

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Інститут інформатики та радіоелектроніки
Факультет комп'ютерних наук та технологій
(повне найменування інституту, факультету)

Кафедра комп'ютерних систем та мереж
(повне найменування кафедри)

Пояснювальна записка

до дипломного проекту (роботи)

бакалавра

(ступінь вищої освіти (освітній ступінь))

на тему **РОЗРОБКА ВЕБ-ФОРУМУ З СИСТЕМОЮ АДМІНІСТРУВАННЯ**

Виконав: студент 4 курсу, групи КНТ-527
спеціальності _____

123 «Комп'ютерна інженерія»

(код і найменування спеціальності)

Освітня програма (спеціалізація) _____

«Комп'ютерна інженерія»

Шумченко Сергій Геннадійович

(прізвище та ініціали)

Керівник Касьян К.М.

(прізвище та ініціали)

Рецензент Морщавка С.В.

(прізвище та ініціали)

м. Запоріжжя
2021 рік

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»
(повне найменування вищого навчального закладу)

Інститут, факультет інформатики та радіоелектроніки, комп'ютерних наук і технологій
 Кафедра «Комп'ютерні системи та мережі»
 Ступінь вищої освіти (освітній ступінь) бакалаврський
 Спеціальність 123 Комп'ютерна інженерія
(код і найменування)
 Освітня програма (спеціалізація) Комп'ютерна інженерія
(назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮ

Завідувач кафедри Кудерметов Р.К.

Р.К.
 “ ” 2021 року

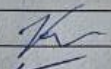
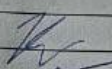
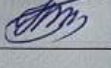
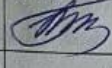
ЗАВДАННЯ
НА ДИПЛОМНИЙ ПРОЕКТ (РОБОТУ) СТУДЕНТА

Шумченко Сергію Геннадійовичу

(прізвище, ім'я, по батькові)

1. Тема проекту (роботи) Розробка Веб-форуму з системою адміністрування
 керівник проекту (роботи) Касьян Костянтин Миколайович, к. т. н., доцент
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)
 затверджені наказом вищого навчального закладу від “17” березня 2021 року № 81
2. Строк подання студентом проекту (роботи) 06 травня 2021 року
3. Вихідні дані до проекту (роботи) опис товару інтернет-магазину
4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)
 - 1) Аналіз сучасних веб-форумів та технічного завдання;
 - 2) Проектування веб-форуму;
 - 3) Розробка бази даних;
 - 4) Розробка веб-форуму.
5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)
 - Діаграма використання веб-форуму;
 - Макети сторінок веб-форуму;
 - Реляційна схема бази даних;
 - Діаграма класів веб-форуму;
 - Інтерфейс системи.

6. Консультанти розділів проекту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконане завдання
1-3	Касьян К.М. к. т. н., доцент		
Нормо-контроль	Щербак Н.В., ст.викладач		

7. Дата видачі завдання 01.03.2021 р.


КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів проекту (роботи)	Примітка
1	Аналіз наявних методів та алгоритмів побудови Веб-формуїв, перегляд отриманих результатів, виділення недоліків та переваг	01.03.2021 р.	
2	Розробка структури системи	15.03.2021 р.	
3	Розробка бази даних	20.03.2021 р.	
4	Розробка веб-форуму	01.04.2021 р.	
5	Розробка системи адміністрування	10.04.2021 р.	
6	Оформлення отриманих результатів у ПЗ	15.04.2021 р.	
7	Оформлення графічного матеріалу	20.04.2021 р.	
8	Оформлення допоміжного матеріалу	01.05.2021 р.	

Студент


(підпис)**С.Г. Шумченко**
(ініціали та прізвище)

Керівник проекту (роботи)


(підпис)**М.К. Касьян**
(ініціали та прізвище)

РЕФЕРАТ

ПЗ: 76 с., 22 рисунках, 11 таблицях, 12 джерел.

CSS, JAVASCRIPT, JQUERY, HTML, WEB, АДМІНІСТРАТОР, БАЗА ДАНИХ, САЙТ, ТЕХНОЛОГІЯ AJAX, ФОРУМ

Об'єкт розробки – веб-форум.

Метою роботи є розробка веб-форуму, аналіз предметної області, аналіз існуючих веб-форумів, моделювання об'єктів проектування, побудова інформаційної моделі, аналіз методики написання форуму з дружнім інтерфейсом, розробка програмного забезпечення.

Предмет розробки – моделі, методи, і інструментальні та програмні засоби проектування та розробки веб-системи.

У даній бакалаврській роботі засобами зв'язки PHP і MySQL був створений форум. Для додавання динаміки на сайт був використаний мову програмування JavaScript і його бібліотека JQuery. Так само була використана популярна на даний момент технологія AJAX. Для вирішення поставленого завдання був обраний програмний пакет Apache, що складається з зв'язки PHP + MySQL і система phpMyAdmin.

Пояснювальна записка складається з 4 розділів. Перший розділ присвячено аналізу технічного завдання. В даному розділі приведені аналіз вихідних даних веб-системи та розроблена специфікація вимог RSS. Здійснюється детальний розбір об'єкта та порівняння з іншими відомими сайтами. Третій розділ описує розробку бази даних. Четвертий розділ присвячено розробці веб-форуму. Описані керівництва програміста та користувача веб-форуму.

ЗМІСТ

Скорочення та умовні позначки	7
Вступ.....	8
1 Аналіз технічного завдання	9
1.1 Аналіз вихідних даних і вимог до веб-форуму.....	9
1.2 Аналіз сучасних веб-форумів	12
1.3 Огляд сучасних технологій розробки веб-форумів	16
2 Проектування веб-форуму	19
2.1 Розробка специфікації вимог SRS	19
2.1.1 Вимоги до веб системи.....	19
2.1.2 Побудова діаграм веб-форуму	23
2.2 Проектування інтерфейсу	28
3 Розробка бази даних.....	31
3.1 Концептуальна модель бази даних.....	31
3.2 Розробка бази даних.....	33
4 Розробка веб форуму.....	37
4.1 Структурна схема веб-форуму	37
4.2 Програмна реалізація веб-форуму.....	39
4.3 Керівництво користувача.....	47
4.4 SEO-оптимізація	49
Висновки.....	53
Перелік джерел посилання	54
Додаток А	56

Плакати:

Плакат 1 - Діаграма використання веб-форуму

Плакат 2 - Макети сторінок веб-форуму

Плакат 3 - Схема бази даних

Плакат 4 - Діаграма класів веб-форуму

Плакат 5 - Інтерфейс веб-системи

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАЧКИ

CMS	-	Content Management System
HTML	-	HyperText Markup Language
JS		Java Script
MVC	-	Model View Control
SRS	-	Software Requirement Specification
UML	-	Unified Modeling Language
БД	-	база даних
ІС	-	інформаційна система
ПЗ	-	програмне забезпечення
ПК	-	персональний комп'ютер
РА	-	розподілена архітектура

ВСТУП

Веб-форум, форум сайту - це особливий клас веб-додатки для організації спілкування користувачів, відвідувачів сайту. Термін відповідає сенсу вихідного поняття «форум». У форумі існують розділи для обговорення. Робота форуму полягає в створенні користувачами тим в розділах і подальшим обговоренням цих тем. Окремо тема - своєрідна тематична гостьова книга.

Внутрішнє розподіл на форумі: «Розділи- теми-повідомлення».

Інформація повідомлень у форумі: «Автор - тема - зміст - дата / час». Повідомлення і всі відповіді на нього утворює гілку (тему, тред, топик, топ).

Метою роботи є розробка веб-форуму, аналіз предметної області, аналіз існуючих веб-форумів, моделювання об'єктів проектування, побудова інформаційної моделі, аналіз методики написання форуму з дружнім інтерфейсом, розробка програмного забезпечення

Для досягнення мети необхідно виконати наступні завдання:

- проаналізувати предметну область і визначити основні поняття, об'єкти і процеси;
- виконати аналіз існуючих веб-форумів: визначити їх функціональні можливості, умови використання і основні характеристики;
- сформулювати основні функціональні можливості і вимоги у вигляді постановки завдання;
- виконати моделювання об'єкта проектування засобами уніфікованої мови моделювання UML;
- побудувати базу даних;
- реалізувати програмну частину веб-форуму.

1 АНАЛІЗ ТЕХНІЧНОГО ЗАВДАННЯ

1.1 Аналіз вихідних даних і вимог до веб-форуму

Принципи форуму. Відхилення від початкової теми обговорення часто заборонене правилами форуму. За дотриманням правил слідкують адміністратори, модератори - учасники, наділені можливістю редагувати, переміщати і видаляти чужі повідомлення в певному розділі або темі, контролювати доступ окремих учасників.

Розмежування доступу. На форумах може застосовуватися гнучке розмежування доступу до повідомлень. Так, на одних форумах читання і створення нових повідомлень доступні будь-яким випадковим відвідувачам, на інших необхідна попередня реєстрація (поширений варіант) - ті і інші форуми називають відкритими. Застосовується і змішаний варіант - коли окремі теми можуть бути доступні на запис всім відвідувачам, а інші - тільки зареєстрованим учасникам. Крім відкритих, існують закриті форуми, доступ до яких визначається персонально для кожного учасника адміністрацією форуму. На практиці - деякі розділи форуму загальнодоступні, а решта доступна обраному колу учасників.

Реєстрація. При реєстрації учасники форуму можуть створювати профілі - сторінки з відомостями про учасника. У своєму профілі учасник форуму може повідомити інформацію про себе, налаштувати свій аватар або автоматично додається до його повідомлень підпис - в залежності від уподобань. Підпис може бути статичним текстом або містити графічні картинки. Більшість форумів має систему особистих повідомлень, що дозволяє зареєстрованим користувачам спілкуватися індивідуально, аналогічно електронній пошті.

Багато форумів при створенні нової теми мають можливість приєднання до неї голосувань або опитувань. При цьому інші учасники форуму можуть проголосувати або відповісти на поставлене в заголовку теми питання, не створюючи нового повідомлення в темі форуму. По методу формування набору тим форуми бувають з динамічним списком тим і з постійним списком тем. У

форумах з динамічним списком тим прості учасники можуть створювати нову тему в рамках тематики форуму.

Як правило, кожен конкретний форум має певну тематику - досить широку, щоб в її межах можна було вести багатопланове обговорення. Часто також кілька форумів зводять воедино в одному місці, яке також називають форумом в широкому сенсі.

Форуми мають можливість пошуку по своїй базі повідомлень. Форум відрізняється від чату розділенням обговорюваних тем і можливістю спілкування не в реальному часі. Це дозволяє обговорювати серйозні теми, надає таким, що відповідає час на обдумування відповіді. Форуми використовуються для консультацій, технічної підтримки і т.п. Форуми майже повністю витіснили новинні групи на базі NNTP (англ. Network News Transfer Protocol) - мережевий протокол для обміну повідомленнями в групах новин. Досить популярний, свого часу спосіб обговорення тем в Інтернеті. Форуми співіснують нарівні з блогами. Ці дві форми спілкування в Інтернеті не поступаються один одному по популярності [1].

Форум - засіб спілкування. Форум, як і гостьова книга є засобом спілкування, тільки він володіє більш широкими можливостями. Якщо трохи перебільшувати, то можна сказати, що форум - це наворочена і вдосконалена гостьова книга.

Але, все ж, форум і гостьова мають істотними відмінностями. Якщо за допомогою гостьової можна трохи розважити відвідувачів, дізнатися пару думок про Ваш сайт, відповісти на питання, то форум - це, в першу чергу, засіб для повторного і багаторазового залучення відвідувачів на сайт.

Технічні моменти. Для роботи форуму необхідна підтримка мов PHP / Perl з боку сервера. Форуми бувають різні, в деяких потрібно спочатку зареєструватися, інші знаходяться у вільному доступі. Більшість форумів можуть працювати тільки при наявності бази даних MySQL, така послуга є тільки на платних хостингах. Є форуми не потребують бази даних, одним з них є ExBB (вся інформація заноситься в текстовий документ і зберігатися в ньому, такий форум важко адмініструвати і управляти його вмістом) [2].

Модератор - це людина, що керує форумом, який є провідним одного або декількох його розділів. Нерідкі випадки, коли на одному форумі відразу кілька модераторів. Бажано щоб модератор був експертом в даній Закарпатській області знань тематики форуму. Модератор може керувати форумом не знаючи пароля до сайту, управління відбувається через спеціальну сторінку і привласнити форум собі модератор вже ніяк не зможе.

Сучасні теми (шаблони), інтуїтивно зрозумілий інтерфейс, область чату, особисті повідомлення і опції «напів-модерується», дискусії - ось деякі речі, які змушують відвідувачів залишатися надовго і регулярно відвідувати форум [3].

Наведемо основні компоненти, які мають бути присутніми в сучасному веб-форумі [4]:

- панель адміністратора / модератора. Форуми зазвичай мають кілька адміністраторів, але кілька модераторів. Адміністратор контролює всю операцію, а модератори допомагають адміністратору підтримувати форум в чистоті і роботі. Наявність окремої панелі управління для модераторів і адміністратора є важливою функцією для розгляду;

- Rich Text Editor. Онлайн-форуми в основному присвячені тексту, але форуми розвиваються як по тексту, так і по візуальному контенту. Наявність розширеного текстового редактора з можливістю додавання зображень і підтримкою HTML - один з хороших способів просування форуму серед нових користувачів»;

- оптимізація для мобільних пристроїв. Більшість користувачів в Інтернеті воліють використовувати свій мобільний телефон через ПК. Таким чином, наявність оптимізованого для мобільних пристроїв адаптивного форуму рекомендується як з точки зору SEO, так і з точки зору користувацького досвіду;

- спам-фільтр. Онлайн-форуми є найкращою метою спамерів, тому наявність надійної системи фільтрації спам-фільтрів є обов'язковою умовою для управління форумами без проблем;

- система рейтингу користувачів. Один з кращих способів підвищити активність користувачів на форумі - створити систему рейтингу користувачів.

Користувачі повинні мати можливість розблокувати нові рівні в залежності від їх взаємодії на форумі;

- налаштування профілю. Надання користувачам можливості налаштувати свій профіль за допомогою аватара і підписи - відмінний спосіб створити унікальне веселе спільнота.

1.2 Аналіз сучасних веб-форумів

Проаналізуємо програмне забезпечення сучасних веб-форумів. Як видно з таблиці 1.1, за перше місце борються платні продукти Invision Power Board (нині IPS Community Suite) і vBulletin. При цьому лідером, нехай і з невеликим відривом, все ж є IPB [5].

Таблиця 1.1 – Програмні продукти для розробки форумів

№	Програмний продукт	Кількість форумів
1	IPB	120 (28%)
2	vBulletin	115 (26,8%)
3	phpBB	87 (20,3%)
4	SMF	30 (7%)
5	Власна розробка	25 (5,8 %)
6	XenForo	19 (4,4 %)
7	PunBB	12 (2,8%)
8	Phorum	4 (0,9%)
9	XMB	2 (0,5%)
10	miniBB	2 (0,5%)

На 3-му місці з невеликим відставанням йде безкоштовний phpBB, що поширюється за ліцензією GNU GPL. Крім нього, також досить популярний SMF (Simple Machines Forum) - спадкоємець YaBB SE, який в свою чергу веде свою історію від YaBB на Perl. Цей движок також безкоштовний, але на відміну від phpBB має ліцензію BSD.

Особливо варто відзначити платний XenForo, перший публічний реліз бета-версії якого відбувся в жовтні 2010 року, а перша стабільна версія вийшла 8 березня 2011 року. Незважаючи на платність і сильну конкуренцію, він зумів потрапити в п'ятірку найпопулярніших движків. Причому деякі форуми відмовилися від раніше використовуваного програмного забезпечення і перейшли на XenForo. Найбільший з них - BMW Club, що містить більше 27 мільйонів повідомлень. Раніше він працював на платному vBulletin [6].

Використання старих двигунів, таких як IkonBoard 2 (Комп'ютерний форум Ru.Board) або YaBB 1 (Absolute Games Forums Central), обумовлено великим віком цих форумів, а також, по всій видимості, консервативністю адміністраторів і (або) користувачів, які не бажають переходити на сучасне програмне забезпечення. Крім того, ці форуми в значній мірі модифіковані та вже мало схожі на оригінальні IkonBoard і YaBB.

Розглянемо використання можливих веб-серверів для форумів. Як можна бачити з таблиці 1.2, більше 2/3 всіх найбільших форумів використовують nginx. Правда, в цей відсоток входять не тільки форуми, що працюють на чистому nginx, але і використовують його як front-end до іншого сервера (як правило, до Apache). Така висока популярність nginx на великих форумах обумовлена його стійкою роботою на високонавантажених проектах [7].

Таблиця 1.2 – Веб-сервери форумів

№	Веб-сервер	Кількість форумів
1	nginx	307 (71,6%)
2	Apache	68 (15,9%)
3	Інший	46 (10,7%)
4	IIS	8 (1,9%)

На чистому Apache працює всього лише близько 16% форумів. IIS (Internet Information Services) від Microsoft має ще меншу поширеність. Його використовують менше 2% форумів. Найбільшим з них є форум SQL.ru, що містить понад 19 мільйонів повідомлень. Значну частину форумів з пункту «Інший або не визначений» складають сайти, які використовують сервіси для захисту від DDoS-

атак - CloudFlare, Qrator, DDoS-GUARD.

Серед кодувань лідирує UTF-8. Дивно те, що Windows-1251 майже не відстає від неї (табл. 1.3). Але і в цьому немає нічого незвичайного, якщо згадати, що в статистиці враховуються найбільші форуми. А найбільші - це, як правило, досить старі форуми, що створювалися ще за часів dial-up'a, коли на рахунку був кожен байт. Варто нагадати, що символи кирилиці займають в UTF-8 по 2 байта, тоді як в Windows-1251 і KOI8-R - по 1 байту.

Таблиця 1.3 – Кодування веб-форумів

№	Кодування	Кількість форумів
1	UTF-8	238 (55,5%)
2	Windows-1251	188 (43,8%)
3	KOI8-R	3 (0,7%)

Ще цікавіше, що до сих пір збереглися форуми, які використовують кодування KOI8-R. Серед найбільших це Farit.ru (більше 8 мільйонів повідомлень), Конференція порталу HiTV (більше 2 мільйонів повідомлень) і Кулінарний форум Kuking (більше 1 мільйона повідомлень) [8].

Підводячи підсумки, можна виділити наступні тенденції за останні кілька років:

- найпопулярнішими двигунами залишаються платні IPB і vBulletin;
- перехід деяких форумів на новий комерційний движок XenForo;
- використання сервера nginx, а також сервісів для захисту від DDoS-атак;
- повільний перехід на кодування UTF-8 при збереженні серйозних позицій у Windows-1251;
- переважна більшість форумів поки ще використовує протокол HTTP, але спостерігається поступовий перехід на HTTPS.

За структурою веб-форуми діляться на наступні різновиди:

- табличні;
- деревовидні;

- універсальні.

Табличні форуми мають строгий порядок в постах відвідувачів форуму. Повідомлення в темах створюють стрічки, тому табличні форуми ще називають стрічковими. Для визначення адресата і адресанта відповіді, для діалогу з ними доступна панель інструментів, а саме звернення до автора, цитування та багато іншого. Завдяки суворій ієрархії в форумі з такою структурою користувачам спілкування не завдає труднощів. Натиснувши на одну з тем форуму, можна побачити всі повідомлення даного топіка. Найпопулярніші теми знаходяться в топі, на перших місцях. Рейтинг створюється за рахунок підрахунку повідомлень і перегляду. Але нюансом табличного веб-форуму варто назвати необхідність в контролі повідомлень, дуже часто в них розміщуються не пов'язані з темою обговорення коментарі. Флудом і флеймом в даному випадку займаються модератори.

Веб-форуми з деревовидної структурою також мають сувору ієрархію повідомлень, але іншого плану, ніж з табличній. У таких форумах все чітко і конкретно, в пості завжди вказується, кому відправляється повідомлення. Легко можна простежити, хто з ким спілкується і до якого відповіді відноситься тій чи іншій коментар. Повідомлення мають вигляд дерева, кожен відповідь і подальші коментарі можуть створити нову гілку обговорення.

Якщо тема не цікава користувачеві, він може просто її проігнорувати. Але в цьому і виявляється мінус такого форуму. Для прочитання повідомлень потрібно кожен раз заходити в заголовок теми, тим самим витрачаючи багато часу. І крім того, можна припустити важливе повідомлення в гілці, яку користувач порахував нецікавою.

Універсальні форуми пропонує користувачам вибрати, в якому режимі працювати: в табличному або деревовидному. Це демократичний підхід, який складним виявляється тільки для новачків.

У кожному веб-форумі є відправник і одержувач інформації. З урахуванням даних ролей форуми діляться на спеціалізовані і неспеціалізовані. А серед них вже виділяють:

- проектні;
- професійні;
- інформаційно-розважальні;
- аматорські.

Проектний форум - це спілкування фахівців, які мають спільні цілі і завдання. Це можуть бути працівники однієї організації або співробітники, які працюють над спільним проектом. Як правило, доступ до участі в такому форумі обмежений або зовсім закритий.

Професійні форуми об'єднують людей однієї професії або сфери діяльності. За допомогою форумів здійснюється обмін досвідом між фахівцями.

Аматорські форми створюють люди зі схожими інтересами, захопленнями і хобі. Спілкування в співтоваристві стає способом провести дозвілля, носить виключно добровільний, і тому, як показує статистика, довгостроковий характер.

Веб-форуми інформаційно-розважального характеру носить виключно функцію обміну інформацією, новинами, чутками, анекдотами або смішними історіями.

Головним достоїнством форумів потрібно назвати об'єднання людей за схожими інтересами: професії, хобі, захоплень. Крім того, спілкування в спільнотах стає приємним і корисним способом провести вільний час. Завдяки форумам кожен користувач може отримати конкретну відповідь на конкретне запитання, поділитися природною посиланням, запитати думку у людей і навіть анонімно поспілкуватися на відверті теми.

1.3 Огляд сучасних технологій розробки веб-форумів

В першу чергу відзначимо, що будь-який сайт складається з двох частин - візуальної і програмної. Над візуальною складовою працюють Frontend-розробники, а над програмною - Backend-фахівці. Перші використовують такі

технології, як HTML, CSS, JavaScript. Для других основними інструментами виступають мови програмування PHP, Python, Ruby. Перелічимо основні сучасні технології створення сайтів:

- HTML, CSS - технології для створення розмітки (HTML) і зовнішнього вигляду сторінки (мова таблиць стилів CSS);

- PHP, JavaScript - мови програмування, що змушують «ожити» сторінку;

- база даних MySQL - відповідає за зберігання текстової інформації;

Notepad ++, Photoshop, Filezilla-спеціальні інструменти для верстки, створення графіки, редагування файлів на хостингу та інших завдань;

CMS - дослівно система управління контентом. Потрібна для зручного управління вмістом сайту, блогу. Зазвичай використовують спеціальні системи за типом WordPress, Joomla, OpenCart.

Далі детально розглянемо, яку значимість грає кожна технологія для створення сайту окремо.

Розмітка і дизайн.

На першому етапі верстальник роблять розмітку документа за рахунок мови розмітки HTML і таблиці стилів CSS. Мова HTML включає безліч тегів. Саме теги підказують браузеру, яким чином необхідно відобразити вміст сайту. Тобто, HTML визначає структуру документа, браузер прочитає всі теги та інструкції, після чого видає на екрані кінцевий результат - сторінку сайту. HTML не є мовою програмування. Це мова гіпертекстової розмітки.

Якщо HTML структурує контент на сторінці, то за допомогою CSS його оформляють візуально, використовуючи кольору, стилі, розміри шрифтів та інших візуальних елементів. На самому початку розвитку інтернет-технологій за допомогою HTML виділяли параграфи, заголовки, але після тільки цього стало недостатньо. Так була створена технологія, що дозволяє формувати текст без зміни структури і вмісту документа. Сьогодні сторінки сайту і контент декорують за рахунок CSS3. Під оформленням прийнято розуміти дизайн, верстку і варіації відображення сайту для пристроїв з різним дозволом екрану.

Мови програмування.

До числа популярних мов програмування відносять PHP, Java, Python, JavaScript. Приблизно 80% програмної частини сьогодні створюють за допомогою PHP. До того ж на базі цієї мови створюють CMS-платформи і PHP фреймворки. На PHP пишуть повноцінні сайти і додатки. Все PHP скрипти виконуються на сервері, і це одна з головних відмінностей мови програмування від HTML, CSS і JavaScript, код яких виконує браузер.

Функціональність ресурсу підвищується з мовою JavaScript. Це відносно молодий мову програмування, який постійно вдосконалюється. З його допомогою роблять просту анімацію, створюють інтерактивні елементи. На основі JavaScript надалі були створені спеціальні бібліотеки, що спрощують роботу з ним. Однією з таких бібліотек є jQuery.

Управління базами даних.

Програмісти використовують різні інструменти і технології для розробки сайтів, в залежності від того, які завдання були поставлені. Незалежно від того, створюється простий сайт на кілька сторінок або ресурс на кілька сотень або тисяч сторінок, буде потрібно база даних (БД). База є набором структурованих даних. Для простоти розуміння БД наведемо простий приклад: фотографія, зроблена камерою телефону, це дані, а галерея, в якій зберігаються знімки, це база даних. Для обробки і зберігання інформації зазвичай використовують MySQL. Фактично це програма з відкритим кодом, яка використовується на сервері SQL. Головна перевага MySQL - висока швидкість обробки даних. Навіть на обробку мільйона записів в таблиці БД потрібні частки секунди.

Для роботи з сайтом необхідна система управління контентом (вмістом сайту), іншими словами - CMS. З її допомогою зручно редагувати текстовий контент, встановлювати різноманітні модулі і плагіни. Відома маса і платних, і безкоштовних CMS під різні потреби і завдання. Велика частина сайтів в мережі управляється за допомогою WordPress, Joomla, OpenCart і 1С-Бітрікс.

2 ПРОЕКТУВАННЯ ВЕБ-ФОРУМУ

2.1 Розробка специфікації вимог SRS

2.1.1 Вимоги до веб системи

Вимоги до програмного забезпечення — набір вимог щодо властивостей, якості та функцій програмного забезпечення, що буде розроблено, або знаходиться у розробці. Вимоги визначаються в процесі аналізу вимог та фіксуються в специфікації вимог, діаграмах прецедентів та інших артефактах процесу аналізу та розробки вимог [9].

Форум (веб-форум, інтернет-форум) - сайт, призначений для проведення онлайн-дискусій. Суть роботи форуму полягає в створенні користувачами (відвідувачами форуму) своїх тем) з їх подальшим обговоренням, шляхом постинга (розміщення) повідомлень всередині цих тем. Окремо взята тема, по суті, являє собою тематичну гостьову книгу. Користувачі можуть коментувати заявлену тему, задавати питання по ній і отримувати відповіді, а також самі відповідати на питання інших користувачів форуму і давати їм поради. Усередині теми також можуть влаштовуватися опитування (голосування), якщо це дозволяє движок. Питання і відповіді зберігаються в базі даних форуму, і в подальшому можуть бути корисні як учасникам форуму, так і будь-яким користувачам мережі Інтернет, які можуть зайти на форум, знаючи адресу сайту, або отримавши його від пошукових систем при пошуку інформації.

Тематика форумів може бути найрізноманітнішою, охоплюючи всі сфери життя, і визначається або власниками форуму або його адміністрацією, або залежить від контингенту учасників. При цьому форуми можуть бути як спеціалізовані, наприклад, присвячені музиці або якого-небудь музичного жанру, а також вузькоспеціалізовані, присвячені будь-якої музичної групи або виконавцю, так і без спеціалізації, в цьому випадку користувачі самі обирають теми цікаві їм. Форуми можуть існувати як автономно, без прив'язки до якого-небудь сайту, так і бути частиною веб-порталів.

Теми однієї тематики об'єднуються, для зручності в відповідні Розділи і, таким чином, найпоширеніша ієрархія веб-форуму: Розділи → Теми → повідомлення.

Як правило, відвідувачам форуму пропонується пройти процедуру реєстрації. Після реєстрації, яка часто вимагає підтвердження самим користувачем або Адміністратором форуму, відвідувач може авторизуватися, отримуючи тим самим переваги і бонуси при доступі до самого форуму і його сервісів. Далі він може редагувати свій профіль - сторінку з відомостями про даний учаснику, на якій він може повідомити інформацію про себе, встановити свій аватар, вказати автоматично додається до його повідомлень підпис. На форумах може застосовуватися надзвичайно гнучке розмежування доступу до повідомлень. Так, на одних форумах читання і створення нових повідомлень доступні неавторизованим відвідувачам (гостям), на інших потрібно для розміщення постів попередня реєстрація, доступна всім бажаючим (найбільш поширений варіант) - ті і інші форуми називають відкритими. Застосовується і змішаний варіант - коли окремі теми можуть бути доступні всім відвідувачам, а інші - тільки зареєстрованим учасникам. Крім відкритих, існують закриті форуми, доступ до яких визначається персонально для кожного учасника адміністраторами форуму. На практиці також нерідко зустрічається варіант, коли деякі розділи форуму загальнодоступні, а інша частина, так звані приховані розділи, доступна лише певні учасники.

За дотриманням правил слідкують модератори та адміністратори - учасники, наділені можливістю редагувати, переміщати і видаляти чужі повідомлення в певному розділі або темі, а також контролювати до них доступ окремих учасників і відвідувачів.

Ключовими функціональними вимогами до інформаційної системи є:

- кроссбраузерність і кроссплатформенність;
- зручний призначений для користувача інтерфейс;
- захищеність від злому;
- проста і зрозуміла авторизація;

- простота в обслуговуванні;
- можливість розширення функціоналу.

В рамках проекту розглядається можливість створювати теми на форумі, участь в обговореннях, оцінювати теми та коментарі інших користувачів, переглядати їх профілі.

Кількість користувачів системи не обмежена. Участь в обговореннях не зареєстрованим читачам форуму не надається.

В рамках виконання проекту необхідно реалізувати такі частини системи як:

- систему авторизації користувачів;
- профілі користувачів;
- поділ рівнів доступу;
- сортування тим за різними критеріями;
- відстеження та відображення активності користувачів.

Відповідно до характеристиками предметної області і загальними принципами роботи веб-форуму необхідно виконувати такі основні завдання:

- створювати розділи форуму;
- створювати теми;
- відповідати в темах;
- лайкати теми / коментарі;
- можливість реєстрації;
- можливість авторизації;
- зберігати дані про реєстрацію користувачів;
- зберігати дані про всі події форуму.

Для експлуатації визначені наступні ролі (табл. 2.1):

- адміністратор (системний адміністратор і адміністратор баз даних);
- відвідувач;
- користувач.

Таблиця 2.1 – Користувачі інформаційної веб-системи

Користувач	Характеристика
Відвідувач	Особа, яка перейшла до сайту з метою перегляду інформації.
Користувач	Особа, яка зареєструвалася на сайті з метою участі в обговоренні
Адміністратор	Особа, яка може редагувати різноманітну інформацію на сайті. Має функції контролю користувачів, редагування відгуків, складання рейтингів.

Клас користувачів «Відвідувачі». Сайт повинен:

- забезпечувати перегляд категорій;
- забезпечити реєстрацію;
- забезпечити зручний та швидкий спосіб пошуку;
- забезпечити зручний спосіб сортування;
- забезпечити зручне читання форуму.

Клас користувачів «Користувачі». Сайт повинен:

- перегляд інформації;
- авторизацію;
- засоби відновлення доступу до профілю;
- особистий кабінет;
- зв'язок з технічною підтримкою;
- забезпечити зручний та швидкий спосіб пошуку;
- забезпечити зручний спосіб сортування;
- забезпечити зручне читання форуму;
- створення тем;
- історію повідомлень;
- можливість коментувати;
- оцінювати коментарі;
- можливість лайкати;
- перегляд рейтингу учасників.

Клас користувачів «Адміністратор». Сайт повинен:

- можливість редагування контенту;
- перегляд статистики;
- редагувати коментарі;
- правка користувачів.

Чим більша і складніша система, тим більшого значення набуває моделювання при її розробленні. Без моделі складну систему неможливо сприйняти як одне ціле. Людське сприйняття складних сутностей є обмеженим. Моделювання системи чи об'єкта дозволяє звузити проблему, зосередивши увагу в кожен момент тільки на певних її аспектах. Складне завдання завжди легше вирішити, якщо розділити його на менші. Правильно обрана модель дозволяє створювати проекти на вищих рівнях абстракції.

Для швидкої і ефективної розробки програмного продукту з мінімальними витратами потрібно залучити робочу силу, вибрати правильні інструменти і визначити правильний напрям роботи. Щоб впоратися з поставленим завданням, беручи до уваги витрати на життєвий цикл системи, необхідно, щоб процес розробки програми був ретельно продуманий і міг бути адаптований до потреб вашого бізнесу і технології [10].

2.1.2 Побудова діаграм веб-форуму

UML (англ. Unified Modeling Language - уніфікована мова моделювання) - мова графічного опису для об'єктного моделювання в області розробки програмного забезпечення, моделювання бізнес-процесів, системного проектування та відображення організаційних структур. В процесі проектування система розглядається з різних точок зору за допомогою моделей, різні уявлення яких подаються у формі діаграм. Діаграма — це графічне представлення безлічі елементів. Зазвичай зображується у вигляді графа з вершинами (сутностями) і ребрами (відносинами). За допомогою діаграм можна візуалізувати систему з різних точок зору.

Проектована система представляється у вигляді безлічі сутностей або акторів, що взаємодіють з системою за допомогою, так званих прецедентів. При цьому актором (actor) або дійовою особою називається будь-яка сутність, що

взаємодіє з системою ззовні. Іншими словами, кожен варіант використання визначає деякий набір дій, який чинять системою при діалозі з актором. При цьому нічого не говориться про те, яким чином буде реалізовано взаємодію акторів з системою. Робота з форумом повинна передбачати: створення і перегляд тим, написання і читання коментарів, оцінку коментарів, фільтрацію контенту. При вході на форум користувач вибирає режим гість (рис. 2.1) або ж входить під своїм обліковим записом і стає користувачем (рис. 2.2.). В разі якщо облікового запису немає, користувач за бажанням може зареєструватися. Всі функції адміністратора форуму наведені на рис. 2.3. Загальна діаграма використання веб-форуму на рисунку 2.4.



Рисунок 2.1 – Діаграма варіантів використання для актора «Відвідувач»

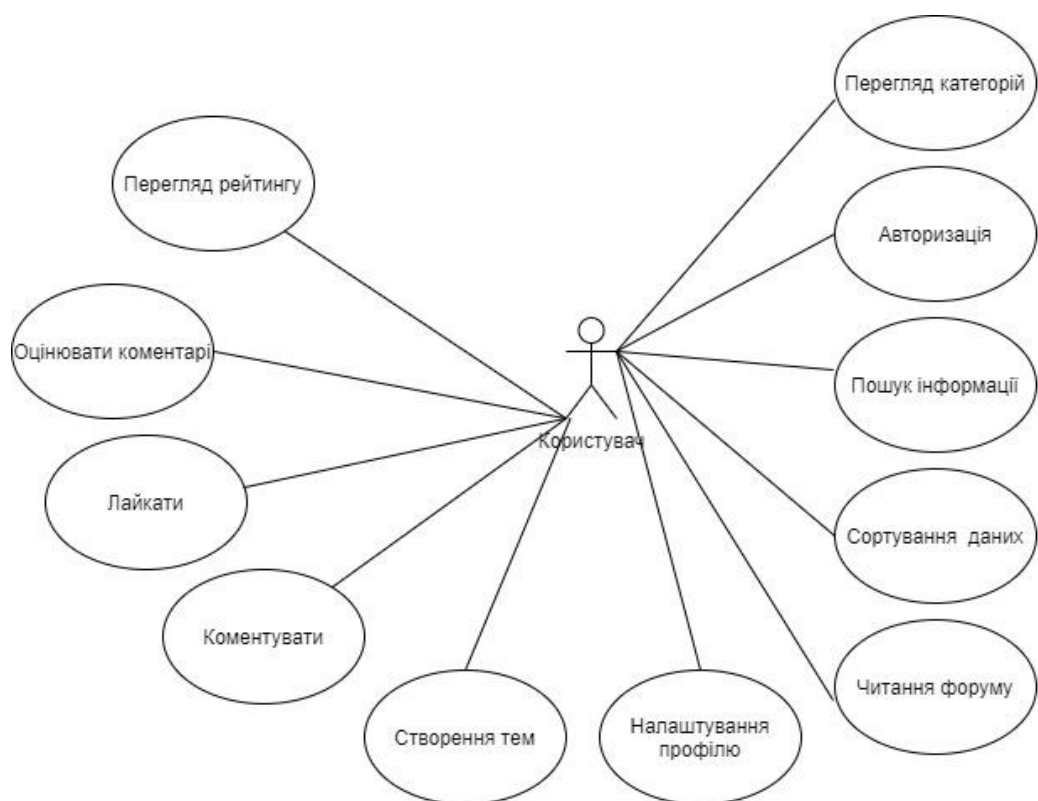


Рисунок 2.2 – Діаграма варіантів використання для актора «Користувач»

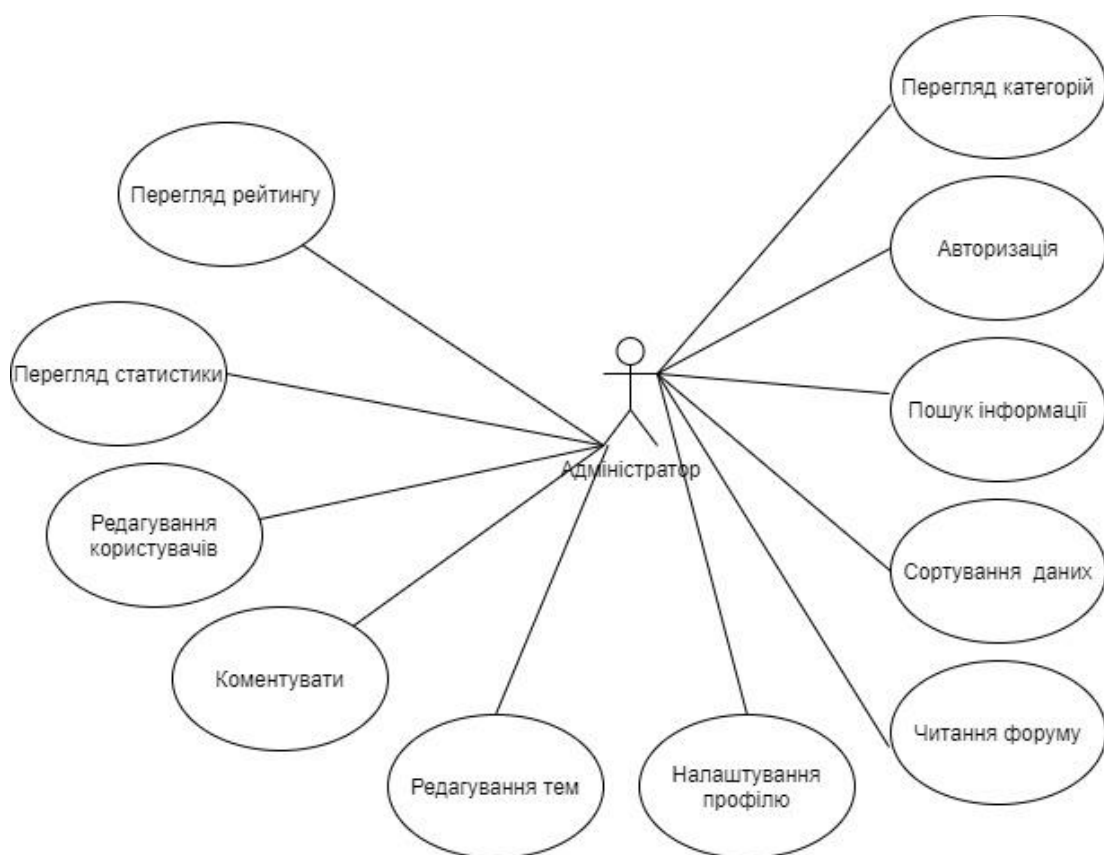


Рисунок 2.3 – Діаграма варіантів використання для актора «Адміністратор»

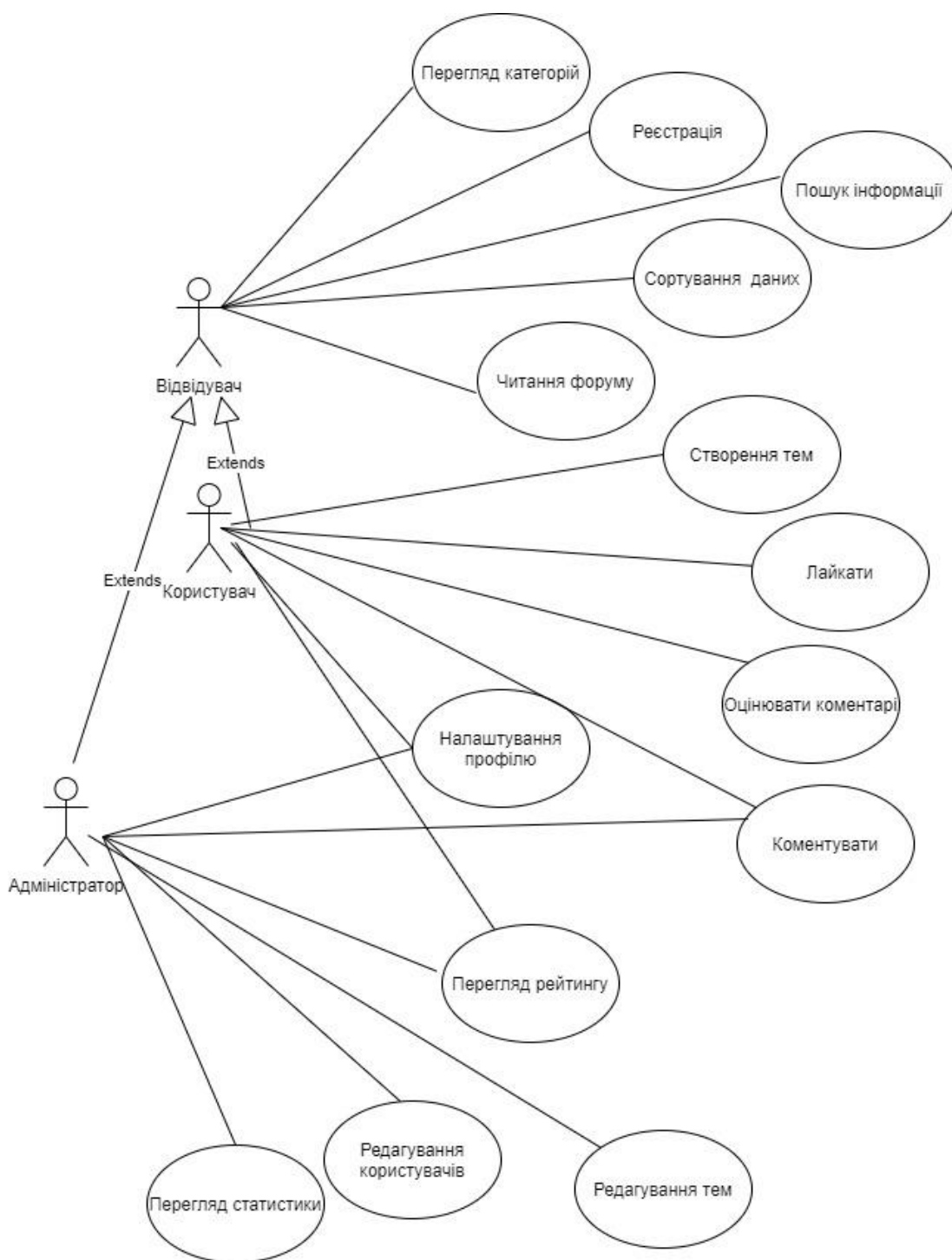


Рисунок 2.4 – Діаграма варіантів використання веб-форуму

Динамічна модель необхідна, щоб відобразити функціональність (поведінку) системи. З цією метою використовуються діаграми, які дозволяють деталізувати особливості алгоритмічної і процедурної реалізації виконуваних системою операцій. Однією з таких діаграм є діаграма діяльності, яка застосовується для

модельовання динамічних аспектів поведінки системи. Діаграма діяльності показує, як потік управління переходить від однієї діяльності до іншої (рис. 2.5-2.6).

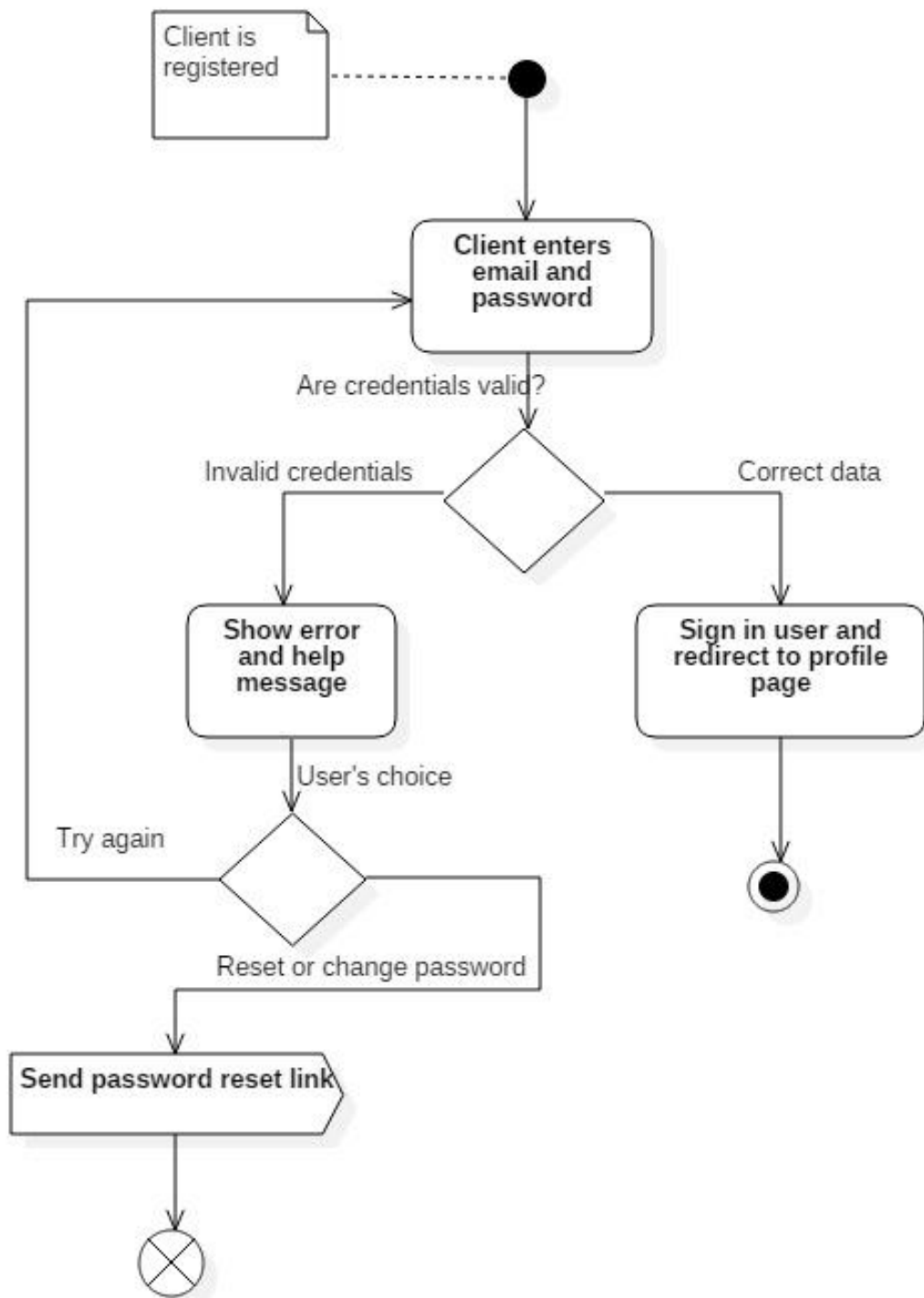


Рисунок 2.5 – Діаграма діяльності «Реєстрація користувача»

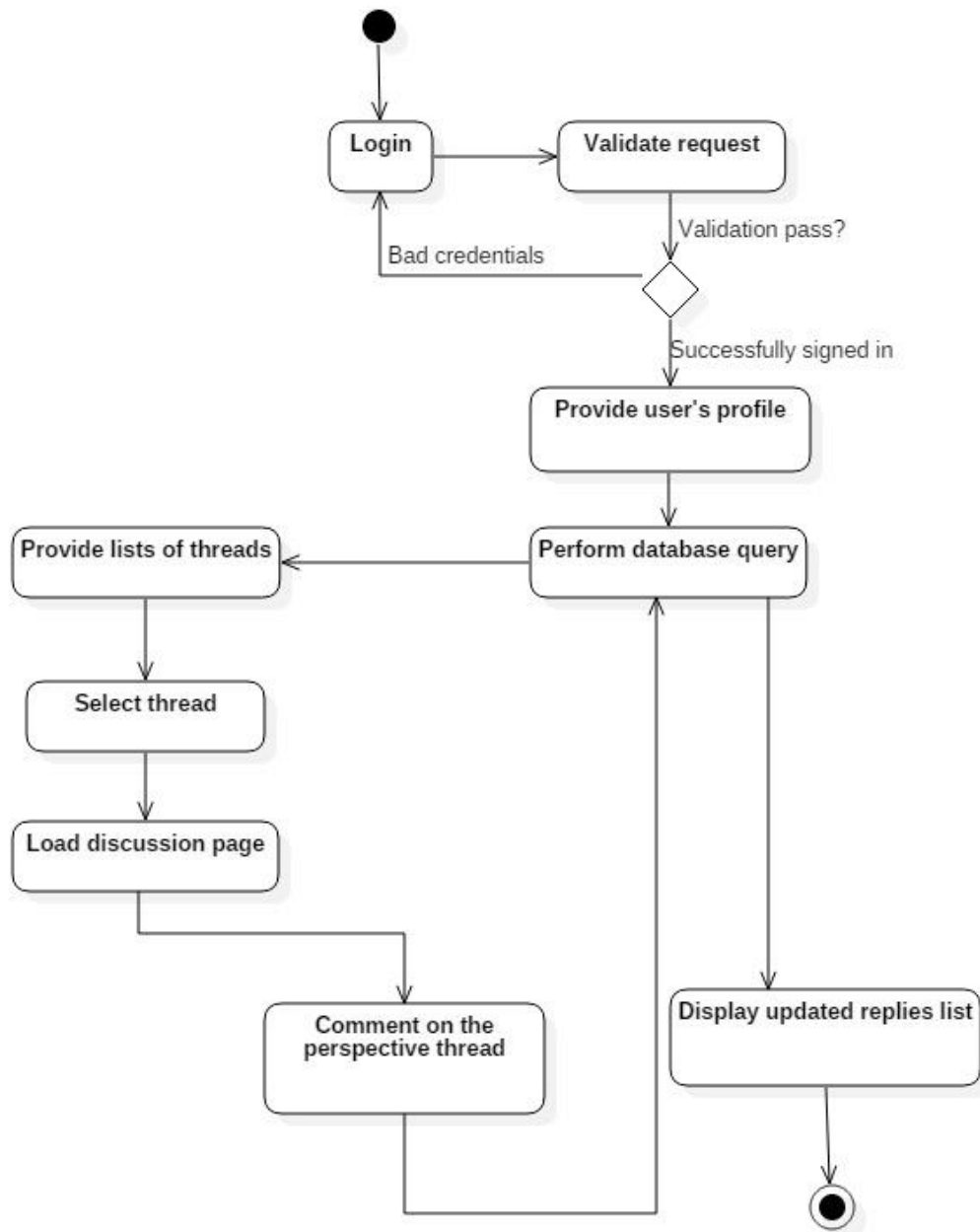


Рисунок 2.6 – Діаграма діяльності «Участь в обговоренні теми»

2.2 Проектування інтерфейсу

Під інтерфейсом програмного забезпечення мають на увазі спосіб взаємодії, який пропонується для споживача і комп'ютера. Саме інтерфейс створює зовнішній образ, надаючи йому конструктивну привабливість. Продумуючи

дизайн, слід врахувати кілька важливих моментів. В цілому, всі вони спрямовані на те, щоб додаток був якомога більш зрозумілим для користувачів. Інтерфейс повинен бути настільки зручний при реалізації, щоб користувачам не знадобилося особливих зусиль, щоб звикнути до нього. В інтерфейсі повинні використовуватися терміни, зрозумілі користувачеві, а об'єкти, керовані системою, повинні бути безпосередньо пов'язані з робочим середовищем користувача [10].

Головну сторінку форуму видно всім користувачам системи. Вона складається з хедера, футера, основної інформації, кнопок «Home», «Реєстрація», «Вхід» (рис. 2.7). Основна інформація поділена на два блоки: категорії форуму та навігацію форумом.

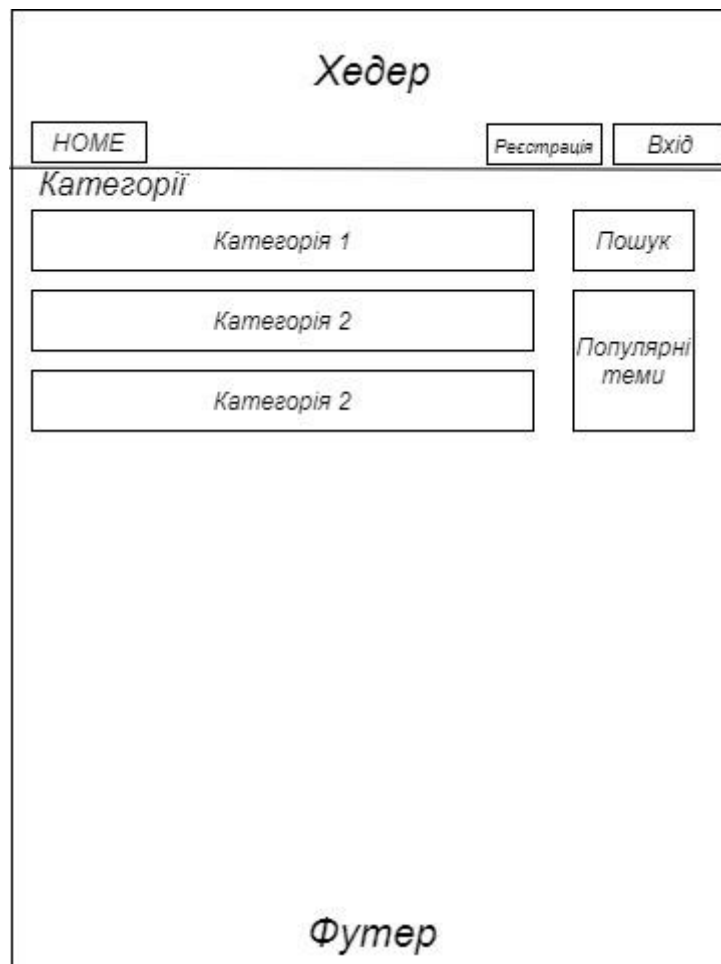


Рисунок 2.7 – Головна сторінка веб-форуму

Відвідувач форуму може зареєструватись або вести облікові дані для того, щоб прийняти участь в обговоренні теми (рис.2.8).

Хедер

[HOME](#)

Вхід

Логін

Пароль

Запам'ятати мене

Приховати моє перебування на форумі

Реєстрація

[Умови реєстрації](#)

Футер

Рисунок 2.8 – Форма Реєстрації/Авторизації

Кожен користувач форуму після реєстрацію може створити власну тему в обраній категорії. Для створення теми потрібно вказати її заголовок, додати опис теми, теги та натиснути кнопку «Зберегти» (рис. 2.9).

Хедер

[HOME](#) [Профіль](#) [Фото](#)

Нова тема

Футер

Рисунок 2.9 – Сторінка «Нова тема»

Зареєстровані користувачі можуть залишати свої коментарі під обраної темою (рис. 2.10).

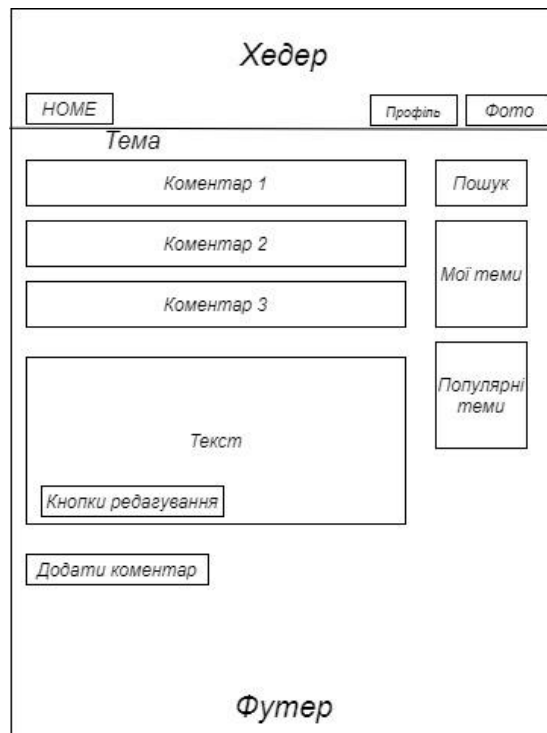


Рисунок 2.10– Сторінка «Коментарі»

3 РОЗРОБКА БАЗИ ДАНИХ

3.1 Концептуальна модель бази даних

На підставі вхідних і вихідних документів, а також вимог, що пред'являються до системи, можна побудувати концептуальну модель даних. Концептуальна модель даних описує сутності бази даних, зв'язку між ними, атрибути і відносини. Сутність - це унікальний неподільний ідентифікований елемент предметної області, представлений безліччю властивостей (атрибутів). Ставлення (зв'язок) сутностей - це асоціація, встановлена між кількома сутностями, і визначає характер їх взаємозалежності.

Концептуальна (змістовна) модель - це абстрактна модель, яка визначає

структуру модельованої системи, властивості її елементів і причинно-наслідкові зв'язки, властиві системі і суттєві для досягнення мети моделювання.

Концептуальна модель складається з сутностей, призначення яких наступне:

- user - містить інформацію про користувача;
- channel - містить інформацію про розподіл форуму;
- like - містить інформацію про оцінку контенту;
- reply - містить інформацію про коментарі;
- thread - містить інформацію про тему форуму;
- activity - містить інформацію про реєстровану подію.

У таблиці 3.1 наведені інформаційні зв'язки між об'єктами предметної області.

Таблиця 3.1 - Зв'язки інформаційних об'єктів

Головний об'єкт	Підлеглий об'єкт	Тип зв'язку
user	activity	1:M
channel	thread	1:M
like	user	M:1
reply	user	M:1
thread	user	M:1
activity	channel	M:1

На рис. 3.1 представлена інформаційно-логічна модель розглянутої БД, побудована відповідно до виявлених інформаційними об'єктами і зв'язками між ними.

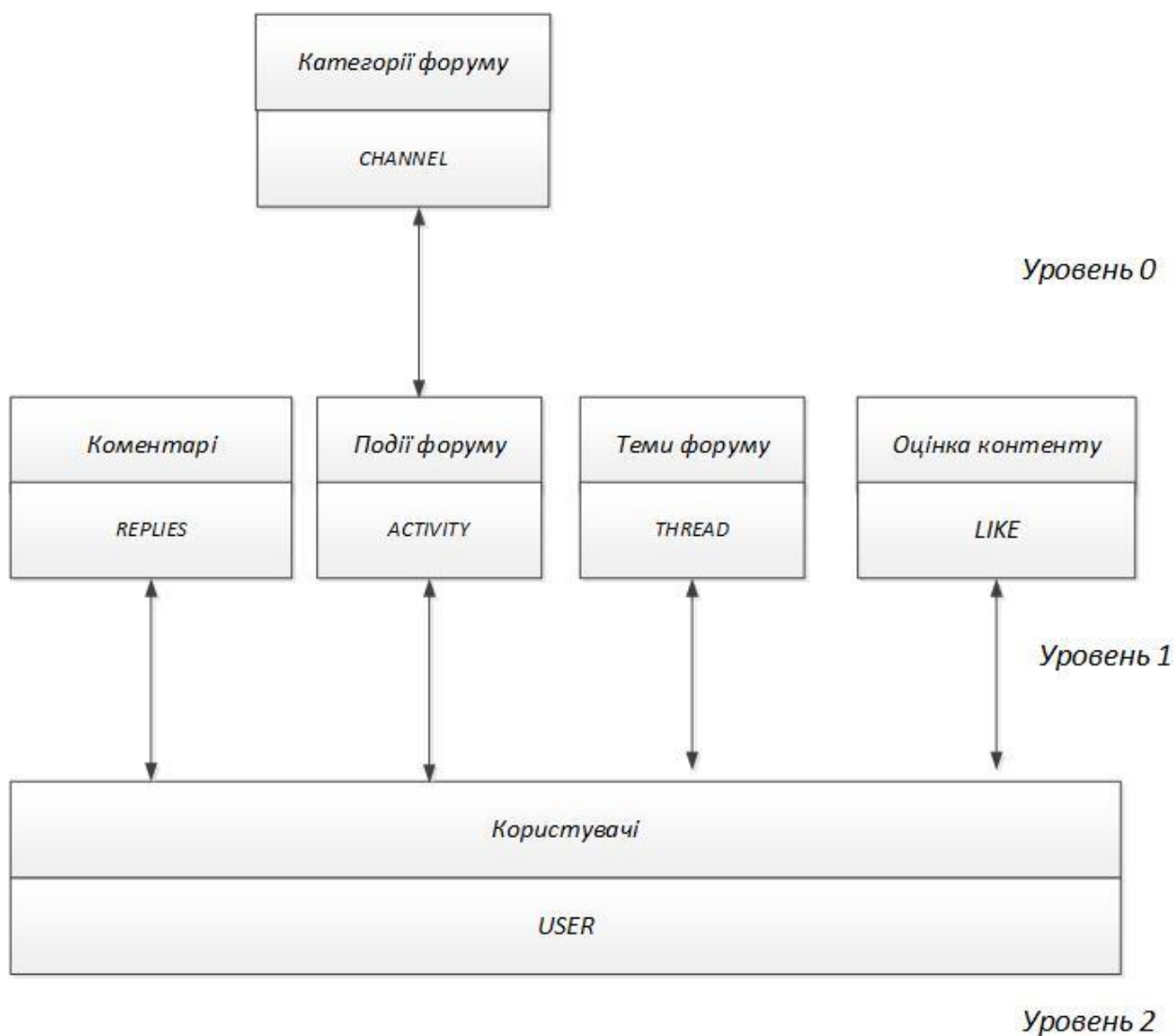


Рисунок 3.1 – Концептуальна модель предметної області

3.2 Розробка бази даних

PHP дозволяє використовувати різні систему управління базами даних, але найбільш популярною на сьогоднішній день в зв'язці з PHP є MySQL. MySQL являє безкоштовне програмне забезпечення, що дозволяє взаємодіяти з базами даних за допомогою команд мови SQL. Раніше вже розглядався процес установки і настройки MySQL [14].

Щоб спростити собі роботу з базами даних MySQL встановимо спеціальний

набір скриптів phpMyAdmin. phpMyAdmin представляє інтуїтивний веб-інтерфейс для управління базами даних MySQL. Використовуючи цей інструмент, набагато легше працювати з базами даних, ніж керувати MySQL через консоль.

Для установки phpMyAdmin завантажимо архів і розпакуємо в папку C: \ localhost, де знаходяться документи для php. Розпаковане папку перейменуємо для стислості в phpMyAdmin.

В розпакованому каталозі phpMyAdmin створимо файл config.inc.php з наступним змістом (лістинг 3.1).

Лістинг 3.1 – Налаштування БД

```
<?php
    $i = 0;
    $i++;
    $cfg['Servers'][$i]['host'] = 'localhost';
    $cfg['Servers'][$i]['extension'] = 'mysqli';
    $cfg['Servers'][$i]['connect_type'] = 'tcp';
    $cfg['Servers'][$i]['compress'] = false;
    $cfg['Servers'][$i]['auth_type'] = 'config';
    $cfg['Servers'][$i]['user'] = 'root';
    $cfg['Servers'][$i]['password'] = 'root';
?>
```

І щоб переконатися, що все налаштовано правильно, в браузері перейдемо до phpMyAdmin, наприклад, <http://localhost:8080/phpmyadmin>.

В процесі розробки було створено 5 таблиць:

- users - таблиця, що зберігає інформації про всіх користувачів на форумі (табл. 3.2);
- channels – таблиця 3.3, що зберігає розділи форуму;
- likes – зберігає інформацію вподобань користувачів (табл. 3.4);
- replies – таблиця 3.5 з інформацією про коментарі;
- threads – теми форуму (табл. 3.6).

Таблиця 3.2 – Структура таблиці "users"

№ п/п	Назва атрибута	Тип даних	Опис
1	ID	INTEGER	Код користувача
2	NAME	VARCHAR(255)	Ім'я / Нік
3	EMAIL	VARCHAR(255)	Е-Mail
4	PASSWORD	VARCHAR(255)	Пароль
5	PHOTO	INTEGER	Фото

Таблиця 3.3 – Структура таблиці "channels"

№ п/п	Назва атрибута	Тип даних	Опис
1	ID	INTEGER	Код категорії
2	NAME	VARCHAR(100)	Назва розділу
3	SLUG	VARCHAR(100)	Slug розділу

Таблиця 3.4 – Структура таблиці "likes"

№ п/п	Назва атрибута	Тип даних	Опис
1	ID	INTEGER	Код лайка
2	USER_ID	INTEGER	Код користувача
3	LIKED_ID	INTEGER	Код об'єкта
4	LIKED_TYPE	VARCHAR(50)	Тип об'єкта

Таблиця 3.5 – Структура таблиці "replies"

№ п/п	Назва атрибута	Тип даних	Опис
1.	ID	INTEGER	Код коментаря
2.	THREAD_ID	INTEGER	Код теми
3.	USER_ID	INTEGER	Код користувача
4.	BODY	VARCHAR(255)	Текст

Таблиця 3.6 – Структура таблиці "threads"

№ п/п	Назва атрибута	Тип даних	Опис
1.	ID	INTEGER	Код теми
2.	USER_ID	INTEGER	Код користувача
3.	CHANNEL_ID	INTEGER	Код розділу
4.	TITLE	VARCHAR(255)	Заголовок теми
5.	BODY	TEXT	Текст теми

Таблиця 3.7 – Структура таблиці "activity"

№ п/п	Назва атрибута	Тип даних	Опис
1.	ID	INTEGER	Код події
2.	SUBJECT_ID	INTEGER	Код об'єкта
3.	USER_ID	INTEGER	Код користувача
4.	ACTIVITY	VARCHAR(50)	Подія
5.	SUBJECT_TYPE	VARCHAR(50)	Модель події

Нормальна форма - якість зв'язку в реляційній моделі даних, що характеризує його з точки зору надмірності, яка потенційно може привести до логічно помилкових результатів вибірки або зміни даних. Нормальна форма визначається як сукупність вимог, яким має задовольняти відношення.

У реляційній моделі відношення завжди знаходиться в першій нормальній формі за визначенням поняття відношення. Що ж стосується різних таблиць, то вони можуть не бути правильними уявленнями відносин і, відповідно, можуть не перебувати в першій нормальній формі. Мінлива відносини знаходиться в другій нормальній формі тоді і тільки тоді, коли вона знаходиться в першій нормальній формі і кожен не ключовий атрибут неприводимого (функціонально повно) залежить від її потенційного ключа. База даних буде знаходитися в третій нормальній формі, якщо вона приведена до другої нормальної форми і кожен не ключовий стовпець незалежний один від одного.

Таким чином, наша база знаходиться в третій нормальній формі, тому що кожен не ключовий стовпець незалежний один від одного. Це наочно видно на діаграмі нашої бази даних на рисунку 3.2.

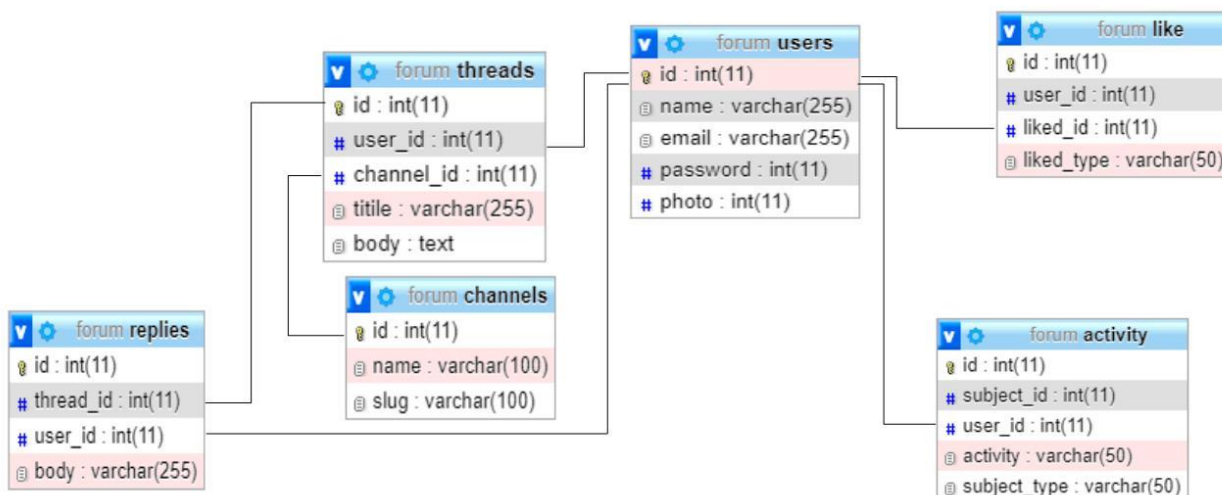


Рисунок 3.2 – Структура бази даних

4 РОЗРОБКА ВЕБ ФОРУМУ

4.1 Структурна схема веб-форуму

Структура сайту - це схема розміщення його основних розділів і сторінок відносно один одного. Це план-схема, яка показує, яким чином будується сайт, логічна зв'язка його сторінок. Такий план-схема зазвичай зображується графічно, у вигляді блоків, пов'язаних стрілочками.

Не варто плутати зовнішню і внутрішню структуру. Зовнішня - це макет сторінки, розташування блоків на ній. Внутрішня ж відображає категорії і приналежність окремих сторінок або матеріалів. Структура сайту залежить від того, який вид сайту взагалі планується робити. Якщо це сайт-візитка, то тут план-схему зробити найпростіше. Тут буде головна сторінка, а від неї йдуть посилання на розділи. У комерційного сайту схема буде набагато складніше. Тут зазвичай три рівня сторінок: головна сторінка сайту, потім категорії, які описують види діяльності, і третій рівень - підвиди. У інформаційного сайту буде схожа структура. Тільки від головної йтимуть основні розділи. А кожна стаття - це буде вже сторінка,

на яку можна потрапити в тому числі і з головною. Причому кожен розділ практично не має інформаційної цінності, це сторінка з посиланнями на статті. У комерційного же сайту кожен розділ містить в собі важливу інформацію. Що стосується інтернет-магазину, це сама складна по своїй організації структура. Тут буде головна сторінка, категорії, підкатегорії і картки товарів. Причому відразу потрібно визначити, які з якостей товару будуть основними, а які - другорядними. Основні стануть підкатегоріями, другорядні - підуть в фільтри.

Добра структура сайту робить його зручним не тільки для відвідувачів. Пошуковим роботам вона теж полегшить роботу, в результаті не буде проблем з індексацією сторінок.

Існує кілька видів структур сайту. Найпростіша лінійна структура - тут кожна сторінка посилається на іншу, одна за одною. Користувач проходить через весь сайт по одному шляху - від початку до кінця. Таку структуру мають Лендінзі, сайти-портфоліо, презентації. Тут основне завдання - провести користувача по точному шляху від початку сайту до кінця, не даючи звернути з нього. Ця ж структура може бути і з відгалуженнями. Використовується, де потрібно познайомити користувача з декількома продуктами, але навігація по сайту теж йде посторінково.

Блокова структура зустрічається вкрай рідко, тому що підходить далеко не для всіх сайтів. В основному - для тих, хто презентує один продукт. Кожен блок описує якусь перевагу, і всі сторінки посилаються один на одного. Тому для просування такої сайт практично готовий, справа за контентом.

Деревоподібна структура - це найпоширеніший вид структури сайту, і зрозуміло чому - вона дозволяє чітко поставити ієрархію сторінок, зв'язати їх з допомогою хлібних крихт. Ця схема є оптимальною для просування сайту, і більшість ресурсів в інтернеті мають в своїй основі саме її.

Весь форум можна розділити 9 сторінок (рис.4.1):

- головна сторінка сайту;
- сторінка реєстрації нового користувача;
- авторизація користувача;

- сторінка списку тем на форумі;
- сторінка перегляду певної теми;
- сторінка для користувачів;
- сторінка пошуку;
- сторінка щоб створити нову тему;
- сторінка відновлення пароля.

При розробці сайту, була написана бібліотека функцій:

- підключення до бази;
- шапка сайту (хедер);
- підвал сайту (футер);
- отримання даних про поточного користувача;
- отримання даних про користувача по його ID.

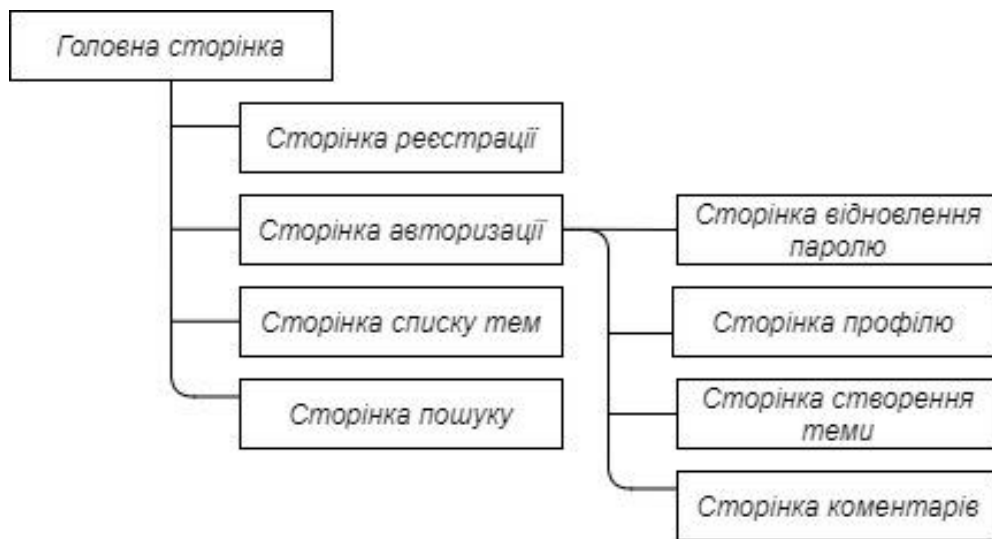


Рисунок 4.1 – Структурна схема веб-форуму

4.2 Програмна реалізація веб-форуму

Структура файлової системи веб-форуму являє собою набір папок-каталогів, виконавчих файлів (html, css, php, javascript документи) і графіки. Будь-який

сучасний браузер підтримує безліч форматів графіки. Найпопулярніші: jpg, png, gif, flash, а також з'явилася можливість виведення векторної графіки svg, canvas. На рисунку 4.2 показано технологічний стек розробленого веб-форуму.

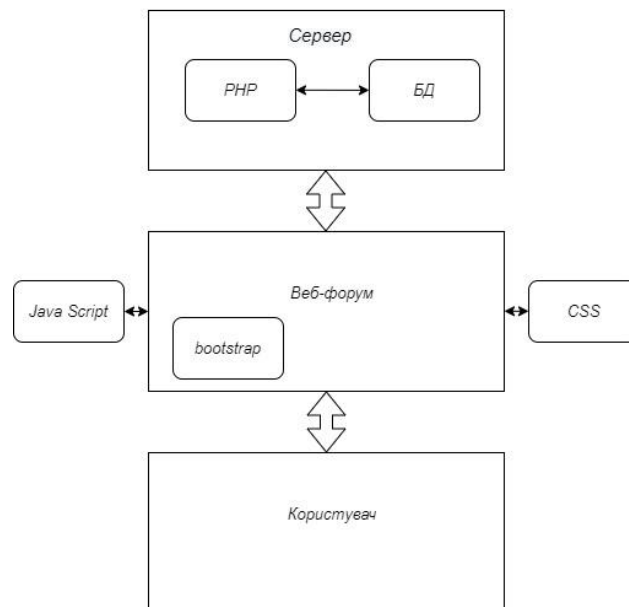


Рисунок 4.2 – Технологічний стек веб-форуму

Система аутентифікації складається з реєстрації і авторизації користувачів.

Реєстрація користувачів лежить в модулі login.php. При відкритті цього модуля, в браузері виводиться 2 поля: для заповнення поштової скриньки користувача і для введення пароля користувача (додаток А.3).

Після того як користувач введе і натисне на кнопку зареєструватися, відправляється AJAX запит на скрипт reg.php, який заносить нового користувача в таблицю, у відповідь скрипт повертає true, якщо все пройшло успішно і користувач доданий в БД або html текст помилок, який виводиться в повідомленні.

Авторизація користувача здійснюється за допомогою модуля login.php, або через панель, яка виводиться в шапці сайту. Модуль, так само як і панель складається з 2 полів: поля для введення поштової скриньки і поля для введення пароля. При авторизації користувача дані із запитом POST надсилаються на скрипт enter.php, який перевіряє правильність входження даних логін - пароль.

Якщо ці дані не збігаються, то відбувається редирект на сторінку `login.php`, в якому за допомогою GET передається тип помилки. Якщо ж все пройшло успішно, і користувач авторизувався, то для нього створюється сесія, в якій існує 1 змінна `id`, в якій зберігається унікальний ідентифікатор користувача, встановлюються або перезаписувати `cookies` і відбувається редирект на сторінку `teams.php`.

Головна сторінка (`index.php`) видно тільки незареєстрованим користувачам, так як зареєстрований користувач завжди перенаправляється на сторінку `forum.php` (додаток А.1-А.2).

При відкритті сторінки `index.php` спочатку відбувається пошук `cookies` користувача. Якщо `cookies` знайдені, то починаємо перевіряти їх. Якщо перевірка пройшла успішно, то заводимо на користувача сесію.

Інакше, виводимо сторінку, для незареєстрованого користувача.

Діаграма класів описує внутрішню структуру сутностей предметної області у вигляді класів і об'єктів (екземплярів класів), визначає інтерфейси класів і відображає взаємозв'язку класів і об'єктів в вигляді відносин. При цьому під класом розуміється абстрактне опис безлічі однорідних об'єктів, що мають однакові атрибути, операції і відносини з об'єктами інших класів; під атрибутом - змістовна характеристика класу, що описує безліч значень, які можуть приймати окремі об'єкти цього класу; під інтерфейсом - іменоване безліч операцій, які характеризують поведінку окремого елемента моделі.

У розробляємо форумі для відображення існуючих в ньому класів використовувалася діаграма класів, яка відображає взаємодію класів між собою. Виділено основні класи: `User`, `Channel`, `Thread`, `Reply`, `Like`, `Activity`. Використовуються інтерфейси `RecordsActivity` і `Likeable`. Їх взаємодія наведено на рисунку 4.3.

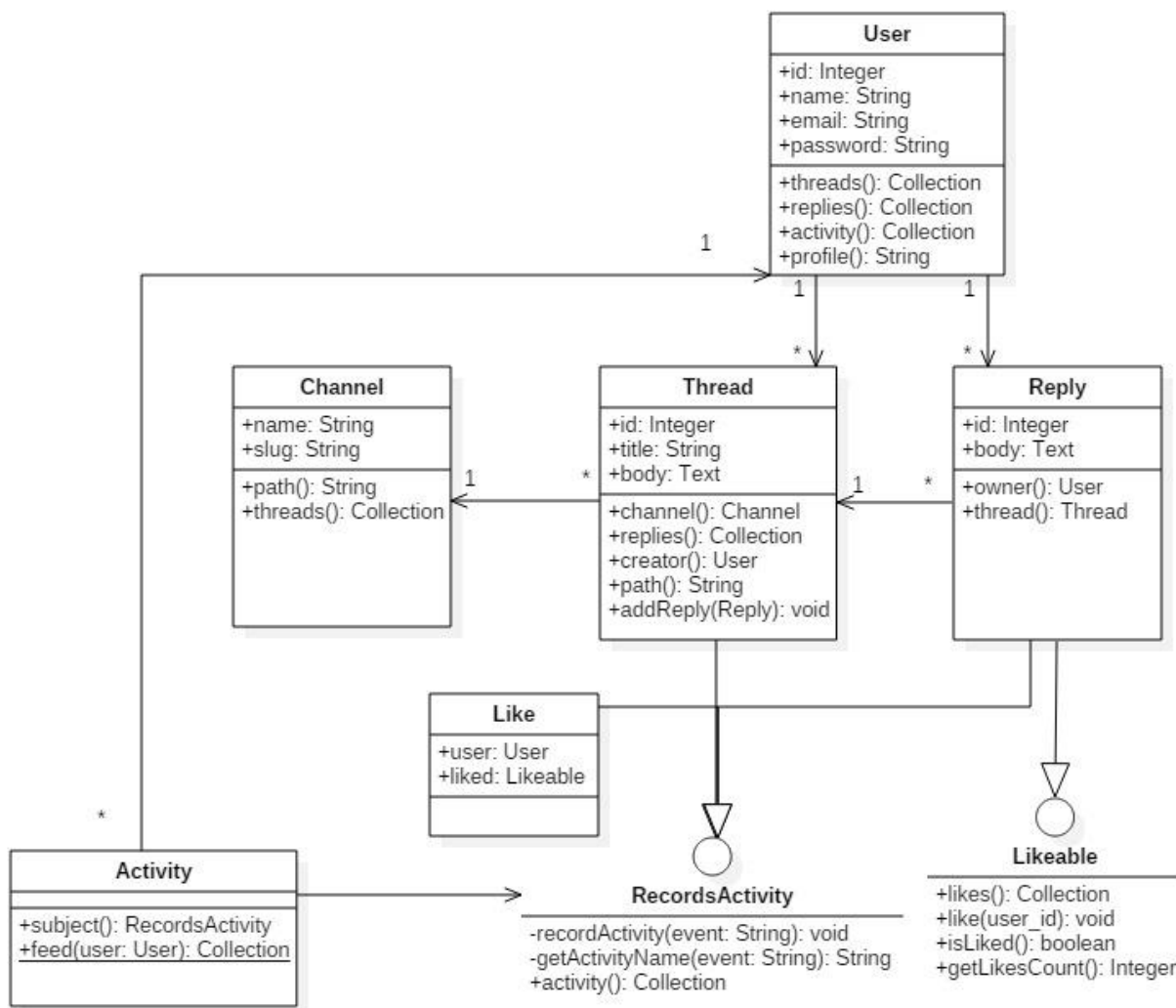


Рисунок 4.3 – Діаграма класів

Каскадні таблиці стилів (Cascading Style Sheets), декларовані як засіб повного контролю над HTML-розміткою, представляються в світлі розглянутої проблеми, як хороший інструмент для організації точного розташування об'єктів на сторінці. CSS дозволяє перевизначити всі властивості будь-якого тега, що призначаються за умовчанням. Стає можливим виконувати вирівнювання текстового блоку щодо сторінки і інших текстових блоків. Використання CSS відкриває нові специфічні можливості, аналогів яким немає з стандартному HTML.

Із застосуванням CSS зростає швидкість створення нової сторінки. Стили, визначені один раз, можуть бути використані необмежену кількість разів в будь-якому місці документа. Важливою деталлю є можливість задати стилі в окремому файлі у вигляді присвоєння різних властивостей тегам і створенні призначених для

користувача стилів. Перевизначення стилів в такому файлі викликає автоматичну зміну стилю відображення всіх об'єктів, для яких застосовувався змінений стиль.

Зупинимося докладніше на способах застосування CSS. Існує чотири способи застосування стилів:

- перевизначення стилю в елементі розмітки;
- розміщення опису стилю в заголовку документа з використанням тега `style`;
- розміщення посилання на зовнішній опис через тег `link`;
- імпорт стилю в документ.

Приклад оформлення стилю для футера веб-форуму наведено на лістингу 4.1.

Лістинг 4.1 - Стиль для футера веб-форуму

```
#myFooter {
    background-color: #3c3d41;
    color: white;
    padding-top: 30px;
}

#myFooter .row {
    margin-bottom: 60px;
}

#myFooter .navbar-brand {
    margin-top: 45px;
    height: 65px;
}

...
@media screen and (max-width: 767px) {
    #myFooter {
        text-align: center;
    }
}

#myFooter{
    flex: 0 0 auto;
    -webkit-flex: 0 0 auto;
}
```

Лістинг стилю форуму наведений у Додатку А.3.

Використання JavaScript при розробці веб-форуму обумовлене наступними причинами. Швидкий для кінцевого користувача: сценарій Java написаний для клієнтської сторони, для підтримки веб-сервера не потрібна підтримка. Він також не потребує компіляції на стороні клієнта, що дає йому певні переваги швидкості. Оскільки сценарій виконується на комп'ютері користувача, в залежності від завдання, результати виконуються майже миттєво. Наприклад, ви можете перевірити будь-який призначений для користувача введення перед відправкою запиту на сервер. Це знижує навантаження на сервер.

Простота: JavaScript відносно простий в освоєнні і реалізації. Він використовує модель DOM, яка забезпечує безліч попередньо функцій для різних об'єктів на сторінках, що робить його легким для розробки сценарію для вирішення користувальницької мети.

Універсальність: JavaScript відмінно працює з іншими мовами і може використовуватися в самих різних додатках. В даний час існує безліч способів використання JavaScript через сервери Node.js. Якщо ви завантажили node.js за допомогою Express, використовуйте базу даних документів, таку як mongodb, і використовуйте JavaScript в інтерфейсі для клієнтів, ви можете створити додаток JavaScript повністю з одного вікна вперед, використовуючи тільки JavaScript.

За допомогою JavaScript в веб-форумі реалізовані функції підключення до БД, відображення ключових елементів редагування коментарів, відправка на сервер даних (лістинг 4.2)

Лістинг 4.2 – Функції JavaScript

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

    /*post message via ajax*/
    $("#reply").on("click", function()
    {
        var message = $.trim($("#message").val()),
            conversation_id =
$.trim($("#conversation_id").val()),
            user_form = $.trim($("#user_form").val()),
            user_to = $.trim($("#user_to").val()),
            error = $("#error");
```

```

        if((message != "") && (conversation_id != "") &&
(user_form != "") && (user_to != ""))
        {
            error.text("Sending...");

$.post("includes/post_message_ajax.php", {message:message, conversatio
n_id:conversation_id, user_form:user_form, user_to:user_to},
function(data) {
            error.text(data);
            //clear the message box
            $("#message").val("");
        });
    });

    //get message
    c_id = $("#conversation_id").val();

    //get new message every 2 second
    setInterval(function()
    {
        //$("#display-
message").load("includes/get_message_ajax.php?c_id="+c_id);

$("#msg_history").load("includes/get_message_ajax.php?c_id="+c_id);
    }, 500);

$("#msg_history").scrollTop($("#msg_history")[0].scrollHeight);
    });

```

Для більш зручного управління конфігурацією і пакетами додатки застосовується файл конфігурації `package.json`. Так, додамо в папку проекту новий файл `package.json` (лістинг 4.3).

Лістинг 4.3 – Файл конфігурації `package.json`

```

{
  "Name": "web-forum",
  "Version": "1.0.0"
}

```

Інформацію про всі додаються пакетах, які використовуються при роботі додатка, додається в секцію `dependencies` (лістинг 4.4).

Лістинг 4.4 – Файл конфігурації package.json – встановлені пакети додатку

```
"dependencies":{
  "classnames": "2.2.6",
  "css-box-model": "^1.2.1",
  "csx": "9.0.0",
  "emojibase-data": "3.2.1",
  "formik": "^2.1.4",
  "highlight.js": "9.15.6",
  "immer": "^2.0.0",
  "karma-viewport": "^1.0.6",
  "lodash": "^4.17.11",
  "lodash.debounce": "^4.0.8",
  "moment": "2.24.0",
  "monaco-editor": "^0.21.2",
  "normalize.css": "8.0.1",
  "react": "^16.9.0",
  "react-aria-live": "2.0.4",
  "react-autosize-textarea": "^7.0.0",
  "react-beautiful-dnd": "12.1.1",
  "react-contenteditable": "^3.3.5",
  "react-day-picker": "^7.4.10",
  "react-dom": "^16.9.0",
  "react-loadable": "5.5.0",
  "react-markdown": "^5.0.3",
  "react-scrolllock": "4.0.1",
  "react-select": "2.4.2",
  "react-spring": "8.0.19",
  "react-virtualized": "9.21.0",
  "reduce-reducers": "1.0.4",
  "redux": "^4.0.4",
  "redux-thunk": "2.3.0",
  "reselect": "4.0.0",
  "shave": "2.5.3",
  "sprintf-js": "1.1.1",
  "stylis": "^4.0.7",
  "swagger-ui-react": "^3.25.0",
  "tabbable": "3.1.1",
  "tar": ">=4.4.2",
  "tsc-watch": "^4.0.0",
  "typescript-fsa": "^3.0.0",
  "typescript-fsa-reducers": "^1.2.0",
  "validator": "10.8.0",
  "webfontloader": "1.6.28"
}
```

4.3 Керівництво користувача

Додаток є сайтом, і щоб почати користуватися ним, необхідно перейти за адресою веб-сайту. Після переходу на сайт, з'явиться головна сторінка сайту як на рисунку 4.4.

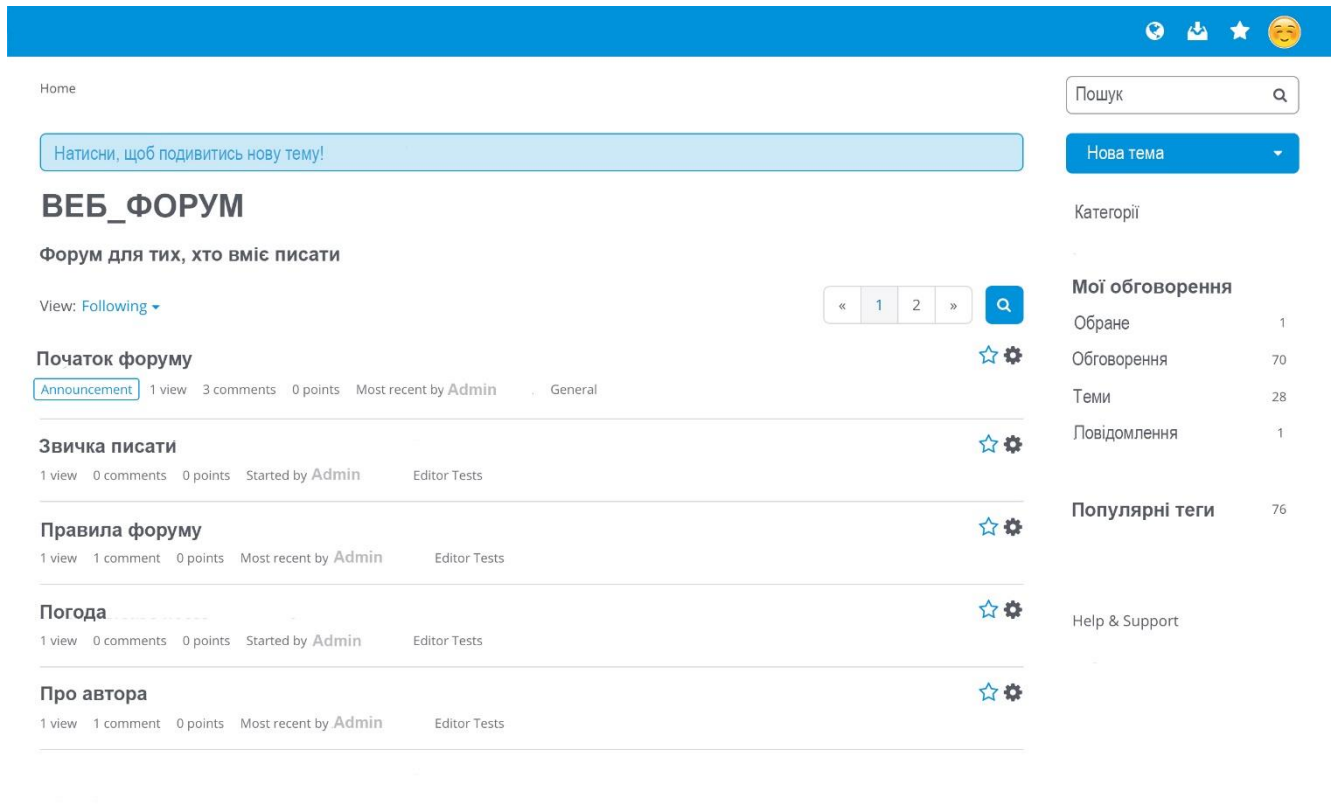


Рисунок 4.4 – Головна сторінка веб-форуму

Оскільки тільки авторизовані користувачі можуть залишати коментарі, то зареєструватися і потім авторизуватися вкрай важливо. Після реєстрації, користувач може увійти, використовуючи email та пароль (рис. 4.5).

Рисунок 4.5 – Форма авторизації

Сторінки створення нової теми та збереження коментаря наведено на рисунках 4.6-4.7.

Рисунок 4.6 – Створення нової теми

Home › Правила форуму

ПРАВИЛА ФОРУМУ

Admin

Правило перше - не має правил

Кментувати | Зберегти | Написати повідомлення автору

Залишити коментар

1. Правила форуму, зазначені нижче, обов'язкові для виконання всіма учасниками форуму без виключень.
2. Правила окремих форумів є доповненнями до загальних правил форуму.
3. Адміністрація залишає за собою право змінювати правила без повідомлення.
4. Доповнення і зміни правил починають діяти з моменту їх опублікування.

Зберегти коментар

Пошук

Нова тема

Категорії

Мої обговорення

Обране	1
Обговорення	70
Теми	28
Повідомлення	1

Популярні теги 76

Help & Support

Рисунок 4.7 – Сторінка «Залишити коментар»

4.4 SEO-оптимізація

Пошукова оптимізація сайту, блогу, форуму, корпоративного або новинного сайту (SEO - Search Engine Optimization) включає в себе ряд дій, спрямованих на поліпшення видимості сайту в пошукових системах: Google, Яндекс та інших.

Процес оптимізації форуму складається з таких частин:

- технічної (створення HTML-карти сайту, редагування файлу robots.txt, настройка URL, верстка, видалення дублікатів сторінок, пошук битих посилань);
- тематичної (підбір ключових слів, оптимізація мета-тегів, заголовків і зображень);

- посилальної: зовнішньої (розміщення статей на зовнішніх майданчиках) і внутрішньої (внутрішня перелінковка - зв'язування сторінок між собою за допомогою посилань).

Файл robots.txt або індексний файл - звичайний текстовий документ в кодуванні UTF-8, діє для протоколів http, https, а також FTP. Файл дає пошуковим роботам рекомендації: які сторінки / файли варто сканувати. Якщо файл буде містити символи не в UTF-8, а в іншому кодуванні, пошукові роботи можуть неправильно їх обробити. Правила, перераховані у файлі robots.txt, дійсні лише щодо того хоста, протоколу і номера порту, де розміщений файл.

Файл повинен розташовуватися в кореневому каталозі в вигляді звичайного текстового документа і бути доступний за основною адресою сайту (лістинг 4.5).

Лістинг 4.5 – Файл robox.txt

```
# Group 1
User-agent: Googlebot
Disallow: /photo/

# Group 2
User-agent: *
Allow: /

Sitemap: http://myfirstforum.com.ua/sitemap.xml
```

Далі на кожній сторінці були додані заголовок сторінки, опис та ключові слова. Ефективність заголовка для SEO-оптимізації сторінки або підвищення конверсії повинна виконувати ряд умов, дотримання яких дозволить уникнути помилок при створенні заголовків сайту:

- довжина заголовка 60 - 100 символів;
- ключові слова на початку заголовка або ближче до початку;
- використовувати менше прямих перерахувань ключових слів;
- стислість, ясність і осмисленість назви сторінки;
- виключити, по можливості, зайві символи (= /! ? * »: <> | + _);
- не вводити в оману відвідувачів, заголовок повинен відповідати вмісту

сторінки;

- не використовувати рідкісні, складні або алегоричні фрази - заголовок повинен бути простим і зрозумілим.

Наприклад для сторінки «Тема» ці пункти будуть такими:

Заголовок сторінки: Обговорення нової теми серед однодумців за темою.

Опис сторінки: Живе спілкування на обрану тему. Дотримуємось всіх умов ведення форуму. Ключові слова: веб-форум, Україна, тема, обговорення, спільнота, спілкування.

Внутрішня структура сайту були оптимізована для пошуку пошукових систем за допомогою розміщення тегів різних рівнів від h1 до h4 включно. h1 використано у хедери та застосовано стиль display: none. Людина його не побачить за ненадібністю, а пошукові системи проаналізують.

У тегу h2 розміщенні всі назви категорій та розділів. У тегу h3 – назви типів за категоріями. Теги іншого рівня впроваджені за потребою.

Для підвищення існуючою ступені оптимізації для пошукових машин у кожний тег зображення було прописано атрибут alt, який застосовується браузером при слабкому підключенні до мережі або якщо користувач відключив завантаження зображень. Крім усього іншого хедер є доступним на будь-якій сторінці та містить гіперпосилання на усі існуючи сторінки.

Семантичне ядро сайту - це список ключових слів, які розподілені на сторінках сайту і за допомогою яких відбувається просування сайту в мережі, це слова та фрази, які найточніше описують вид діяльності, товари або послуги, що пропонує певна веб-сторінка. Така серцевина зазвичай складається з високочастотного ключового слова та інших, які ранжуються за частотою використання. Ядро є основою сайту, від нього залежить, чи буде веб-сторінка в ТОП видачі пошукових систем.

Для оптимізації знаходження сайту потенційними покупцями у мережі Інтернет було розроблене семантично ядро, що є впорядкованим списком запитів користувачів на вибрану тему відсортовані від найбільш частого запиту до найменшого (табл. 4.1).

Таблиця 4.1 – Перелік ключових слів для пошуку «веб-форум»

Keyword	KW. difficulty	Volume	CPC (\$)	PPC comp., %	Results	Social domains
форум веб моделей	5.515	320	0.08	43	416000000	facebook, youtube
веб форум	4.096	210	0.034	4	48100000	facebook, wikipedia
веб кам форум	5.001	140	0.08	45	339000	facebook, wikipedia, youtube
веб модели форум	N/A	50	0.16	75	0	
веб модель форум	3.774	50	0.19	60	6040000	wikipedia, youtube, facebook
форум веб дизайна	0.795		0.014	11	3180000000	youtube, wikipedia, pinterest
форум веб мастеров	3.152	20	0	10	20700000	vk
форум веб дизайнеров	1.071	20	0	29	4670	vk, wikipedia
работа веб моделью форум	5.292	20	0.303	87	283000	vk, facebook, wikipedia
форум веб разработчиков	1.611	20	0	16	692000	vk, wikipedia

Для подальшого розвитку веб-форуму обов'язково потрібно обрати теми форуму. За обраної темою додати спеціалізовані теги та ключові слова.

ВИСНОВКИ

В даному бакалаврському проєкті була розроблена веб-система веб-форуму. Кожен модуль має своє функціональне призначення, яке дозволить створювати, коментувати різні теми форуму.

В ході роботи над бакалаврським проєктом було проведено аналіз вихідних даних і вимог до системи, на основі якого в подальшому будувався вибір апаратного та програмного забезпечення системи. Наступні завдання були реалізовані:

- проаналізовано предметна область і визначені основні поняття, об'єкти і процеси;
- виконано аналіз існуючих веб-форумів: визначено їх функціональні можливості, умови використання і основні характеристики;
- сформульовані основні функціональні можливості і вимоги у вигляді постановки завдання;
- виконано моделювання об'єкта проєктування засобами уніфікованої мови моделювання UML;
- побудована база даних;
- розроблена програма частина веб-форуму;
- проведена SEO-оптимізація налаштувань веб-форуму.

Таким чином, можна констатувати, що поставлена мета бакалаврського проєкту досягнута.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Елманова Н.З. Borland C++Builder 3.0. Архитектура «клиент-сервер», многозвенные системы и Internet-приложения. – М.: Диалог – МИФИ, 1999. – 240 с.
2. Хабибуллин И.Ш. Разработка web-служб средствами Java. – СПб.: БХВ-Петербург, 2003. – 400 с.
3. Токмаков Г. П. Базы данных и знаний. Проектирование баз данных по технологии «клиент-сервер» и разработка клиентских приложений: Учебное пособие. - Ульяновск; УлГТУ, 2005. - 143 с.
4. Гэри Д. Java Server Pages. - М.: "Вильямс", 2002. - 464 с.
5. Флэнаган. Д. JavaScript полное руководство; справочник по самому популярному языку программирования, 7-е издание. – К.: Диалектика, 2020. – 722 с.
6. Сравнение фронтенд-фреймворков в реальных условиях [Электронный ресурс]. – М.: Редакция techrocks.ru, 2017.- Режим доступа: <https://techrocks.ru/2017/12/24/front-end-frameworks-comparison/>
7. Котеров Д.В. PHP7: Полное руководство / Д. В. Котеров, И. В. Симдянов. - СПб: Питер, 2016. - 1088 с.
8. Форсье Д. Django: Разработка веб-приложений на Python. - СПб: Питер, 2009. - 458 с.
9. Стаффер М. Laravel. Полное руководство 2-е издание.- СПб: Питерб 2020.- 512 с.
10. Файни К. SQL: Руководство по изучению языка. – М.: ДМК, 2019. - 456 с.
11. Пономаренко Н. Н. Введение в веб-программирование. HTML-разметка. – Харьков: ХАИ, 2013. – 49 с.
12. Зудилова Т. В., Буркова М. Л. Web-программирование HTML. Учебное пособие. / Т. В. Зудилова, М. Л. Буркова. - СПб: НИУ ИТМО, 2012. – 70 с.

13. Kravets R. Typical ways of web-communities development / R. Kravets, A.M. Peleschyshyn, Yu. Syerov. // Proceedings of the International Conference on Computer Science and Information Technologies, CSIT`2006, September 28th-30th, Lviv, Ukraine. – P.56–58.
14. Серов Ю.О. Моделювання поведінки та класифікація учасників Веб-спільнот на основі нечітких множин // Вісник Національного університету “Львівська політехніка”. – 2008.– № 610 – С. 218–228.
15. Welser H. T. Visualizing the Signatures of Social Roles in Online Discussion Groups / Howard T. Welser, Eric Gleave, Danyel Fisher, Marc Smith [Electronic resource] – Access mode: <http://www.cmu.edu/joss/content/articles/volume8/Welser/>.
16. Brandes U. Exploratory Network Visualization: Simultaneous Display of Actor Status and Connections / Ulrik Brandes, Jörg Raab, Dorothea Wagner [Electronic resource] – Access mode: <http://www.cmu.edu/joss/content/articles/volume2/BrandesRaabWagner.html>.
17. Freeman L. Visualizing Social Networks [Electronic resource] – Access mode: <http://www.cmu.edu/joss/content/articles/volume1/Freeman.html>.
18. Пелешишин А.М. Позиціонування сайтів у глобальному інформаційному середовищі (монографія). – Львів: Вид-во Нац. ун-ту “Львівська політехніка”, 2007. – 260 с.
19. Page, The PageRank citation ranking: Bringing order to the Web / Page, Lawrence; Brin, Sergey; Motwani, Rajeev and Winograd, Terry [Electronic resource] – Access mode: <http://ilpubs.stanford.edu:8090/422/1/1999-66.pdf>

ДОДАТОК А

Лістинги сторінок веб-форуму

Додаток А.1 – Головна сторінка

```

<?php

session_start();
include_once 'includes/dbh.inc.php';
define('TITLE', "Dashboard| -");

$companyName = "Franklin's Fine Dining";

function strip_bad_chars( $input ){
    $output = preg_replace( "/[^a-zA-Z0-9_-]/", "", $input);
    return $output;
}

if(!isset($_SESSION['userId']))
{
    header("Location: login.php");
    exit();
}

include 'includes/HTML-head.php';
?>
    <link href="css/list-page.css" rel="stylesheet">
    <link href="css/loader.css" rel="stylesheet">
</head>

<body onload="pageLoad()" >

    <div id="loader-wrapper">
    <img src='img/500.png' id='loader-logo'>
        <div class="loader">
            <div class="loader_bar"></div>
            <div class="loader_bar"></div>
            <div class="loader_bar"></div>
            <div class="loader_bar"></div>
            <div class="loader_bar"></div>
            <div class="loader_ball"></div>
        </div>
    </div>

    <div id="content" style="display: none">

        <?php include 'includes/navbar.php'; ?>

        <div class="container-fluid">
            <div class="row">
                <div class="col-sm-3" >

                    <?php include 'includes/profile-card.php'; ?>

                </div>

                <div class="col-sm-7" >

                    <div class="text-center p-3">
                        
                    </div>
                </div>
            </div>
        </div>
    </div>

```



```

        <h2 class='text-muted'>DASHBOARD</h2>
        <br>
    </div>

    <ul class="nav nav-tabs justify-content-center" id="myTab"
role="tablist">
        <li class="nav-item">
            <a class="nav-link active" id="forum-tab" data-toggle="tab"
href="#forum" role="tab"
            aria-controls="forum" aria-selected="true">Recent
Forums</a>
        </li>
        <li class="nav-item">
            <a class="nav-link" id="blog-tab" data-toggle="tab"
href="#blog" role="tab"
            aria-controls="blog" aria-selected="false">Recent
Blogs</a>
        </li>
        <li class="nav-item">
            <a class="nav-link" id="poll-tab" data-toggle="tab"
href="#poll" role="tab"
            aria-controls="poll" aria-selected="false">Recent
Polls</a>
        </li>
        <li class="nav-item">
            <a class="nav-link" id="event-tab" data-toggle="tab"
href="#event" role="tab"
            aria-controls="event" aria-selected="false">Recent
Events</a>
        </li>
    </ul>

    <br>

    <div class="tab-content" id="myTabContent">

        <div class="tab-pane fade show active" id="forum"
role="tabpanel" aria-labelledby="forum-tab">

            <div class="d-flex align-items-center p-3 my-3 text-white-
50 bg-purple rounded shadow-sm">
                
                <div class="lh-100">
                    <h1 class="mb-0 text-white lh-100">Latest Forums</h1>
                </div>
            </div>

            <div class="row mb-2">

                <?php
                    $sql = "select topic_id, topic_subject, topic_date,
topic_cat, topic_by, userImg, idUsers, uidUsers, cat_name, (
                        select sum(post_votes)
                        from posts
                        where post_topic = topic_id
                    ) as upvotes
                    from topics, users, categories
                    where topics.topic_by = users.idUsers
                    and topics.topic_cat = categories.cat_id
                    order by topic_id desc, upvotes asc
                    LIMIT 6";

```

```

$stmt = mysqli_stmt_init($conn);

if (!mysqli_stmt_prepare($stmt, $sql))
{
    die('SQL error');
}
else
{
    mysqli_stmt_execute($stmt);
    $result = mysqli_stmt_get_result($stmt);

    while ($row = mysqli_fetch_assoc($result))
    {
        echo '<div class="col-md-6">
            <div class="card flex-md-row mb-4
shadow-sm h-md-250">
                <a
href="posts.php?topic='.$row['topic_id'].'">
                    
                <div class="card-body d-flex
flex-column align-items-start">
                    <strong class="d-inline-block
mb-2 text-primary text-center ml-auto">
                        <i class="fa fa-chevron-up"
aria-hidden="true"></i><br>'.$row['upvotes'].'
                    </strong>
                    <h6 class="mb-0">
                        <a class="text-dark"
href="posts.php?topic='.$row['topic_id'].'">
                            .substr(ucwords($row['topic_subject']),0,15).'...</a>
                        </h6>
                    <small class="mb-1 text-
muted">'.date("F jS, Y", strtotime($row['topic_date'])).'</small>
                    <small class="card-text mb-
auto">Created By: '.ucwords($row['uidUsers']).'</small>
                    <a
href="posts.php?topic='.$row['topic_id'].'">Go To Forum</a>
                    </div>
                </div>
            </div>';
    }
}
?>

</div>

</div>

<div class="tab-pane fade" id="blog" role="blog" aria-
labelledby="blog-tab">

    <div class="d-flex align-items-center p-3 my-3 text-white-
50 bg-purple rounded shadow-sm">
        

```

```

<div class="lh-100">
  <h1 class="mb-0 text-white lh-100">Latest Blogs</h1>
</div>

<div class="row mb-2">

  <?php
    $sql = "select * from Blogs, users
            where blogs.blog_by = users.idUsers
            order by blog_id desc, blog_votes asc
            LIMIT 6";
    $stmt = mysqli_stmt_init($conn);

    if (!mysqli_stmt_prepare($stmt, $sql))
    {
      die('SQL error');
    }
    else
    {
      mysqli_stmt_execute($stmt);
      $result = mysqli_stmt_get_result($stmt);

      while ($row = mysqli_fetch_assoc($result))
      {
        echo '<div class="col-md-6">
              <div class="card flex-md-row mb-4
              shadow-sm h-md-250">
                <div class="card-body d-flex
                flex-column align-items-start">
                  <strong class="d-inline-block
                  mb-2 text-primary">
                    <i class="fa fa-thumbs-up"
                    aria-hidden="true"></i> '.$row['blog_votes'].'
                  </strong>
                  <h6 class="mb-0">
                    <a class="text-dark"
                    href="blog-page.php?id='.$row['blog_id'].'">'.substr($row['blog_title'],0,10).'...</a>
                  </h6>
                  <small class="mb-1 text-
                  muted">'.date("F jS, Y", strtotime($row['blog_date'])).'</small>
                  <small class="card-text mb-
                  auto">'.substr($row['blog_content'],0,40).'...</small>
                  <a href="blog-
                  page.php?id='.$row['blog_id'].'">Continue reading</a>
                </div>
                <a href="blog-
                page.php?id='.$row['blog_id'].'">
                  
                </a>
              </div>
            </div>';
      }
    }
  ?>

</div>

```

```

</div>

<div class="tab-pane fade" id="poll" role="poll" aria-
labelledby="poll-tab">

    <div class="d-flex align-items-center p-3 my-3 text-white-
50 bg-purple rounded shadow-sm">
        
        <div class="lh-100">
            <h1 class="mb-0 text-white lh-100">Latest Polls</h1>
        </div>
    </div>

    <div class="my-3 p-3 bg-white rounded shadow-sm">

        <?php

            $sql = "select p.id, p.subject, p.created, p.poll_desc,
p.locked, (
                    select count(*)
                    from poll_votes v
                    where v.poll_id = p.id
                    ) as votes
                    from polls p
                    order by votes desc
                    LIMIT 5";

            $stmt = mysqli_stmt_init($conn);

            if (!mysqli_stmt_prepare($stmt, $sql))
            {
                die('SQL error');
            }
            else
            {
                mysqli_stmt_execute($stmt);
                $result = mysqli_stmt_get_result($stmt);

                while ($row = mysqli_fetch_assoc($result))
                {

                    echo '<a href="poll.php?poll='.$row['id'].'">
                    <div class="media text-muted pt-3">
                        
                        <p class="media-body pb-3 mb-0 small
lh-125 border-bottom border-gray ">
                            <strong class="d-block text-gray-
dark">'.ucwords($row['subject']).'</strong></a>
                            '.date("F jS, Y",
strtotime($row['created'])).'
                            <br><br>
                            <span class="text-primary" >
                                '.$row['votes'].' User(s)
                            </span>
                        </p>
                        <span class="text-right">';

                    if($row['locked'] === 1)

```

```

        {
            echo '<br><b class="small text-
muted">[Locked Poll]</b>';
        }
        else
        {
            echo '<br><b class="small text-
success">[Open Poll]</b>';
        }

        echo '</span>
            </div>';
    }
}
?>

</div>

</div>

<div class="tab-pane fade" id="event" role="event" aria-
labelledby="event-tab">

    <div class="d-flex align-items-center p-3 my-3 text-white-
50 bg-purple rounded shadow-sm">
        
        <div class="lh-100">
            <h1 class="mb-0 text-white lh-100">Upcoming Events</h1>
        </div>
    </div>

    <div class="my-3 p-3 bg-white rounded shadow-sm">

        <?php

            $sql = "select event_id, event_by, title, event_date,
event_image
                from events
                where event_date > now()
                order by event_date asc
                LIMIT 5";
            $stmt = mysqli_stmt_init($conn);

            if (!mysqli_stmt_prepare($stmt, $sql))
            {
                die('SQL error');
            }
            else
            {
                mysqli_stmt_execute($stmt);
                $result = mysqli_stmt_get_result($stmt);

                while ($row = mysqli_fetch_assoc($result))
                {
                    $earlier = new DateTime(date("Y-m-d"));
                    $later = new DateTime($row['event_date']);
                    $diff = $later->diff($earlier)->format("%a");

```

```

page.php?id='.$row['event_id'].'">
    <div class="media text-muted pt-3">
        
        <p class="media-body pb-3 mb-0 small
lh-125 border-bottom border-gray">
        <strong class="d-block text-gray-
dark">'.ucwords($row['title']).'</strong></a>
        '.date("F jS, Y",
strtotime($row['event_date'])).'<br><br>
        <span class="text-primary" >'.$diff.'
days remaining </span>
    </p>
    <span class="text-primary text-
right">';

$_SESSION['userId'] == $row['event_by'])
    if ($_SESSION['userLevel'] == 1 ||
    {
        echo '<a href="includes/delete-
event.php?id='.$row['event_id'].'&page=forum" >
            <i class="fa fa-trash" aria-
hidden="true" style="color: red;"></i>
            </a>
            </span>';
        }
    else
    {
        echo '</span>';
    }
    echo '</span>
    </div>';
}
?>

</div>

</div>

</div>

<div class="col-sm-2">

    <div class="text-center p-3 mt-5">
        <a href="team.php" target="_blank">
            <i class="creator-icon fa fa-users fa-5x" aria-
hidden="true"></i>
            </a>
            <p><br>THE CREATORS</p>
        </div>

        <a href="forum.php" class="btn btn-warning btn-lg btn-block">-
Forum</a>
        <a href="hub.php" class="btn btn-secondary btn-lg btn-block">-
Hub</a>
        <br><br><br>
        <a href="create-topic.php" class="btn btn-warning btn-lg btn-
block">Create a Forum</a>

```

```

        <a href="create-blog.php" class="btn btn-secondary btn-lg btn-
block">Create a Blog</a>

        </div>
    </div>
</div>
<?php include 'includes/footer.php'; ?>
</div>

<script src="js/jquery.min.js"></script>
<script src="js/bootstrap.min.js" ></script>

<script>
    var myVar;

    function pageLoad() {
        myVar = setTimeout(showPage, 4000);
    }

    function showPage() {
        document.getElementById("loader-wrapper").style.display = "none";
        document.getElementById("content").style.display = "block";
    }
</script>

</body>
</html>

```

Додаток А.2 – Сторінка для зареєстрованих користувачів

```

<?php

    session_start();
    require 'includes/dbh.inc.php';

    define('TITLE', "Forum | -");

    if(!isset($_SESSION['userId']))
    {
        header("Location: login.php");
        exit();
    }

    include 'includes/HTML-head.php';
?>

    <link rel="stylesheet" type="text/css" href="css/list-page.css">
</head>

<body style="background: #f1f1f1">

    <?php include 'includes/navbar.php'; ?>

    <main role="main" class="container">

```

```

<div class="d-flex align-items-center p-3 my-3 text-white-50 bg-purple rounded
shadow-sm">
  
  <div class="lh-100">
    <h1 class="mb-0 text-white lh-100">- Forums</h1>
    <small>Spreading Ideas</small>
  </div>
</div>

<div class="my-3 p-3 bg-white rounded shadow-sm">
  <h5 class="border-bottom border-gray pb-2 mb-0">Top Categories</h5>

  <?php

    $sql = "select cat_id, cat_name, cat_description, (
            select count(*) from topics
            where topics.topic_cat = cat_id
            ) as forums
            from categories
            order by forums desc, cat_id asc
            LIMIT 5";
    $stmt = mysqli_stmt_init($conn);

    if (!mysqli_stmt_prepare($stmt, $sql))
    {
        die('SQL error');
    }
    else
    {
        mysqli_stmt_execute($stmt);
        $result = mysqli_stmt_get_result($stmt);

        while ($row = mysqli_fetch_assoc($result))
        {

            echo '<a href="topics.php?cat='.$row['cat_id'].'">
                <div class="media text-muted pt-3">
                    
                    <p class="media-body pb-3 mb-0 small lh-125 border-bottom
border-gray ">
                        <strong class="d-block text-gray-
dark">'.ucwords($row['cat_name']).'</strong></a>
                        <br>'. $row['cat_description'].'
                    </p>
                    <span class="text-right text-primary">
                        Forums: '. $row['forums'].' <i class="fa fa-book" aria-
hidden="true"></i><br>';

                if ($_SESSION['userLevel'] == 1)
                {
                    echo '<a href="includes/delete-
category.php?id='.$row['cat_id'].'&page=forum" >
                        <i class="fa fa-trash" aria-hidden="true" style="color:
red;"></i>
                        </a>
                    </span>';
                }
                else
                {
                    echo '</span>';
                }

                echo '</div>';
            }
        }
    }
}

```



```

        if ($_SESSION['userLevel'] == 1)
        {
            echo '<small class="d-block text-right mt-3">
                <a href="create-category.php" class="btn btn-primary">Create
Category</a>';
        }
        else
        {
            echo '<small class="d-block text-right mt-3">';
        }
    ?>

    <a href="categories.php" class="btn btn-primary">All Categories</a>
</small>

</div>

<div class="my-3 p-3 bg-white rounded shadow-sm">
    <h5 class="border-bottom border-gray pb-2 mb-0">Top Forums</h5>

    <?php

        $sql = "select topic_id, topic_subject, topic_date, topic_cat, topic_by,
userImg, idUsers, uidUsers, cat_name, (
            select sum(post_votes)
            from posts
            where post_topic = topic_id
            ) as upvotes
        from topics, users, categories
        where topics.topic_by = users.idUsers
        and topics.topic_cat = categories.cat_id
        order by upvotes desc, topic_id asc
        LIMIT 5";
        $stmt = mysqli_stmt_init($conn);

        if (!mysqli_stmt_prepare($stmt, $sql))
        {
            die('SQL error');
        }
        else
        {
            mysqli_stmt_execute($stmt);
            $result = mysqli_stmt_get_result($stmt);

            while ($row = mysqli_fetch_assoc($result))
            {
                echo '<a href="posts.php?topic='.$row['topic_id'].'">
                    <div class="media text-muted pt-3">
                        
                        <p class="media-body pb-3 mb-0 small lh-125 border-bottom
border-gray">
                            <strong class="d-block text-gray-
dark">'.ucwords($row['topic_subject']).'</strong></a>
                            <span class="text-
warning">'.ucwords($row['uidUsers']).'</span><br>
                            '.date("F jS, Y", strtotime($row['topic_date'])).'
                        </p>
                        <span class="text-primary text-center">
                            <i class="fa fa-chevron-up" aria-hidden="true"></i><br>
                            '.$row['upvotes'].'<br>';
            }
        }
    }

```

```

        if ($_SESSION['userLevel'] == 1 || $_SESSION['userId'] ==
$row['idUsers'])
        {
            echo '<a href="includes/delete-
forum.php?id='.$row['topic_id'].'&page=forum" >
                <i class="fa fa-trash" aria-hidden="true" style="color:
red;"></i>
                    </a>
                </span>';
        }
        else
        {
            echo '</span>';
        }
        echo '</span>
                </div>';
    }
}
?>

<small class="d-block text-right mt-3">
    <a href="create-topic.php" class="btn btn-primary">Create A Forum</a>
    <a href="topics.php" class="btn btn-primary">All Forums</a>
</small>

</div>
</main>

<?php include 'includes/footer.php'; ?>

<script src="js/jquery.min.js"></script>
<script src="js/bootstrap.min.js"></script>
</body>
</html>

```

Додаток А.3 – Сторінка авторизації

```

<?php

session_start();
define('TITLE','KLiK');

function strip_bad_chars( $input ){
    $output = preg_replace( "/[^a-zA-Z0-9_-]/", "", $input);
    return $output;
}

if(isset($_SESSION['userId']))
{
    header("Location: index.php");
    exit();
}

include 'includes/HTML-head.php';
?>

</head>

<body>

<section id="cover">
    <div id="cover-caption">
        <div class="container">
            <div class="col-sm-10 offset-sm-1">
                <img src='img/200.png'>
                <h5 class="text-white">Spreading Ideas</h5>
            </div>
        </div>
    </div>

```

```

<br>
<?php
    if(isset($_GET['error']))
    {
        if($_GET['error'] == 'emptyfields')
        {
            echo '<div class="alert alert-danger" role="alert">
                <strong>Error: </strong>Fill In All The Fields
            </div>';
        }
        else if($_GET['error'] == 'nouser')
        {
            echo '<div class="alert alert-danger" role="alert">
                <strong>Error: </strong>Username does not exist
            </div>';
        }
        else if ($_GET['error'] == 'wrongpwd')
        {;
            echo '<div class="alert alert-danger" role="alert">
                <strong>Error: </strong>Wrong password -
                <a href="reset-pwd.php" class="alert-link">Forgot
Password?</a>
            </div>';
        }
        else if ($_GET['error'] == 'sqlerror')
        {
            echo '<div class="alert alert-danger" role="alert">
                <strong>Error: </strong>Website error. Contact
admin to have it fixed
            </div>';
        }
    }
    else if(isset($_GET['newpwd']))
    {
        if($_GET['newpwd'] == 'passwordupdated')
        {
            echo '<div class="alert alert-success" role="alert">
                <strong>Password Updated </strong>Login with your
new password
            </div>';
        }
    }
    ?>
    <form method="post" action="includes/login.inc.php" class="form-inline
justify-content-center">
        <div class="form-group">
            <label class="sr-only">Name</label>
            <input type="text" id="name" name="mailuid"
                class="form-control form-control-lg mr-1"
placeholder="Username">
        </div>
        <div class="form-group">
            <label class="sr-only">Email</label>
            <input type="password" id="password" name="pwd"
                class="form-control form-control-lg mr-1"
placeholder="Password">
        </div>
        <input type="submit" class="btn btn-dark btn-lg mr-1" name="login-
submit" value="Login">
    </form>
    <br><a href="signup.php" class="btn btn-light btn-lg mr-1">Signup</a>
    <br><br>
    <div class="position-absolute login-icons">
        <a href="contact.php">
            <i class="fa fa-envelope fa-2x social-icon" aria-
hidden="true"></i>
        </a>
        <a href="contact.php">

```

```

                                <i class="fa fa-github fa-2x social-icon" aria-
hidden="true"></i>
                                </a>
                                </div>

                                </div>
                                </div>
                                </div>
                                </section>

                                <script src="js/jquery.min.js"></script>
                                <script src="js/bootstrap.min.js" ></script>
                                </body>
                                </html>

```

Додаток А.3 - Стили веб-форуму

```

.btn-primary, .btn-block {
    color: #fff;
    padding: 10px 34px;
    background-color: #4f5173;
    border-color: #fdfffb;
}

.btn-primary:hover, .btn-block:hover {
    color: #cee8f0;
    padding: 10px 34px;
    background-color: #404264;
    border-color: #000;
}

.btn-primary:hover, .btn-primary:focus, .btn-primary:active, .btn-primary.active,
.open .dropdown-toggle.btn-primary {
    color: #cee8f0;
    padding: 10px 34px;
    background-color: #404264;
    border-color: #000;
}

.btn-default {
    color: #000;
    padding: 12px 36px;
    background-color: #fff;
    border-color: #313438
}

.btn-default:hover {
    color: #000;
    padding: 12px 36px;
    background-color: #fff;
    border-color: #000;
    box-shadow: 8px 5px 5px #888
}

.btn-default:hover, .btn-default:focus, .btn-default:active, .btn-default.active,
.open .dropdown-toggle.btn-default {
    color: #fff;
    background-color: #000;
    border-color: #fff;
    padding: 12px 36px;
    box-shadow: 8px 5px 5px #888
}

::selection {

```

```
        background: #000;
        color: #fff
    }

    ::-moz-selection {
        background: #000;
        color: #fff
    }

    .card {
        padding: 1.3rem;
        position: relative;
        display: -ms-flexbox;
        display: flex;
        -ms-flex-direction: column;
        flex-direction: column;
        min-width: 100px;
        word-wrap: break-word;
        background-color: #fff;
        background-clip: border-box;
        border: 1px solid rgba(0, 0, 0, .125);
        border-radius: .25rem;
        box-shadow: 3px 3px 10px 1px rgba(0, 0, 0, 0.1);
    }

    .post-header {
        margin-bottom: 2.2rem;
        margin-top: -1rem;
    }

    .tag {
        padding: 1px 10px 1px 8px;
        color: #e6faff;
        background: #4f5173;
    }

    .post-content {
        padding-top: 1rem;
        padding-bottom: 1.5rem;
    }

    .date {
        margin: -1.3rem;
        background: #636363;
        color: #e6faff;
        margin-bottom: 1rem;
        padding-left: 1rem;
    }

    .span-post-no {
        text-align: right;
        margin-right: 1rem;
        float: right;
    }

    .post {
        margin-bottom: 2rem;
    }

    .likes {
        background: #636363;
        color: #e6faff;
        border-radius: .1rem;
        padding-left: 1rem;
        margin: -1.3rem;
    }

    a {
        color: #ecec;
        text-decoration: none;
    }
}
```

```

a:hover {
  color: #d4525c;
  text-decoration: none;
}

.quote {
  background: #adaebfd6;
  color: #fdfdfd;
  padding-left: .4rem;
  display: block;
  margin: 1rem;
}

.quote-user {
  color: #191b1b;
  font-weight: 600;
}

table {
  margin-bottom: 1.2rem;
}

.blk-link {
  color: black;
}

th {
  text-align: left;
}

/* @media only screen and (max-width: 580px) {*/
.user {
  background: #8386af1c;
  padding: 30px;
  margin-bottom: 40px;
}

/*}*/

.profile-link {
  background: #8386af2e;
  color: #000000;
  padding-top: .1rem;
  padding-bottom: .1rem;
  padding-left: .2rem;
  padding-right: .2rem;
  border-radius: .2rem;
}

li a {
  color: #efefef;
}

.breadcrumb-item.active {
  color: #ff9000;
  font-weight: 600;
}

.breadcrumb {
  background-color: #636363;
  box-shadow: 3px 3px 10px 1px rgba(0, 0, 0, 0.1);
}

.page-item.active .page-link {
  z-index: 1;
  color: #e6faff;
  background-color: #4f5173;
  border-color: #4a4b5a;
  box-shadow: 3px 3px 10px 1px rgba(0, 0, 0, 0.3);
}

```

```
.page-link {
  position: relative;
  display: block;
  padding: .5rem .75rem;
  margin-left: -1px;
  line-height: 1.25;
  color: #4f5152;
  background-color: #fff;
  border: 1px solid #d5d6d8;
  box-shadow: 3px 3px 10px 1px rgba(0, 0, 0, 0.1);
}

textarea.form-control {
  height: auto;
  box-shadow: 3px 3px 10px 1px rgba(0, 0, 0, 0.1);
}

.user-img {
  width: 140px;
  height: 140px;
  object-fit: cover;
  border-radius: 50%;
}

.vote {
  position: absolute;
  right: 0;
  bottom: 40px;
}

.vote-count {
  font-size: 30px;
}

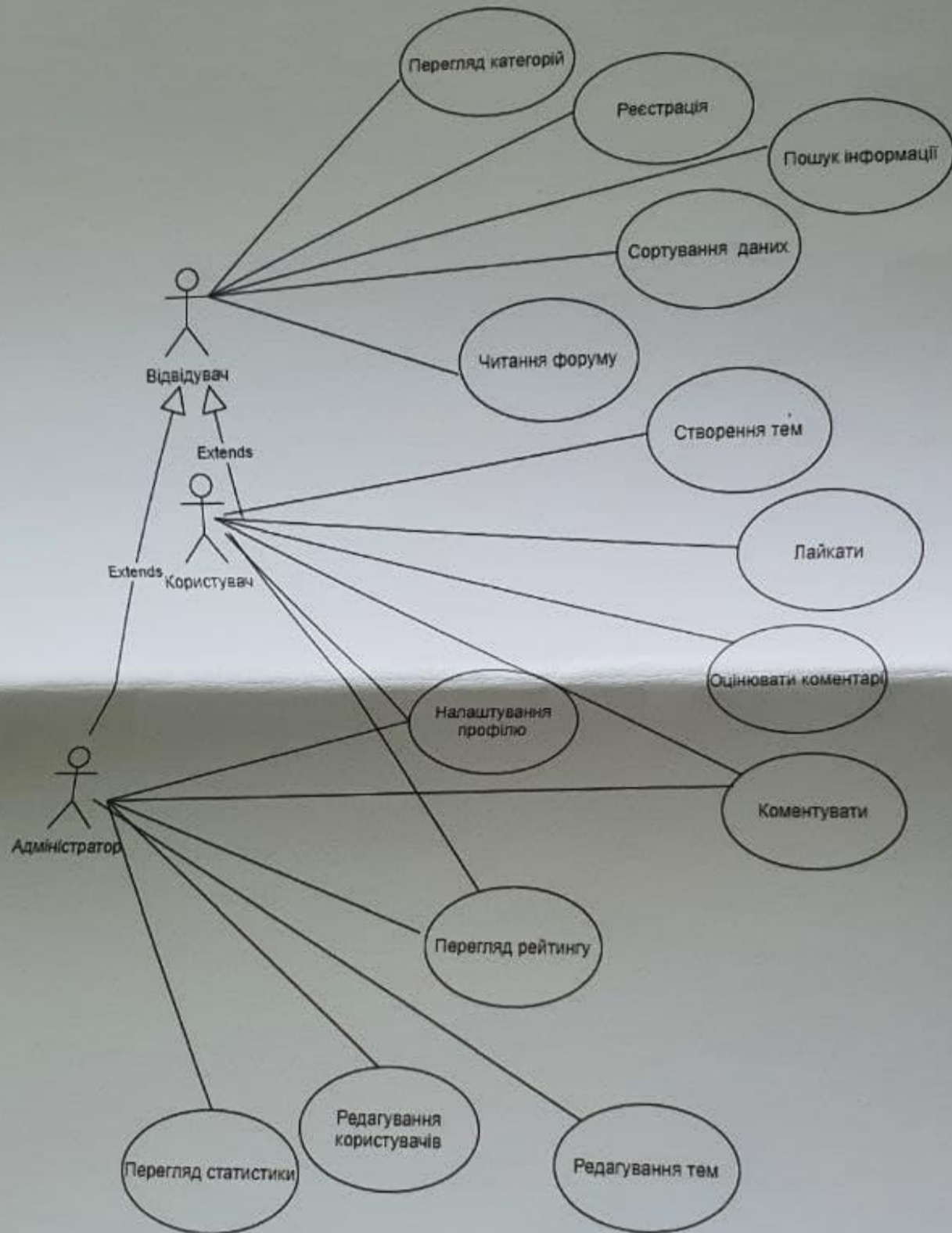
a .fa {
  color: #ff9000;
}

a .fa-trash {
  color: red;
}

a .fa:hover {
  color: #FFC382;
}

.fa {
  color: grey;
}
```

Діаграма варіантів використання веб-форуму



Зам.	Лист	№ докум.	Підп.	Дата
Розроб.		Шумченко С.Г.	<i>[Signature]</i>	
Перев.		Касьян К.М.	<i>[Signature]</i>	
Т.контр.				
Н.контр.		Щербак Н.В.	<i>[Signature]</i>	
Затв.		Кудряметов Р.К.	<i>[Signature]</i>	

13.02070849.00045 ПЛІ

Розробка Веб-форуму з системою адміністрування
Діаграма варіантів використання веб-форуму

Лист	Маса	Масит.
Лист 1		Листів 1

НУ «Запорізька політехніка»
КНТ-527

13.02070849.00045 ПЛ2

Макети сторінок веб-форуму

Хедер

HOME Реєстрація Вхід

Категорії

Категорія 1 Пошук

Категорія 2 Популярні теми

Категорія 2

Футер

Рисунок 1 – Головна сторінка веб-форуму

Хедер

HOME

Вхід

Логін

Пароль

Залишити мене

Приховати мій перебування на форумі

Вхід

Реєстрація

Умови реєстрації

Реєстрація

Футер

Рисунок 2 – Форма Реєстрації/Авторизації

Хедер

HOME Профіль Фото

Нова тема

Обрати категорію Пошук

Назва теми Мої теми

Опис

Кнопки редагування

Теги

Зберегти

Футер

Рисунок 3 – Сторінка «Нова тема»

Хедер

HOME Профіль Фото

ТЕМА

Коментар 1 Пошук

Коментар 2 Мої теми

Коментар 3 Популярні теми

Текст

Кнопки редагування

Додати коментар

Футер

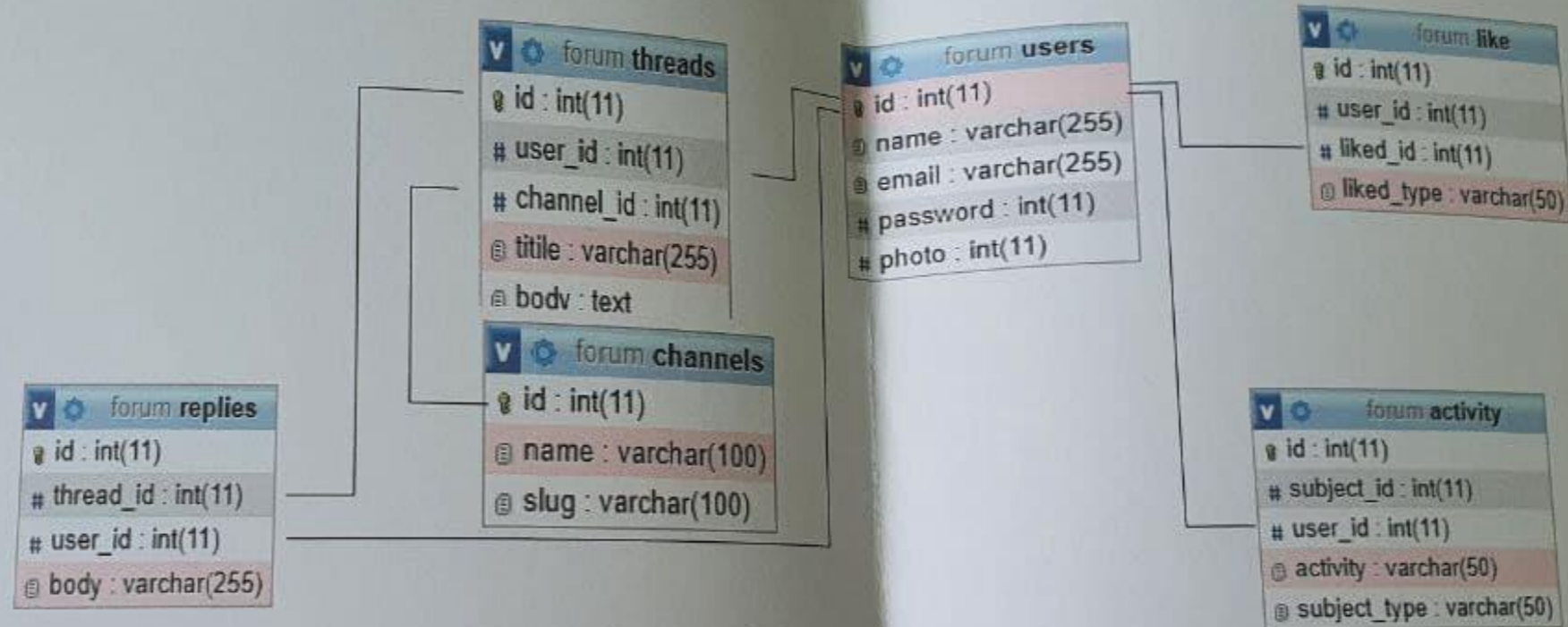
Рисунок 4 – Сторінка «Коментарі»

13.02070849.00045 ПЛ2

Зам.	Лист	Лі док.	Підп.	Дата	Лист	Місяц	Місяць
Розроб.	Шумченко С.Г.						
Перев.	Касьян К.М.				Лист 1		Листів 1
Т.контр.					НУ «Запорізька політехніка»		
Н.контр.	Щербак Н.В.				КНТ-527		
Затв.	Кудерметов Р.К.						

13.02070849.00045 ПЛІЗ

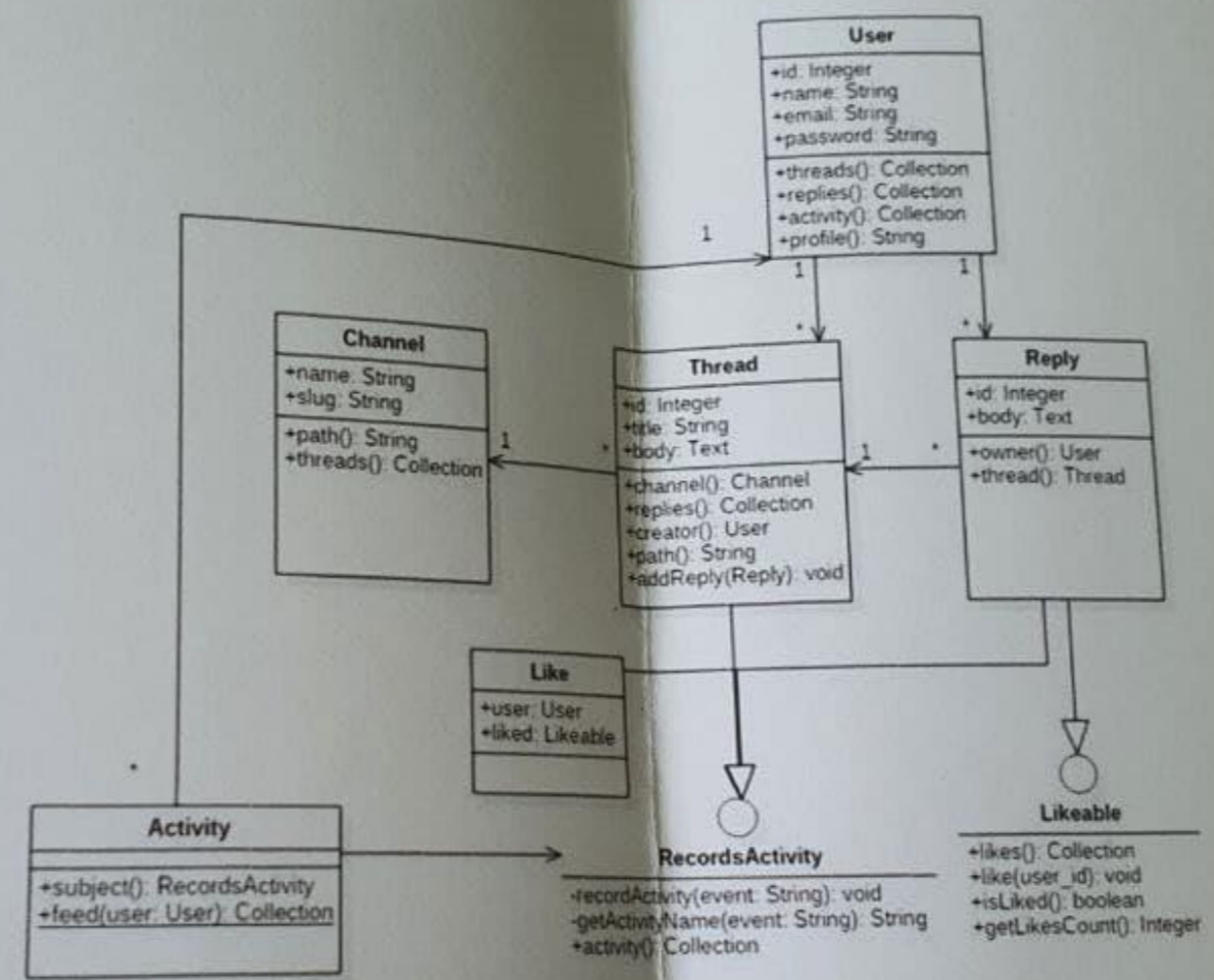
База даних Веб-форуму



					13.02070849.00045 ПЛІЗ		
					Розробка Веб-форуму з системою адміністрування		
					База даних веб-форуму		
Зам.	Лист.	№ докум.	Підп.	Дат.	Лист 1	Лист 2	Лист 3
Розроб.		Шумченко С.Г.	<i>[Signature]</i>				
Перев.		Касян К.М.	<i>[Signature]</i>				
Т.контр.							
Н.контр.		Щербак Н.В.	<i>[Signature]</i>				
Затв.		Кудряшова Р.К.	<i>[Signature]</i>				

ІТУ «Запорізька політехніка»
КНТ-527

Діаграма класів



					13.02070849.00045 ПЛ14		
Зам.	Лист	№ докум.	Підп.	Дата	Розробка Веб-форуму з системою адміністрування		
Розроб.	Шумченко С.Г.						
Перев.	Касьян К.М.				Діаграма класів		
Т.контр.							
Н.контр.	Щербак Н.В.				НУ «Закарпатська політехніка» КНТ-527		
Затв.	Куберметов Р.К.						

13.02070849.00012 ПЛ15

Інтерфейс веб-форуму

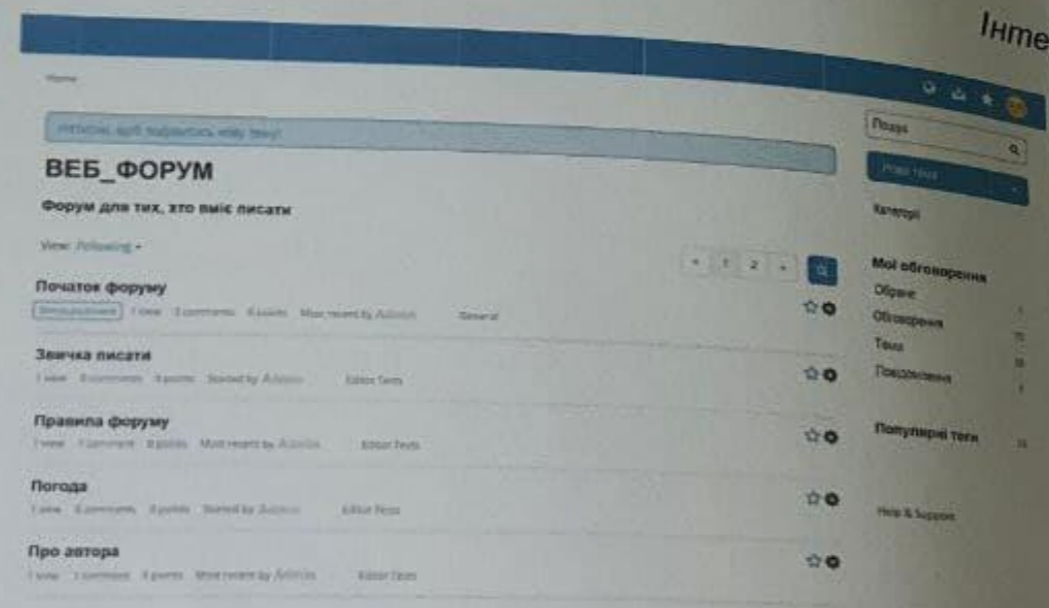


Рисунок 1 – Головна сторінка веб-форуму

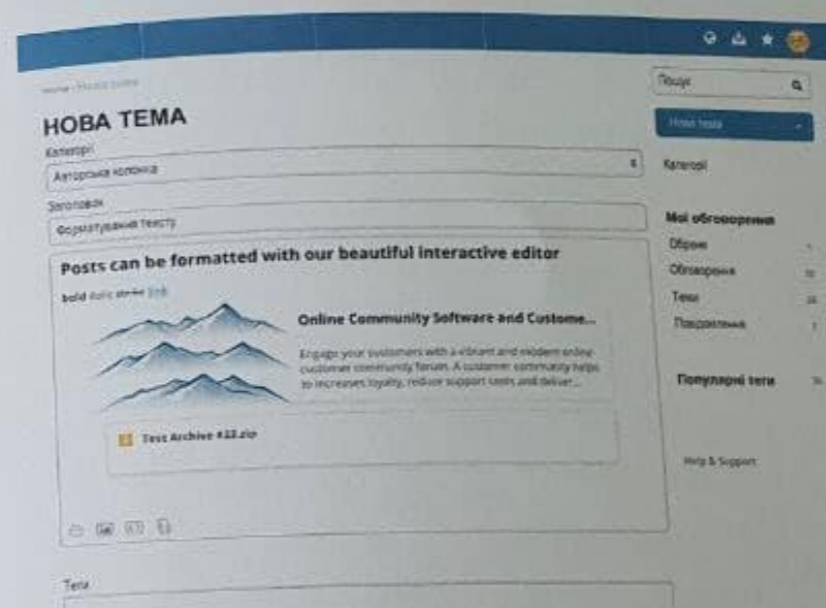


Рисунок 2 – Створення нової теми

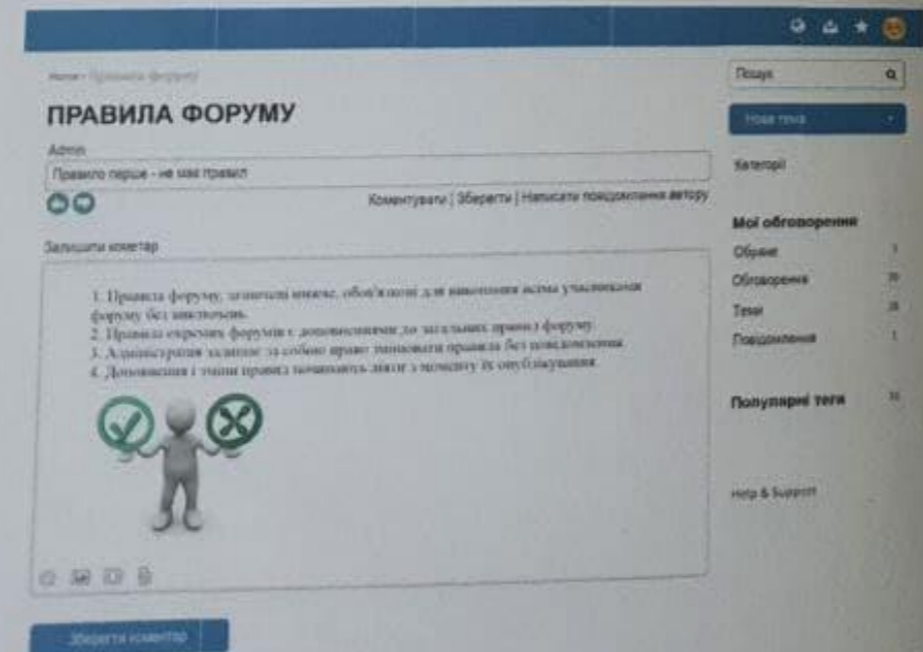


Рисунок 3 – Сторінка «Залишити коментар»

13.02070849.00045 ПЛ15

Розробка Веб-форуму з системою адміністрування
Інтерфейс веб-форуму

Зам.	Лист	№ докум.	Підп.	Дата
Розроб.		Шумченко С.Г.	<i>[Signature]</i>	
Перев.		Касьян К.М.	<i>[Signature]</i>	
Т.контр.				
Н.контр.		Щербак Н.В.	<i>[Signature]</i>	
Затв.		Кудерметов Р.К.	<i>[Signature]</i>	

Лист	Місяц	Місяць
Лист 1		Листів 1

НУ «Запорізька політехніка»
КНТ-527