> Натиснення кнопки «В мене ідея!» Заповнення усіх важливих форм Натиснення кнопки «Зберегти» 	Успішне додавання новини па перевірку адміністратору	Виконано

Продовження таблиці 4.1

5	Пошук новини	<ul style="list-style-type: none"> • Заповнення форми пошуку • Натиснення кнопки «Пошук» 	Перехід на сторінку з результатами пошуку	Виконано
---	--------------	--	---	----------

В таблиці 4.2 наведено результати розподілу функціональних тестових випадків та наборів текстових даних для них за варіантами використання, на прикладі.

Таблиця 4.2

Розподіл функціональних тестових випадків та наборів текстових за варіантами використання

Варіанти використання	Тестові випадки	Тестові данні
Check Authorization	1	2
Check option "Add_news"	1	4
Check option "view_news"	1	7
Check option "view news_search_result"	1	3
Check option "registration"	1	2
Загалом:	5	26

4.1.2 Тестування сумісності

Тестування сумісності – необхідний етап тестування, у якому перевіряється коректність роботи програмного продукту у більшості середовищ. Оскільки програма «Екоуніверситет» є веб-орієнтованою, то тестування буде проводитись на сумісність із такими браузерами:

- Internet Explorer
- Opera
- Google Chrome
- Vivaldi

- Mozilla FireFox
- Microsoft Edge

У ході тестування успішно пройшли 19 із 20 усіх можливих тестів, отже тестування можна рахувати частково успішним.

Дефекти виявлені при тестуванні:

Дефект №1: В браузері Vivaldi, на головній сторінці не відображало необхідних шрифтів.

Опис: при відображенні сайту на усіх браузерах необхідно зберігати цілісність та стабільність система, а саме: коректність відображення усіх модулів, визначені шрифти та їх розміри.

Усунення дефекту: було додано у код такий запис - `<meta http-equiv="Content-Type" content="text/html; charset=utf-8 " />`.

4.1.3 Тестування продуктивності

Основним показником продуктивності є час, затрачений на очікування відповіді сервера при виконанні операцій користувачем. Основні показники можна дізнатися перевіривши систему на швидкодію.

Також для перевірки загальної продуктивності системи було виконано такі види тестування:

- Тестування стабільності;
- Конфігураційне тестування;
- Навантажувальне тестування;
- Стрес-тестування;

Результати тестування: 15 із 15 тестів успішно пройшли даний тип тестування. Час з'єднання із сервером – 0,14 с., час відповіді сервера – 0,32 с. Виходячи з отриманих результатів, тестування вважається пройденим.

4.1.3 Тестування GUI

Графічний інтерфейс користувача – основна оболонка яку буде бачити людина при користуванні сайтом. [12] Задачею тестування графічного інтерфейсу є виявлення помилок наступного характеру:

- помилки в функціонуванні інтерфейсу;
- незворотні помилки при взаємодії з інтерфейсом;
- втрата чи спотворення даних при передачі через елементи інтерфейсу;
- невідповідність інтерфейсу до документації.

Функціональне тестування графічного інтерфейсу буде проводитися у п'ять етапів:

- аналіз вимог до користувацького інтерфейсу;
- розробка тест плану для перевірки;
- виконання тестів;
- порівняння відношення створеного інтерфейсу до вимог;
- створення та оголошення результатів тестування.

Результати тестування: при проведенні тестів графічного інтерфейсу користувача було перевірено:

- відповідність інтерфейсу до ряду евристик Нільсена, а саме:
 - прозорість стану системи – користувач завжди знає де він знаходиться, завдяки так званим «хлібним крихтам» у нижньому блоці сайту;
 - схожість системи з реальним оточенням – при створенні системи було використано прості та зрозумілі терміни на усіх сторінках сайту;
 - запобігання помилкам – при помилкових діях користувача буде відображено підказку з даними, які необхідно виконати для нормального функціонування системи;
- відповідність інтерфейсу до вимог вказаних у завданні;
- зручність експлуатації інтерфейсу.

Усі пункти тестування успішно пройдені та відповідають вимогам. Користувацький інтерфейс повністю відповідає поставленим цілям.

4.2 Розгортання програмного продукту

При створенні сайту використовувався локальний сервер з доменним ім'ям localhost, але для повноцінного функціонування сайту та доступу користувачів необхідно розмістити його на Інтернет-хостинг.[7] Для розташування на хостингу необхідно:

1. обрати хостинг та доменне ім'я сайту;
2. через FTP менеджер розмістити усі необхідні файли на хостингу;
3. перенести базу даних на хостинг за допомогою представлених власником спеціалізованих програм;
4. налаштувати права доступу до конфігурацій сайту;

Після виконаних вищеперелічених дій усі користувачі, які мають адресу сайту матимуть доступ до перегляду інформації та статей на сайті.

4.3 Інструкція користувача

Інструкція користувача є невід'ємною частиною при навчанні роботи з створеним програмним продуктом. Структура даного розділу буде включати поетапний опис роботи із компонентами системи. На рисунку 4.3 зображено головну сторінку сайту.

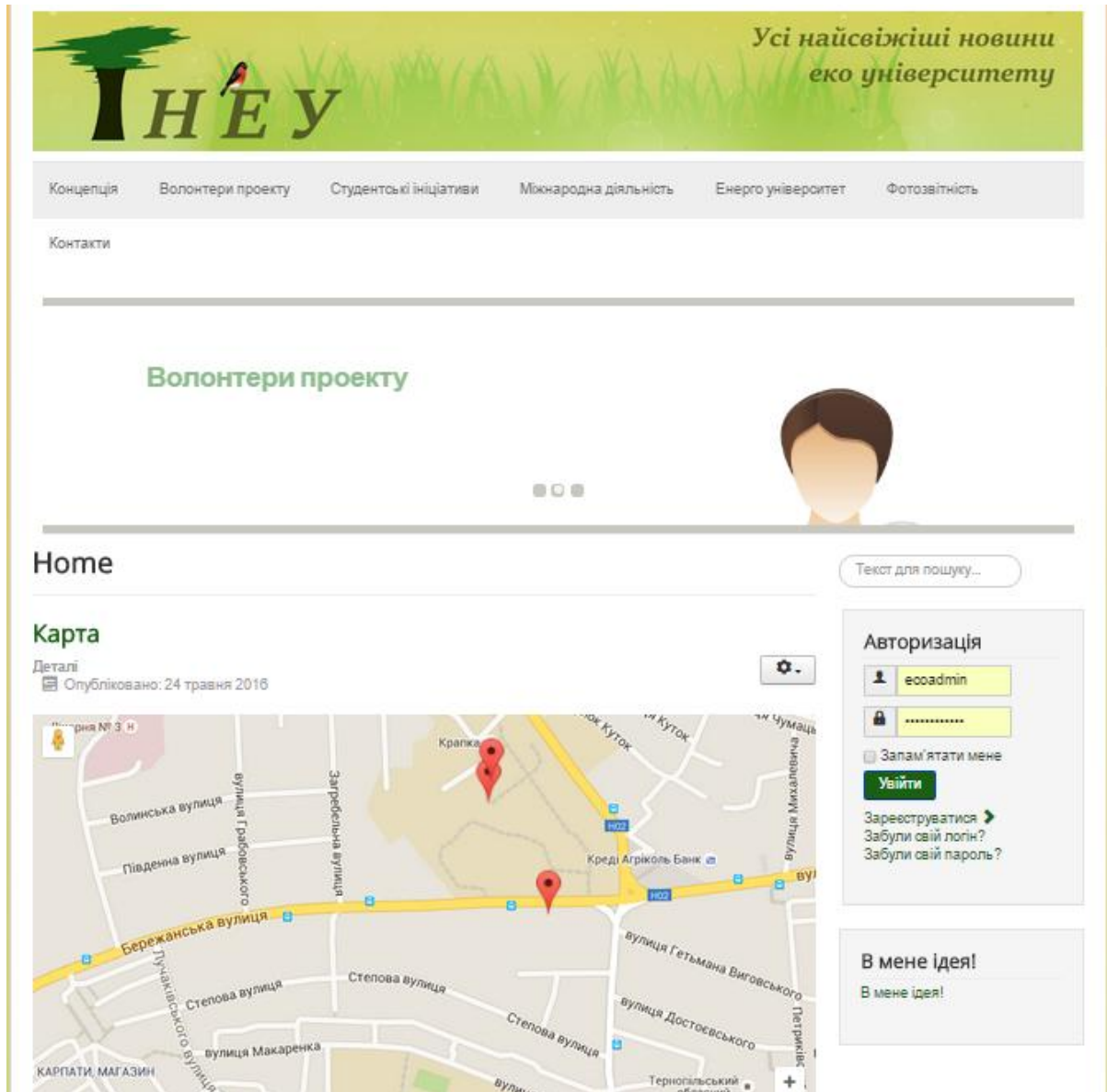


Рисунок 4.3 – Головна сторінка сайту

4.3.1 Додавання матеріалу на сайт

Для додавання матеріалу на сайт є два шляхи:

1. Додавання матеріалу адміністратором сайту

Першим етапом є отримання прав адміністратора на сайті. Для цього необхідно перейти на сторінку сайту та у стрічці посилання написати: eco/administrator.php

У вікні яке відкриється необхідно ввести логін та пароль облікового запису адміністратора (рис.4.4)

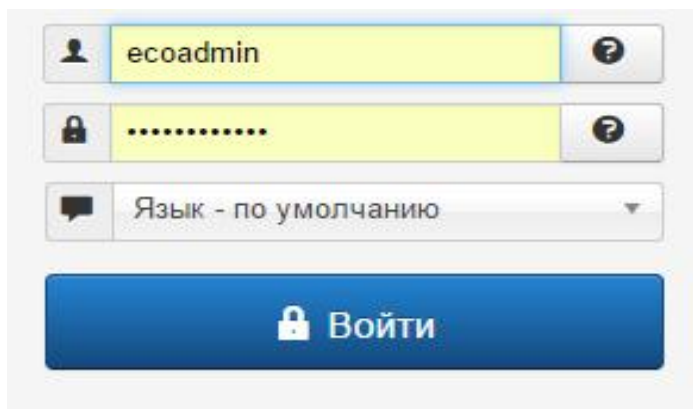


Рисунок 4.4 – Авторизація адміністратора

Після чого відкриється головне адміністраторське меню. У головному вікні, на верхній панелі навігації потрібно вибрати пункт «Матеріали». Далі у спливаючому меню необхідно вибрати пункт «Менеджер матеріалів» та натиснути на «Додати матеріал» (рис. 4.5).

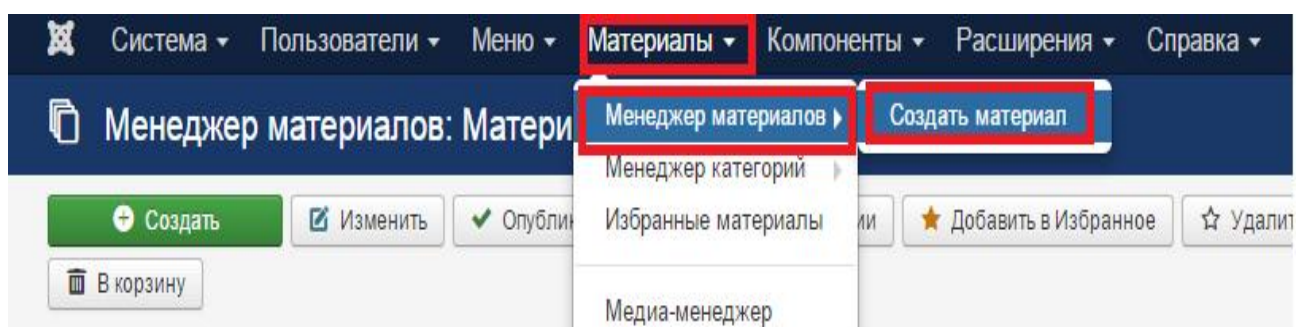


Рисунок 4.5 – Додавання матеріалу за допомогою менеджера

Після виконаних дій відбудеться перехід на сторінку із додаванням новини(рис.4.6).

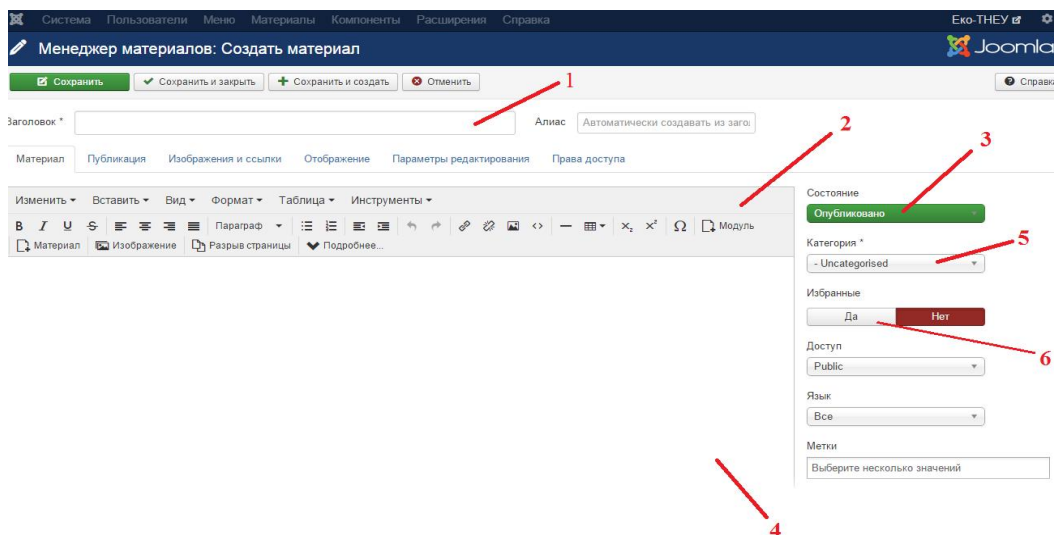


Рисунок 4.6 – Інтерфейс додавання матеріалу на сайт

Як видно на рисунку 4.6 інтерфейс представлений у вигляді функціональних блоків, кожен з яких відповідає за вигляд матеріалу на сторінці сайту. Присутні такі блоки:

1. Назва матеріалу, яка буде відображатися у заголовку статті.
2. Редактор тексту.
3. Статус новини (опублікована/неопублікована).
4. Форма вводу тексту матеріалу.
5. Категорія, у якій буде знаходитися матеріал.
6. Можливість додавання матеріалу на головну сторінку.

Після заповнення усіх необхідних полів потрібно натиснути кнопку «Зберегти».

При умові заповнення усіх ключових полів система покаже повідомлення про успішне додавання матеріалу (рис 4.7).

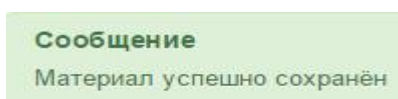


Рисунок 4.7 – Повідомлення про успішну публікацію матеріалу

2. Додавання матеріалу зареєстрованим користувачем

Для додавання матеріалу користувачем необхідно авторизуватися на сайті за допомогою форми авторизації (рис 4.8)

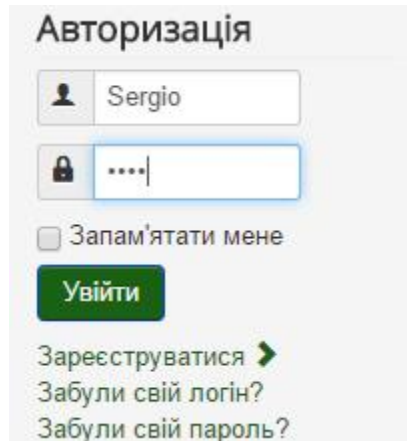


Рисунок 4.8 – Панель авторизації користувача

Після успішної авторизації необхідно натиснути на кнопку «В мене ідея!» (рис.4.9)




Рисунок 4.9 - Кнопка подання новини зареєстрованим користувачем

При додаванні матеріалу без прав адміністратора буде доступний не весь функціонал системи, а лише основні моменти необхідні для подальшої публікації. На рисунку 4.10 зображено основні компоненти потрібні для наповнення.

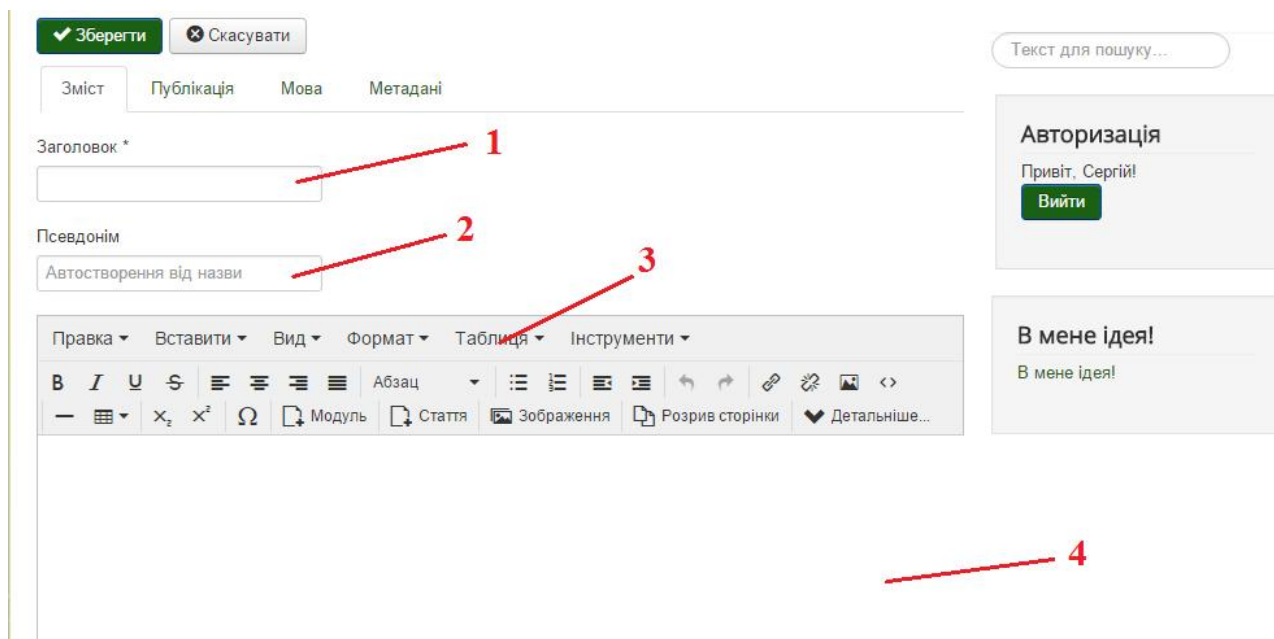


Рисунок 4.10 – Додавання матеріалу користувачем

1. Назва матеріалу.
2. Псевдонім користувача.
3. Меню форматування тексту.
4. Поле для вводу тексту матеріалу.

При додаванні матеріалу користувачем не можна вибрати категорію для публікації та статус публікації, цей функціонал доступний лише адміністратору. Користувач лише відправляє матеріал на перегляд, редагування та затвердження.

Щоб матеріал який написав зареєстрований користувач потрапив на сайт, адміністратор повинен переглянути його у менеджері матеріалів. Після попереднього перегляду необхідно обрати категорію матеріалу та натиснути на кнопку «Зберегти» (рис 4.11).

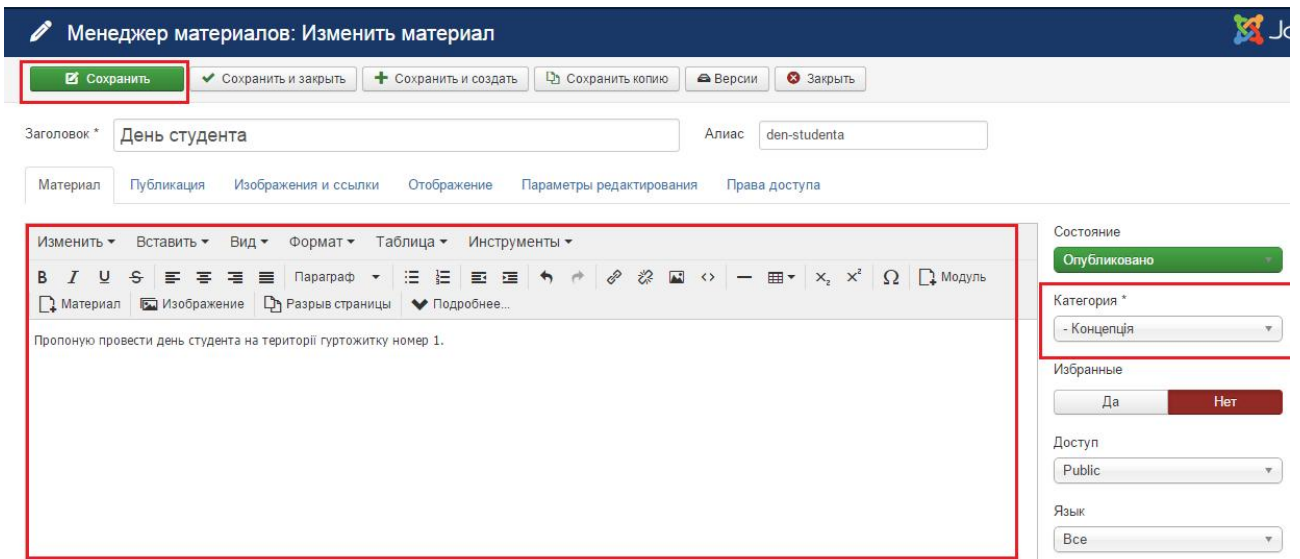


Рисунок 4.11 – Публікація матеріалу адміністратором

4.3.2 Реєстрація нового користувача

Новий користувач сайту розпізнається системою як гість, тому для використання додаткового функціоналу потрібно зареєструватися у системі.

Щоб зареєструватися необхідно у блоці сайту «Авторизація» обрати пункт меню «Зареєструватися» (рис 4.12).

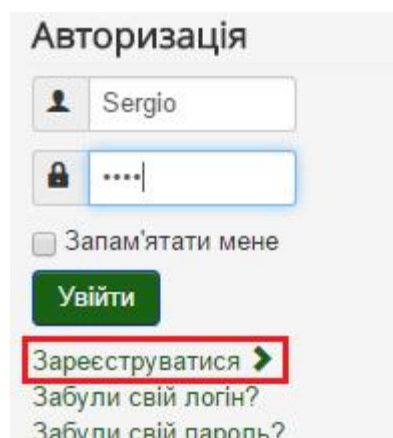


Рисунок 4.12 – Реєстрація нового користувача

Після натискання на посилання відбудеться перехід на форму реєстрації, де необхідно заповнити усі ключові поля позначені “*” (рис 4.13).

Реєстрація користувача

* Обов'язкове поле

Ім'я *

Логін *

Пароль *

Підтвердити пароль *

Адреса електронної пошти: *

Підтвердити адресу електронної пошти: *

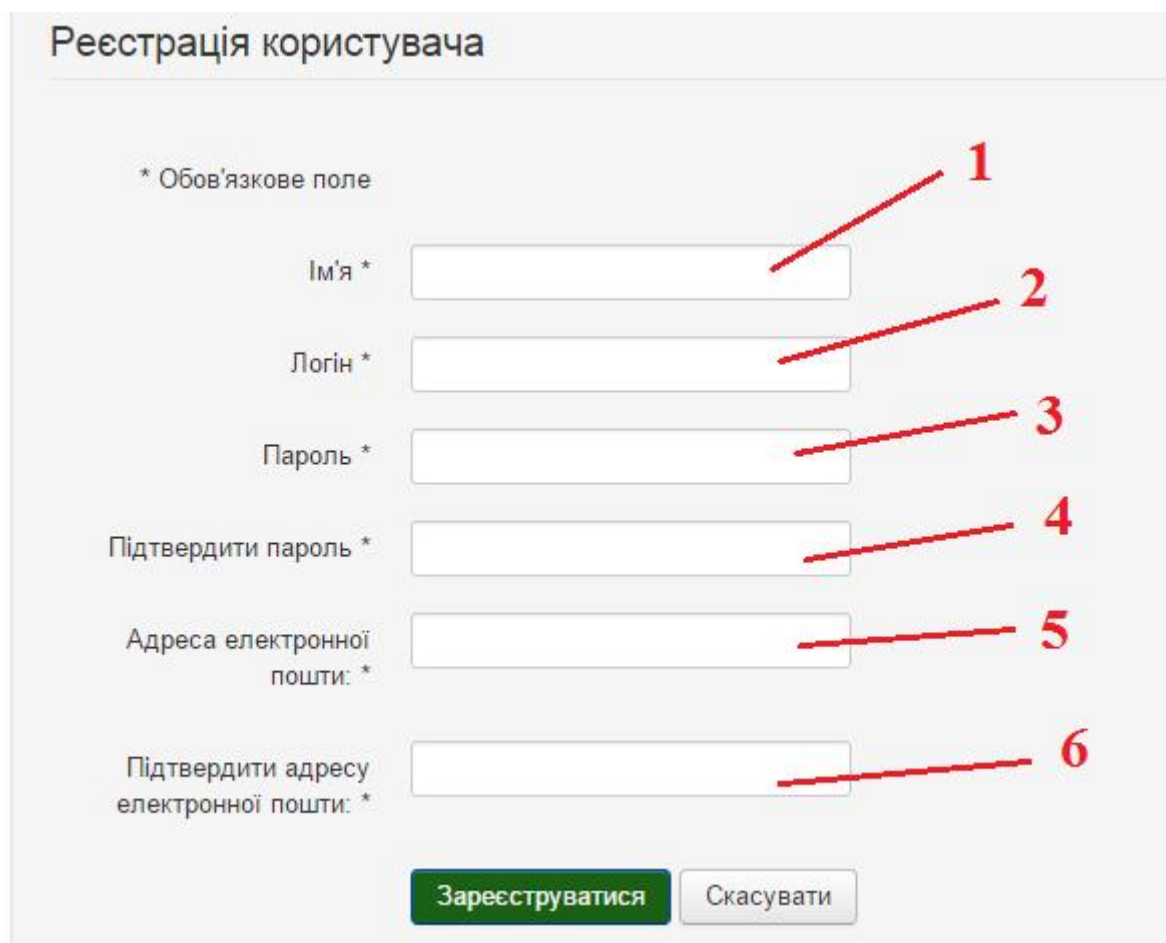
The image shows a registration form titled "Реєстрація користувача". It includes a legend for required fields, six input fields with asterisks, and two buttons. Red lines with numbers 1 through 6 point to the input fields for Name, Login, Password, Confirm Password, Email, and Confirm Email respectively.

Рисунок 4.13 – Форма реєстрації

На даній форм представлені такі поля:

1. Ім'я користувача, яке буде відображатися адміністратору при надсиланні матеріалу.
2. Логін, необхідний для авторизації на сайті.
3. Пароль користувача.
4. Підтвердження паролю.
5. Електронна адреса користувача.
6. Підтвердження електронної адреси.

Після введення усіх необхідних даних і натискання кнопки «Зареєструватися» на екрані з'явиться повідомлення про успішну реєстрацію (рис. 4.14).

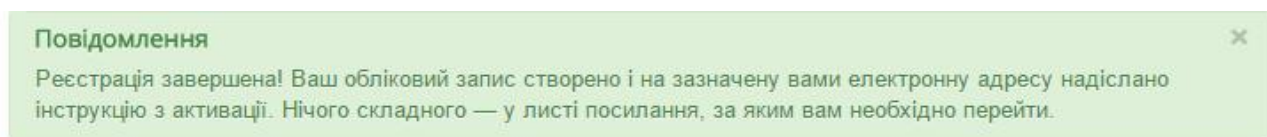


Рисунок 4.14 – Повідомлення про успішну реєстрацію

На головній сторінці сайту присутня карта із маркерами, на якій представлено місцезнаходження заходів на території студентського містечка. Додавання маркерів на карту доступне лише адміністратору у відповідному менеджері модулів (рис. 4.15).

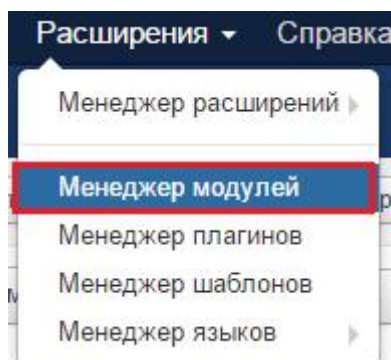


Рисунок 4.15 – Менеджер модулів

Після вибору пункту меню відбудеться посилання на сторінку із встановленими модулями, де потрібно обрати «WF MAPS». Перед адміністратором відкриється панель управління модулем, у якому потрібно перейти на вкладку «Налаштування списку маркерів» та заповнити усі поля (рис 4.16).

Модуль Описание Привязка к пунктам меню Права доступа Расположение и настройка элементов управления **Настройка списка маркеров**

_____ Маркер 1 _____

Маркер включен Да Нет

Имя маркера 1

Широта 2

Долгота 3

Иконка маркера Выбрать

Описание 4

Рисунок 4.16 – Додавання нового маркера

1. Назва маркеру.
2. Позичіонування маркеру – широта.
3. Позичіонування маркеру – довгота.
4. Короткий опис маркеру.

Після заповнення усіх необхідних полів на головній сторінці сайту з'явиться карта із маркером (рис. 4.17).

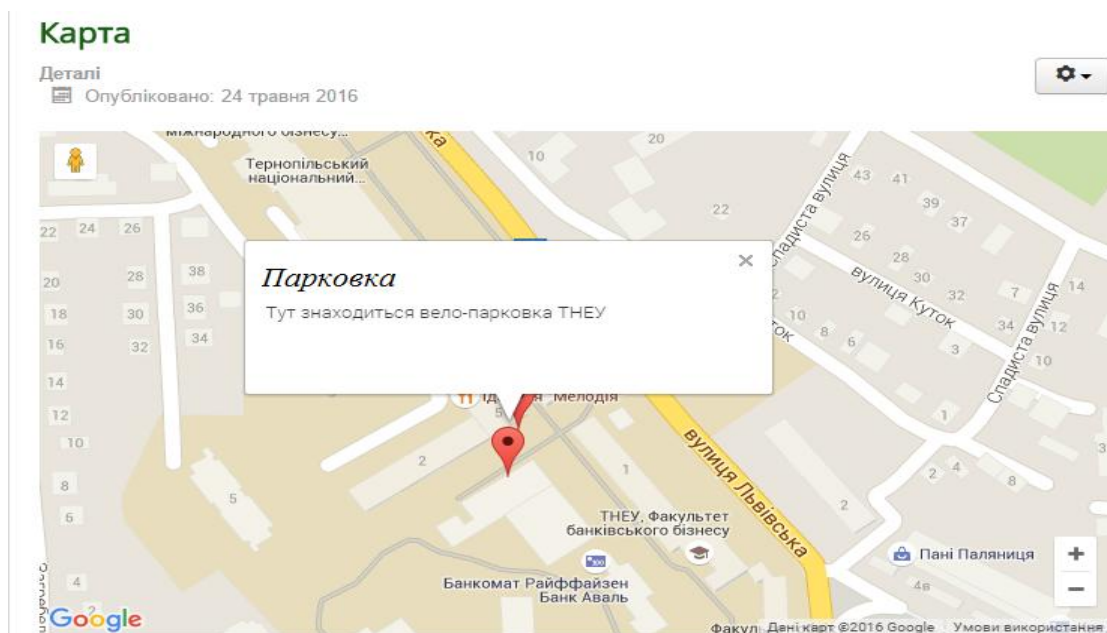


Рисунок 4.17 – Результат додавання маркера

Висновки до четвертого розділу

У розділі тестування сформовані основні етапи та плани тестування програмного продукту, також проведено різного роду типи тестування для перевірки надійності, стабільності та зручності впроваджуваної системи.

ВИСНОВОК

Процес подання новини на доступний для усіх користувачів сервіс є досить простою, але водночас корисною функцією для інформування клієнтів про усі екологічні заходи на території університету, тому для детального аналізу та автоматизації було обрано саме цю тему.

Проектування та створення інтернет-сайту «Екоуніверситет» є досить хорошим показником серйозного ставлення навчального закладу по відношенню до проблеми еко – забрудненості. Головним чином впровадження даного програмного продукту демонструє зацікавленість університету у покращенні екологічного стану як довкілля так і території навчального закладу, що призведе до покращення навчального процесу та життя студентів на території студентського містечка.

Зробивши детальний аналіз предметної області та проаналізувавши правильність підходу до вибору архітектури додатку можна сказати, що створена система не є фінансово затратною та зручною у використанні.

Програмна реалізація даної сфери діяльності може використовуватися для швидкого, зручного та ефективного інформування користувачів про останні новини та майбутні заходи на території навчального закладу. Оскільки програма є веб-орієнтованою, вона не прив'язана до робочого місця користувача та може використовуватися з будь якої точки світу де присутнє з'єднання з мережею Інтернет.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. PHP – найбільш популярна мова для веб програмування [Електронний ресурс] Режим доступу: <http://asaweb.com.ua/ua/php-5/>
2. Створення бази даних в phpMyAdmin [Електронний ресурс] – Режим доступу: <http://programming.in.ua/web-design/allphp/139-createdatabase.html>
3. Об'єктно-орієнтовне програмування і PHP 5[Електронний ресурс] – Режим доступу: <http://phpist.com.ua/php/17-oopphp>
4. Плавна прокрутка сторінки з допомогою JavaScript[Електронний ресурс] – Режим доступу: <http://ruszura.in.ua/korysni-efekty-dlya-sajtu/plavna-prokrutka-storinky-z-dopomohoyu-javascript.html>
5. Как изменить шаблон joomla[Електронний ресурс] – Режим доступу: <http://www.site-do.ru/joomla/joomla16.php>
6. Етапи створення модулів в joomla 3[Електронний ресурс] – Режим доступу: <http://joomla-create.ru/uroki/raznoe/1516-kak-sozdat-modul-v-joomla-3>
7. Урок 7: Перенос сайта Joomla на хостинг[Електронний ресурс] – Режим доступу: <http://cms-joomla-help.com/kak-sozdat-sajt/urok-7-perenos-sajta-joomla-na-khosting>
8. Майкл Моррісон. Вивчаємо PHP і MySQL, 2010. – 256 с.
9. Стивен Хольцнер. PHP у прикладах. / Стивен Хольцнер . М.: 000 «Бином-Пресс», 2007 г. Пер. з англ. 352 с
10. Ларрі Ульман. Ульман Л. Основи тестування:/Ларрі Ульман. Пер. з англ. - М.: ДМК Пресс, 2001. -288 с.
11. Томсон Лаура. Розробка Web додатків за допомогою MySQL і PHP: Пер. з англ. /Лаура Томсон, Люк Вел
12. Гутманс Э., Баккен С, Ретанс Д. GUI тестування./ Пер. з англ. СПб: Символ- Плюс, 2006. 704 с.
13. Оманец А. С. Функціональні та нефункціональні вимоги до ПЗ/ Андрій Степанович Оманец. – Київ, 2007. – 207 с.

14. Степанюк М. А. Взаємодія PHP та MySQL / М. А. Степанюк, Є. О.

Олежинн, А. В. Петрук. – Одеса: Маяк, 2009. – 314 с.

15. Види тестування ПЗ : дис. канд. техн. наук : 7.05010301 / . – Київ, 2006. –

139 с.

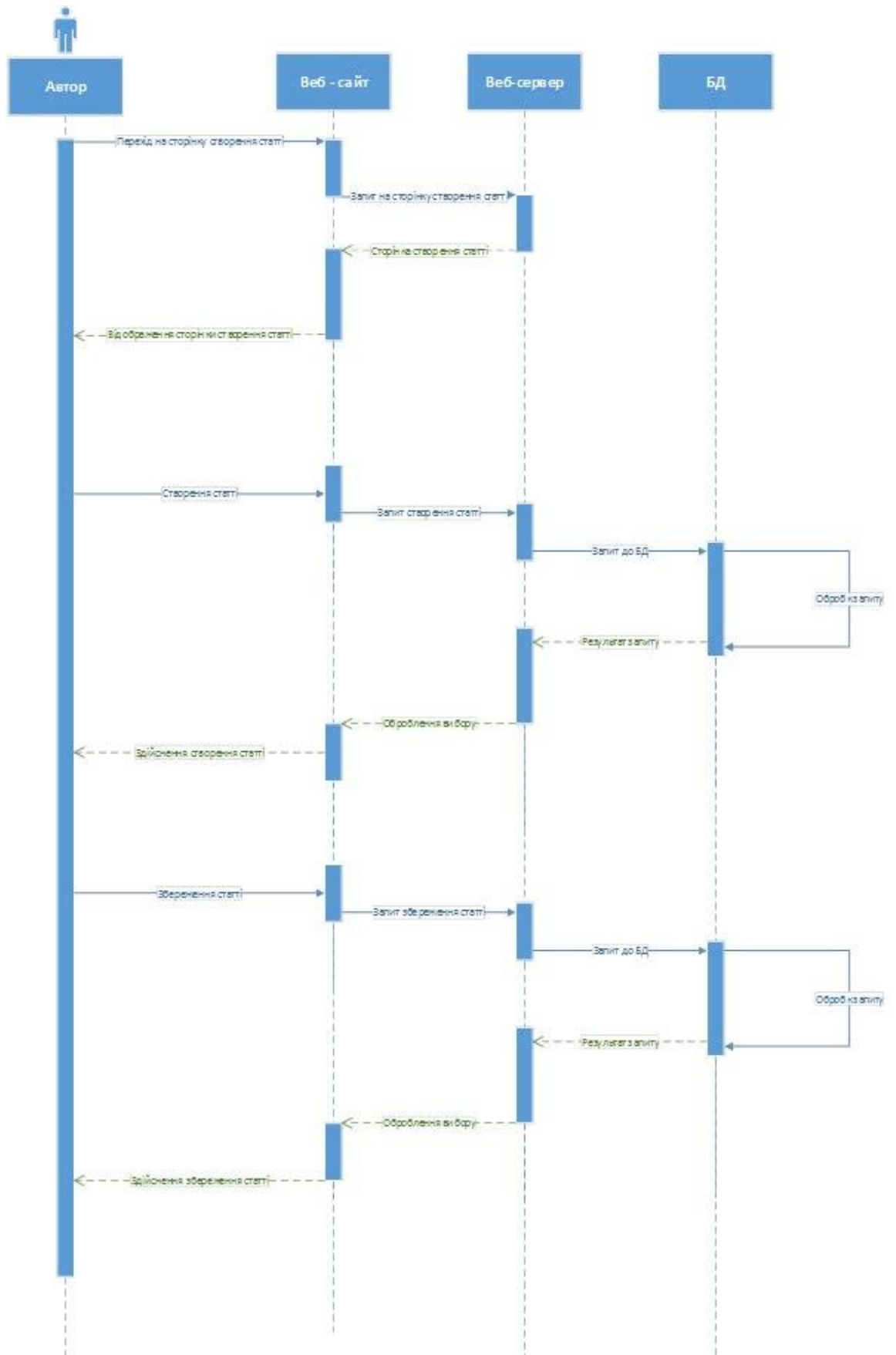
ДОДАТОК А

Глосарій

Термін	Опис терміну
Основні поняття та критерії предметної області та проекту	
Новина	оперативне інформаційне повідомлення, яке містить суспільно важливу та актуальну інформацію, що стосується певної сфери життя суспільства загалом чи окремих його груп.
Публікація	надрукована, видана, єдина за формою і змістом робота.
Веб-сайт	сукупність веб-сторінок, доступних у мережі Інтернет, які об'єднані як за змістом, так і за навігацією.
Інтернет	всесвітня система взаємополучених комп'ютерних мереж, що базуються на комплекті Інтернет-протоколів.
Браузер	програмне забезпечення для комп'ютера або іншого електронного пристрою, що дає можливість користувачеві взаємодіяти з текстом, малюнками або іншою інформацією на гіпертекстовій веб-сторінці.
Користувачі системи	
Гість	Має можливість переглядати наукові статті, завантажувати випуски журналу на ПК.
Користувач	Створює наукові статті, відправляє статті на рецензування.
Адміністратор	Здійснює керування сайтом, публікує рецензовані статті.
Вхідні та вихідні документи	
Вимоги статті	Містить вимоги до оформлення статті.

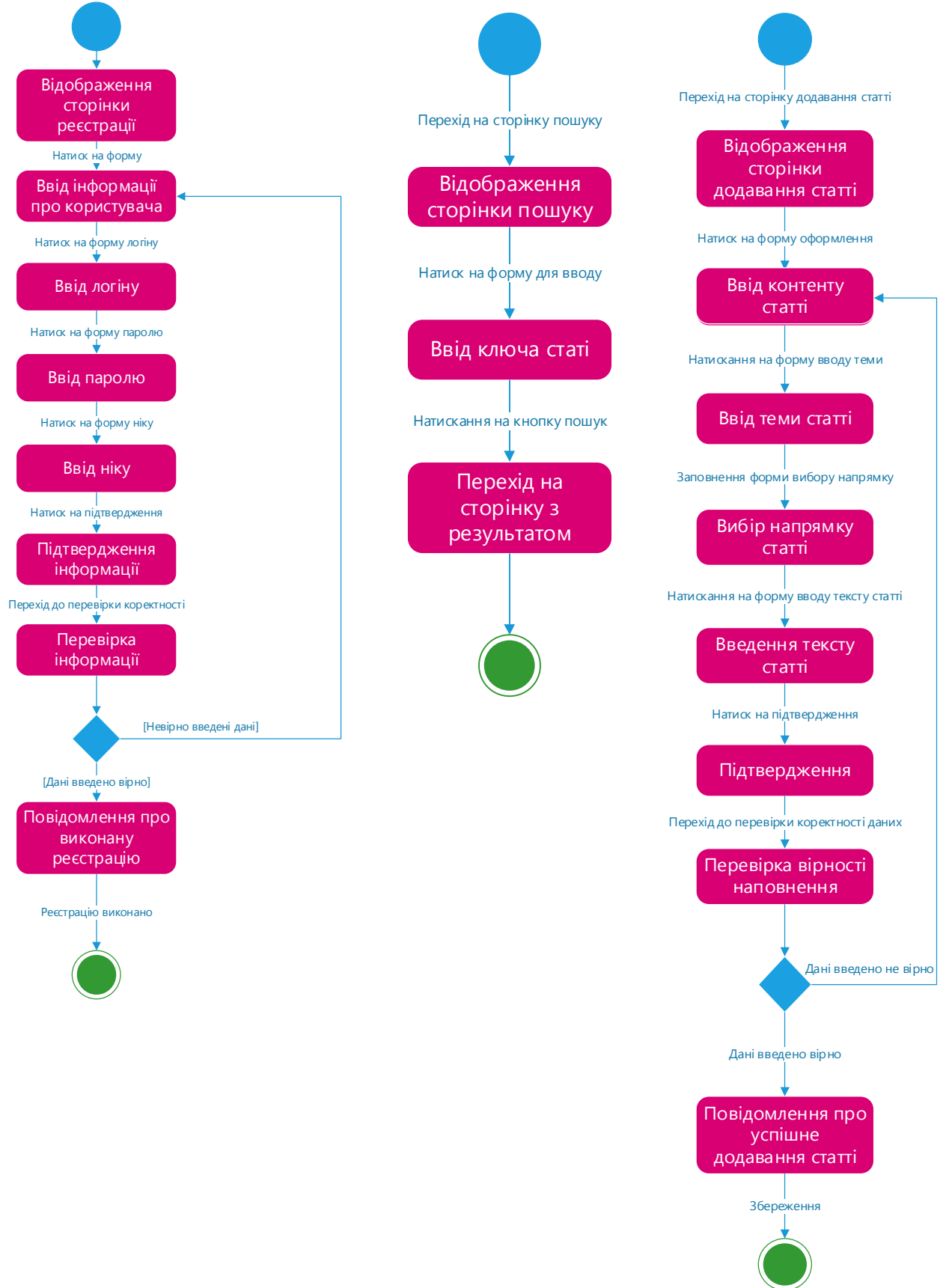
ДОДАТОК Б

Діаграма послідовностей для функції «Подання новини»



ДОДАТОК В

Діаграми діяльності для функцій «Реєстрація користувача», «Пошук статей» та «Додавання статті»



ДОДАТОК Г

Результат фізичного проектування БД

```
CREATE TABLE Material_Category
(
id_category int(10) NOT NULL AUTO_INCREMENT,
Name varchar(30) NOT NULL,
PRIMARY KEY (id_category)
);
CREATE TABLE User_type (
id_user_type int(10) NOT NULL AUTO_INCREMENT,
Name varchar(255) NOT NULL,
Required varchar(255) NOT NULL,
PRIMARY KEY (id_user_type)
);
CREATE TABLE `User` (
id_user int(10) NOT NULL AUTO_INCREMENT,
Name varchar(30) NOT NULL,
Surname varchar(30) NOT NULL,
Login varchar(30) NOT NULL,
Password varchar(30) NOT NULL,
User_typeid_user_type int(10) NOT NULL,
PRIMARY KEY (id_user)
);
CREATE TABLE Materials (
id_material int(10) NOT NULL AUTO_INCREMENT,
Name varchar(30) NOT NULL,
Content varchar(2550) NOT NULL,
Create_date date NOT NULL,
Userid_user int(10) NOT NULL,
Categoryid_category int(10) NOT NULL,
```

PRIMARY KEY (id_material)

);

ALTER TABLE Materials ADD INDEX FKMaterials970897 (Categoryid_category), ADD CONSTRAINT FKMaterials970897 FOREIGN KEY (Categoryid_category) REFERENCES Material_Category (id_category);

ALTER TABLE `User` ADD INDEX FKUser596068 (User_typeid_user_type), ADD CONSTRAINT FKUser596068 FOREIGN KEY (User_typeid_user_type) REFERENCES User_type (id_user_type);

ALTER TABLE Materials ADD INDEX FKMaterials838721 (Userid_user), ADD CONSTRAINT FKMaterials838721 FOREIGN KEY (Userid_user) REFERENCES `User` (id_user);

ДОДАТОК Д

Лістинг коду

```
.nav-collapse .dropdown-menu a {
    padding: 9px 15px;
    font-weight: bold;
    color: #555;
    -webkit-border-radius: 3px;
    -moz-border-radius: 3px;
    border-radius: 3px;
}

.nav-collapse .nav > li > a:hover,
.nav-collapse .nav > li > a:focus,
.nav-collapse .dropdown-menu a:hover,
.nav-collapse .dropdown-menu a:focus {
    background-color: #f2f2f2;
}

.navbar-inverse .nav-collapse .nav > li > a:hover,
.navbar-inverse .nav-collapse .nav > li > a:focus,
.navbar-inverse .nav-collapse .dropdown-menu a:hover,
.navbar-inverse .nav-collapse .dropdown-menu a:focus {
    background-color: #111111;
}

.nav-collapse.in .btn-group {
    margin-top: 5px;
    padding: 0;
}

.nav-collapse .dropdown-menu {
    position: static;
    top: auto;
    left: auto;
    float: none;
    display: none;
    max-width: none;
    margin: 0 15px;
    padding: 0;
    background-color: transparent;
    border: none;
    -webkit-border-radius: 0;
    -moz-border-radius: 0;
    border-radius: 0;
    -webkit-box-shadow: none;
    -moz-box-shadow: none;
    box-shadow: none;
}
```

ДОДАТОК Е

Головна сторінка

```

<?php
defined('_JEXEC') or die;

$app      = JFactory::getApplication();
$doc      = JFactory::getDocument();
$user     = JFactory::getUser();
$this->language = $doc->language;
$this->direction = $doc->direction;
$params = $app->getTemplate(true)->params;

$option = $app->input->getCmd('option', '');
$view   = $app->input->getCmd('view', '');
$layout = $app->input->getCmd('layout', '');
$task   = $app->input->getCmd('task', '');
$itemid = $app->input->getCmd('Itemid', '');
$sitename = $app->get('sitename');

$doc->setHtml5(true);

if($task == "edit" || $layout == "form" )
{
    $fullWidth = 1;
}
else
{
    $fullWidth = 0;
}

JHtml::_('bootstrap.framework');
$doc->addScript($this->baseurl . '/templates/' . $this->template . '/js/template.js');

$doc->addStyleSheet($this->baseurl . '/templates/' . $this->template . '/css/template.css');

$userCss = JPATH_SITE . '/templates/' . $this->template . '/css/user.css';

if (file_exists($userCss) && filesize($userCss) > 0)
{
    $doc->addStyleSheetVersion('templates/' . $this->template . '/css/user.css');
}

JHtml::_('bootstrap.loadCss', false, $this->direction);

if ($this->countModules('position-7') && $this->countModules('position-8'))
{
    $span = "span6";
}
elseif ($this->countModules('position-7') && !$this->countModules('position-8'))
{
    $span = "span9";
}
elseif (!$this->countModules('position-7') && $this->countModules('position-8'))

```



```
{
    $span = "span9";
}
else
{
    $span = "span12";
}

if ($this->params->get('logoFile'))
{
    $logo = '<img src="" . JUri::root() . $this->params->get('logoFile') . "" alt="" . $sitename . "" />';
}
elseif ($this->params->get('sitetitle'))
{
    $logo = '<span class="site-title" title="" . $sitename . "">' . htmlspecialchars($this->params->get('sitetitle')) . '</span>';
}
else
{
    $logo = '<span class="site-title" title="" . $sitename . "">' . $sitename . '</span>';
}
?>
```

ДОДАТОК Є

Зміни Index.php

```
if ($this->countModules('position-7') && $this->countModules('position-8'))
{
    $span = "span6";
}
elseif ($this->countModules('position-7') && !$this->countModules('position-8'))
{
    $span = "span9";
}
elseif (!$this->countModules('position-7') && $this->countModules('position-8'))
{
    $span = "span9";
}
else
{
    $span = "span12";
}

if ($this->params->get('logoFile'))
{
    $logo = '<img src="" . JUri::root() . $this->params->get('logoFile') . "" alt="" . $sitename . "" />';
}
elseif ($this->params->get('sitetitle'))
{
    $logo = '<span class="site-title" title="" . $sitename . "">' . htmlspecialchars($this->params->get('sitetitle')) . '</span>';
}
else
{
    $logo = '<span class="site-title" title="" . $sitename . "">' . $sitename . '</span>';
}
?>
```